

**CROSS COUNTRY EVENT MANAGEMENT SYSTEM FOR MALAYSIAN
SECONDARY SCHOOL**

KHEW LI TIEN


**A project report submitted in partial fulfilment of the
requirements for the award of Bachelor of Science
(Hons.) Software Engineering**

**Lee Kong Chian Faculty of Engineering and Science
Universiti Tunku Abdul Rahman**

April 2021

DECLARATION

I hereby declare that this project report is based on my original work except for citations and quotations which have been duly acknowledged. I also declare that it has not been previously and concurrently submitted for any other degree or award at UTAR or other institutions.

Signature : 

Name : Khew Li Tien

ID No. : 17UEB04367

Date : 15 April 2021

APPROVAL FOR SUBMISSION

I certify that this project report entitled “**Cross Country Event Management System for Malaysian Secondary School**” was prepared by **Khew Li Tien** has met the required standard for submission in partial fulfilment of the requirements for the award of Bachelor of Science (Honours) Software Engineering at Universiti Tunku Abdul Rahman.

Approved by,

Signature : *D.gunavathi*

Supervisor : Ms. Gunavathi a/p Duraisamy

Date : 07 May 2021

Signature : _____

Co-Supervisor : _____

Date : _____

The copyright of this report belongs to the author under the terms of the copyright Act 1987 as qualified by Intellectual Property Policy of Universiti Tunku Abdul Rahman. Due acknowledgement shall always be made of the use of any material contained in, or derived from, this report.

© 2021 Year, Khew Li Tien. All right reserved.

ACKNOWLEDGEMENTS

I would like to thank everyone who had contributed to the successful completion of this project. I would like to express my gratitude to my research supervisor, Ms Gunavathi a/p Duraisamy for her invaluable advice, guidance and her enormous patience throughout the development of the research.

In addition, I would also like to express my gratitude to my loving parents and friends who had helped and given me encouragement during the completion of the system and this report.

ABSTRACT

Cross country event can be an enriching experience for most Malaysian secondary school students. However, the management for such event can be exhaustive due to the sizeable load of recording and sorting large groups of runners while maintaining focus under the hot weather of Malaysia. In addition to most Malaysian secondary school employing manual logging method for recording runners, the result of recording can be less imprecise and slow. The solution to an exact recording and automation of task for event personnel is by employing dedicated tracking equipment such as chips and tracking mat, but will remain as a poor choice for financial management. This project intends to ease the event management process and reduce the downsides of current common recording methods through the usage of QR code scanning for recording participants, and through providing tools for finalizing ranking based on the recorded data. Rapid Application Development methodology was adopted and functionalities prototyping was thoroughly emphasized and performed during the system development. The system was developed with the guidance of the requirements elicited through the evaluation from the survey completed by SMJK Catholic PJ and exploratory study on available resources. User Acceptance Test and System Usability Test was performed with several participants of secondary schools and most had identified that the system is able to perform the core tasks with ease of learning. Additionally, evaluation done such as items encompassing unit testing, integration testing, system testing and static code reviewed had positively indicated that the system is stable. In conclusion, the project is believed to have achieved all the elicited objectives of developing a cross country event management system that overcomes the weakness of manual logging method by employing QR scanning for recording and providing tools for finalizing results.

TABLE OF CONTENTS

DECLARATION	ii
APPROVAL FOR SUBMISSION	iii
ACKNOWLEDGEMENTS	v
ABSTRACT	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES	xiii
LIST OF SYMBOLS / ABBREVIATIONS	xvi
LIST OF APPENDICES	xvii

CHAPTER	TITLE	PAGE
1	INTRODUCTION	1
	1.1 Introduction	1
	1.2 Problem Statement	2
	1.2.1 Accuracy: Manual ranking tracking by teachers can be inaccurate	2
	1.2.2 Speed and size: Manual ranking tracking efficiency deteriorates over time and participants size	2
	1.2.3 Cost: Sophisticated tracking equipment can cause financial constraint for secondary schools	2
	1.3 Project Goal	3
	1.4 Project Objectives	3
	1.5 Final Solution	4
	1.6 Final Approach	6
	1.6.1 Rapid Application Development (RAD)	6
	1.7 Project scope	7
	1.7.1 Target User	7

	1.7.2	Modules Covered	7
	1.7.3	Modules Not Covered	9
2		Literature Review	10
	2.1	Software methodology	10
	2.1.1	Agile methodology	10
	2.1.2	Traditional approach	10
	2.1.3	Differences between Rapid Application Development (RAD), Agile and traditional approach.	11
	2.2	Native and hybrid mobile application	12
	2.2.1	React Native	12
	2.2.2	Differences between Native and Hybrid mobile application	12
	2.3	Database	14
	2.3.1	Relational database vs non-relational database	14
	2.3.2	Selection of suitable NoSQL provider	15
	2.3.3	Selection of suitable SQL database	16
	2.4	Existing systems	17
	2.4.1	Active Endurance	17
	2.4.2	MyLaps	20
	2.4.3	JomRun	24
	2.4.4	Existing system comparison	25
3		Methodology and Project Management	28
	3.1	Development methodology	28
	3.2	Project plan	30
	3.2.1	Work Breakdown Structure	30
	3.2.2	Gantt Chart	31
	3.3	Development tools	33
	3.3.1	Programming Language	33
	3.3.2	Framework	33
	3.3.3	Databases	34
	3.3.4	Other Tools	35

4	Project Specification	36
4.1	System Overview	36
4.2	Requirements Discovery Method	37
4.3	Requirement Specification	39
	4.2.1 Functional Requirements	39
	4.2.2 Non-functional Requirements	41
5	Project Design	42
5.1	Software Architecture Design	42
5.2	Database Design	44
	5.2.1 Entity Relationship Diagram (ERD)	44
5.3	Use Case	45
	5.3.1 Use Case diagram	45
	5.3.2 Use-Case specification	45
5.4	User Interface Design	58
	5.4.1 Web Application	58
	5.4.2 Mobile Application	68
6	Project Implementation	77
6.1	Route Design	77
6.2	API endpoints	78
6.3	Data Flow Management	80
	6.3.1 Redux State Management	80
	6.3.2 Redux State Management Dictionary	83
	6.3.3 Route Data Flow	97
7	Testing and Evaluation	98
7.1	Test Plan	98
	7.1.1 Purpose	98
	7.1.2 Scope	98
	7.1.3 Test Items	98
	7.1.4 Features to be Tested	99
	7.1.5 Features Not to be Tested	100

7.1.6	Test Approach and Strategy	101
7.1.7	Item Pass Fail Criteria	101
7.1.8	Test Deliverable	104
7.1.9	Entry Criteria	105
7.1.10	Exit Criteria	105
7.2	Manual Testing	105
7.2.1	Unit Test	105
7.2.2	Integration Test	111
7.2.3	System Test	114
7.3	User Acceptance Test	116
7.4	System Usability Test	117
7.5	Static Code Review	118
8	Conclusion and Discussion	120
8.1	Conclusion	120
8.2	Limitation	120
8.3	Recommendation	121
	REFERENCES	122
	APPENDICES	124

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.2.2.1	: Differences in Native and Hybrid Mobile Application Based On Criteria	12
Table 2.3.1.1	: Table of Differences Between Relational Database and Non-Relational Database	14
Table 2.3.2.1	: Table of Differences Between Firebase, MongoDB and RethinkDB	15
Table 2.3.3.1	: Table of Differences Between MariaDB, Oracle and MySQL	16
Table 2.4.4.1	: Table of Differences Between All The Existing Systems Studied	25
Table 6.2.1	: List of API Endpoints For Both Web and Mobile Applications	78
Table 6.3.2.1.1.1	: Table of Redux Components Created For The Global App Container	83
Table 6.3.2.1.2.1	: Table of Redux Components Created For The CreateEvent Container	85
Table 6.3.2.1.3.1	: Table of Redux Components Created For The Event Container	87
Table 6.3.2.1.4.1	: Table of Redux Components Created For The Register Participants Container	89
Table 7.2.1.1.1	: Account Management Module Unit Test Case	106
Table 7.2.1.2.1	: Create Event Component Unit Test Cases	107
Table 7.2.1.2.2	: Update Event Feature Unit Test Cases	108
Table 7.2.1.2.3	: Event Page Unit Test Cases	109
Table 7.2.1.2.4	: Register Participant Feature Unit Test Cases	110
Table 7.2.1.3.1	: Record Participant Module Unit Test Cases	110
Table 7.2.1.4.1	: Search Module Unit Test Cases	111

Table 7.2.2.1.1	: Event Module and Register Participant Module Integration Test	112
Table 7.2.2.2.1	: Search Module and Event Module Integration Test	112
Table 7.2.2.3.1	: Register Participant and Record Participant Integration Test	112
Table 7.2.2.4.1	: Event Management and Record Participant Integration Test	113
Table 7.2.2.5.1	: Account Management and Record Participant Integration Test	113
Table 7.2.3.1.1	: Event Preparation Flow System Test	114
Table 7.2.3.2.1	: Event Progression Flow System Test	115
Table 7.2.3.3.1	: Event Progression Flow System Test	115
Table 7.3.1	: Test Result of User Acceptance Test	116
Table 7.4.1	: Result of System Usability Test	117
Table 7.6.1	: Summary Of Manual Testing	119

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 1.5.1	: Figure Describing Process Flow of The Cross Country Management System	4
Figure 1.5.2	: Diagram of Flux Architecture	5
Figure 1.5.3	: Diagram of Client-Server Architecture as Top View of the System	5
Figure 1.6.1.1	: RAD Flow Diagram (LucidChart.com, n.d.)	6
Figure 1.7.2.1	: Modules for Cross Country Management System's Web Application	7
Figure 1.7.2.2	: Modules for Cross Country Management System's Mobile Application	8
Figure 2.4.1.1	: Path Flow of Active Applications For A Complete Sports Event	17
Figures 2.4.1.2	: Path Flow For Active Endurance Experience Mobile Application	18
Figures 2.4.1.3	: Screenshots of ActiveWorks Web Application Home Page and Event Page	19
Figure 2.4.1.4	: Screenshot of ActiveWorkss People Section	19
Figure 2.4.1.5	: Screenshot of Active Endurance Timing Overview Screen	20
Figure 2.4.2.1	: Screenshot of MyLaps CMS Main Page	21
Figures 2.4.2.2	: Screenshot of MyLaps CMS Further Event Configuration	21
Figures 2.4.2.3	: Screenshots of Sporthive Live Mobile Application	22
Figures 2.4.2.4	: Illustration of Event Day Operations and Sample <i>Bibtag</i> (mylaps.com)	23
Figure 2.4.3.1	: Screenshot of JomRun Mobile Application	24
Figure 3.1.1	: RAD Flow Diagram (LucidChart.com, n.d.)	28

Figure 3.2.1.1	: Work Breakdown Structure For Cross Country Event Management System Based On RAD Methodology	30
Figure 3.2.2.1	: List of Planned Tasks	31
Figures 3.2.2.2	: Gantt Charts of The Planned Tasks	32
Figure 4.1.1	: Figure Describing Process Flow of The Cross Country Management System	36
Figure 5.1.1	: Flux Software Architecture Design	42
Figure 5.1.2	: Client-Server Architecture Design	43
Figure 5.2.1.1	: Entity Relationship Diagram For CCEMS	44
Figure 5.3.1.1	: Use Case Diagram For Cross Country Event Management System.	45
Figure 5.4.1.1	: Visitor Page	58
Figure 5.4.1.2	: Visitor Login Form	59
Figure 5.4.1.3	: Visitor Registration Form	59
Figure 5.4.1.4	: User Home	60
Figure 5.4.1.5	: Create Event First Step	60
Figure 5.4.1.6	: Create Event Second Step	61
Figure 5.4.1.7	: Create Event Third Step	61
Figure 5.4.1.8	: User's All Event List Page	62
Figure 5.4.1.9	: Organizer Upcoming Event Page	63
Figure 5.4.1.10	: Organizer Ongoing Event Page	63
Figure 5.4.1.11	: Completed Event Page	64
Figure 5.4.1.12	: Event Page Edit State	64
Figure 5.4.1.13	: Participant List Page	65
Figure 5.4.1.14	: Participant List QR View Page	65
Figure 5.4.1.15	: Register Participant Page	66

Figure 5.4.1.16 : User Profile Page	67
Figure 5.4.1.16 : Search Result Page	67
Figure 5.4.2.1 : Entry Screen	68
Figure 5.4.2.2 : Home page	69
Figure 5.4.2.3 : Event List Screen	70
Figure 5.4.2.4 : Create Event Screen	71
Figure 5.4.2.5 : Completed Event Screen	72
Figure 5.4.2.6 : Organizer Upcoming Event Screen	73
Figure 5.4.2.7 : Organizer Ongoing Event Screen	74
Figure 5.4.2.8 : Record Participant Screen Successful Scan	75
Figure 5.4.2.9 : Record Participant Screen Successful Record	76
Figure 6.1.1 : Web Application Route Diagram	77
Figure 6.1.2 : Mobile Application Route Diagram	77
Figure 6.3.1.1 : Snippet of Redux Action For Handling Retrieve Event Data In Web Application	81
Figure 6.3.1.2 : Snippet For Async Redux State, Action and Reducer Definitions In Mobile Application	81
Figure 6.3.1.3 : Snippet For Async Action Definition In Mobile Application	82
Figure 6.3.1.4 : Diagram of Redux State Management Flow	82
Figure 6.3.2.2.1 : Snippet For The Slice Building of Event Redux Wrapper	91
Figure 6.3.2.2.2 : Snippet For The Slice Building of User Redux Wrapper	92
Figure 6.3.2.2.3 : Snippet For Usage of Redux Slice In Login	92
Figure 6.3.3.1 : Dataflow Diagram for the Mobile Application	97
Figure 7.5.1 : Review Result For Web Application	118
Figure 7.5.2 : Review Result For Mobile Application	118

LIST OF SYMBOLS / ABBREVIATIONS

<i>RAD</i>	Rapid Application Development
<i>APP</i>	application
<i>UML</i>	Unified Modelling Language
<i>UI</i>	User Interface
<i>CCEMS</i>	<i>Cross Country Event Management System</i> (The project application name)
<i>UAT</i>	User Acceptance Test
<i>SUT</i>	System Usability Test
<i>AWS</i>	Amazon Web Service
<i>RDS</i>	Amazon Relational Database Service

LIST OF APPENDICES

APPENDIX	TITLE	PAGE
APPENDIX A:	Result of survey	124
APPENDIX B:	Samples of runners' bib implemented by SMJK Katholik PJ	131
APPENDIX C:	Code Review From CodeBeat	133
APPENDIX D:	User Acceptance Test and System Usability Test	134

CHAPTER 1

INTRODUCTION

1.1 Introduction

Cross country event is included as one of the mandatory Malaysian secondary school event to be participated as stated within the ‘1 Murid 1 Sukan’ principle with intention to promote talents, sportsmanship and sport participation of Malaysian students (Kementerian Pelajaran Malaysia, 2011). The historical view on recreational physical activities such as running has always been positive, with individuals gaining more benefits than disadvantages in a long term. Several authors such as Lee et al.(2014) are in support of running activities as beneficial fitness training, stating runners are 45% less prone to cardiovascular risk compared to non-runners.

A common Malaysian secondary school cross country event is guarded by teachers and student personnel at each checkpoint, in which the passing runners’ ID and time are recorded using the method of the school’s choosing. The event is concluded once the ranking of the top runners has been tabulated. Cross country event organized by secondary school often are not for monetary purposes. Hence, most secondary schools adopt a manual logging method for the event, whereby personnel guarding at each checkpoint shall record the passing runners’ number onto a physical notebook. The process is then followed with lengthy operations of merging data and ranking students manually.

Manual logging that is commonly employed is extremely vulnerable to human-made errors. Event personnel can be overwhelmed when attempting to record a group of packed participants. Additionally, efficiency of manual recording will eventually deteriorate over time. Any mismanagement by the secondary school on the ranking can be detrimental toward students’ motivation in cultivating their sport interest, which in turn will be defeating the purpose of ‘1 Murid 1 Sukan’ initiative.

If the event is managed using a dedicated tracking equipment, the event can be completed smoothly and with satisfaction. However, funding is an obvious obstacle for secondary school to prepare the equipment. Alternatively, a dedicated cross country event management application can be created to overcome the inefficiency and ineffectiveness of manual logging method.

1.2 Problem Statement

1.2.1 Accuracy: Manual Ranking Tracking by Teachers Can Be Inaccurate

A majority of secondary schools' cross country events log students' ranking manually by hand as observed by the responsible teachers in which human mistake may come into effect. Factors such as environment distractions or being overwhelmed by multiple entries in a short period of time can affect accuracy of logging manually. As a result, arrival detail recorded can be imprecise to the actual data and personnel can mistakenly record or organize the wrong information and ranking for the participants. The system developed should be able to quickly and precisely record participants data to help them better organize ranking.

1.2.2 Speed and Size: Manual Ranking Tracking Efficiency Deteriorates Over Time and Participants Size

Logging of data takes time and energy. In a high traffic area during peak time, manual logging may hinder runners' speed, resulting in temporary queue(s) being formed. Runners should not be restrained from moving forward due to the personnel's speed. From management's viewpoint, it can be energy taxing for personnel to guard under hot weather. Therefore, it is often observed that participants beyond the ranking eligible for rewards are not recorded to save energy and to avoid the hassle of organizing a large number of handwritten data. The system developed should make it easy and quick to record and organize data for the personnel.

1.2.3 Cost: Sophisticated Tracking Equipment Can Cause Financial Constraint For Secondary Schools

Professional marathon events often use equipment such as 'timing chips' or 'tracking pad' to accurately log the progress of all participants. In average, American marathon event participants pays around 13USD for marathon utilities such as tracking bibs and sporting accessories, in which the cost will inversely increase in decreasing proportion of event scale (Page, 2020). This is yet to factor in the cost of renting tracking pads for checkpoints use. Reusable gadgets and chips can be costly to be purchased, maintained and ensure it is returned for the event next year. Given that the cross-country event is organized by public secondary school and the demographics and sponsors are usually the teachers, students and parents, tracking equipment would be a poor choice of school funds usage. Hence, middle schools often find volunteers to help manually log

participants ranking. The system developed need to utilize available equipment to reduce cost.

1.3 Project Goal

The project goal is to create a standard management system for secondary school to manage their cross country events and to quickly and accurate record passing runner at each checkpoint using QR code scanning or manual entry method. The system created will be easily accessible through web and using user's own mobile devices.

1.4 Project Objectives

The main goal of the project is to create a system that assist Malaysian secondary schools in organizing any cross-country event.

1. To implement recording techniques such as QR code scanning facilitated using android device camera and manual entry method to easily record passing participants.
2. To improve tracking time accuracy and logging speed by designing mobile application functionality that automatically create timestamp based on current time of scanning and recording.
3. To save cost for hardware-based tracking equipment by using user's available android mobile phone as the tracking equipment.
4. To facilitate accurate event administration by providing tools for producing finalized ranking for a completed event.
5. To create a functional database accessible through both web application and mobile application that keep record of all cross country events and its matching student data.

1.5 Final Solution

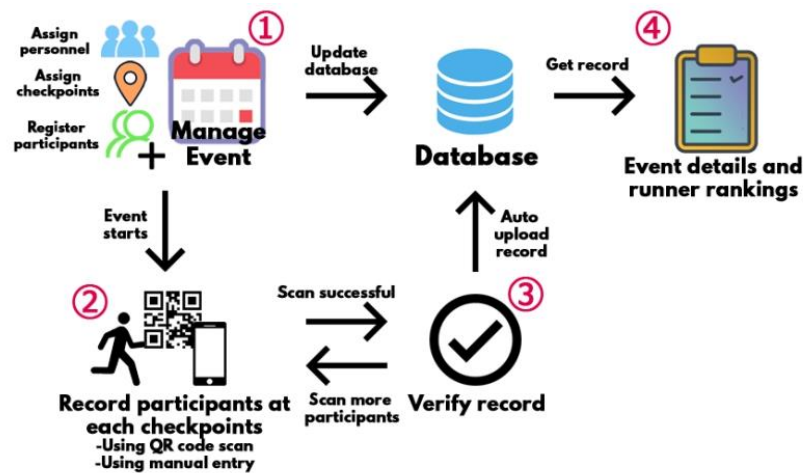


Figure 1.1.1: Figure Describing Process Flow of the Cross Country Management System

The process flow for the cross country event organized using the developed system is shown as above. All users and event organizer will be able to create their own cross country event management system using both mobile and web application. The creation process involves obtaining basic event information, assigning the assisting personnel, and adding participant categories with their required checkpoints. Once the event has been created, the event organizer and event personnel will be able to start registering participant for the event using the web application. The registration process has the functionality to generate the necessary QR Code for scanning later. Other than that, event organizer will have access to update, delete, start and stop event.

The event can only be started by the event organizer using the mobile application. Once the start event button has been clicked, the event is started. Event organizers and personnel will be able to start scanning the QR code of the passing participants at any selected checkpoints. Manual entry method is also accepted providing the correct information has been filled during recording. All records are saved automatically into the database, and will generate the finalized result and ranking once the event organizer ends the event. The completed event will not be editable anymore and all users will have access to view the event results in the event page.

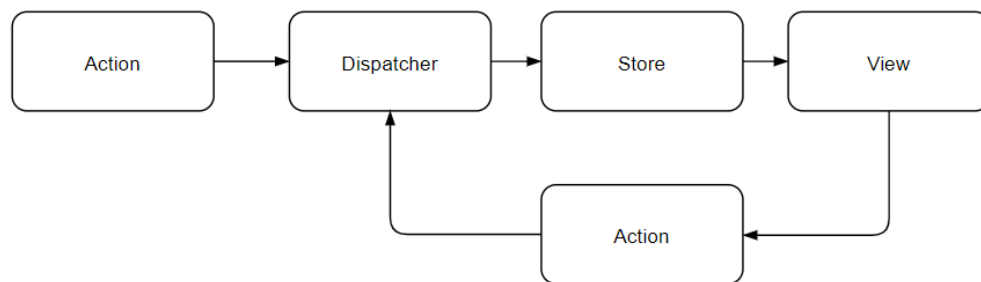


Figure 1.2.2: Diagram of Flux Architecture

The diagram above shows the flux architecture that has been chosen to develop the system. Flux architecture is chosen as the system to be developed will be utilizing React Js and React Native framework for the web and mobile application respectively. Flux architecture favours the unidirectional data flow of the react framework, with emphasis on the dispatcher being the central controller for updating various stores in the view (facebook.github, n.d.). The View in React framework is consisting of multiple components, and all the components are handled by one or many dispatchers. Each dispatcher will register the actions called and modify the appropriate data in store, which will be reflected in view.

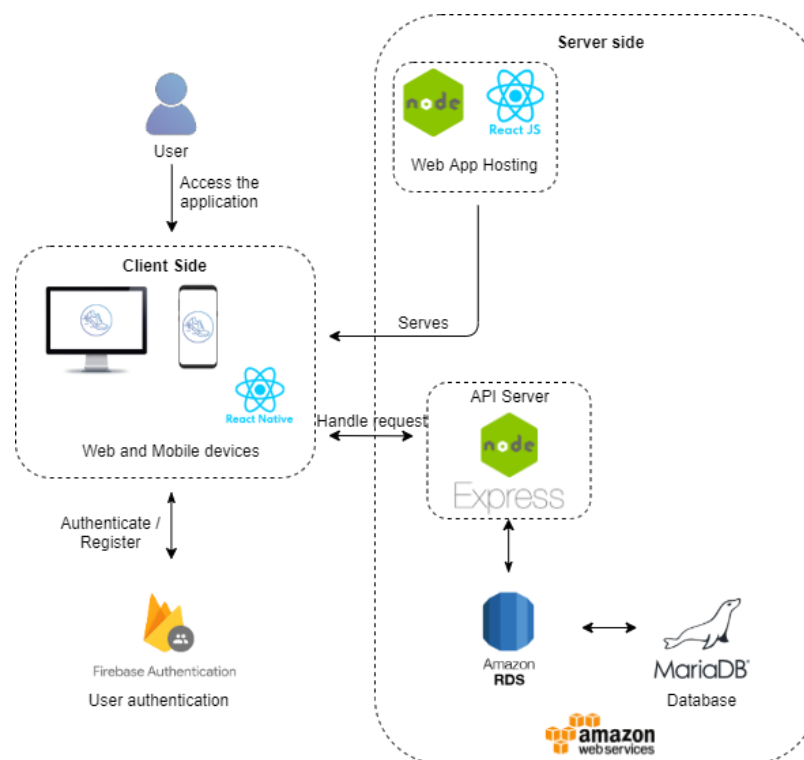


Figure 1.3.3: Diagram of Client-Server Architecture as Top View of the System

At a top view, the overall system behaves similarly to a client-server architecture. All client-side operations are performed on the user's devices. According to the actions

dispatched, the system will perform communication with the server side in order to retrieve the required information. Express is used to create the API gateway for this system, which will be used as a communication medium between the client-side with the database. The database chosen for this project is the MariaDB database, hosted in the cloud using Amazon Web Service (AWS)'s Relational Database Service (RDS). The web application is currently hosted locally using Node, therefore there may be limitation to the access of the website. Firebase authentication service is also utilized in this system in order to facilitate user sign in using regular email and Gmail.

1.6 Final Approach

1.6.1 Rapid Application Development (RAD)

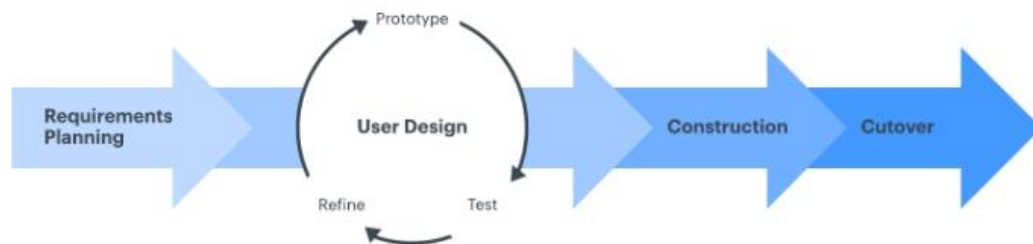


Figure 1.4.1.1: RAD Flow Diagram (LucidChart.com, n.d.)

The development methodology used for the Cross Country Event Management System is the Rapid Application Development (RAD). RAD is chosen due to its iterative development nature where the system's feasibility continues to improve on each prototyping iteration. Division of project into multiple segments or prototypes will facilitate discussion with stakeholders on the iteration's available features, therefore increasing confidence of product launch. RAD has very good time management as the methodology emphasizes on delivering a product with the adequate working set of requirements in time (Geambasu et al., 2011). RAD will bring advantage in system complexity as scopes are usually well defined, user involvement is active, and test coverages are usually extended to full features in each prototype.

RAD has different phases, each with unique responsibilities. Project requirements gathering and analysis are done during the planning phase to create a sufficient initial fundamental knowledge of the system to be built. From planning and analysis, the prominent problems of current cross country events are identified to be the inefficiency and ineffectiveness of the manual logging method.

On each prototyping cycle or user design cycle, multiple features are developed and are reviewed by the stakeholders to decide how to proceed for the next iteration. As most software projects are dynamic, it is required during the prototyping cycle to analyse the new requirements, propose a design and implement them. RAD's prototyping process is considered complete when the prototype finally satisfies the stakeholders, where it will move on to construction and cutover or testing to produce the final product and prepare for launch.

1.7 Project scope

1.7.1 Target User

The target users for this management system are the organizers and event personnel from secondary school who wish to manage their cross country digitally. The target users can be from organizers who wish to employ QR Code scanning using their own mobile device as a method of recording passing participants at each checkpoint.

1.7.2 Modules Covered

The section below shall describe the module implemented in the system and the responding functionality provided from said modules. As the system is available as a web application and mobile application, most modules will share the similar functionalities between both platforms. The modules with the same functionalities under two platforms at the: event management, account management and search. Participants management and record participants modules will only be implemented in web application and mobile application respectively. Diagrams depicting the module covered for web and mobile application will be included in this section.

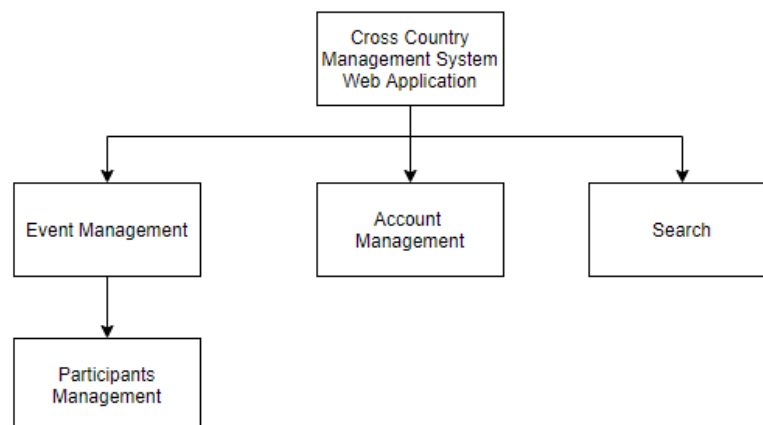


Figure 1.5.2.1: Modules for Cross Country Management System's Web Application

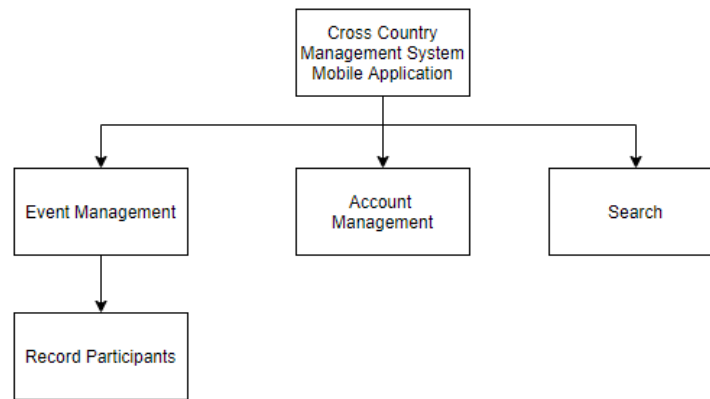


Figure 1.6.2.2: Modules for Cross Country Management System's Mobile Application

1.7.2.1 Event Management Module

The event management module is responsible for event related operations. All users of the system are allowed to create a new cross country event under this module, and the event will be recorded into the database once all necessary information of the events have been filled. The modules should recognize the position of the user within the event and provide the authorized functionalities. The main organizer of the event should have all the functionalities for an event, which would be update event information, the events' groups and the events' assisting personnel, delete event, start event and end event. The module would recognize the status of the event and display the information necessary, which affects the functionalities and the rendering of the result table.

1.7.2.2 Participants Management Module

The participants management module is implemented under the event management module of web application, and will be used to register new participants to the events. Registration of participants will require the registrar to enter the required field such as the participant's sports house and event group. A successful registration of the participants will record the participants in the database and generate the appropriate QR code of the participants. Participant listing has also been implemented under this module to display and manage the participants of this event. The module will only be accessible to the event organizer.

1.7.2.3 Record Participants Module

The record participants module is implemented under the event management module of the mobile application, and will be used to record the time of participants at a checkpoint during the event. The module facilitates QR scanning and manual data entry for recording the participants. The module however, will only be accessible to the event organizer and event personnel during an active event.

1.7.2.4 Account Management Module

The users of the system will be able to modify their account information under this module. The information that can be changed are the display name and password. This module is also responsible for keeping user information in the AWS database and the firebase authentication database.

1.7.2.5 Search Module

Search module have been implemented in both web and mobile applications to facilitate the searching of existing events.

1.7.3 Modules Not Covered

Certain additional functionality that may serve benefits to the management system are not implemented. The upload and download event data features can be useful for quickly populating a new event with the necessary information, such as the participants list and recording list. However, this feature was not implemented due to time constraint and the inconvenience stemming from maintaining the data integrity during upload. A bind Gmail option will not be made available in the profile page due to the nature of firebase authentication of automatically binding an email to the existing Gmail if the user registered under the same Gmail domain and clicked the login via Gmail option. Record participants initially decided in the Final Year Project 1 document were to be implemented in the mobile application for the registration of participants via QR scanning, however this functionality have been moved to web application to ease the download and printing of the generated runner QR code. In addition to that, score generation for sport house by student ranking as defined in the initial requirements specification were omitted due to unsure calculation of the score methods and unsuitability as a generally employed system.

CHAPTER 2

Literature Review

2.1 Software Development Methodology

2.1.1 Agile Methodology

Agile software development is an iterative approach to software development. This approach establishes high involvement and communication between team, developers and the stakeholders to create the best solution. As an iterative approach, agile development splits a project into multiple tasks and iterations to be completed which are called 'sprints'. Those sprints are usually expected to be completed within 2-week periods (Laubheimer,2017). Sprint's task may vary differently but is highly focused on the implementation of a specific function. Differentiating tasks creates an emphasis on following strict schedule and priority, hence each sprint is expected to be completed on time before proceeding into the next iteration. In agile methodology, stakeholder satisfaction level of the prototype or product is frequently measured and new requirements or changes may surface. Using agile methodology, estimation of project completion time is difficult and often inaccurate as projects need to meet both project objectives and stakeholders' approval. Unlike most methodology, Agile highlights the development of User Experience (UX) design as compliance to a holistic development.

2.1.2 Traditional Approach

Traditional approach such as the Waterfall model leans toward a predictive approach as it is heavily dependent on careful planning from the beginning of the development cycle to elicit the correct requirements (Stoica, Mircea and Ghilic-micu, 2013). Beyond the planning phase, interaction with stakeholders is very low, hence the likelihood of changing requirements during development stages is very low and is often only considered when stumped with flawed architecture. The well-defined structure of Waterfall approach allows predictable outcomes from different stages, hence tasks and members allocation are easily manageable. Conversely, quality control is difficult and time consuming due to rigid practice and timelines. Stakeholders will only be presented with the final product at project completion.

2.1.3 Differences Between Rapid Application Development (RAD), Agile and Traditional Approach.

Due to the iterative nature of RAD and Agile, user involvement during the project development is much higher than the traditional approach of the Waterfall model. As a result, there is a difference in adaptive strength to comply with the dynamic business requirement changes to produce a fitting end product. Inability of the traditional approach to manage unprecedented changes can result in high restructuring costs as there may be difficulty in altering the rigid schedule established. Documentation quality is generally better in iterative approach as code comments are written more often on each prototype to facilitate future visit.

Paraphrasing from Stoica, Mircea and Ghilic-micu (2013), Agile has higher degree of autonomy as it is expected of project members to adapt with the changes correctly, whereas RAD may have higher focus on alignment to ensure the project is moving toward how it is defined in the written objectives. There is a strong need in communication with stakeholders in Agile compared to RAD, hence it is expected that meetings and change of requirement happens more frequently. Due to the nature of high stakeholder involvement, the complexity of the project will eventually go beyond the initial requirements due to expansion of scope from each 'feelable' prototype (Difference Between Prototype Model and RAD Model, 2019).

Despite RAD emphasis on time frame delivery, the schedule of delivery is generally looser as Agile 'Sprints' are delivered more frequently (Idesis, 2019). Quality of product may differ as RAD high emphasis on speed delivery seeks to complete a software project with sufficient set of requirements whereas Agile has weaker control in team due to priority in achieving high customer satisfaction. Due to expansion of scope from Agile method, there is a substantial increase in risk when implementing new features that are incompatible with existing structures and interdependencies. On the other hand, RAD defined a clearer scope from planning phases that can serve as guidelines to prevent implementing unmanageable changes unless necessary. It also should be noted that Agile 'sprints' will often deliver individual features that concentrate on unit test and feature test but tend to miss information from system test.

2.2 Native and Hybrid Mobile Application

2.2.1 React Native

The React Native framework is an open source project by Facebook first released in 2015. React Native aims to “Learn once, write everywhere”, allowing developers to bridge the gap between iOS and Android using a single language. Being able to cross platform, React Native can be cost effective for companies to hire developers to build the application. React Native uses JavaScript programming language to write application code. As JavaScript are usually not compile-able by the native platform themselves, React Native utilizes the JavaScriptCore, which would provide both iOS and android platform direct access to the JavaScript engine that translates React Native codes into a runnable mobile application.

Given React Native is written in JavaScript, many other libraries can be used to smoothen the application. For example, Redux state management has proven to be effective in handling data sharing across screens and is able to overcome performance issues such as from excessive rendering (Peal,2018).

2.2.2 Differences Between Native and Hybrid Mobile Application

Table 2.1.2.1: Differences in Native and Hybrid Mobile Application Based On Criteria

Native Mobile Application	Criteria	Hybrid Mobile Application
Android or iOS	Platform	Both Android and iOS
Android Studio, Swift	Tools	React Native
Short if application is intended for one platform	Production cost and time	Longer as it needs to cater to both platform
Fairly matured	Maturity	Fairly new
Full support/documentation on respective platform's features	Support	Less support/documentation available online
Easy to implement as the framework is intended for the platform itself.	Hardware capability	Need to ensure the correct access / data retrieval / coding to different platform hardware
Good	Performance	Mediocre
Smaller	Application size	Bigger

Hybrid application encompassing both Android and iOS platforms is beneficial in the essence of saving time in writing code, as deployability can be a critical strength in marketing a product (Mohammadi Kho'i and Jahid, 2016). For small businesses, developing a hybrid application can encompass all the needs as the cost and time can be cheaper than developing 2 native applications, while also being able to provide all the basic functions required. However, if cost and timing is not a factor, native application will provide a better performance and easier maintenance workflow. Citing from Peal (2018), AirBnB eventually switched from React Native hybrid application back to native application due to issues extending from difficulty in production to cumbersome debugging. Given React Native uses the Javascript language, its unsafe typing can increase likelihood of error during refactoring especially when the prop has a common name and is essential to be processed by multiple components.

Maturity can be an important deciding factor for choosing the correct framework. Given that React Native framework is fairly new compared to its native counterpart, there may be limited documentation regarding the best approach into handling certain errors (Axelsson and Carlström, 2016). Community and resources availability is lesser for React Native than native applications which can stretch development time to resolve trivial issues and make production estimation unpredictable. As an application becomes more complex, hybrid applications such as React Native may fail to provide sufficient support to implement the correct plugin or library for the features to be implemented. This dilemma enhances when the application is required to utilize hardware across platform as different platforms hardware function calls can be different in naming, returned value or functionality. There can be inconsistency in bridging functionalities between the platforms, leading to unexpected bugs. Native applications designed for specific platforms will inevitably perform better and have smaller application size as the development works closer with the technology itself.

Based on the information found, the React Native has been chosen for hybrid mobile application. Despite the weaknesses, React Native will enable easy development for both platforms at once. The development for iOS platform can be begun easily after the completion of android platform. Maturity of the community and ease of finding a solution may not be concerned as the system to be developed is not expected to be too complex.

2.3 Database

2.3.1 Relational Database vs Non-Relational Database

Table 2.2.1.1: Table of Differences Between Relational Database and Non-Relational Database

Relational Database	Criteria	Non-relational database
SQL (Structured Query Language)	Type	NoSQL (Not Only SQL)
MySQL, MS-SQL	Example	Firebase realtime, MongoDB
ACID (Atomic, Consistent, Isolated, Durable)	Properties	BASE (Basic Availability, Soft-state, Eventual consistency)
Must conform to column rules	Schema and Flexibility	Follows key-value pair
Increasingly difficult and expensive to scale	Scalability	Easy and cheaper to scale
Vertical	Scaling orientation	Horizontal
Excellent	Complex query handling	Weak
Selection, insertion, update	Data manipulation	Object-oriented retrieval

Efficiency of organizing and managing database is based on the type of database implemented. The characteristics and differences between relational database such as SQL and non-relational database such as NoSQL can be described as the table shown above. Between SQL and NoSQL, SQL has a need for strong adherence to fixed schematic representation, indicating that every information must follow a set of rules such as data type and length defined in the table (Pore and Pawar, 2015). NoSQL overcome the limitation from fixed representation by implementing a key-value pair concept, hence NoSQL is more dynamic compared to SQL as it allows for flexible information insertion. However, SQL conforming to columns rules indicates that there will be strong data integrity within the table, it is less likely to have an orphan data. Given that SQL follows a table format, a vertical scaling is implied. Vertical scalability

becomes difficult to manage as retaining a single server will require better processing power to handle the load of processing a large volume of data. Alternative to processing power is to split database into multiple server. However, SQL needs to maintain data integrity when splitting server, hence additional steps need to be taken to ensure a correct implementation. NoSQL follows a horizontal scaling that is more cost-effective in handling with and expanding to multiple servers. NoSQL is less suitable for handling complex query compared to SQL as SQL has a stronger query language and interface. Structuring a NoSQL may be difficult as the complexity of the data increases, which may lead to requiring additional code to restructure and combine multiple retrieval in order to generate the desired single ling result. On contrary, SQL may allow easy structuring of result within a single query and can insert multiple data according to select statement directly (Sudiartha et al., 2020).

2.3.2 Selection of Suitable NoSQL Provider

Table 2.3.2.1: Table of Differences Between Firebase, MongoDB and RethinkDB
(Khedkar et al., 2017)

Database	Firebase	MongoDB	RethinkDB
Data model	JSON	JSON	BJSON
Operating system	Cloud hosted	Linux,OS X, Solaris, Windows	Linux, Windows, OS X
Cloud platform	Google	AWS, dotCloud, Compose.io	Many cloud platform
Query complexity	Impossible to query field with some properties	Querying may not support join	Good

According to Khedkar et al. (2017), Firebase, MongoDB and RethinkDB all have unique and distinctive advantages that cater to different needs. All 3 databases can be hosted in the cloud, however firebase console platform located in the cloud too would limit the access to export user data unless permitted. Between all 3 databases, RethinkDB is best at handling complex queries whereas Firebase capability at handling complex queries is the worst and is dependent on how the application database is structured.

Unique features are provided by each database. RethinkDB excels at maintaining table availability as having three or more servers will enable failover protection that pushes any secondary node to represent the unavailable primary node. Failover prevention can dramatically reduce the risk of server downtime. MongoDB consolidating data will enable a better scalability and performance.

2.3.3 Selection of Suitable SQL Database

Table 2.4.3.1: Table of Differences Between MariaDB, Oracle and MySQL

Database	MariaDB	Oracle	Microsoft SQL
Distributed Partition	Yes	Partitioning	Yes
Backup and restore	MariaDB backup	Yes	Yes
Point-in-time rollback	MariaDB flashback	Oracle Flashback	No
Stored procedure	PL/PSM	PL/SQL	Transact-SQL
Cloud host	Amazon Relational Database Service	Oracle	Microsoft

According to (MariaDB, 2019), distributed partitioning can be done to spread a database into multiple constituent parts, therefore increasing the throughput and availability of the database. Between the three databases, Microsoft SQL is unable to perform distributed partitioning. All three databases are capable of backup, however Oracle and MariaDB are able to undo previous transactions through rollback instead of directly restoring an older version of the database. However, only the stored procedure of Oracle is capable of performing dynamic SQL query and running transactions within a stored procedure. Microsoft SQL is able to run similar queries but will require to create a prepared execution instead.

Using the information above, the final database chosen be the MariaDB relational database. MariaDB's point-in-time rollback may be more convenient for the development of a medium-small scaled project. Scalability of the project is not a concern due to the size of userbase. Data integrity may be important for the management system to properly retrieve a whole set of data without any orphan data.

The complex query handling may serve beneficial when trying to directly update the result of any events.

2.4 Existing Systems

Studies were conducted on existing systems to understand the flow of operation encompassed within a sports event organized. A single system may consist of multiple platforms and applications to complete a full event.

2.4.1 Active Endurance

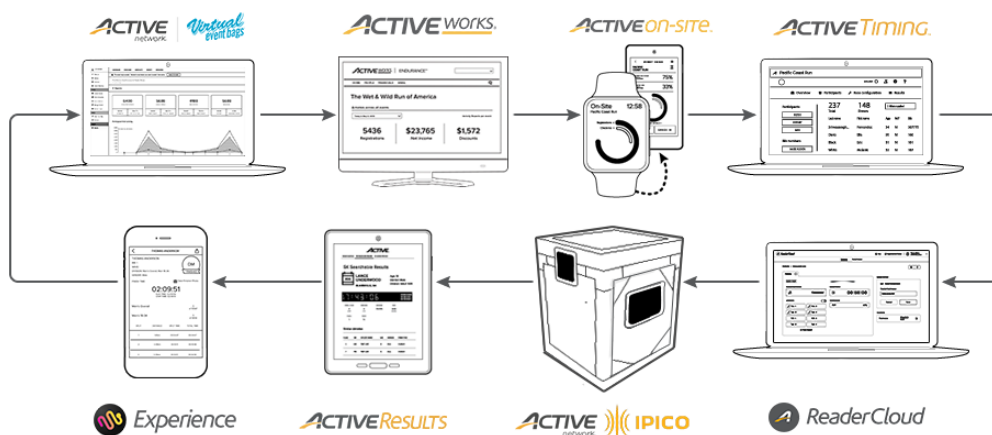
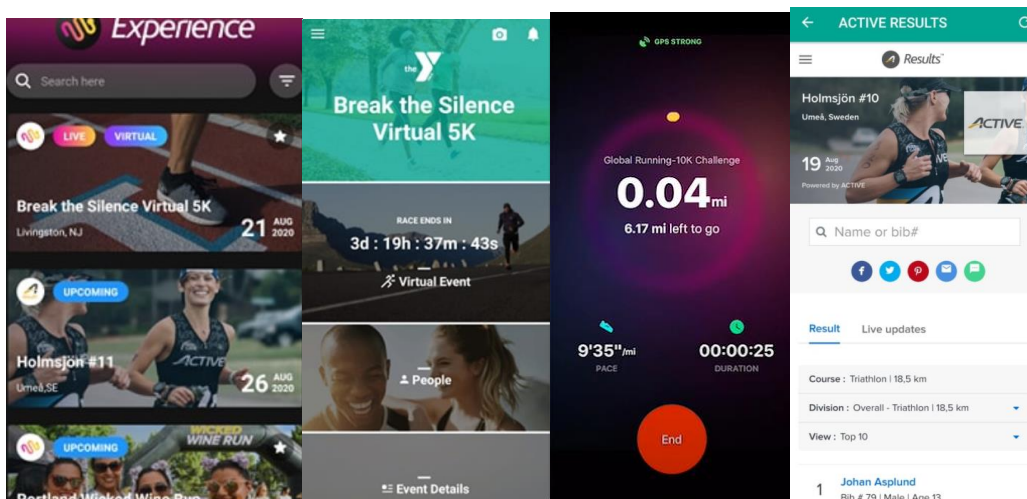


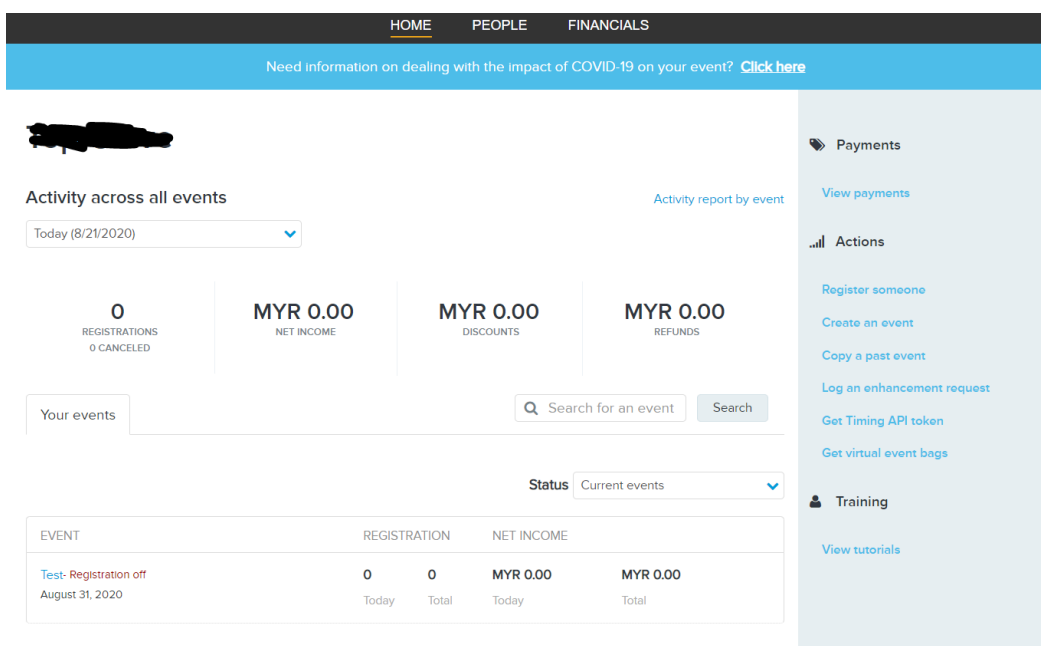
Figure 2.1.1.1: Path Flow of Active Applications For A Complete Sports Event. (Activeendurance.com, n.d.)

Active Network is a technology platform provider that helps organizations to organize and manage their sport events. Different Active applications are catered for different purposes which separates the workflow between management and participants. Active Network also provides tools to analyse the event based on data gathered, creating reports of participant performance and event budget auditing.



Figures 2.2.1.2: Path Flow For Active Endurance Experience Mobile Application

Client operations such as viewing events and registration are all done using the Active Endurance Experience mobile application. Potential participants can navigate through the mobile application by selecting upcoming events listed in the main screen or search an event via search bar. Upon entering the event screen, users have functionalities to check the event details, joined participants, photos and latest event updates. Upcoming events and live events have registration options available. Once registered, the type of event will determine how the event would be completed. A virtual running event would start tracking user activities using the device’s sensors once initiated. On-site running events will be managed by Active Timing software instead. Everybody can view the results of the event via the mobile application and Active Result page.



Welcome Li Tien | Help | Settings | Choo

HOME PEOPLE FINANCIALS

Test

August 12, 2020

Dashboard Setup Participants Invitations Teams Tools Reports Express registrations Virtual event

Before you can start collecting registrations, you must finish your event setup and activate registration.

[Continue setup](#)

Event overview

Event setup

Test

8/31/2020 - 8/31/2020

Klcc

Lot No. 241, Level 2, Suria KLCC, Kuala Lumpur City Centre.


Jalan Ampang

50450

Malaysia

ACTIVE.com listing

ACTIVE.com listing



Here's your unique QR code, which links to your ACTIVE.com listing. Click the image to download or view a larger version.

Event is not published

A link to the event listing will be available after event setup is complete.

Event setup

- Event summary
- Event details
- Restrictions and settings
- Registration categories
- Self-registration team setup
- Self-edit setup
- Form questions
- Giving
- Additional purchases
- Waivers
- Emails sent out
- ACTIVE.com listing
- Activation

Figures 2.3.1.3: Screenshots of ActiveWorks Web Application Home Page and Event Page

The main event management application is ActiveWorks. Using ActiveWorks, event managers can create repositories for any upcoming sports event hosted by their organization. During event creation, all event details must be specified, such as selecting the event location, setting up event restrictions and event group divisions, specifying method of payment and etc. Participant registration can be done here or automated using other software. Additional tools are provided to analyse the financial part and general information of the event.

HOME PEOPLE FINANCIALS

People

1 Result

NAME	EMAIL ADDRESS	PHONE	CITY	STATE	COUNTRY	GENDER
Li Tien Khew	dougk24601@gmail.com	60113184	Klang		Malaysia	

Figure 2.4.1.4: Screenshot of ActiveWorkss People Section

The event management application allows the organization to add personnel to the database. All personnel have access to the events organized and are able to update the

events. During event tasks and checkpoints are not assigned in ActiveWorks management system.

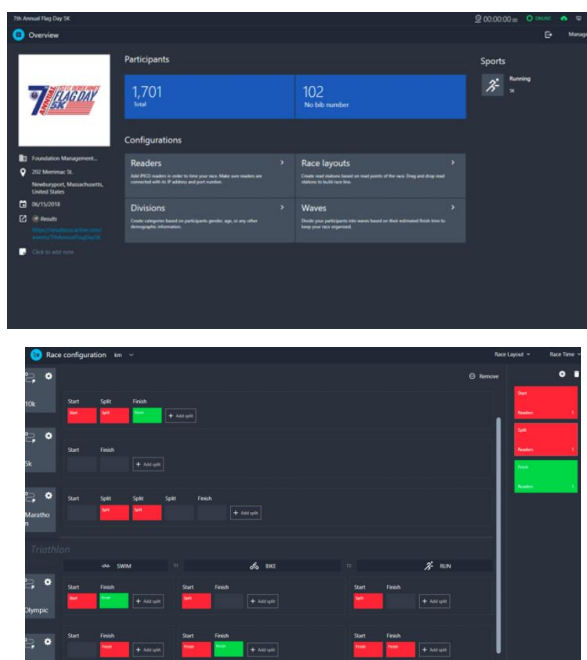
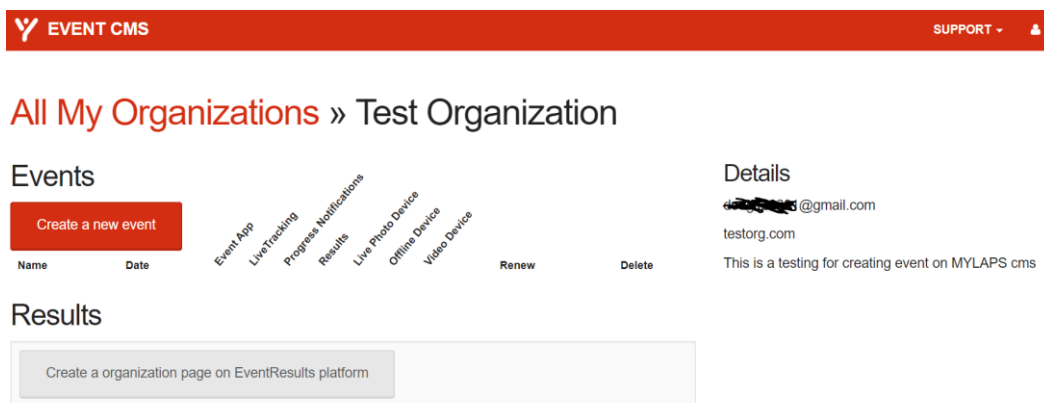


Figure 2.5.1.5: Screenshot of Active Endurance Timing Overview Screen

Active Endurance Timing software is responsible for all operations during the event. Event personnel are required to configure the software to integrate with their race timing equipment. Event personnel will determine the start time for different divisions and sync the checkpoints within the race tracks with the correct equipment. Every successful scan by the timing equipment will update the software with the corresponding participant details. Event error reporting and handling are done using this software too.

2.4.2 MyLaps

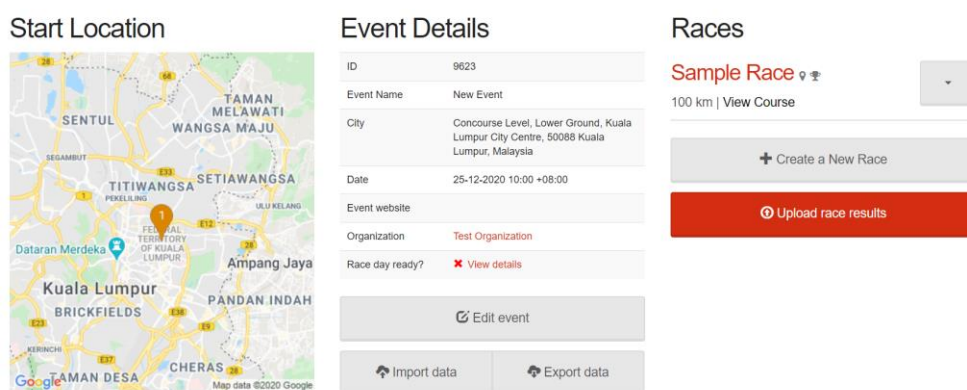
MyLaps is a service provider for managing sports events ranging from running, cycling, triathlon, skating and motorsports. The management system used is the MyLaps Content Management system (CMS), and the participants interface used is the two different mobile applications: EventApp and Sporthive Live.



The screenshot shows the MyLaps CMS interface. At the top, there is a red header with the 'EVENT CMS' logo on the left and 'SUPPORT' with a user icon on the right. Below the header, the main heading reads 'All My Organizations » Test Organization'. The page is divided into three main sections: 'Events', 'Results', and 'Details'. The 'Events' section features a red 'Create a new event' button and a list of event types: Event App, Live Tracking, Progress Notifications, Results, Live Photo Device, Online Device, and Video Device. The 'Results' section has a button to 'Create a organization page on EventResults platform'. The 'Details' section shows user information: 'testorg@gmail.com', 'testorg.com', and a note: 'This is a testing for creating event on MYLAPS cms'.

Figure 2.6.2.1: Screenshot of MyLaps CMS Main Page

Using the MyLaps CMS, organizations can set up any new upcoming sport events. Organizers can choose to select which mobile application to promote the event on. After selecting the main application, organizers would be able to customize the landing page of the event when accessed through said application. Organizers are required to enter all the necessary event information detailing the event's proceedings. Live tracking option is available to select if the event is able to provide the required devices. Event organizers can include a redirect link for event registration if it is set up externally. Once everything is set, the event will be published online to the mobile application and interested participants may register for the event.

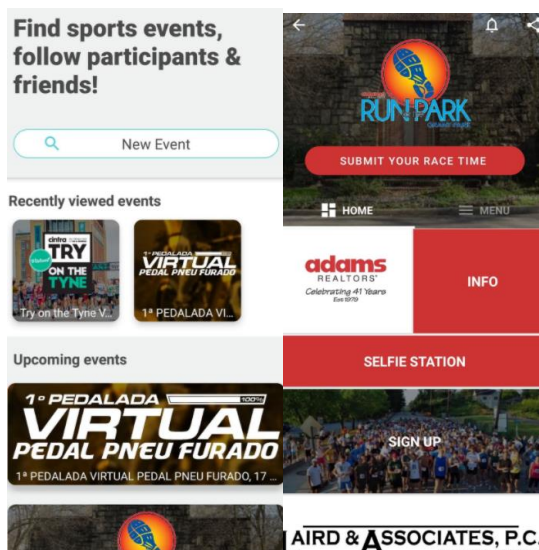


The screenshot displays the 'Further Event Configuration' page, divided into three columns: 'Start Location', 'Event Details', and 'Races'. The 'Start Location' column shows a Google Map of Kuala Lumpur with a red pin at the 'Concourse Level, Lower Ground'. The 'Event Details' column contains a form with the following information: ID: 9623, Event Name: New Event, City: Concourse Level, Lower Ground, Kuala Lumpur City Centre, 50088 Kuala Lumpur, Malaysia, Date: 25-12-2020 10:00 +08:00, Event website: (empty), Organization: Test Organization, Race day ready?: ✘ View details. Below the form are buttons for 'Edit event', 'Import data', and 'Export data'. The 'Races' column shows a dropdown menu set to 'Sample Race', a distance of '100 km | View Course', a '+ Create a New Race' button, and a red 'Upload race results' button.

Figures 2.7.2.2: Screenshot of MyLaps CMS Further Event Configuration

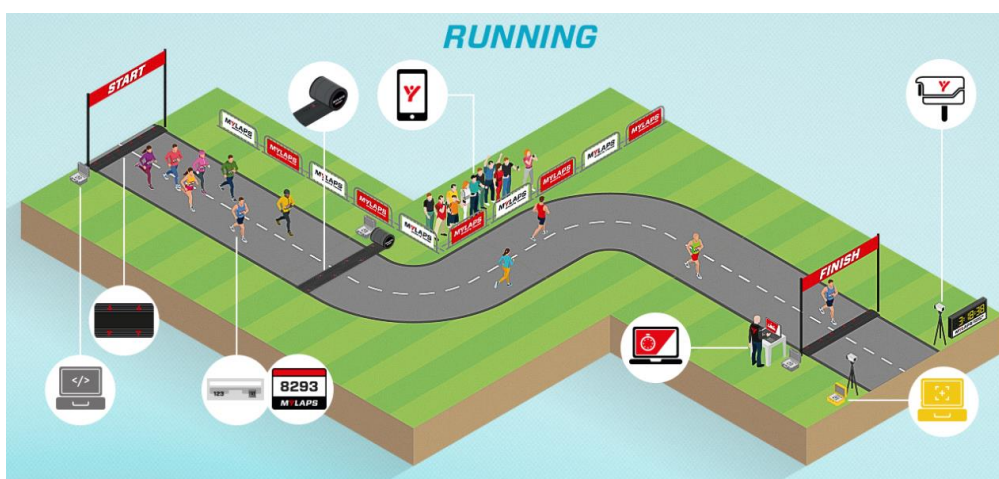
Once the event's base information has been set up, the organization will be informed on more required details that need to be uploaded. The required documents may include the warrant for establishing said running event at the location and etc. Organizers must upload the participant list in csv and pinpoint the course track in xml

file. The set up needs to be completed 3 days prior to the event day. Once the event is ‘race day ready’, organizers can start the event and begin event day operations.



Figures 2.8.2.3: Screenshots of Sporthive Live Mobile Application

Users will be able to view all upcoming events at the main screen of Sporthive Live. For past events, users will be required to manually search for the event by the search bar. Clicking on the event thumbnail will redirect users into the event’s dashboard, consisting all event details. Registration button will redirect user to a designated page for registration.





Figures 2.9.2.4: Illustration of Event Day Operations and Sample *Bibtag*
(mylaps.com)

On-site event utilizes MyLaps's unique equipment to function. Every participants will be assigned with their respective *bibtag* containing their registration information. *Bibtag* emits UHF signal that will be picked up by the detection mat, thus registering that the participant has passed the checkpoint. MyLaps's timing software will be used to collect all the participants' timing and uploaded to the database. Timing software works along with other MyLaps's software, allowing for live tracking functionality and finalization of result.

A MyLaps virtual running event does not require any additional equipment. Participants will be able to use their device to track their activities using the device's in-built hardware such as the phone's accelerometer. Organizers can choose to track participants' activities using Sporthive Live application's in-built tool, or they may opt for specific data/file upload. Finalized results can be accessed from the mobile application's result section or from MyLaps's web application.

2.4.3 JomRun

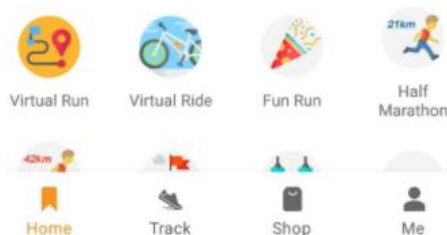
Running Events Malaysia

Featured

See More >



18 - 18
SEP OCT
**Kitty Meow Virtual
Half Marathon**
Anywhere



Registration Closed

**Ultimate Penang Runner Challenge
2020**

05 Jul 2020 (Sunday) | 06:30 am
(GMT +08:00)

Teluk Bahang Dam, Penang

From only RM51.65!

Winner Prizes for 12KM



Tritiq Sdn Bhd

ABOUT

EVENT POSTPONED

Let's challenge yourself with Ultimate Penang Runner Challenge 2020!

Figure 2.10.3.1: Screenshot of JomRun Mobile Application

Figures 2.4.3.1 show the interface of JomRun mobile application. Users are able to filter upcoming events by category. JomRun application has many advertisements to promote major upcoming events and has a shop section to purchase merchandises or sports equipment. Tracking functionality is available during the virtual run event. Using the 'Track' feature from the homepage will start tracking user running activities but the result cannot be used for event upload. Entering the landing screen of any upcoming events will preview all event information in a scrollview. Clicking on registration usually redirects users to an external link for payment. Once registration is completed, the receipt will be added into the user profile. Depending on the event, receipt details stored and functionality provided will vary, thus users need to refer to the event organizer's instructions.

2.4.4 Existing System Comparison

Table 2.5.4.1: Table of Differences Between All The Existing Systems Studied

Criterion	Active Endurance	MyLaps	JomRun
Event creation	Only premium users can create events	All users can create events.	No event creation.
Event management	Event organizer only.	Event organizer only	Managed by application owner.
Participants registration method	Register participants internally.	Register participants internally.	Redirects participant to external registration platform.
Supports on-site event operation	Supported	Supported	Not supported
Method of on-site tracking	Equipment signal detection	Equipment signal detection, GPS tracking	N/A
Equipment of tracking	Any matching timing equipment for on-site events. Tag/Chips, detection mat, other dedicated detection equipment.	Strictly MyLaps's tracking equipment for on-site events. Bib with tag, BibTag decoder, detection mat, network kit, Pro Chip, ProChip smart decoder, timing and scoring software	Mobile device accelerometer or data upload for virtual run.
Cost of on-site equipment	Quotation from company, varies by event size.	Quotation from company, varies by event size.	N/A

Target Market	Professional athlete, casual runner	Professional athlete, casual runner	Casual runner
Geographic segmentation	United States	United States	Malaysia

A common characteristic shared among the three systems is the separation of participant platforms from the management platform. Most of the participant platforms are provided via mobile applications, which deliver operations such as view and search event details. The mobile applications are all free to install, but incur additional cost for the registration of chosen events.

Between the three, only MyLaps provides a free to use management platform. Active Endurance requires payment to complete the event setup process while JomRun fails to provide any event management system. The participants registration for both MyLaps and Active Endurance are done using their own system, while JomRun will redirect users to the organizer's domain for contact.

MyLaps and Active Endurance facilitate on-site event operation, while JomRun is primarily used for virtual run events. Both MyLaps and Active Endurance seek to run events for professional athletes, therefore dedicated tracking equipment has to be used. Active Endurance systems will be able to accept data from their list of permitted timing equipment, while MyLaps will strictly uses tracking equipment of their own brand. Event runners will be required to equip the tracking chips that are interpretable by the decoders. Therefore, to properly complete any event, results from dedicated tracking equipment will have to be used.

Two hardware functionalities ventured is the QR scanning capability and accelerometer. QR code scanning functionality is only implemented by MyLaps and Active Endurance for participant attendance verification during event day. All three systems utilize the device's accelerometer to track users' running activity for virtual run. The geographic segmentation of Active Endurance, MyLaps and JomRun are United States, United States and Malaysia respectively.

As the system to be developed will be used by Malaysian secondary school, sensitivity of tracking accuracy can be more lenient than the system studied. The system developed will be employing QR scanning to record participants at any checkpoint for on-site cross country events, as adapted from MyLaps and Active

Endurance method of QR scanning for participants event day check in. The runner tag containing the QR code can be printed by the school or by the participants themselves. Event organizers and personnel will be using their own android device as equipment for recording participants, hence saving the cost for the event. All users of the system will be able to create and host events.

CHAPTER 3

Methodology and Project Management

3.1 Development Methodology

The development methodology chosen is the Rapid Application Development (RAD) methodology. RAD has 4 major phases as illustrated below:

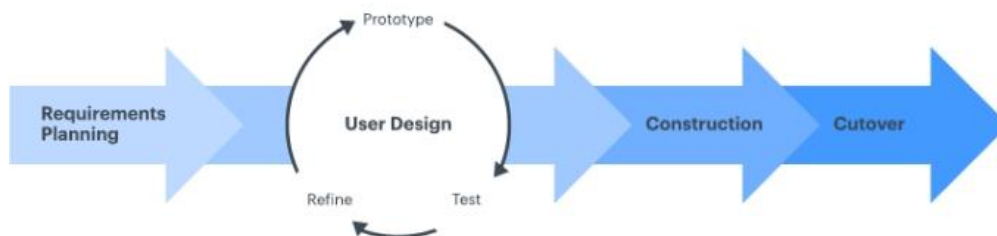


Figure 3.1.1: RAD Flow Diagram (LucidChart.com, n.d.)

For this Cross Country Event Management system, the project begins with the planning phase. Within the planning phase, the project basis needs to be established. Scope and objectives were to be specified to define the essential works, predetermined output and result of this project. Proposed solution and approach is also confirmed to determine the best software architecture, methodology and tools to be used for implementing the project. Requirements are elicited from the secondary school cross country event organizers to document the required features and functions of the project. Project works will be decomposed into many smaller, specific and more manageable tasks via work breakdown structure. Researches are done to analyse the project problems and gain understanding on the best approach for a perfect delivery. Once all the basis has been confirmed for planning, the project will move on to the user design phase.

The user design phase's objective is to create functional prototypes for the systems that achieves the requirements elicited. The user design phase generally consists of 3 processes: prototype, test and refine. Diagrams of the management system such as use case, sequence and database diagrams will be modelled and refined along the user design phase. Every prototyping process is done by confirming the system models of the prototype, then building said component or system from the model using a development tool. Each prototype consists of new features or changes implemented for the components of the defined modules.

Once the components or system are created, testing will be done on the prototype to ensure it is working as intended in requirements. The prototype must be evaluated during the testing of each iteration to verify if it is up to standard before moving onto the construction phase. Testing of the prototype may include unit testing and integration testing, but may differ depending on the new changes introduced to that prototype iteration. For this project, most testing completed will be manual testing to consider the end user's viewpoint. If the prototype is lacklustre, refining process must be done. The refine process may include tasks such as improving or repurposing the designed style, model, modules and prototype.

In the construction phase will be short as the prototype has been continuously improved in the user design phase. In this phase, emphasis will be put into completing the actual system and ensuring that it can be deployed with any issue. Integration and system testing must not fail in this phase. The cross country event management system should be guaranteed to work in an actual environment with real-life operations.

Cutover phase is done after construction phase. The system undergoes acceptance test by the stakeholders to determine if the system is in compliance with all requirements, needs, processes and design. For this course, the project will be considered as complete after the completion of acceptance test and will be presented to the responding supervisor and moderator.

Under RAD, if the system is within or beyond satisfactory, the system will be pushed for launch. A proper domain will be used to host the web application and the APK for mobile application would be built. Supports will be provided for the users to use the system and existing bugs will be corrected.

3.2 Project Plan

3.2.1 Work Breakdown Structure

Work breakdown structure is done based on Rapid Application Development Methodology (RAD).

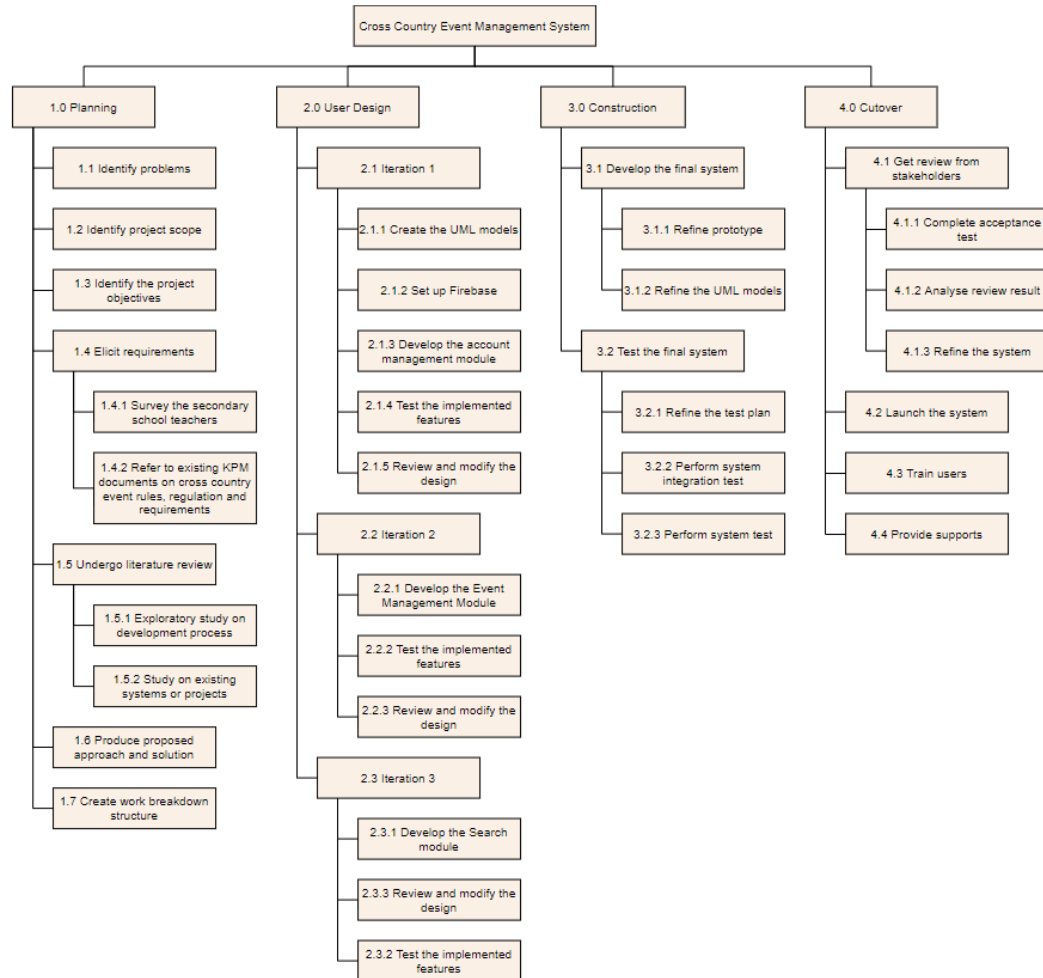
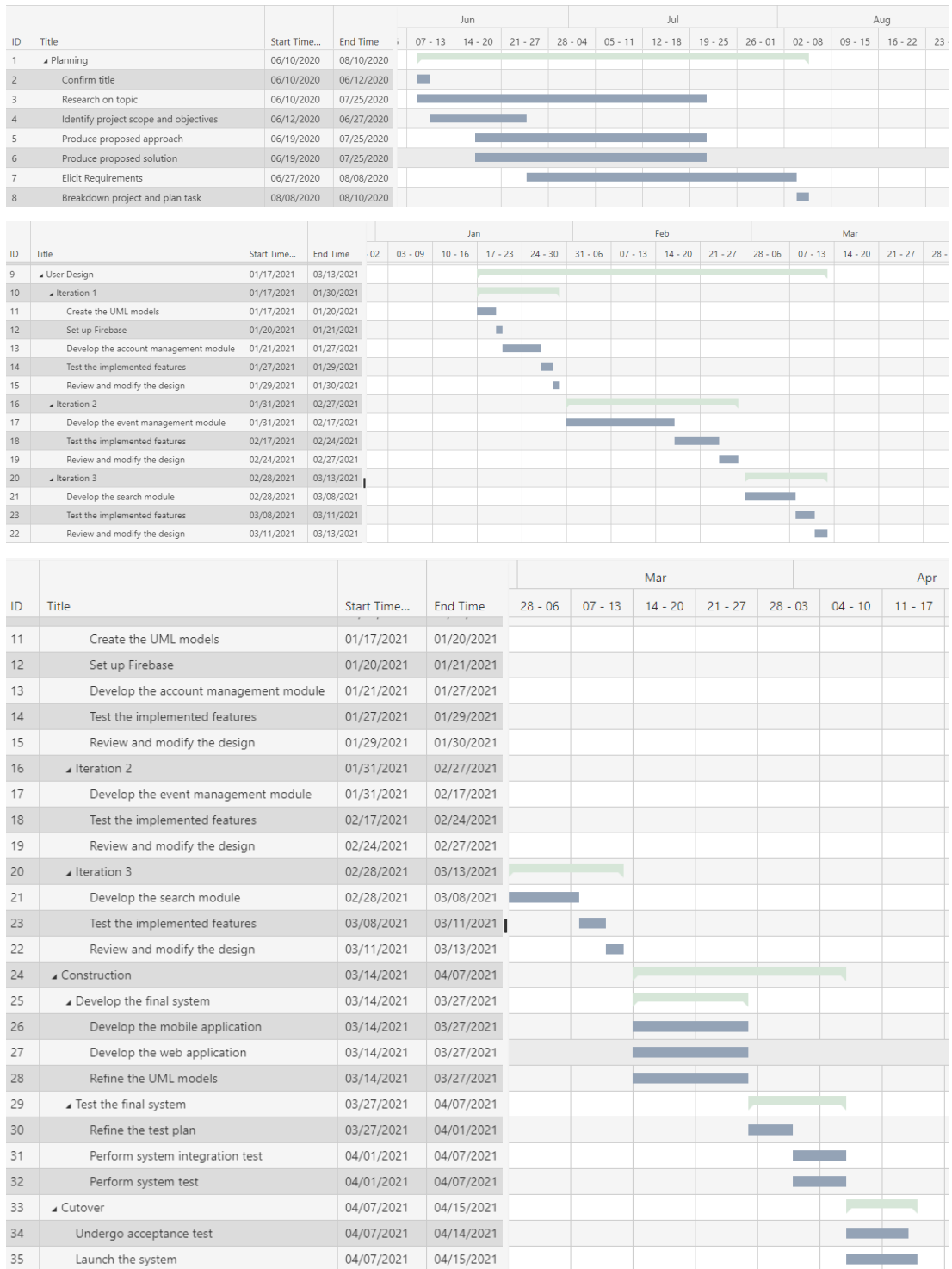


Figure 3.2.1.1: Work Breakdown Structure For Cross Country Event Management System Based On RAD Methodology

3.2.2 Gantt Chart

ID	Title	Start Time...	End Time
1	▾ Planning	06/10/2020	08/10/2020
2	Confirm title	06/10/2020	06/12/2020
3	Research on topic	06/10/2020	07/25/2020
4	Identify project scope and objectives	06/12/2020	06/27/2020
5	Produce proposed approach	06/19/2020	07/25/2020
6	Produce proposed solution	06/19/2020	07/25/2020
7	Elicit Requirements	06/27/2020	08/08/2020
8	Breakdown project and plan task	08/08/2020	08/10/2020
9	▾ User Design	01/17/2021	03/13/2021
10	▾ Iteration 1	01/17/2021	01/30/2021
11	Create the UML models	01/17/2021	01/20/2021
12	Set up Firebase	01/20/2021	01/21/2021
13	Develop the account management module	01/21/2021	01/27/2021
14	Test the implemented features	01/27/2021	01/29/2021
15	Review and modify the design	01/29/2021	01/30/2021
16	▾ Iteration 2	01/31/2021	02/27/2021
17	Develop the event management module	01/31/2021	02/17/2021
18	Test the implemented features	02/17/2021	02/24/2021
19	Review and modify the design	02/24/2021	02/27/2021
20	▾ Iteration 3	02/28/2021	03/13/2021
21	Develop the search module	02/28/2021	03/08/2021
23	Test the implemented features	03/08/2021	03/11/2021
22	Review and modify the design	03/11/2021	03/13/2021
24	▾ Construction	03/14/2021	04/07/2021
25	▾ Develop the final system	03/14/2021	03/27/2021
26	Develop the mobile application	03/14/2021	03/27/2021
27	Develop the web application	03/14/2021	03/27/2021
28	Refine the UML models	03/14/2021	03/27/2021
29	▾ Test the final system	03/27/2021	04/07/2021
30	Refine the test plan	03/27/2021	04/01/2021
31	Perform system integration test	04/01/2021	04/07/2021
32	Perform system test	04/01/2021	04/07/2021
33	▾ Cutover	04/07/2021	04/15/2021
34	Undergo acceptance test	04/07/2021	04/14/2021
35	Launch the system	04/07/2021	04/15/2021

Figure 3.3.2.1: List of Planned Tasks



Figures 3.4.2.2: Gantt Charts of The Planned Tasks

3.3 Development Tools

3.3.1 Programming Language

3.3.1.1 JavaScript

JavaScript has been used extensively when developing the application in both React Js and React Native framework. JavaScript is primarily used for creating and calling the appropriate functions for both web and mobile applications' components.

3.3.1.2 MySQL

MySQL is an abbreviation for “My”, the name of the founder, Michael’s daughter and “SQL” which stands for Structured Query Language (Husky Intelligence, n.d.). MySQL has been used for managing communication and data exchange with the MariaDB database.

3.3.2 Framework

3.3.2.1 React Js

React Js is a Javascript framework for creating web user interfaces by composing components, states and props. Most pages are structured as a single container, and within each container consist of components with different functionalities. Redux has been used alongside React to facilitate better state management between components.

3.3.2.2 React Native

JavaScript framework such as React Native is used for creating and rendering any hybrid mobile application. As this project is to be released for android mobile devices, only the android files and configuration for the React Native project are updated for changes. Similar to React Js, the project is built using components, states and props defined. Libraries developed by both React Native main developers and the community will be used to facilitate the QR scanning.

3.3.2.3 Express

Express is a Node.js framework that provides a common set of utilities which will be used for building servers (Labs, 2020). The project’s API Gateway is built on the Express server to facilitate API calls for handling requests such as data exchanges.

3.3.2.4 Bootstrap

Bootstrap is used extensively during the web application development to quickly build common components without needing to define the basic html, css and javascript file. Many components provided by bootstrap have been used in this project such as the Modal, flexbox, pagination and class stylings. The component and utilities make it easy to create a clean looking user interface. In addition to that, bootstrap can offer responsive layout for better rendering of the page when accessed from either mobile device or desktop.

3.3.3 Databases

3.3.3.1 Amazon Relational Database Service

Amazon Relational Database Service (RDS) is a cloud database solution provided by Amazon Web Service (AWS). As AWS is hosted in the cloud, the database will be available as long as it is switched on in the AWS console. As the service is provided by a known company, the database is less likely to face unintended down time. In addition, AWS provides additional tools and features such as setting up VPC and group which makes database connection management easier.

3.3.3.2 MariaDB

MariaDB is a relational database that is used for this project. This database has been chosen due to its clear table structuring and its distinct storage engines such as InnoDB that is good for querying by row and ColumnStore that is better for querying data by index. The MariaDB is hosted on Amazon's RDS.

3.3.3.3 Firebase

Firebase Authentication provided by Firebase is used in this project to partially manage user sign in and register. Although the user data will also be stored in MariaDB, Firebase is used for authenticating users for both regular email sign in and Gmail service sign in. Email address verification service is provided but will not be implemented in this project until it is ready for launch to allow easy testing during development.

3.3.4 Other Tools

3.3.4.1 Github

Github is used as the version control system for this project. All project changes are committed regularly to the repositories and versioned with tags to keep track of the versions. The project can be cloned and reinitialized for development or usage in another machine. GitHub is also used as fail safe to allow rollback in case critical and defected changes have been made to the local copy. GitHub is commonly used by the software development industry for work collaboration.

CHAPTER 4

Project Specification

4.1 System Overview

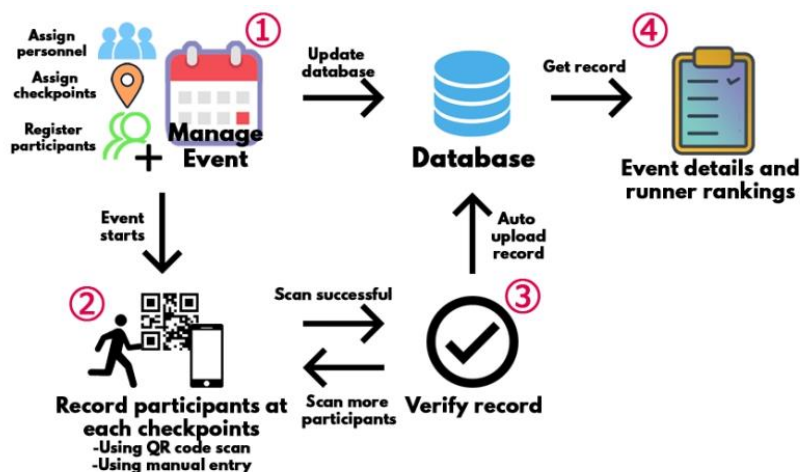


Figure 4.1.1: Figure Describing Process Flow of The Cross Country Management System

The process of a cross country event is distinguishable in 4 steps: First, set up or manage the event. Second, record participants at checkpoints. Third, verify the new record. Finally, retrieve the finalized list. Manage event concerns the event management modules of web and mobile. Event organizer will create the cross country event either using mobile or web application and input relevant information to complete the event creation process. Assigning personnel by adding users by their CCEMS ID can be done during the event creation process. Participant groups and their total required checkpoints can be added during the creation process to distinguish the categories of runners. Should the organizer make any mistake during the creation process, the event will be available for update as long as it has not been started or ended yet.

To register participants or runners for the events, the record participants module of the web application is used. The personnel or organizer would be able to achieve that by using the register participants function in the event page. After registering the participants, the QR Code will be displayed for the user and the image files can be saved for creating their own QR bibs for the event. Registered participants can be found again in the participants list, where the QR code can be viewed and

downloaded again. Update of participants is possible in the participant list as long as the event has not been started or ended yet.

The event organizer can choose when to start or end the event. During the event, the mobile application's record participant module will be used. The module should detect QR code on the runner's bib and display runner information. The personnel should verify if the runner information is correct, then the timestamp of the runner passing the checkpoint will be uploaded to the database. Personnel can choose to manually enter the data if detection fails. Event organizer can finalize and close the event using either platform to prevent further entry. Event management module should retrieve the event's participant information and produce a ranking list.

For users who are not interested in participating or managing the event, they are able to search for the existing event in the search page. The search functionality is available for both registered and unregistered users. In addition to that, registered users will be able to view their top 3 upcoming or recent events that they were assigned to assist or have created in the home page. Event list containing the user's events will also be made available in the event list page.

4.2 Requirements Discovery Method

The requirements discovery method will be elaborated and concluded in this section. Two discovery methods were used to identify the requirements for this project, which are the exploratory study and the survey research. The main requirement discovery method is the exploratory study. Multiple similar event management systems were examined to determine the workflow of their application. The importance of this method is to recognize the pattern and narrow down the core features to be implemented in this project.

The exploratory study on existing systems revealed that most well-established marathon system providers employ high-end equipment to perform precise tracking of participants at any given checkpoint. The equipment identified was the "timing chips / tags" placed on the runners, and the detection mat or machines placed at the checkpoints. Precise and quick tracking were possible using the equipment due to consistent frequency emission and detection. However, to employ such a method would require the high-end equipment to be present, which can cause financial constraint upon purchase.

The purpose of using the equipment was to scan and record participants after all. It has been noted that QR scanning was employed by MyLaps and Active Endurance for admitting the participants before the event start and that the QR scanning method can be performed using any mobile devices. Therefore, it is possible to eliminate the need for dedicated tracking equipment and utilize QR scanning for the recording of participants at a checkpoint instead. The requirement of using QR scanning for participant recording was elicited. On a similar notion, eliminating the need for dedicated equipment will also significantly reduce the equipment cost as the only equipment required is the user's personal mobile device.

From the exploratory study on the existing system, inspiration was drawn from their event creation and event navigation process. It is a requirement that the system must allow users to create an event for the system. The study has also narrowed down some fundamental information that is required such as title, data, time, location as they help provide basic impressions for a visitor of that event page. All runner category elicitation were done during the event creation process, and not any other part of the module.

Besides similar systems, exploratory studies were also conducted on the architectures and the methodology of the systems. The purpose of conducting the study is to choose the best methodology that suits the pacing of the project development, and to choose the appropriate system architectures, frameworks, libraries and database to be used for the system. Under the architecture study, the difference between native mobile and hybrid mobile development were ventured, which can be concluded as: native mobile hold inherent advantage over hybrid mobile application due to code familiarity to the device's hardware and build, along with smaller packaging size due to the application serving for a single platform. Despite precaution on the difficulty in accessing hardware functionality using hybrid mobile development, the flexibility of deploying in both mobile platforms at once can be a significant time saver. Requirement was elicited to signify that only the android hardware will be focused on when implementing the mobile application with a hybrid framework. Maturity of the react framework is not a concern as the system may not be handling complex operations. The database to be chosen were cloud based relational database due to ease of query and acceptable performance due to the size of the application to be developed.

The second discovery method used in this project is the stakeholders survey. A survey consisting of questions regarding basic procedures of the event is created via

both open ended questions and close ended questions. The purpose of the survey is to identify the appropriate moment in which the developed application can serve its purpose. Besides that, the survey also serves to establish the core functionality required by the stakeholders. The survey has been completed by teachers and event assistants of SMJK Katholik PJ. Runner bib received from the respondent is included in appendix B, and the result of survey is included in appendix A.

Through the requirement discovery methods, the core requirements have been identified and listed in section 4.2. It has been revealed that the methods employed by the school is through manual logging methods, and digital logging method at the stations. Handwritten data will be passed to the judge for digital logging where the ranking of the students was generated. Overall, the average time taken to record a student is 2 seconds with a rare tendency to mis-record the first 150 students of the checkpoints. The difficult part of the methods employed by the schools is difficulty in tracking students when the participants come in group, therefore rubber bands were used to identify if the student had passed a checkpoint. Several concerns unrecognized from the initial assumptions made were also obtained from the stakeholders. The unrecognized concerns were the requirements to attribute scores into the eligible winning participants' sports house (rumah sukan) and to tally up all the scores obtained by each sport house. Using this information, requirement of implementing QR scanning were elicited to handle the issue of the school's current weak point in handling the scanning of participants that comes in group. It is believed that QR scanning may require less effort from the personnel and is faster.

4.3 Requirement Specification

4.2.1 Functional Requirements

1. The system should let users register for account.
2. The system should let users login into their account using the registered account information.
3. The system should let registered users perform updates to their account.
4. The system should let users to search for past finished cross-country event details.
5. The system should let users sort the past cross-country event details based on provided criteria.

6. The system should let registered users create and configure a new upcoming cross-country event to be managed.
7. The system should let event organizers assign registered users as the event personnel for their upcoming event.
8. The system should let event organizers set the number of checkpoints and checkpoints required by each group.
9. The system should let event organizers to make amendment to an existing event via mobile and web application interface.
10. The system should highlight the error of event configuration and prevent the wrong amendment from being uploaded into the database.
11. The system should allow event organizers to define the group of runners for an event.
12. The system should allow event organizers and event personnel to view the status and details of their event.
13. The system should start timing the event once the event organizer starts the event and stops when the event organizer ends the event.
14. The system should allow event personnel to register participants into the database using the web application before the event starts.
15. The system should generate QR code based on the participants information.
16. The system should only allow event personnel and record participants at a checkpoint using mobile application via either QR Code scanning or manual data entry.
17. The system should recognize and prevent the recording of recorded participants for the same checkpoint.
18. The system should connect to the designated API gateway for running all CRUD operation.
19. The system should be able to perform operations on Amazon Relation Database MariaDB as defined by the instructions.
20. The system should allow registered users to view their profile information.
21. The system should allow event personnel to configure the list of event participants before the event starts.

22. The system should generate the final result ranking and display the result ranking in the event detail page after the event's completion.
23. The system should provide interface for showing the list of events that are created by the user and events that are assigned as personnel for the user.

4.2.2 Non-Functional Requirements

1. The system should be developed in English language.
2. The mobile application should operate in android device of version 8.0 and above with minimum of 64mb available storage.
3. The web application should run in Mozilla web browser version 38.0 and above.
4. The web and mobile application should connect to and synchronize with the same cloud database.
5. The system should utilize android mobile camera for QR Code scanning.
6. The web application should utilize local storage and sessions to keep data.
7. The system should allow user to register using Google account.
8. The system should have login persistence.
9. The system should display the relevant errors on poorly filled form fields.

CHAPTER 5

Project Design

5.1 Software Architecture Design

The main architecture design employed for this system is the Flux architecture design. Flux is the application architecture used by Facebook to develop the client-site web application as it properly describes React framework's components and the unidirectional data flow (Facebook.github, n.d.). As the majority of the cross country event management system is developed using React framework, flux architecture better depicts the structure and relationship of the systems components compared to other architecture patterns.

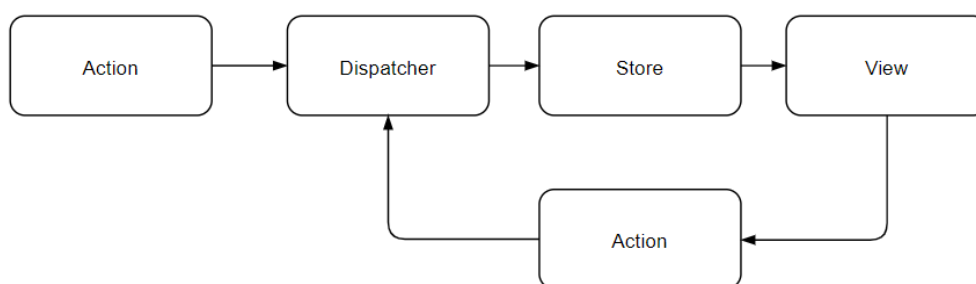


Figure 5.1.1: Flux Software Architecture Design

The flux architecture is derived and modified from Model-View-Controller (MVC) architecture pattern, with modification to acknowledge that each 'view' from react, or a react component, will be performing unidirectional data flow update with the model directly without the assistance of controller. The architecture also acknowledges that a single screen will have multiple components or views, some of which are used multiple times in a single page with different props, and most are re-rendered during state changes.

Both mobile and web application begin with an action initially, which will perform dispatch to the necessary functions for rendering the route, navigations, containers and components. The dispatcher also includes performing requests with the API gateway in order to retrieve database information such as the user details and event details in order to populate the store (consisting of the component's state and the redux state) with the information. After the page's views have been rendered, the user can perform UI action on the components to trigger dispatch for the appropriate response.

Transition from one page into the next page are handled by navigation dispatch and the appropriate dispatch for passing props or updating redux state.

The software architecture design from a top view may indicate a design similar to a client-server architecture design. The client side will exchange information with the server side in order to facilitate the complete workflow of the system. The overview of such design can be described with the diagram below:

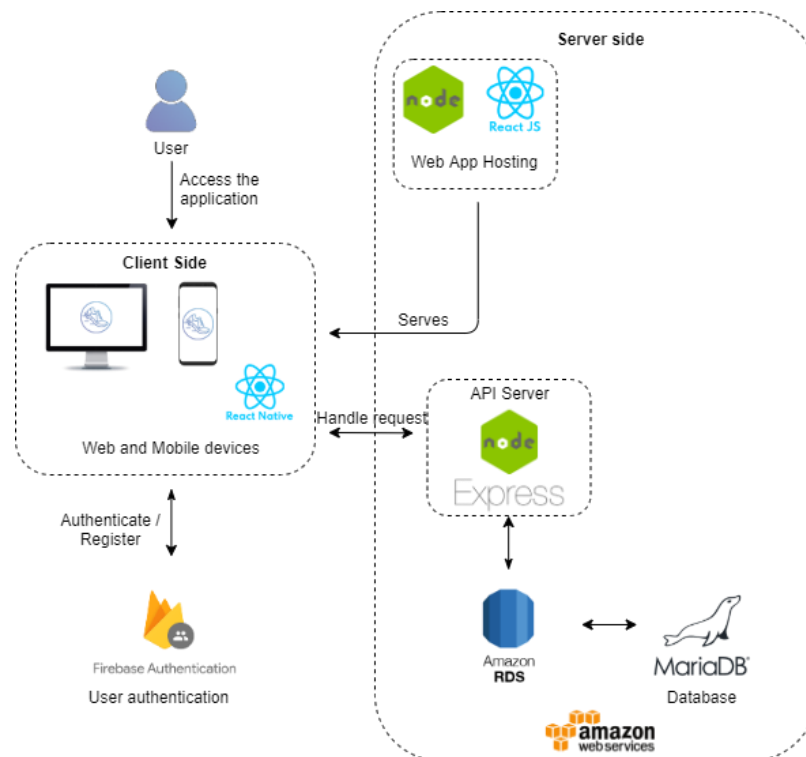


Figure 5.2.2: Client-Server Architecture Design

The client side mainly consist of the devices used by the users to access the applications of the system, which is the web application and the mobile application. The mobile application built using react native will compile the application code and bridge with the native modules in order to render the views. Additionally, the framework may utilize the hardware functionality of camera to perform QR scanning as permitted.

For the server side, the API server serves as the gateway for handling all requests. As mentioned, Express framework is used to host the API gateway. The web application is temporary hosted using Node.Js and React framework during development. Both the API server and web application server are hosted locally, therefore the server needs to be initialized and may suffer downtime influenced by the host server's condition.

Amazon Relational Database Service (RDS) is used for hosting MariaDB in a cloud environment. All database queries and operation are communicated using the API gateway, and the results will be returned appropriately back to the client side. Most management for MariaDB hosting is configured through Amazon Web Service (AWS) console, this include setting up the proxy, VPC and permitted IP groups.

5.2 Database Design

5.2.1 Entity Relationship Diagram (ERD)

ERD illustrates the relationship between the tables of the database. There are a total of 8 main tables used for the database. The event result must be referenced to an existing data of other tables. Despite the relationship, foreign keys are not implemented yet to exchange data integrity for data insertion and update flexibility. 'kb_sum_of_checkpoints' is a 'knowledge base' table, which is used as a table for providing information on the arithmetic series of checkpoints for completion validation.

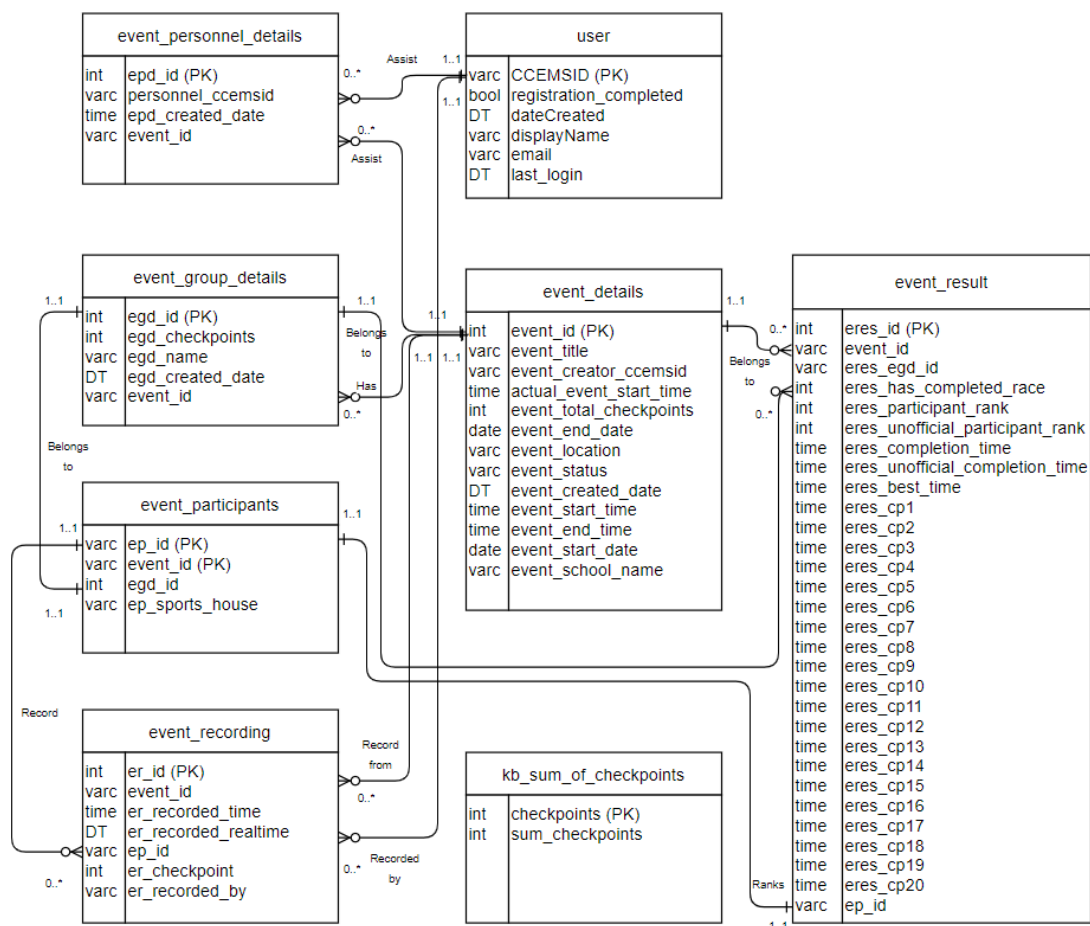


Figure 5.3.1.1: Entity Relationship Diagram for CCEMS

5.3 Use Case

Use case diagram is used to define the interactions of different groups of system users with the system. Each system user category has different functionalities, but may inherit the functionalities of the generalized category. A public user is a user that has not logged in with a registered account. A registered user is a logged in user and can be assigned as an event personnel for none or many events, or choose to create upcoming events. The event creator will be the event organizer and cannot be reassigned. An event organizer has the moderating functionalities for the event they created and is generalized as event personnel as they have access to event day functionalities too. After the use case diagram is drawn, the use case specification is defined to describe the details and flow of how the use cases are performed.

5.3.1 Use Case Diagram

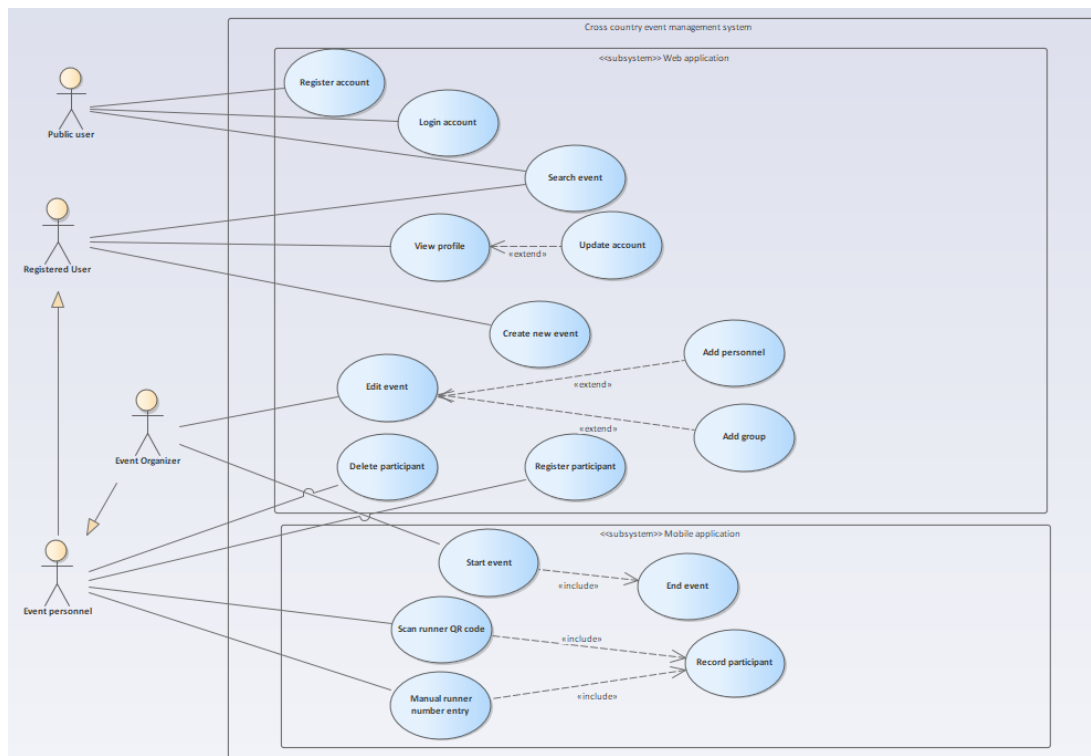


Figure 5.4.1.1: Use Case Diagram For Cross Country Event Management System.

5.3.2 Use-Case Specification

5.3.2.1 Register Account

Use case name	Register account
Use case ID	A1

Relevant functional requirement	1. The system should let users register for account.	
Priority	Medium	
Source	-	
Primary business actor	Public User	
Description	This use case describes how a non-registered user can register for an account to use this system.	
Precondition	-	
Trigger	User clicks on the register an account button.	
Course of events	Actor actions	System responses
	1.The user clicks on register an account 3.The user enters all required details 4.The user clicks on register button	2.System shows the registration screen 5.System performs validity check based on data and clashing username. 6.System display message confirming that the registration is successful.
Alternative course of events	5 The system found invalid data or clashing username 5.1 The system display invalid message 5.2 The system highlights the invalid field	
Post condition	The system registered the new account and log the new user in.	

5.3.2.2 Login Account

Use case name	Login account
Use case ID	A2
Relevant functional requirement	2. The system should let users login into their account using the registered account information.
Priority	High
Source	-

Primary business actor	Public User	
Description	This use case describes how an user can login into their account	
Precondition	-	
Trigger	User entered the login screen	
Course of events	Actor actions	System responses
	1.The user enters the login screen 3.The user enters username and password or click on sign in using other method 3. The user clicks on sign in or sign in using other method	4.The system fetches the matching account 5.The system returns user to main screen
Alternative course of events	4 The system couldn't find any matching account 4.1 The system displays invalid message	
Post condition	The user is logged in with the correct personal details.	

5.3.2.3 View Profile

Use case name	View profile	
Use case ID	A3	
Relevant functional requirement	20. The system should allow registered users to view their profile information.	
Priority	High	
Source	-	
Primary business actor	Registered user	
Description	This use case describes how an user can login into their account	
Precondition	User is logged in	
Trigger	User clicks on profile	
Course of events	Actor actions	System responses

	1.The user enters the profile screen	2.The system fetches the account's details 3.The system displays the result in a specific format
Alternative course of events	-	
Post condition	The profile screen displays all the correct information.	

5.3.2.4 Update Account

Use case name	Update account	
Use case ID	A4	
Relevant functional requirement	3. The system should let registered users perform updates to their account.	
Priority	High	
Source	Extends: Use case ID A3 - View profile	
Primary business actor	Registered User	
Description	This use case describes how can user can update their account.	
Precondition	User performed use case id A3 and is in profile screen	
Trigger	User clicks on 'update profile' button.	
Course of events	Actor actions	System responses
	1.The user clicks the 'update profile' button in the profile screen. 2.The user updates any field. 3.The user clicks on confirm update button.	4.The system updates the account information in the database.
Alternative course of events	4 The system found field(s) that was keyed in incorrectly 4.1 The system displays error message 4.2 The system highlights the invalid field(s)	
Post condition	The user account information is updated.	

5.3.2.5 Search Event

Use case name	Search event	
Use case ID	S1	
Relevant functional requirement	4. The system should let users to search for past finished cross-country event details.	
Priority	High	
Source	-	
Primary business actor	Public User, Registered User	
Description	This use case describes how any user can find an existing event in the database	
Precondition	-	
Trigger	User enters the result searching screen.	
Course of events	Actor actions	System responses
	1.The user enters the result searching screen. 2.The user enters the event detail for searching. 3.The user clicks on search event. 6.The user clicks any event from the displayed list.	4.The system retrieves the relevant results. 5.The system presents a list of relevant event. 7.The system retrieves the selected event's data. 8.The system displays the result.
Alternative course of events	4 The system was unable to find any relevant result. 4.1 The system display 'No result found' message in the result screen.	
Post condition	The selected event's detail is displayed.	

5.3.2.6 Create New Event

Use case name	Create new event
Use case ID	E1
Relevant functional requirement	6. The system should let registered users create and configure a new upcoming cross-country event to be managed.

Priority	High	
Source	-	
Primary business actor	Registered User	
Other participating actors	-	
Description	This use case describes how a registered user can create management repository for an upcoming cross-country event.	
Precondition	-	
Trigger	User clicks on 'Create new event' button.	
Course of events	Actor actions	System responses
	1.The user enters the create event screen. 2.The user follows all guideline and fill in all required details.	3.The system checks for conflicting field. 4.The system registers the new event into the database and adds the new event into the relevant users' directories.
Alternative course of events	3 The system found field(s) that was keyed in incorrectly 3.1 The system displays error message 3.2 The system highlights the invalid field(s)	
Post condition	The new event is created.	

5.3.2.7 Edit Event

Use case name	Edit event
Use case ID	E2
Relevant functional requirement	6. The system should let registered users create and configure a new upcoming cross-country event to be managed.
Priority	High
Source	-
Primary business actor	Event organizer

Description	This use case describes how an event organizer user can edit an upcoming event's details.	
Precondition	The event organizer must have an upcoming event.	
Trigger	User clicks on 'Edit event' button.	
Course of events	Actor actions	System responses
	<ol style="list-style-type: none"> 1.The user enters select an upcoming event event. 2.The user chooses the edit event option. 3.The user follows all guideline and modify any details. 4.The user clicks on 'Update event' button 	<ol style="list-style-type: none"> 5.The system checks for conflicting field. 6.The system updates the event in the database.
Alternative course of events	<ol style="list-style-type: none"> 3 The user wants to add new personnel <ol style="list-style-type: none"> 3.1 The user clicks on add new personnel 3.2 The user performs use case ID E3 5 The system found field(s) that was keyed in incorrectly <ol style="list-style-type: none"> 5.1 The system displays error message 5.2 The system highlights the invalid field(s) 	
Post condition	The event is updated with new information.	

5.3.2.8 Add Personnel

Use case name	Add personnel
Use case ID	E3
Relevant functional requirement	7. The system should let event organizers assign registered users as the event personnel for their upcoming event.
Priority	High
Source	Extends: Use case ID E2 - Edit event
Primary business actor	Event organizer
Description	This use case describes how an event organizer can add new personnel to assist in an upcoming event.

Precondition	-The event organizer must have an upcoming event. -The event organizer was performing use case ID E2	
Trigger	User clicks on 'Add personnel' button.	
Course of events	Actor actions	System responses
	1.The user clicked on 'Add new personnel' from edit event screen. 2.The user enters the relevant personnel details. 3.The user clicks on 'Confirm add personnel' button	4.The system searches for the registered user in the database using the details entered. 5.The system adds the personnel into the event and adds the event into the personnel's list of events.
Alternative course of events	4 The system was unable to find the matching user. 4.1 The system displays error message 4.2 The system highlights the invalid field(s)	
Post condition	The event is updated with new personnel and the personnel have access to functions for the event.	

5.3.2.9 Register Participant

Use case name	Register participants
Use case ID	E5
Relevant functional requirement	14. The system should allow event personnel to register participants into the database before the event starts.
Priority	High
Source	-
Primary business actor	Event personnel
Description	This use case describes how an event organizer or an event personnel can add new participants or runner for the event.
Precondition	-The event organizer must have an upcoming event.
Trigger	User clicks on 'Register participants' button.

Course of events	Actor actions	System responses
	<p>1.The user clicked on ‘Register participant’ button at the event page.</p> <p>2.The user enters the relevant participant information.</p>	<p>3.The system search for similar participants in the database.</p> <p>4. The system save the participants into the database.</p>
Alternative course of events	<p>3 The system found a matching participant with similar data.</p> <p>3.1 The system disables the register button.</p>	
Post condition	The event is updated with new participant.	

5.3.2.10 Start Event

Use case name	Start event	
Use case ID	E6	
Relevant functional requirement	13. The system should start timing the event once the event organizer starts the event and stops when the event organizer ends the event.	
Priority	High	
Source	-	
Primary business actor	Event organizer	
Description	This use case describes how an event organizer can start the event during event day.	
Precondition	The event must be an upcoming event.	
Trigger	User clicks on ‘Start event’ button.	
Course of events	Actor actions	System responses
	<p>1.The user clicks on ‘Start event’ button from the event screen.</p>	<p>2.The event starts timing and event day features are enabled for all event personnel.</p>
Alternative course of events	2. The event is unable to start	

	2.1 The system display error suggesting possible cause stopping the event from starting.
Post condition	The event is starts and event personnel have access to event features.

5.3.2.11 End Event

Use case name	End event	
Use case ID	E7	
Relevant functional requirement	13. The system should start timing the event once the event organizer starts the event and stops when the event organizer ends the event.	
Priority	High	
Source	Included by: Use case ID E6 – Start event	
Primary business actor	Event organizer	
Description	This use case describes how an event organizer can end and conclude an event.	
Precondition	The event must be a currently progressing event.	
Trigger	User clicks on ‘End event’ button.	
Course of events	Actor actions	System responses
	1.The user clicks on ‘End event’ button from the event screen. 3.The user confirms to end event.	2.The system asks for confirmation to end the event. 4.The system finalizes the event and disables selected event features for event personnel.
Alternative course of events	-	
Post condition	The event is concluded and finalized.	

5.3.2.12 Register Participant

Use case name	Register participant
---------------	----------------------

Use case ID	E8	
Priority	Medium	
Source	-	
Primary business actor	Event personnel	
Description	This use case describes how an event personnel can register a participating runner into the system.	
Precondition	The event must exist in the database.	
Trigger	User clicks on 'Register participant' button.	
Course of events	Actor actions	System responses
	<p>1.The user clicks on 'register participant' button from the event screen.</p> <p>2.The user can choose to scan the QR Code of the participant.</p> <p>3.The user enters additional details of the participant.</p>	4.The system registers the participant into the database.
Alternative course of events	-	
Post condition	The event database contains additional information of the participants.	

5.3.2.13 Delete Participant

Use case name	Delete participant
Use case ID	E11
Relevant functional requirement	21. The system should allow event personnel to configure the list of event participants before the event starts.
Priority	Medium
Source	-
Primary business actor	Event personnel

Description	This use case describes how an event personnel can delete a participating runner from the event.	
Precondition	The event must exist in the database. The participant must have been registered for the event.	
Trigger	User clicks on 'Register participant' button.	
Course of events	Actor actions	System responses
	1.The user clicks on 'participant list' button from the event screen. 2.The user locates the participant and click on the icon with 'trashcan' image.	3. The system delete the participants from that event in the database.
Alternative course of events	-	
Post condition	The participant has been removed from the event's participant table.	

5.3.2.14 Scan Runner QR Code

Use case name	Scan runner QR Code	
Use case ID	E9	
Relevant functional requirement	16. The system should only allow event personnel and record participants at a checkpoint using mobile application via either QR Code scanning or manual data entry.	
Priority	High	
Source	-	
Primary business actor	Event personnel	
Description	This use case describes how an event personnel can scan the qr code of a participant during event day.	
Precondition	The event must have started.	
Trigger	User is on record participant screen.	
Course of events	Actor actions	System responses

	<p>1.The user enters the record participant screen.</p> <p>2.The user aligns the phone camera to the QR code.</p> <p>5.The user clicks on the ‘record’ button of the ‘Snackbar’ to confirm record.</p>	<p>3.The mobile camera detects the QR Code.</p> <p>4.The system creates a ‘snackbar’ message to display the runner information with ‘record’ button as option.</p> <p>6.The system recognize the record invoke and saves the participants into the database. A ‘snackbar’ containing the ‘success’ message to indicate successful record.</p>
Alternative course of events	-	
Post condition	The system displays the runner information with ‘record’ option besides it.	

5.3.2.15 Manual Runner Number Entry

Use case name	Manual runner number entry
Use case ID	E10
Relevant functional requirement	16. The system should only allow event personnel and record participants at a checkpoint using mobile application via either QR Code scanning or manual data entry.
Priority	High
Source	-
Primary business actor	Event personnel
Description	This use case describes how an event personnel can manually enter the participant code during event day.
Precondition	The event must have started.

Trigger	User is on record participant screen.	
Course of events	Actor actions	System responses
	1.The user enters the record participant screen. 2.The user enters the runner code in the field below. 3.The user clicks on the manual entry section's 'record' button,	4. The system detects recordable information and creates 'Snackbar' message with 'success' message to indicate successful record.
Alternative course of events	-	
Post condition	The 'record' option for manual entry becomes clickable.	

5.4 User Interface Design

5.4.1 Web Application

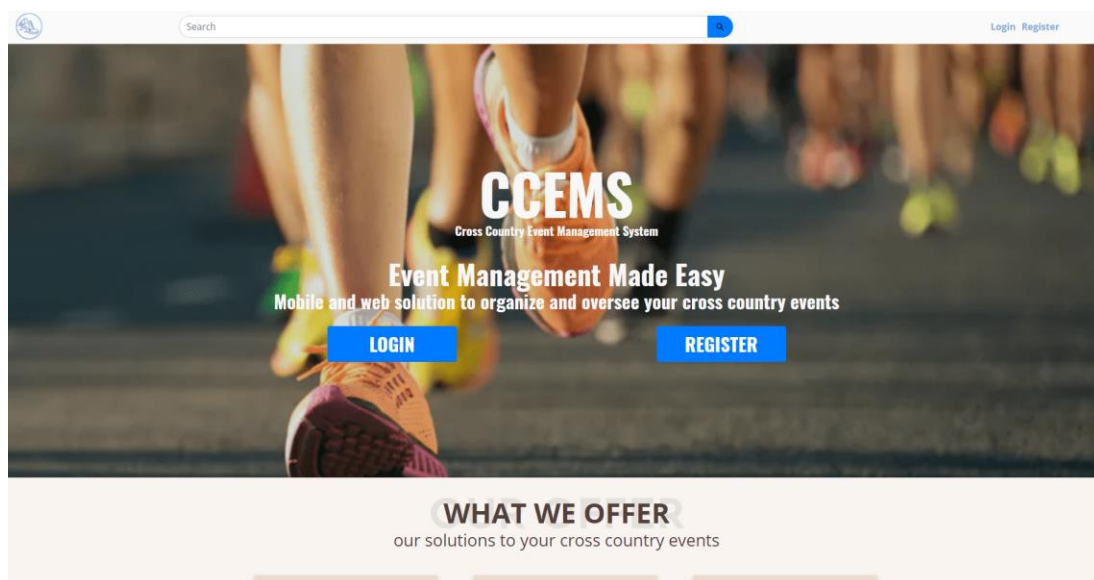


Figure 5.5.1.1: Visitor Page

Use case ID	-
Functional requirement number	-
Description	Screen above shows the page of non-logged in users. Through this page, the user can choose to search, login or register.

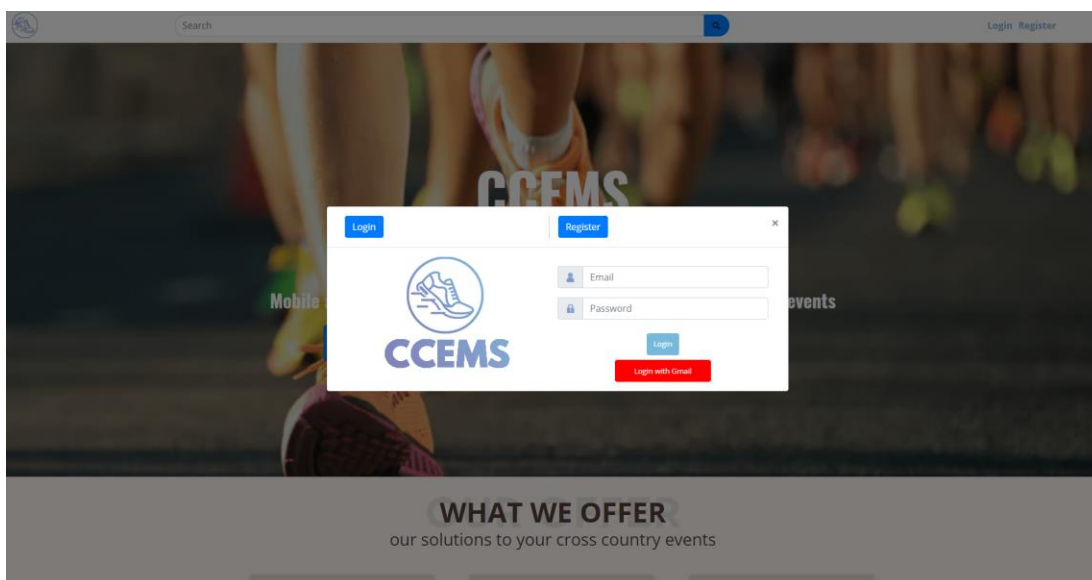


Figure 5.6.1.2: Visitor Login Form

Use case ID	A2
Functional requirement number	2
Description	Screen above shows the form to login into the CCEMS web application. User will be logged in with valid credential.

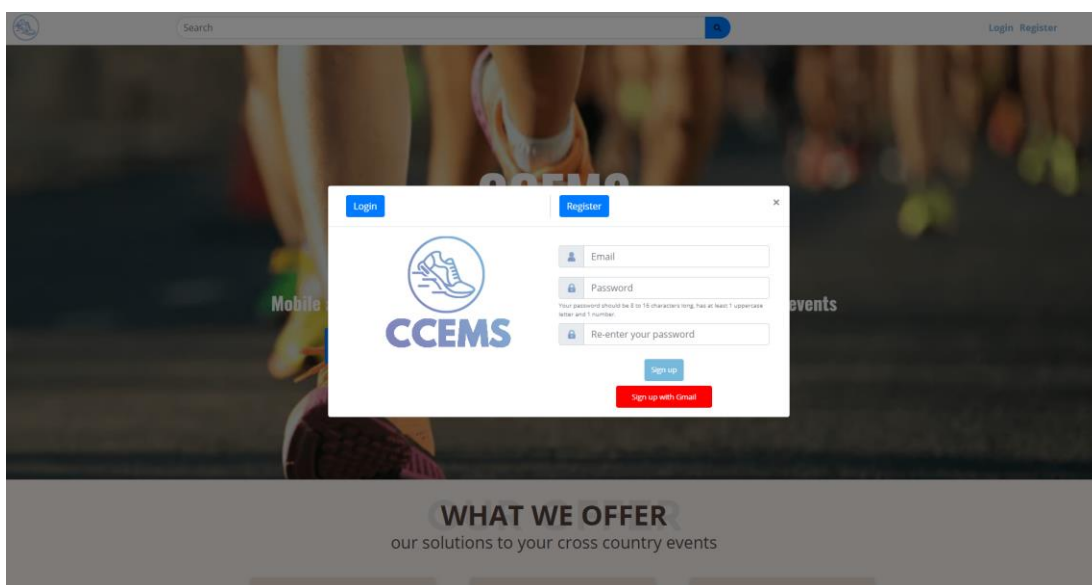


Figure 5.7.1.3: Visitor Registration Form

Use case ID	A1
-------------	----

Functional requirement number	1
Description	Screen above shows the form with relevant fields to register a CCEMS account using the web application.

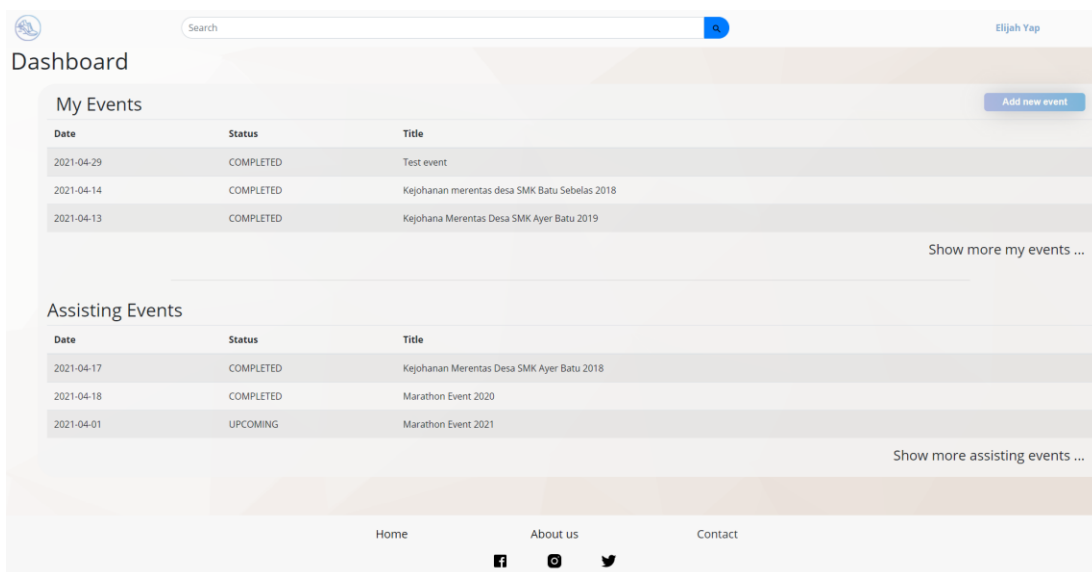


Figure 5.8.1.4: User Home

Use case ID	-
Functional requirement number	23
Description	Screen above shows home page after an user has logged in. Upcoming 3 events created by the user and assigned as personnel is shown in the dashboard.

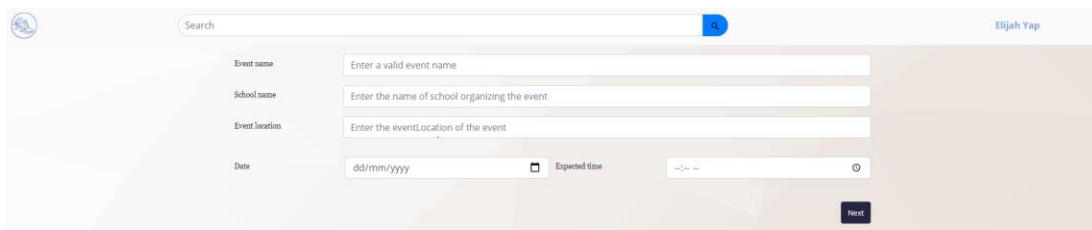


Figure 5.9.1.5: Create Event First Step

Use case ID	E1
Functional requirement number	6, 10

Description	Screen above shows the first step of event creation process using the web application. Users are required to fill the basic detail of the event in this step. Incorrectly filled form should display error message below the field
-------------	--

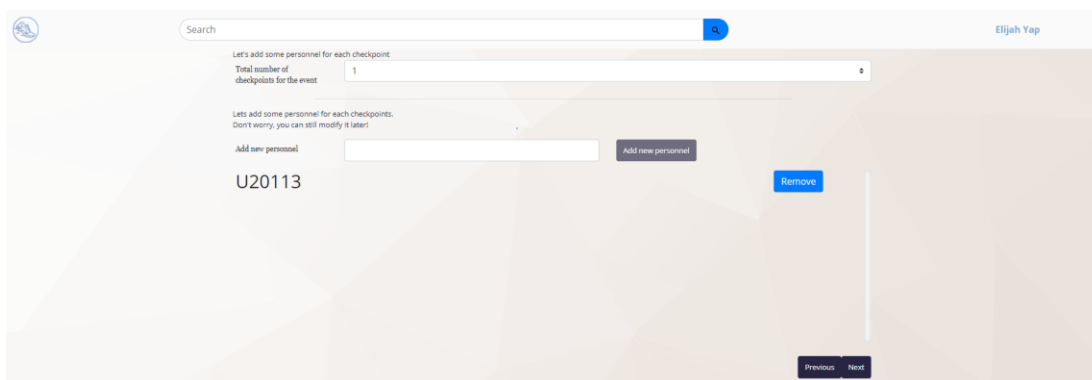


Figure 5.10.1.6: Create Event Second Step

Use case ID	E1, E3
Functional requirement number	2,7,8
Description	Screen above shows the second step of event creation process using the web application. Users are required to register assigned personnel for the event.

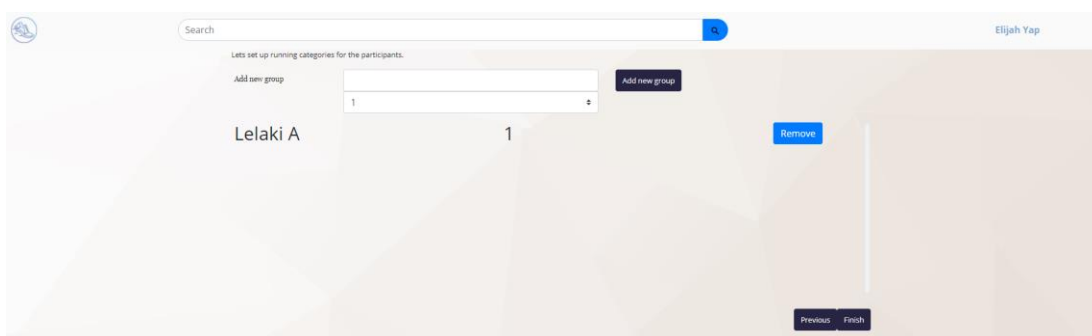


Figure 5.11.1.7: Create Event Third Step

Use case ID	E1, E4
Functional requirement number	2,8,11
Description	Screen above shows the third and last step of event creation process using the web application. Users are required to

	appoint group(s) for the event and the checkpoint required for each group to be passed during event.
--	--

No.	Date	Status	Title	Creator
1	2021-04-29 10:00 PM	COMPLETED	Test event time	U20113
2	2021-04-18 04:20 PM	COMPLETED	Wowee marathon 3	U56174
3	2021-04-17 04:20 PM	COMPLETED	Wowee How	U29647
4	2021-04-15 12:37 PM	UPCOMING	Test event name	U20113
5	2021-04-14 11:32 PM	COMPLETED	Sample event	U20113
6	2021-04-13 12:00 PM	ONGOING	Test event mobile	U20113
7	2021-04-13 12:00 PM	COMPLETED	Test event mobile	U20113
8	2021-04-13 11:59 PM	COMPLETED	Test event name 144	U20113
9	2021-04-12 12:00 PM	UPCOMING	TEST MOBILE EVENT	U20113
10	2021-04-07 01:11 AM	UPCOMING	Test event name	U20113

Figure 5.12.1.8: User's All Event List Page

Use case ID	S1
Functional requirement number	5
Description	Screen above shows the list of events that are created or assigned as personnel by the users. Click on the item will redirect user to the event page.

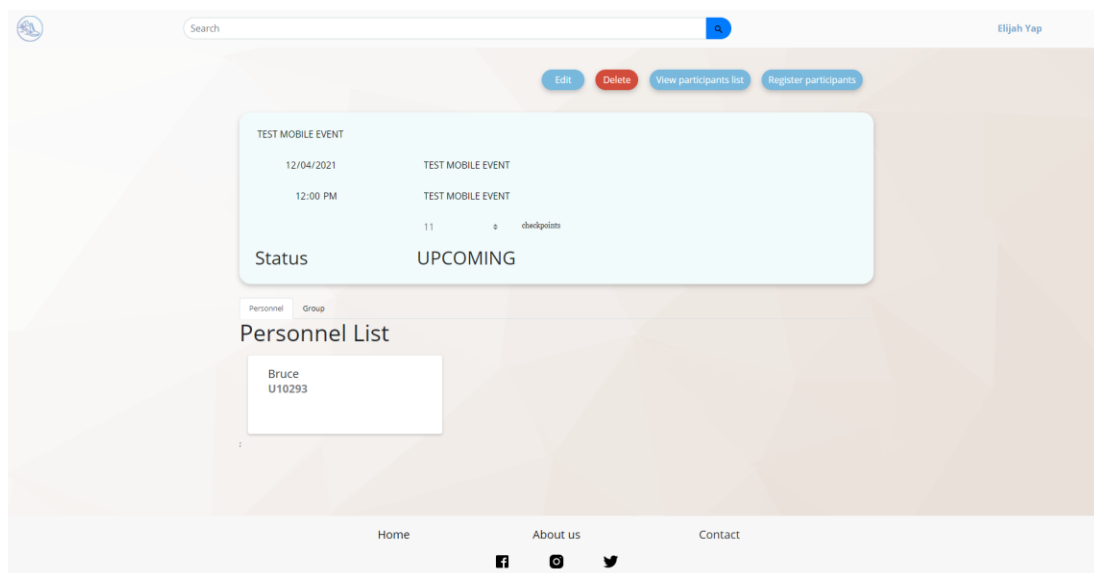


Figure 5.13.1.9: Organizer Upcoming Event Page

Use case ID	E2
Functional requirement number	12
Description	Screen above shows the event detail page for an organizer's upcoming event in web application. Edit and delete functionality is available and when clicked will trigger the E2 use case. View participant list and register participant functionality are also available for the functionality as implied.

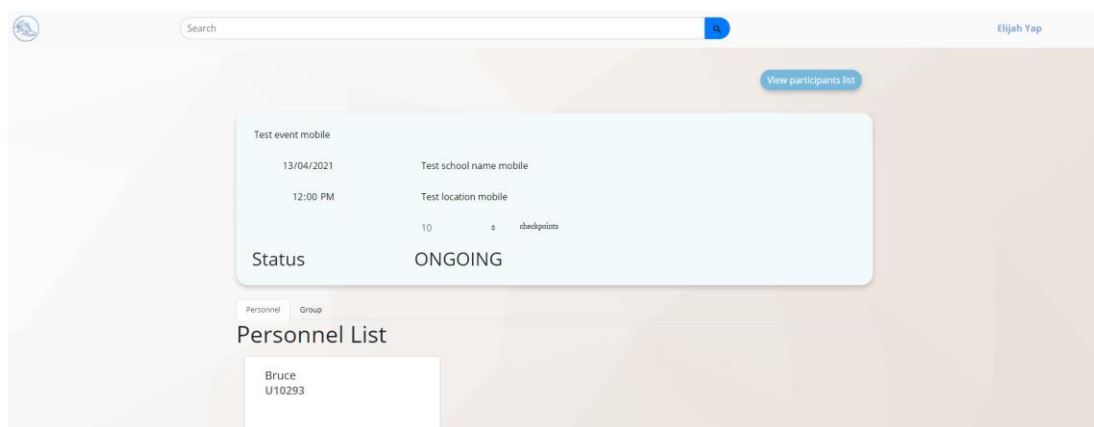


Figure 5.14.1.10: Organizer Ongoing Event Page

Use case ID	-
Functional requirement number	12

Description	Screen above shows the event detail page of an ongoing event in web application. Edit, delete or add participant functionalities will not be available.
-------------	---

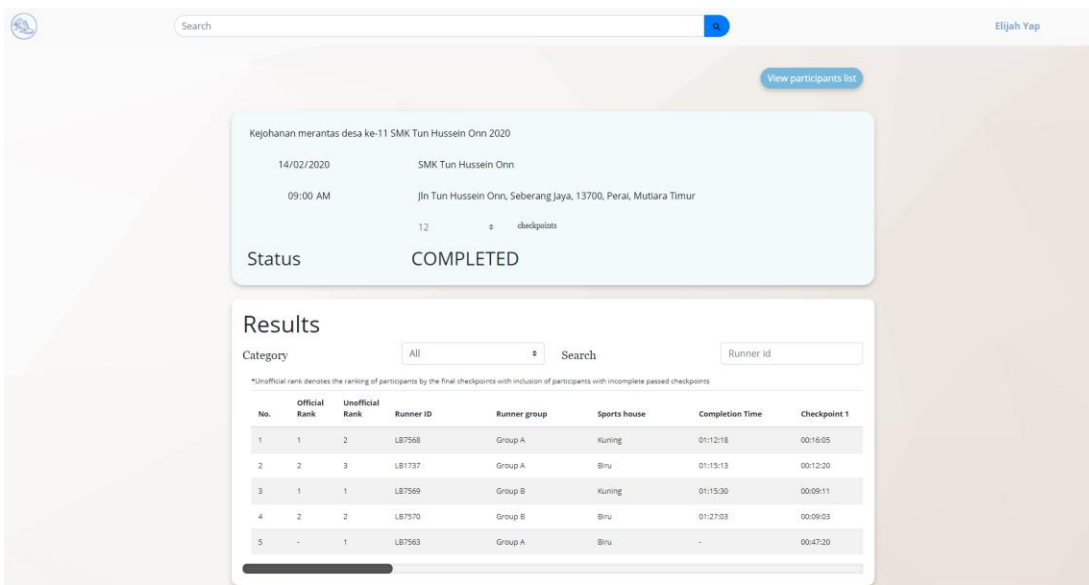


Figure 5.15.1.11: Completed Event Page

Use case ID	-
Functional requirement number	12, 22
Description	Screen above shows the event detail page for a completed event in web application. Result ranking will be shown in the page and can be sorted according to the functionalities provided. Runner name will not be shown in the result preview as to enforce runner’s privacy.

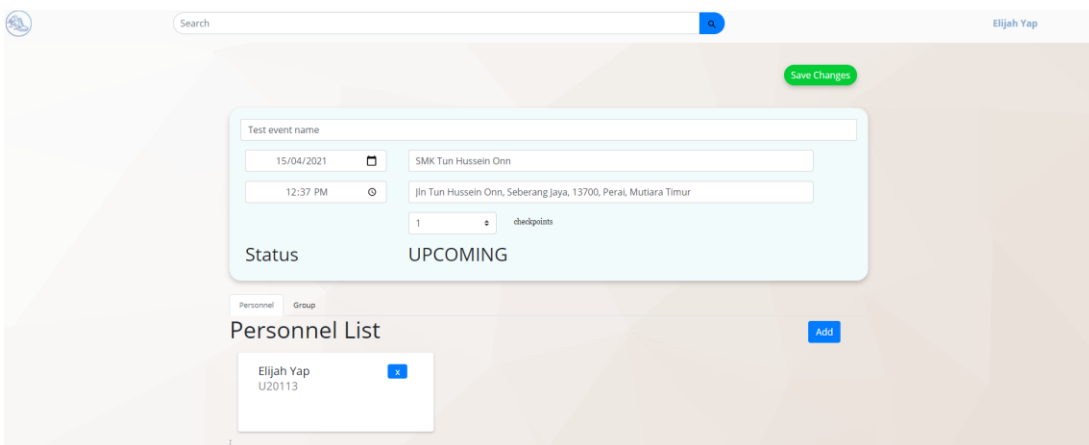


Figure 5.16.1.12: Event Page Edit State

Use case ID	E2, E3, E4
Functional requirement number	9
Description	Screen above shows the edit event detail screen after clicking on the edit event button. Basic event details, personnel list and group list can be updated accordingly. Clicking on save changes will overwrite the event detail in the database with the new data.

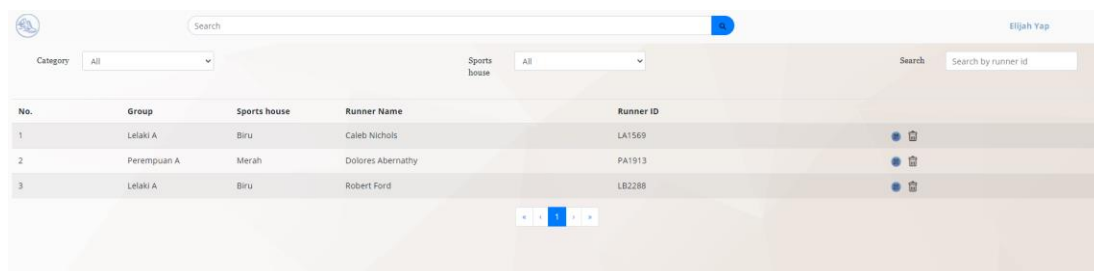


Figure 5.17.1.13: Participant List Page

Use case ID	E11
Functional requirement number	21
Description	Screen above shows the participant list page. The event participants details are shown here but cannot be updated anymore. The participants can be deleted. Clicking on the QR icon will show the runner's QR image. Participant name will only be shown in the participants list.



Figure 5.18.1.14: Participant List QR View Page

Use case ID	-
Functional requirement number	15

Description	Screen above shows the QR image of a participant after clicking on the QR icon in the participant list page. The image can be downloaded.
-------------	---

Figure 5.19.1.15: Register Participant Page

Use case ID	E8
Functional requirement number	14, 15
Description	Screen above shows the participant registration page. New participant can be added into the event by filling in the field and clicking the register participant button. One registered, their QR code generated will be displayed below.

The figure consists of two screenshots of a user profile page. The top screenshot shows the 'Edit' mode, with an 'Edit' button in the top right corner. The form contains three input fields: 'Display name' with the value 'Elijah Yapp', 'Email' with the value 'elijah@gmail.com', and 'Current password' with a masked password '*****'. The bottom screenshot shows the 'Save changes' mode, with a 'Save changes' button in the top right corner. The form contains four input fields: 'Display name' with the value 'Elijah Yapp', 'Email' with the value 'elijah@gmail.com', 'Password' with a masked password, and 'Confirm password' with a masked password.

Figure 5.20.1.16: User Profile Page

Use case ID	A3, A4
Functional requirement number	3, 20
Description	Screen above shows the user profile page without and with edit mode enabled respectively. User can choose to update their display name and password for the CCEMS system.

The figure shows a search result page with a search bar containing 'SMK Convent' and a search icon. Below the search bar, it says 'Searching results for SMK Convent (1 items...)'. The results are displayed in a table with the following columns: No., Date, Status, School, Title, and Creator. The table contains one row of data.

No.	Date	Status	School	Title	Creator
1	2021-07-28 01:37 PM	UPCOMING	SMK Tun Hussein Onn	Kejuhanan Merentas Desa SMK Convent Kajang 2021	U20113

Figure 5.21.1.16: Search Result Page

Use case ID	S1
-------------	----

Functional requirement number	4
Description	Screen above shows the page after searching for event using the search bar using the web application. All events containing the search string will be displayed as a list.

5.4.2 Mobile Application

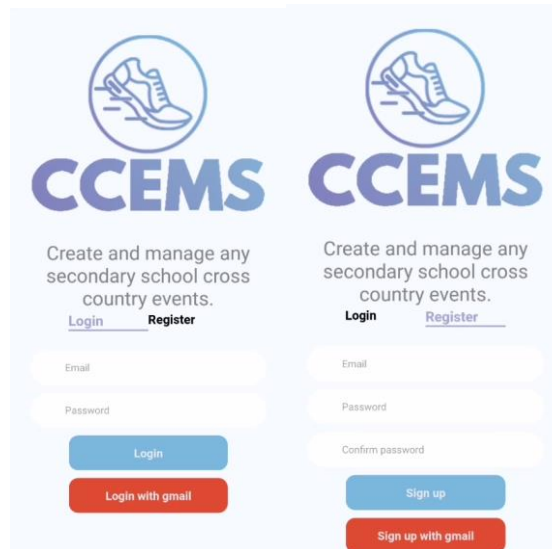


Figure 5.22.2.1: Entry Screen

Use case ID	A1, A2
Functional requirement number	1, 2
Description	Screen above shows the login and register screen for the mobile application. User will be logged in after registering or logging in.

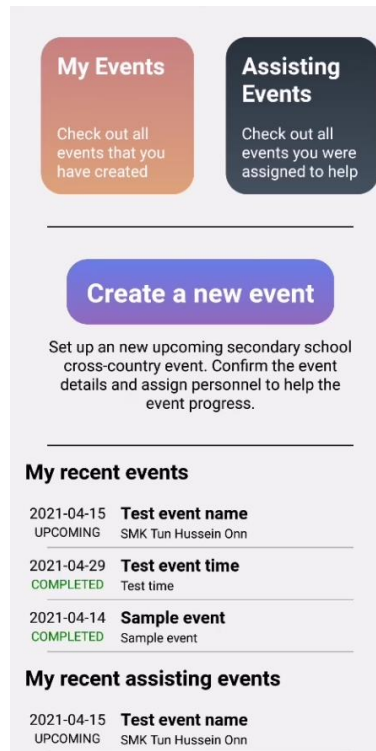


Figure 5.23.2.2: Home page

Use case ID	-
Functional requirement number	23
Description	Screen above shows home page after a user has logged in through the mobile application. Upcoming 3 events created by the user and assigned as personnel is shown in the dashboard.

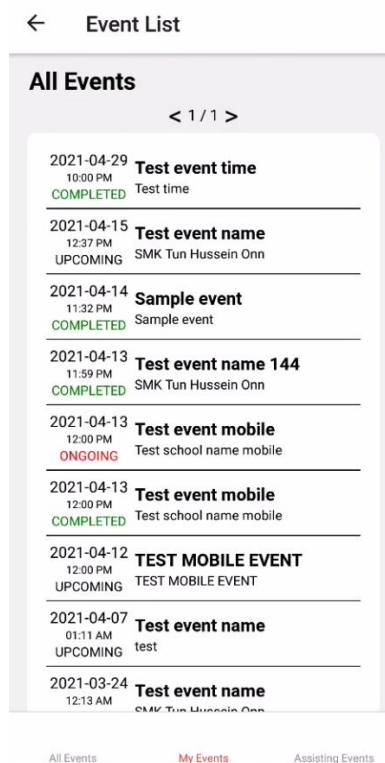


Figure 5.24.2.3: Event List Screen

Use case ID	-
Functional requirement number	23
Description	Screen above shows the event list screen for the mobile application. User can choose to navigate between different event list category by clicking on the tabs below.

Step 1 : Event Details
Let's get some basic information for the cross country event!

Event title

School name

Location

Date 2021-04-18

Expected start time 12:00pm

Number of checkpoints 1

Step 2 : Assign personnels
Add some users using their CCEMS ID to help the event progress!

Personnel ID

Step 3 : Create groups
Define the group or categories that your participants can compete for in this cross-country event!

Group name

Group's checkpoints 1

Figure 5.25.2.4: Create Event Screen

Use case ID	-
Functional requirement number	2, 6, 7, 8, 10, 11
Description	Screen above shows the event creation process through the mobile application. This is an extension from the use case E1, E3, E4 that have been implemented in mobile application instead of web application.

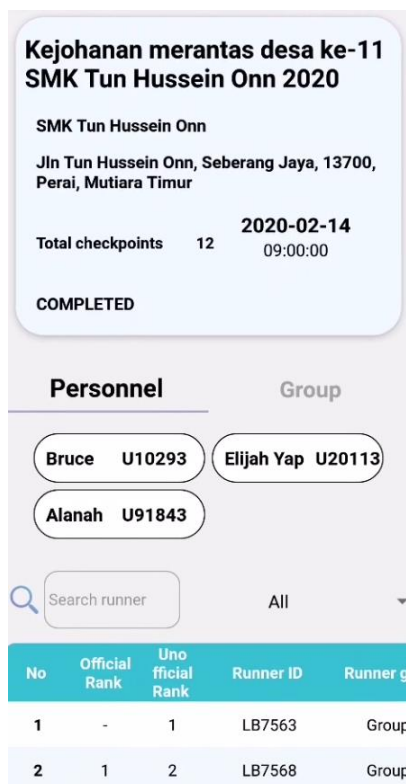


Figure 5.26.2.5: Completed Event Screen

Use case ID	-
Functional requirement number	12, 22
Description	Screen above shows the event detail page for a completed event in mobile application. Result ranking will be shown in the screen and can be sorted using the functionalities provided.



Figure 5.27.2.6: Organizer Upcoming Event Screen

Use case ID	E6
Functional requirement number	13
Description	Screen above shows the event detail page for an organizer's upcoming event in mobile application. Edit and delete functionality is available and when clicked will trigger the E2 use case. The event can be started by clicking the start event button. When the event is started, the event status will be updated and the event timer will start.

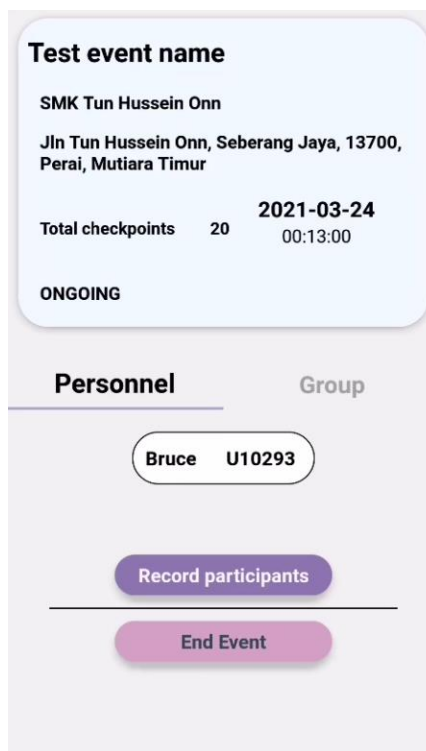


Figure 5.28.2.7: Organizer Ongoing Event Screen

Use case ID	E7
Functional requirement number	13
Description	Screen above shows the event detail page for an organizer's ongoing event in mobile application. Event organizer and personnel can enter the recording screen by clicking the record participants button. Event organizer can end the event and generate a finalized event result ranking by clicking on end event button.

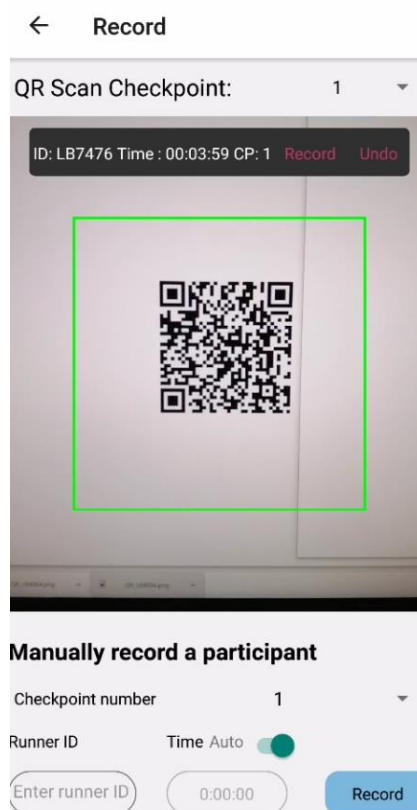


Figure 5.29.2.8: Record Participant Screen Successful Scan

Use case ID	E9, E10
Functional requirement number	16
Description	Screen above shows the runner record screen with a successful QR code scan. A successful QR code scan should result in a record confirmation snackbar showing at top of the screen, given that the runner has not been scanned before for the checkpoint. Clicking on the 'record' of the snackbar will save the runner data into database. Manual data entry will not trigger a snackbar, but instead will show the successful message.

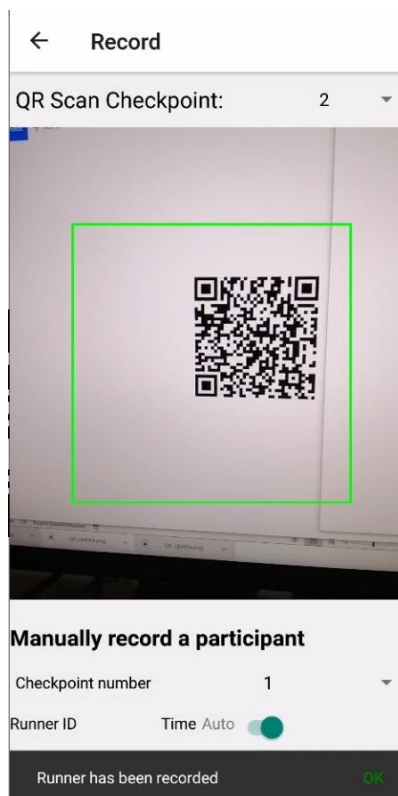


Figure 5.30.2.9: Record Participant Screen Successful Record

Use case ID	E9, E10
Functional requirement number	16, 17
Description	Screen above shows the record screen with successful scan notification at the bottom. A recorded runner will not be re-registered by the system as shown with the lack of confirm record snackbar on top.

CHAPTER 6

Project Implementation

6.1 Route Design

The web application and mobile application routing or navigation have been mapped out to highlight the possible flow path when using the system.

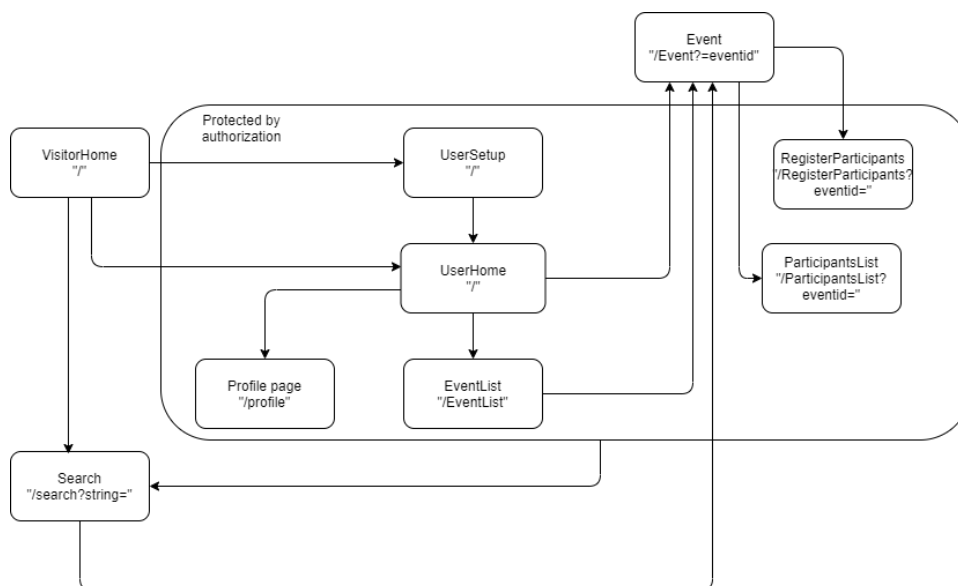


Figure 6.1.1: Web Application Route Diagram

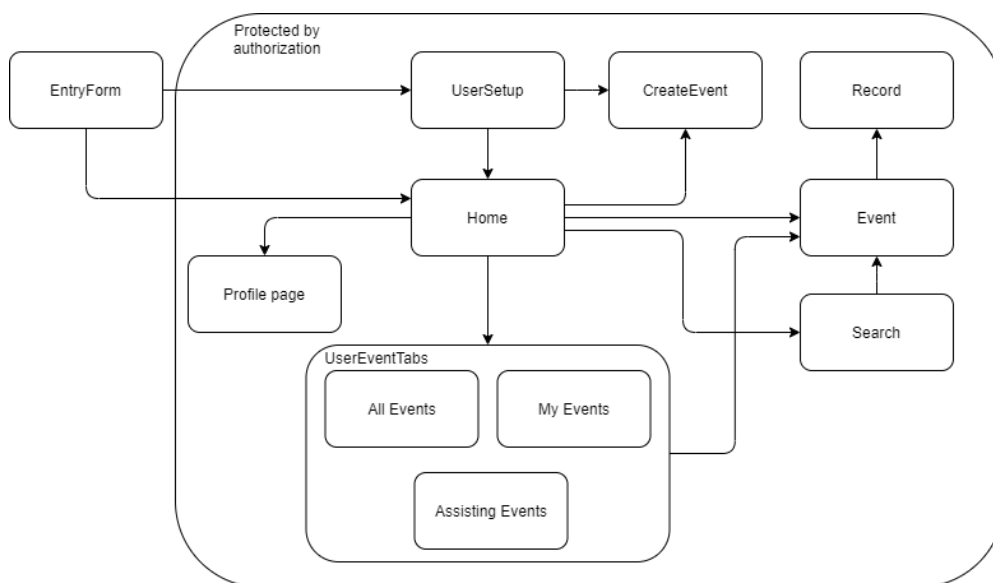


Figure 6.2.2: Mobile Application Route Diagram

6.2 API endpoints

API endpoints indicate the URL for web and mobile application to connect to the API gateway for retrieving the required information or to perform CRUD operation in the database. The most commonly method used is POST, followed with PUT and DELETE methods. GET operation is avoided as most of the operations are not idempotent requests or repeating requests, and there is security disadvantage of identifying payload by URL using GET compared to obtaining payload via JSON using POST. A list of the API endpoints used by both web and mobile application to exchange data with the database can be viewed below:

Table 6.1.1: List of API Endpoints For Both Web And Mobile Applications

Users		
Method	Route	Description
POST	/req/getUser	Retrieve the user data based on the email payload.
POST	/req/signUp	Register the new user into the database.
PUT	/req/UserSetupConfirmation	Update the new user's display name and their confirmation status.
PUT	/req/updateUser	Update the user's personal information to the database.
POST	/req/dashboardMyEvent	Retrieve data for the top 3 upcoming or most recent events for the My Events section of the homepage.
POST	/req/dashboardAssistingEvent	Retrieve data for the top 3 upcoming or most recent events for the Assisting Events section of the homepage.
POST	/req/eventList	Retrieve the list of events that the user is involved as organizer or personnel.
Events		
Method	Route	Description

POST	/req/eventData	Retrieve the event data using the event ID.
POST	/req/createEvent	Create a new record of event and the new records of group and personnel in the database.
PUT	/req/updateEvent	Update the event details, personnel details and group details for the event.
DELETE	/req/deleteEvent	Remove the event details and its corresponding group and personnel details.
POST	/req/registerParticipant	Register a new participant for an event.
POST	/req/getParticipantList	Retrieve the list of participants for the event.
DELETE	/req/deleteParticipant	Remove the participant from the event.
POST	/req/beginEvent	Update the status of the event to 'ONGOING'.
POST	/req/endEvent	Update the status of the event to 'COMPLETED' and generate records of event results.
POST	/req/getParticipantRecords	Retrieve the participant records.
POST	/req/recordParticipant	Record the participant's timing and details from at checkpoint into the database.
POST	/req/getEventResult	Retrieve the result for the selected event.
Utilities		
Method	Route	Description
POST	/req/search	Retrieve the list of events that may fit the search string.

POST	/req/checkCCEMSAvail	Retrieve the count of the CCEMSID in the user table based on the payload data to check if the CCEMSID is available or has been assigned.
POST	/req/checkRunnerId	Retrieve the count of runner ID in the participant table based on the payload data to check if the runner ID is available or has been assigned.

6.3 Data Flow Management

6.3.1 Redux State Management

The mobile and web application for CCEMS were developed using React Native and React Js respectively. To facilitate smooth data flow or transfer between screens and pages, redux is used in the development of CCEMS. Redux is a state management tools that keeps the values of defined variables within redux store which can be retrieved on another components or pages. The main objective of using redux is to handle or pass the user and event data within the system without requiring the page to create a new request on every page load or page navigation.

In a simpler term, redux will centralize the data for the applications and create data flow between the redux store with the application containers. Data flow by passing props or states between containers or views is less implemented as it may be more flexible and efficient to allow the containers to subscribe to data directly from the redux store. The instances where redux's flexibility may come in handy is when user is trying to returning to the previous container with new information or updating multiple containers concurrently.

Besides that, the set of functions or actions can be defined using redux action, which can be called in any components of the application without redeclaring. Some async function call such as to create a request to the API gateway were create within redux action, as it can be used to directly update the redux store with the request results using the redux reducers.

The most appropriate use case of an async redux action is implemented in the web application's function to retrieve event data and dispatch the correct action according to the result of request. Redux reducer dispatched shall update the states within redux store, whether if it has failed or succeeded in fetching data from the API

gateway. The redux action will be called upon the page navigation and populate the data accordingly. The snippet for said redux action is shown as below:

```
// Action creator to dynamically dispatch different function based on scenario instead of returning action object.
export const retrieveData = () => async (dispatch, getState) => {
  const urlData = new URLSearchParams(location.search);
  const eventId = urlData.get('eventId');
  const res = await fetch('/req/eventData', {method: 'POST', body:JSON.stringify({'eventId':eventId}), headers: { "Content-Type": "application/json"}});
  const myData = await res.json();
  const eventDataLength = myData[0].length;

  // Dispatch for when HTTP request is not valid.
  if(res.status != 200){
    dispatch({
      type: RetrieveDataInvalid,
      payload: {status: res.status, statusText: res.statusText},
    });
  }
  // Dispatch for when there is no data on the event id.
  else if(eventDataLength !=1){
    dispatch({
      type: DataStructureInvalid,
      payload: {statusText: 'Event does not exist'}
    });
  }
  // All is well. Dispatch to update redux state
  else {
    dispatch({
      type: RetrieveData,
      payload: myData,
    });
  }
}
```

Figure 6.3.1.1: Snippet of Redux Action For Handling Retrieve Event Data In Web Application

Similar implementation of async redux action is implemented for the mobile application. In mobile application, the most appropriate use case of such definition would be the retrieval of event list information. This is because the event list is separated into three main categories: All Events, My Events and Assisting Events. The rendering of each event listing pages is rendered through tab navigation, however it may not be necessary for the application to perform a new fetch request every time the users switches between the tabs.

```
import {
  buildAsyncState,
  buildAsyncReducers,
  buildAsyncActions,
} from '@thecodingmachine/redux-toolkit-wrapper'
import FUEL from '@/Services/User/FetchUserEventList'

export default {
  initialState: buildAsyncState('FUEL'),
  action: buildAsyncActions('user/FUEL', FUEL),
  reducers: {...buildAsyncReducers({
    errorKey: 'FUEL.error',
    loadingKey: 'FUEL.loading',
  })},
  fulfilled: (state, {payload, type}) =>{
    // Once the data is fulfilled, update redux state data with the async action's payload data
    console.log('[Reducer: FetchUserEventList] User event list fetched with payload:');
    console.log(payload);
    state.FUELTank = payload
  }
}
```

Figure 6.4.1.2: Snippet For Async Redux State, Action And Reducer Definitions In Mobile Application

The code above indicates how the async redux is built for the mobile application. 'redux-toolkit-wrapper' library is used to compile the redux operations (store, action and reducers) together to allow easy building and dispatch. The library will create a dedicated async state declared as 'FUEL' to contain the error and loading key for the async redux operation. Action imported from the service directory will be called for the API data request and the reducer shall update the redux state accordingly when the result have been returned by the redux action. The action definition for API request in the mobile application is shown as below:

```
import api, { handleError } from '@/Services/ccemsIndex'

export default async (data) => {
  // Modify this to check if all parameters are present
  if (!data) {
    return handleError({ message: 'Data CCEMS ID is required' })
  }
  console.log('Data to FUEL\'s asyncaction');
  console.log(data);
  console.log('From FUEL service, category');
  console.log(data.category);

  const CCEMSID = data.CCEMSID;
  const category = data.category;
  const page = data.page;
  const order = data.order;
  const myData = {'CCEMSID': CCEMSID}

  const response = await api.post('req/eventList', {"CCEMSID":CCEMSID, "page":page, "order":order, "category":category})
  console.log('response done')
  console.log(response.data);
  return response.data
}
```

Figure 6.5.1.3: Snippet For Async Action Definition In Mobile Application

The code for async action above includes a simple API request call for retrieving the event list data according to the payload provided from the mobile screen. Once the request is successful, the response data is returned back to the previous slice for redux state update.

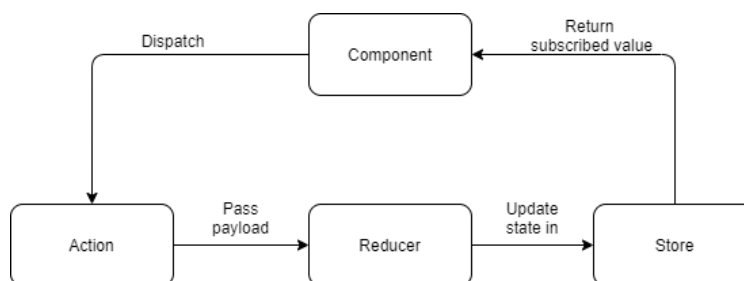


Figure 6.6.1.4: Diagram of Redux State Management Flow

The redux state management flow can be summarized as above. Component will dispatch action on event call. The action, whether async or not, will create and pass the payload for the reducer. Reducer is responsible for determining the difference between the previous redux state and new redux state. The reducer will update the

redux state in store with the payload information. As the store has been updated, the new changes will be returned to the subscribing components.

6.3.2 Redux State Management Dictionary

A list of the redux definition and their functionalities description will be covered below:

6.3.2.1 Web Application

Dissection of the column relationship and the assumption made can be defined as below:

- Location used for redux store indicates which container is currently subscribing to the state. Subscription is achieved using useSelector
- Location used for redux reducer indicates which redux action will call the reducer.
- Location used for redux action indicates which containers will dispatch the action.
- Redux components for web application are declared inside its respective component directory for separation of concerns.

6.3.2.1.1 App

Table 6.2.2.1.1.1: Table of Redux Components Created For The Global App Container

App			
Type	Name	Description	Location used
Store	globalState.userData: { CCEMSID:", displayName:", email:", registration_completed: 0,}	Contain the information of current user	EntryForm, Event, CreateEvent, ParticipantsList, RegisterParticipants, Home, UserSetup, ProfilePage,
Reducer	UserData	Updates the field in globalState.userData	signUpUser action, autoRetrieveUserData action, manualPushUserData action

Action	signUpUser	Create CCEMSID and create request for insertion of new user into the database. Uses UserData reducer to update the globalState.userData with new user information.	EntryForm
Action	autoRetrieveUserData	POST request to database to retrieve user data and dispatched via UserData reducer to update the globalState.userData with current session's user information.	ConditionalRouting
Action	manualPushUserData	Dispatch UserData reducer to update globalState.userData with payload from the page.	Login, SignUp
Reducer	SignOut	Updates the field in globalState.userData with empty value	SignOut action
Action	SignOut	Dispatch SignOut reducer.	NavBar

6.3.2.1.2 Create Event

Table 6.3.2.1.2.1 : Table of Redux Components Created For The CreateEvent Container

CreateEvent			
Type	Name	Description	Location used
Store	CreateEvent.currentPage: 1	Contains the state of current page value of event creation process.	CreateEvent, firstPage, secondPage, thirdPage
Reducer	NextPage	Increase the count of CreateEvent.currentPage by 1.	nextPage action
Reducer	PrevPage	Decrease the count of CreateEvent.currentPage by 1.	prevPage action
Action	nextPage	Dispatch NextPage action.	CreateEvent, firstPage, secondPage, thirdPage
Action	prevPage	Dispatch PrevPage action.	CreateEvent, firstPage, secondPage, thirdPage
Store	CreateEvent.firstPageData: { eventName:", schoolName:", eventLocation:", eventDate:", eventTime:", },	Contains the states of create event's first step.	CreateEvent, firstPage
Reducer	SaveFirstPage	Update CreateEvent.firstPageData with payload data.	saveFirstPage action

Action	saveFirstPage	Dispatch SaveFirstPage reducer with payload from the caller.	firstPage
Store	CreateEvent.secondPage Data: { totalCheckpoints:20, personnels:[], },	Contains the states of create event's second step.	CreateEvent, secondPage, thirdPage
Reducer	SaveSecondPage	Update CreateEvent.secondPage Data with payload data.	saveSecondPage action
Action	saveSecondPage	Dispatch SaveSecondPage reducer with payload from the caller.	secondPage
Store	CreateEvent.thirdPageData: {group:[]},	Contains the states of create event's third step.	CreateEvent, thirdPage
Reducer	SaveThirdPage	Update CreateEvent.thirdPageData with payload data.	saveThirdPage action
Action	saveThirdPage	Dispatch SaveThirdPage reducer with payload from the caller.	thirdPage
Reducer	DetachCreateEvent	Updates all CreateEvent redux state value to empty	detachCreateEvent action
Action	detachCreateEvent	Dispatch DetachCreateEvent. Used for clearing the CreateEvent redux state when leaving the create event page.	CreateEvent

6.3.2.1.3 Event

Table 6.4.2.1.3.1 : Table of Redux Components Created For The Event Container

Event			
Type	Name	Description	Location used
Store	Event.isValid: true	Contains the information of the validity of current event.	Event
Store	<pre> fetchedEventVariables: { eventDet: {}, personnelDet: [], groupDet: [{}, {}], removedPersonnelDet: [], removedGroupDet: [],},}; </pre>	Contains the information of the event and additional information for when editing the event.	Event
Reducer	RetrieveDataInvalid	Updates Event.isValid to false.	retrieveData action
Reducer	DataStructureInvalid	Updates Event.isValid to false.	retrieveData action
Reducer	RetrieveData	Updates Event.isValid and Event.fetchedEventVariables with the payload data.	retrieveData action
Action	retrieveData	Async function to retrieve data from the API gateway, and dispatch the reducer according to the result of the request. If the result is not successful with status code $\neq 200$, dispatch RetrieveDataInvalid. If the retrieved event details does not exact to 1 record,dispatch DataStructureInvalid. If all is well, dispatch	Event

		RetrieveData to update event details.	
Reducer	UpdatePersonnel	Updates event.fetchedEventVariables.personnelDet with payload of new personnel array objects.	updatePersonnel action
Action	updatePersonnel	Dispatch UpdatePersonnel reducer with payload containing new array objects of event personnel.	Event
Reducer	UpdateGroup	Updates event.fetchedEventVariables.groupDet with payload of new personnel array objects.	updateGroup action
Action	updateGroup	Dispatch UpdateGroup reducer with payload containing new array objects of event personnel.	Event
Reducer	AddRemoveGroup	Updates event.fetchedEventVariables.removedGroupDet with payload data.	addRemoveGroup action
Action	addRemoveGroup	Dispatch AddRemoveGroup reducer with payload containing information the group objects that have been removed during edit.	Event
Reducer	AddRemovePersonnel	Updates event.fetchedEventVariables	addRemovePersonnel action

		s.removedPersonnelDet with payload data.	
Action	addRemovePersonnel	Dispatch AddRemovePersonnel reducer with payload containing information the group objects that have been removed during edit.	Event
Reducer	DetachEvent	Updates Event state with empty values.	detachEvent action
Action	detachEvent	Dispatch DetachEvent. Used for clearing the Event redux state when leaving the event page.	Event

6.3.2.1.4 Register Participants

Table 6.5.2.1.4.1 : Table of Redux Components Created For The Register Participants Container

RegisterParticipants			
Type	Name	Description	Location used
Store	Event.isValid: true	Contains the information of the validity of current event.	RegisterParticipants
Store	fetchedEventVariables: { eventDet:{}, personnelDet:[], groupDet:[{}], removedPersonnelDet: [], removedGroupDet:[],	Contains the information of the event and additional information for when editing the event.	RegisterParticipants

	}, };		
Reducer	RetrieveDataInvalid	Updates Event.isValid to false.	retrieveData action
Reducer	DataStructureInvalid	Updates Event.isValid to false.	retrieveData action
Reducer	RetrieveData	Updates Event.isValid and Event.fetchedEventVariables with the payload data.	retrieveData action
Action	retrieveData	Async function to retrieve data from the API gateway, and dispatch the reducer according to the result of the request. If the result is not successful with status code $\neq 200$, dispatch RetrieveDataInvalid. If the retrieved event details does not exact to 1 record, dispatch DataStructureInvalid. If all is well, dispatch RetrieveData to update event details.	RegisterParticipants
Reducer	DetachEvent	Updates Event state with empty values.	detachEvent action
Action	detachEvent	Dispatch DetachEvent. Used for clearing the Event redux state when leaving the event page.	RegisterParticipants

6.3.2.2 Mobile Application

Redux components for the mobile application are declared in the global scope. It is observed that the mobile application utilizes redux less compared to the web application due to the defined navigation flow of the mobile system. It is less likely for a user in mobile application to skip the pages between the start point and the end point compared to web application. Redux state is maintained through single page application, meaning that data will be lost when the page is refreshed for web application. Mobile application redux state will only be lost when the user closes and restarts the mobile application. Therefore, prop passing in mobile application will have greater data integrity and is used more often compared to in the web application.

Most actions and reducers in mobile application are export default, therefore the naming of the redux component is declared during import in the index store. The store, action and reducer are bundled together under a wrapper for redux with similar objective. As such, the wrapper will then be built into slice using the redux-toolkit-wrapper library. The built slice will be available for dispatching and selecting in the containers. The implementation of the build slice functionalities on the imported wrapper can be defined as the snippets below:

```
import { buildSlice } from '@thecodingmachine/redux-toolkit-wrapper'
import EFED from './FetchEventData'
import PassEventID from './PassEventID'
import UpdateEnableEdit from './UpdateEnableEdit'
import UpdatePersonnel from './UpdatePersonnel'
import UpdateGroup from './UpdateGroup'

// This state is common to all the "event" module, and can be modified by any "event" reducers
const sliceInitialState = {
  item: {},
  passedEventId: '',
  eventDet: [],
  personnelDet: [],
  groupDet: [],
  enableEdit: false,
}

export default buildSlice('event', [EFED, PassEventID, UpdateEnableEdit, UpdatePersonnel, UpdateGroup], sliceInitialState).reducer
```

Figure 6.7.2.2.1: Snippet For The Slice Building of Event Redux Wrapper

```

import { buildSlice } from '@thecodingmachine/redux-toolkit-wrapper'
import FetchOne from './FetchOne'
import FUEL from './FetchUserEventList'
import FUD from './FetchUserData'
import FDAE from './FetchDashboardAssistingEvent'
import FDUE from './FetchDashboardUserEvent'
import CELC from './ChangeEventListCategory'

// This state is common to all the "user" module, and can be modified by any "user" reducers
const sliceInitialState = {
  FUELTank: 'initial',
  item: {},
  FDAE: {},
  FDUE: {},
  CCEMSID:'',
  displayName:'',
  email:'',
  registration_completed:'',
  EventListCategory: '',
}

export default buildSlice('user', [FetchOne, FUEL, FUD, FDAE, FDUE, CELC], sliceInitialState).reducer

```

Figure 6.8.2.2.2 : Snippet For The Slice Building of User Redux Wrapper

The built slice can be used in the container, such as utilizing selector to retrieve the redux state and using dispatch to run the redux action declared in the wrapper. An example of the usage can be seen as below:

```

const user = useSelector((state) => state.user);
const dispatch = useDispatch();
const [password, setPassword] = useState('');
const [email, setEmail] = useState('');

const LoginFunc = () => {
  console.log(password)
  console.log(email)
  if(password != null && email != null){
    auth()
    .signInWithEmailAndPassword(email, password)
    .then(() => {
      console.log(auth().currentUser);
      dispatch(FetchUserData.action(auth().currentUser.email));
      navigation.navigate('Home', null);
    })
    .catch(error => {
      manageError(error.code);
    });
  }
}

```

Figure 6.9.2.2.3: Snippet For Usage of Redux Slice In Login

The code above demonstrated that useSelector has been used to retrieve redux state named 'user', and dispatch for FetchUserData action with the payload has been made after a successful login. The dispatched action will update the mobile application's 'user' redux state.

6.3.2.2.1 App

The state management directory for mobile application can be defined as below:

App.js			
Type	Name	Description	Location used
Store	<pre>user: { FUELTank: 'initial', item: {}, FDAE: {}, FDUE: {}, CCEMSID: "", displayName: "", email: "", registration_completed: "", EventListCategory: ""},</pre>	The redux store for event is initialized when the mobile application ran.	EntryForm, Home, EventList, Event, CreateEvent, RecordParticipants
Store	<pre>event: { item: {}, passedEventId: "", eventDet: [], personnelDet: [], groupDet: [], enableEdit: false,}</pre>	The redux store for user is initialized when the mobile application ran.	Event, RecordParticipants
Store	<pre>export default FUD: { item: [], errorKey: null, loadingKey: null}</pre>	Stores async redux state information such the fetch status into errorKey and loadingKey, and request	Redux wrapper in /Store/User/FetchUserData
Store	<pre>export default FUEL: { item: [],</pre>	data into item array if no additional reducer is defined.	Redux wrapper in /Store/User/FetchUserEventList

	errorKey:null, loadingKey:null }		
Store	export default FDEA:{ item:[], errorKey:null, loadingKey:null }		Redux wrapper in /Store/User/FetchDashboardAssistingEvent
Store	export default FDUE:{ item:[], errorKey: null, loadingKey: null }		Redux wrapper in /Store/User/FetchDashboardUserEvent
Store	export default EFED:{ item:[], errorKey: null, loadingKey: null }		Redux wrapper in /Store/Event/FetchEventData
Action	<i>FUD / FetchUserData</i> or export default	Perform async request to retrieve user data using email, then dispatch the result into FUD reducer. Retrieved from /Service/User/FetchUserData.	Redux wrapper in /Store/User/FetchUserData
Reducer	<i>FUD / FetchUserData</i> or export default	Update the async redux state's errorkey and loadingkey, and redux state's user:{CCEMSID, displayName, email, registration_completed} with the payload. Retrieved from /Store/User/FetchUserData	Redux wrapper in /Store/User/FetchUserData

Action	<i>FUEL / FetchUserEventList or export default</i>	Perform async request to retrieve user event list data using CCEMSID, then dispatch the result into FUEL reducer. Retrieved from /Service/User/FetchUserEventList.	Redux wrapper in /Store/User/FetchUserEventList
Reducer	<i>FUEL/ FetchUserEventList or export default</i>	Update the async redux state's errorkey and loadingkey, and redux state's user.FUELTank with the payload consisting of array of event data objects. Retrieved from /Store/User/FetchUserData	Redux wrapper in /Store/User/FetchUserEventList
Action	<i>FDEA / FetchDashboardAssistingEvent or export default</i>	Perform async request to retrieve the list of assisting events using CCEMSID, and dispatch result into FDEA reducer. Retrieved from /Service/User/FetchDashboardAssistingEvent	Redux wrapper in /Store/User/FetchDashboardAssistingEvent
Reducer	<i>FDEA/ FetchDashboardAssistingEvent or export default</i>	Update the async redux state's errorkey and loadingkey, and redux state's user.FDEA with the payload consisting of array of event data objects. Retrieved from /Store/User/FetchDashboardAssistingEvent	Redux wrapper in /Store/User/FetchDashboardAssistingEvent

Acti on	<i>FDUE / FetchDashboard UserEvent or export default</i>	Peform async request to retrieve the list of user's own created events using CCEMSID, and dispatch result into FDUE reducer. Retrieved from /Service/User/FetchDashboardUserEvent	Redux wrapper in /Store/User/FetchDashboardUserEvent
Red ucer	<i>FDUE/ FetchDashboard UserEvent or export default</i>	Update the async redux state's errorkey and loadingkey, and redux state's user.FDUE with the payload consisting of array of event data objects. Retrieved from /Store/User/FetchUserData	Redux wrapper in /Store/User/FetchDashboardUserEvent
Acti on	<i>EFED / FetchEventData or export default</i>	Peform async request to retrieve the event data using event id, and dispatch result into EFED reducer. Retrieved from /Service/Event/FetchEventData	Redux wrapper in /Store/Event/FetchEventData
Red ucer	<i>EFED / FetchEventData or export default</i>	Update the async redux state's errorkey and loadingkey, and redux state's event's eventDet, groupDet, and personnelDet with the payload consisting object of event variables. Retrieved from /Store/User/FetchUserData	Redux wrapper in /Store/Event/FetchEventData

Action	UpdateGroup	Create a reference action to dispatch the wrapper's reducers for updating the groupDet redux state.	Redux wrapper in /Store/Event/UpdateGroup
Reducer	UpdateGroup	Update the event.groupDet redux state with the payload data.	Redux wrapper in /Store/Event/UpdateGroup
Action	UpdatePersonnel	Create a reference action to dispatch the wrapper's reducers for updating the personnelDet redux state.	Redux wrapper in /Store/Event/UpdatePersonnel
Reducer	UpdatePersonnel	Update the event.personnelDet redux state with the payload data.	Redux wrapper in /Store/Event/UpdatePersonnel

6.3.3 Route Data Flow

As mentioned in 6.3.1 Redux State Management, the usage of props passing via navigation or route is rarely implemented. For web application, all state management is done using redux. For mobile application, there is only several instances where passing props using navigation params is implemented. Majority of the state and data management are handled by a central redux store. The route data flow for the developed mobile application can be seen as below:

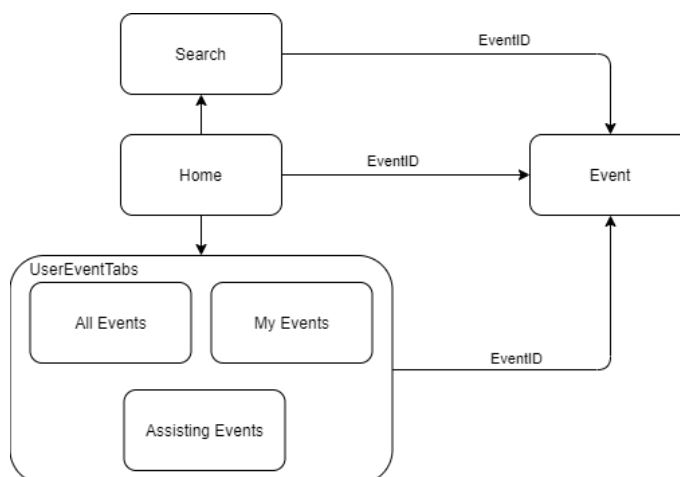


Figure 6.10.3.1: Dataflow Diagram for the Mobile Application

CHAPTER 7

Testing and Evaluation

7.1 Test Plan

7.1.1 Purpose

This section entails the strategy proposed and implemented to undergo the necessary investigation and testing on the modules of the developed cross country event management system. This is to ensure that all required functionalities have been properly implemented and are up to the required standards. The planning should correlate to the primary objectives of the system's creation with the intention to find and rectify potential defects or deficiency of the system. Below includes the objectives for this test plan:

- To identify the features to be tested and not to be tested with justification.
- To describe the testing strategies employed for the testing process.
- To express on the expected deliverable from testing

7.1.2 Scope

Testing scope is defined as the extent of functionalities or features that will be tested in this development life cycle. As the system to be developed encompasses a “full stack” development, i.e. both front end and back end of the web and mobile application has to be developed, the coverage of features to be tested will be in regard of the objectives of this project. Core functionalities such as the event management module, will be the representative for the success of this, and will be greatly focused on during the testing phase. Therefore, features that are primarily aesthetic and contribute insignificant deal to the core functionality of the system will not be emphasized, or will be disregarded temporarily for this testing cycle. The next few sections should include a detailed elaboration on the list of features to be tested and not to be tested for this testing cycle.

7.1.3 Test Items

The objective of this project is to create a management system for handling cross country events and to develop the necessary tools for tracking participants and finalizing the result of the event. Therefore, the core modules to be tested will be the

event management modules, register participants modules and record participants modules. Functionalities which fall under the modules will be the features to be tested. Functionalities from remaining modules of account management and search will be placed at lower test priority or be included in the features not to be tested. Features not to be tested will also include the non-functional requirements elicited which are less feasible for testing. However, exceptions will be made to functionalities such as login and register of the remaining module as they play a significant role in authenticating the user for the usage of the system.

7.1.4 Features To Be Tested

The features to be tested as placed from highest priority or risk to the lowest priority of risk:

- **Create new event (High risk)**
Create new event is the core functionality of this project to create an event record that is used for managing the process of a cross country event. Testing involved for this functionality are verifying that all fields (items include event basic details, event participant group, and event assigned personnel) can be filled and handled properly, navigation to the next page or container is possible, and the creation process can be completed with success of the created event showing up under the user's event list.
- **Update event (High risk)**
Update event must be properly implemented to allow the event organizer to make modification to the event details, event group details and the list of assigned assisting personnel for the event. Testing done must verify that all fields can be updated, and the new updated information will show up properly within the event page.
- **Register participants (High risk)**
Participant must be registered to allow proper QR scanning and event result generation. Testing must ensure that the register participants fields can be filled, the generation of the participant QR code is possible, and the QR image can be downloaded.
- **Record participants (High risk)**

Record participants are done during an ongoing event to log the time of a participant at a checkpoint. Testing must ensure that the mobile application is able to utilize the android camera to perform QR scanning, and the detection of the participant using QR scanning with the correct timestamp can be verified. Additionally, the manual record method must be tested to ensure the data has been logged successfully.

- Start event and end event (High risk)

Start event and end event decides if the ongoing event functionalities will be available for usage. Testing must be done to ensure that an upcoming event can be started by the event organizer, following with the enabling of all ongoing event features. The ongoing event must be verified that it can be ended.

- Preview result (Medium risk)

Result generation and result preview functionality must be implemented in order to properly mark the closure of a cross country event. Testing must be done to ensure that the result generation is completed after the event ends, and the result should be displayed on the event page.

- Login (Medium risk)

Users must login into the system to use all of the mobile application functionalities and most of the web application functionalities. Testing must be done to ensure that login through the web and mobile application is possible.

- Register (Medium risk)

Users must be able to register an account to use this system. Testing must be done to ensure that the registration of new users can be performed with success and that the user should be able to login into the registered account in the future.

- Search (Low risk)

All users must be able to search events to view the details of the event. If the event is completed, the users must be able to view the event in the search list.

7.1.5 Features Not To Be Tested

- Update account (Low risk)

Update account functionality is performed to change the user's display name and the user's password. Testing is deferred for the update account functionality as the display name updated is mainly cosmetic. Update password

functionality will be more appropriate if it was to reset password for users who have forgotten their password.

- **Sorting (Low risk)**

Sorting is performed on the event list and event result to organize the data into the criteria that is desired. No testing is done on this feature as it can be considered as a cosmetic feature.

7.1.6 Test Approach and Strategy

Black box testing method for functional testing is employed for this project. Black box testing strategy is performed through manual testing, a testing method that examines the result of the system through the viewpoint of an end user. This testing method will not be examining the code written to evaluate the test result, but instead will be evaluating based on the type of output produced using a defined set of inputs. The testing will be considered as successful if the system correctly produces the desired output using the set of given inputs.

The testing level to be covered will be of the unit test, integration test and system test and finally the user acceptance test (UAT). Unit testing is employed for examining the functionality of a single and isolated feature, component or module. Integration testing is performed to ensure that the interaction between two or more components or modules are as defined. System testing will evaluate if all components are able to work together and produce the desired process flow for the system. UAT is performed with users falling under the targeted demographic of the system to determine if the system is in compliance with handling real-world scenarios.

On a similar notion, usability testing is done with the same UAT audience to determine the system's ease of use and navigation. Code review has also been done using CodeBeat to measure the technical prowess of the system developed. All testing will be detailed in their respective section.

7.1.7 Item Pass Fail Criteria

Criteria is defined to establish the condition where a module testing would be considered as 'pass' or 'fail'. The criteria are as below:

Table 7.1.7.1: Table of Test Pass / Fail Criteria for Each Modules

Module name	Pass Criteria	Fail criteria
Event management module	<ol style="list-style-type: none"> 1. Users can enter the create event page. 2. Users can navigate between the event page of the web application. 3. Users can fill the fields available on the event page. 4. Events can be successfully created and shown in the user event list. 5. Event page displays the correct event information 6. Event page should recognize the role of the users for the event, and display the authorized functionalities. 7. Event page should recognize the status of the event and display the appropriate functionalities. 8. Event page should recognize a completed event and display the result list. 9. Event organizers should be able to update their own event with success. 10. Event organizers should be able to delete their own event. 11. Users should be able to view their own event list. 12. Users should be able to sort their event list with 'My events', 'Assisting events' and 'All events' category. 13. Event organizers should be able to start and end an event. 14. Entering an invalid event page should show the 	<ol style="list-style-type: none"> 1. The create event page is inaccessible. 2. User is unable to navigate between the event page of the web application despite having no field with error. 3. The form field cannot be updated. 4. Events cannot be created or are not shown at all. 5. Event page retrieves the wrong event details, personnel details or group details. 6. Functionalities are displayed despite lack of authority or mismatching event status. 7. Event cannot be deleted successfully. 8. Event cannot be updated successfully. 9. Event list and dashboard display empty list despite knowing event has been created. 10. Event cannot be started or ended.

	appropriate error message.	
Register participants module	<ol style="list-style-type: none"> 1. Event organizers and event personnel should be able to view the participants list. 2. Event organizers and event personnel should be able to add new participants for the event. 3. Event organizers and event personnel should be able to delete participants from the participants list. 4. Event organizers and event personnel should be able to view the QR code of event participants in the participants list. 5. The module should be able to display the QR code after the participants are registered. 6. The module should allow the download of QR code as an image file. 	<ol style="list-style-type: none"> 1. Event organizers or event personnel cannot view the participants list of an incomplete event. 2. Register participant functionality did not properly register a participant. 3. The generate QR code functionality during registration and in participants list does not work. 4. The QR image cannot be downloaded.
Record participants module	<ol style="list-style-type: none"> 1. Event organizers and event personnel should only be able to enter the record page when the event status is 'ONGOING'. 2. Event organizers and event personnel should be able to successfully record a participant. 3. Scanning should detect the QR on the runner bib and generate a pop out 'snackbar' containing runner ID, checkpoint number, and the 	<ol style="list-style-type: none"> 1. All event user were able to access the record participant screen. 2. The system could not recognize a new participant on a new checkpoint during scanning. 3. No add participant record pop out is shown after successful scanning of a new participant of a new checkpoint. 4. Manual entry recording did not show pop out 'snackbar' that the

	<p>timestamp of the record instance.</p> <ol style="list-style-type: none"> 4. Manual entry recording form should be able to be filled. 5. User must be able to toggle between auto-time and self-filled time. 6. Manual entry recording should display a pop out 'snackbar' on successful logging. 7. Users should be able to change the current checkpoint for QR scanning. 8. Users' device should not display a pop out 'snackbar' when rescanning the same runner that has already been scanned at the same checkpoint. 	<p>recording was successful.</p> <ol style="list-style-type: none"> 5. All fields and options cannot be updated or toggled in the recording page.
Account management module	<ol style="list-style-type: none"> 1. The user will be able to register an account using a web and mobile application. 2. The user will be able to login into an account from a web and mobile application. 3. The appropriate error message will be shown during empty field. 	<ol style="list-style-type: none"> 1. Registration could not be completed despite meeting the field rules. 2. User cannot login into their account despite entering the correct credential. 3. No error message or clarity were shown for bad entry during login and registration.
Search module	<ol style="list-style-type: none"> 1. The correct events list will be shown using the search string. 	<ol style="list-style-type: none"> 1. The event list shown does not match or resembles any similarity.

7.1.8 Test Deliverable

Test deliverable will be included in the corresponding sections of the items. The list of expected test deliverables is:

- Test cases and test results
- User Acceptance Test Result
- System Usability Test Result

- Static Code Review

7.1.9 Entry Criteria

- All software tools have been properly installed and can function properly.
- Hardware functionality for mobile applications have been properly configured for the APK.
- All features to be tested have been developed.
- Expected test cases and test data are ready.

7.1.10 Exit Criteria

- All high severity bugs have been rectified.
- All test cases have been covered.
- At least 90% of the unit test case should pass.
- At least 95% of the integration and system test case should pass.

7.2 Manual Testing

Manual testing or known as black box testing is the method employed for performing investigation and testing on the system for this project. This method will be examining the functionalities from an end user's viewpoint, and therefore will not involve examination and evaluation on the system code. Four level of testing are done using manual testing.

7.2.1 Unit Test

Unit testing is performed on each of the modules, relevant components and features that are responsible in achieving the objective of the project. Only features to be tested as listed in section 7.1.4 will be included in the unit test, with high priority modules or features to have a more detailed testing. Both web and mobile application will be tested separately, however the tabulation of test cases will be merged if the test execution share high similarity.

7.2.1.1 Account Management Module

Table 7.1.1.1.1: Account Management Module Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Empty register field validation.	1.Register is clicked with all empty field.	Error message is displayed.	Pass	Pass
Email format validation	1.Register with email in non-email format.	Error message under the field is displayed.	Pass	Pass
Confirm Password Validation	1.Enter the mismatch confirmed password.	Error message is displayed under the field.	Pass	Pass
Password format validation.	1.Register with password in non-conformance to the required format.	Error message is displayed under the field.	Pass	Pass
Successful Register	1.Register with all field under correct format.	User is redirected to registered user's set display name page.	Pass	Pass
Empty login field validation.	1.Login is clicked with all empty field.	Error message is displayed.	Pass	Pass
Password checking	1.Login with the wrong password.	Error message is displayed that say either email or password is wrong.	Pass	Pass
Email checking	1.Login with the wrong email.	Error message is displayed that say either email or password is wrong	Pass	Pass
Successful login	1.Login with correct credential.	User is redirected to registered user home page.	Pass	Pass
Logout	1.Click on user's name on navbar. 2.Click on logout.	User is successfully logged out and redirected to visitor page.	Pass	Pass

7.2.1.2 Event Management Module

Table 7.2.1.2.1: Create Event Component Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Empty field validation	1.Click on next page with all field left empty.	Error message is displayed on all the unfilled field.	Pass	Pass
Past date validation.	1.Picks date that are dated before today's date. 2.Click on next page.	Error message is displayed.	Pass	Pass
Complete First Page	1.Fill all first page field with correct information and format 2.Click on next page.	Second page of create event is shown.	Pass	N/A
Add new personnel	1.Any letter is entered into the personnel CCEMS ID field. 2.Add personnel button is clicked.	The personnel is registered for the event.	Pass	Pass
Personnel removal	1.Add new personnel. 2.Click the remove button.	The personnel is removed from the list.	Pass	Pass
Complete second page	1.Next page button is clicked.	Third page of the create event is shown.	Pass	N/A
Test back button	1.Click on back button on the second page and third page of the create event page.	User is redirect to one steps behind. All information are maintained in both previous page and current page.	Pass	N/A
Add new group	1.Select any checkpoints. 2.Enter any name for the group. 3.Click add group.	The new group with checkpoint is added to the list below.	Pass	Pass
Remove group	1.Add new group 2.Click on the remove button for the group.	The group is removed from the list.	Pass	Pass
Update checkpoint with group	1.Add a new group with 10 checkpoints.	The group with exceeding checkpoint count is removed.	Pass	Pass

	2.Return to second steps, and set maximum event checkpoints to 5. 3.Go to third step.			
Create event	1.Complete all information for all three steps. 2.Click on the finish button. 3.Click save on confirmation alert.	An alert message is shown when the finish button is clicked. User will be redirected to homepage when the save button of alert box is clicked.	Pass	Pass

Table 7.3.1.2.2: Update Event Feature Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Validate edit button	1.Enter event page of an event created by the account. 2.Click on edit button of event page.	All event information turned into editable field. The functionalities on top of the page is the 'Save Change' button.	Pass	Pass
Empty field validation.	1.Enter edit mode in event page. 2.Click on save changes with empty field on the page.	A dialog box pops out informing the user the changes cannot be saved.	Pass	Pass
Delete event	1.Enter an event page created by the account. 2.Click on the delete button.	The event is deleted and cannot be found in the search or event list.	Pass	Pass
Validate save change	1.Enter edit mode in event page. 2.Update all field of the event with new information. 3.Remove existing groups and personnel. 4.Add new groups and personnel. 5. Click on save changes.	The new information is displayed on the event page after a page reload.	Pass	Pass

Table 7.4.1.2.3: Event Page Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
User view for upcoming event	1.Enter an upcoming event as a non-event-related user.	The event details, personnel details and group details are displayed.	Pass	Pass
Personnel view for upcoming event	1.Enter an upcoming event in which the user is assigned as personnel.	The event details, personnel details, group details, register participants button and participants list button are displayed.	Pass	Pass
Event organizer view for upcoming event.	1.Enter an upcoming event that was created using the account.	The event details, personnel details, group details, and the buttons for edit, delete, register participant and participant list are displayed.	Pass	Pass
Completed event page view	1.Enter any completed event as a user. 2.Enter another completed event as assigned personnel. 3.Enter another completed event as the event organizer.	All three roles will observe the save structure of page. The event result is available.	Pass	Pass
Test record participant button as event organizer.	1.Enter an ongoing event that was created using the account. 2.Locate and click on the record participant button.	The user is redirected to the record participant screen.	N/A	Pass
Test record participant button as event personnel.	1.Enter an ongoing event that the user was assigned as personnel. 2.Locate and click on the record participant button.	The user is redirected to the record participant screen.	N/A	Pass

Table 7.5.1.2.4: Register Participant Feature Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Empty field validation.	1.Click on the register button with all field left empty.	The register button cannot be clicked.	Pass	N/A
Empty runner group validation.	1.Enter a register participant page with no group added for the event. 2.Fill all field. 3.Click on the register button.	The register button cannot be clicked.	Pass	N/A
Empty runner ID validation.	1.Fill all field except for runner id. 2.Click on the register button.	The register button cannot be clicked.	Pass	N/A
Successful register participant.	1.Select the runner group. 2.Select the sports house. 3.Fill the runner id prefix. 4.Generate a runner ID. 5.Click on register participants button.	The participant has been registered with QR code shown below.	Pass	N/A

7.2.1.3 Record Participant Module

Table 7.6.1.3.1: Record Participant Module Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Manual record with empty field validation.	1.Click on the record button with all field left empty.	Nothing happens.	N/A	Pass
Manual record with invalid runner ID.	1.Enter a random string for the runner ID field. 2.Click on the record button.	Nothing happens.	N/A	Pass
Successful Manual Record	1.Fill all field for manual entry.	A 'snackbar' with success message is displayed.	N/A	Pass

	2.Click on the record button.			
Invalid QR code scan	1.Scan a QR image that is not created from the register participant.	Nothing happens.	N/A	Pass
Successful QR Scan	1.Scan a runner QR that has not been scanned for the checkpoint. 2.Click on 'Record' located on top of the pop out 'snackbar'	A 'snackbar' with the runner id, checkpoint and time stamp is shown, together with option to Record and Undo on the 'snackbar'. A new 'snackbar' informing that the user has been added after 'Record' is clicked.	N/A	Pass
Rescan of QR at same checkpoint.	1.Scan a QR that has already been scanned with the same checkpoint.	Nothing happens.	N/A	Pass

7.2.1.4 Search Module

Table 7.7.1.4.1: Search Module Unit Test Cases

Test case	Test Execution	Expected Result	Web App Pass/Fail	Mobile App Pass/Fail
Search All	1. Click on the search button without any string.	All events from the system will be displayed in a list.	Pass	Pass
Search event	1. Enter any string on the search field and click on the search button.	A list of events with relevance to the string is shown.	Pass	Pass

7.2.2 Integration Test

Integration testing is conducted between several modules to confirm that the functionalities that are implemented in each individual modules or components are able to communicate with or influence with another modules. The confirmation usually involves verifying that certain functionalities previously not available on a different module will be available upon changes made to one of the modules.

7.2.2.1 Event Module and Register Participant Module

Table 7.8.2.1.1: Event Module and Register Participant Module Integration Test

Test case	Update event group and register participant
Test Procedure	<ol style="list-style-type: none"> 1. Enter the event page as an event organizer. 2. Enter edit mode. 3. Add new participant group. 4. Save changes. 5. Enter the register participant page. 6. Select the newly added group.
Expected Result	The new group and existing groups are properly shown on the register participant module.
Pass/Fail	Pass

7.2.2.2 Search Module and Event Management Module

Table 7.9.2.2.1: Search Module and Event Module Integration Test

Test Case	State pass verification between event list and event page
Test Procedure	<ol style="list-style-type: none"> 1. Enter any search string and click on the search button. 2. Click on any event from the event list.
Expected Result	User is redirected to the selected event.
Pass/Fail	Pass

7.2.2.3 Register Participants Module and Record Participants Module

Table 7.10.2.3.1: Register Participant and Record Participant Integration Test

Test Case	Verify registered participants can be recorded during event
Test Procedure	<ol style="list-style-type: none"> 1. Enter the event registration page. 2. Register a new participant. 3. Saves the QR code. 4. Start the same event where the participant is registered in. 5. Enter the recording screen. 6. Scan the QR code. 7. Confirm Record the participant

Expected Result	The generated QR of the participants of the event can be scanned and recorded using the mobile application in an ongoing event.
Pass/Fail	Pass

7.2.2.4 Event Management Module and Record Participants Module

Table 7.11.2.4.1: Event Management and Record Participant Integration Test

Test Case	Verify recorded participants will be generated in the finalized participants ranking list.
Test Procedure	<ol style="list-style-type: none"> 1. Register a participant for an event. 2. Start the same event where the participant is registered in. 3. Record the participant at all required checkpoint. 4. Return to event page. 5. End the event. 6. Refresh the event page. 7. Locate the participant in the result ranking list.
Expected Result	The participant's runner ID should show up in the result ranking list in the completed event.
Pass/Fail	Pass

7.2.2.5 Account Management Module and Record Participants Module

Table 7.12.2.5.1: Account Management and Record Participant Integration Test

Test Case	Verify assigned personnel will be able to record participants in an ongoing event.
Test Procedure	<ol style="list-style-type: none"> 1. Create a sub-account. 2. Copy the CCEMSID. 3. Log out of sub-account and login into the main account. 4. Create a new event and add the sub-account as event personnel by its CCEMSID. 5. Login into main account in the mobile application. 6. Locate the same event and start the event. 7. Log in as sub-account using mobile application. 8. Locate and enter the same event.

	9. Click on record participant.
Expected Result	Account recognized as assisting personnel for the event will be able to enter the record participant screen of an ongoing event.
Pass/Fail	Pass

7.2.3 System Test

System test is conducted on the web and mobile application to ensure that the system can be run similarly to how the process flow is defined.

7.2.3.1 Event Preparation

Table 7.13.3.1.1: Event Preparation Flow System Test

Test Case	Full process of preparing an upcoming event.
Test Procedure	<ol style="list-style-type: none"> 1. Register and login into an account from the web application. 2. Enter the create an event page. 3. Complete all fields with required information. 4. Add in new groups and personnel for the event. 5. Complete the creation process. 6. Locate and enter the newly created event in event list. 7. Click on register participants. 8. Fill in the required fields and register a new participant. 9. Save the QR code of the participants. 10. Enter the participant list from the event page. 11. Logout.
Expected Result	<ol style="list-style-type: none"> 1. Register and login can be completed 2. Newly created event is present in My Event list. 3. Status of the event is 'UPCOMING'. 4. Registered participant can be located in the participant list. 5. Logout is successful.
Pass/Fail	Pass

7.2.3.2 Event Progression

Table 7.14.3.2.1: Event Progression Flow System Test

Test Case	Full process on event day operation
Test Procedure	<ol style="list-style-type: none"> 1. Login into an account with upcoming event using the mobile application. 2. Locate and enter the event page. 3. Click on the start event button. 4. Click on the record participant button. 5. Record participants by scanning and manual entry. 6. Return to event page. 7. End the ongoing event. 8. Refresh the screen. 9. Locate and find the scanned participants.
Pre-condition	<ol style="list-style-type: none"> 1. An event has already been created using the account. 2. Participants have been registered for the event using web application.
Expected Result	<ol style="list-style-type: none"> 1. Login for mobile application can be completed. 2. Event can be started. 3. Record participant screen can be entered by the event organizer. 4. QR scanning and manual entry can be completed successfully as indicated by a 'success snackbar' notification. 5. Event can be ended. 6. Correct result list is generated.
Pass/Fail	Pass

7.2.3.3 Checkpoint Guarding

Table 7.15.3.3.1: Event Progression Flow System Test

Test Case	Full process on event day operation
Test Procedure	<ol style="list-style-type: none"> 1. Login into an account using the mobile application. 2. Locate and enter the ongoing event. 3. Click on the record participant button. 4. Record participants by scanning and manual entry.

Pre-condition	<ol style="list-style-type: none"> 1. The account has been added as assisting personnel for the event. 2. The event has been started by the event organizer.
Expected Result	<ol style="list-style-type: none"> 1. Login for mobile application can be completed. 2. Record participant screen can be entered by the event personnel. 3. QR scanning and manual entry is successfully as indicated by a 'success snackbar' notification.
Pass/Fail	Pass

7.3 User Acceptance Test

User Acceptance Test (UAT) has been conducted with total of 7 participants consisting of 2 secondary school teachers and 5 secondary school students. The UAT scenario designed is based on the core functionalities and adapted from system testing. It should be noted that functionalities tested in the web application will not be repeated for mobile application, and some scenarios have to be trimmed as to ensure that the participants are able to complete all the UAT scenarios within 1 hour or within their available time. The UATs are conducted on both face-to-face mode and digital mode using Zoom with share control. Most participants have verified that the functionalities of the system are working, and did not face much issue navigating through the site. Comments were given by the participants on how the system can be further improved. Full sheets can be located in appendix D. The statistic for the UAT can be summarized as below:

Table 7.16.1: Test Result of User Acceptance Test

Web Application		
Test Module	Total passed	Total conducted
Register and Login	7	7
Event creation	7	7
Event Update	7	7
Participants Registration	7	7
Participants Management	7	7
Event Search	7	7
View Completed Event	7	7

Mobile Application		
Test Module	Total passed	Total conducted
Login	7	7
Event Start	7	7
Event Recording	7	7
Event End	7	7

7.4 System Usability Test

System Usability Test has been conducted with the same audience of UAT after the completion of the previous test. The testing is done by completing a survey form to express their opinion on the overall usefulness, ease of use, ease of learning and satisfaction. The survey created for the System Usability Test is adapted from Arnold M. Lund (2001)'s USE questionnaire, which consists of a total 30 questions group by 4 factors aforementioned. The scale ranges from strongly disagree of 1 to strongly agree of 7. The completed result of the system usability test can be located at appendix D.

Table 7.17.1: Result of System Usability Test

Tester	Factors (Scale of 1 to 7)			
	Usefulness	Ease of Use	Ease of Learning	Satisfaction
1	5.125	6.091	7	5.5
2	5.125	5.182	6	4.5
3	4.75	5.636	6.5	5.168
4	6.375	6.182	6.25	6.168
5	5.375	4.812	5.5	4.5
6	5.375	5.091	4.5	5.167
7	6.125	6.091	6	6
Average	5.464	5.584	5.964	5.367

The result indicated that the system scored highest in ease of learning due to the simplicity of navigation and clear instruction text. The system scored lowest in satisfaction as the system generally does not give much enriching experience or pushes the extent of usage outside the purpose of completing the tasks. The system can be

improved on the satisfaction aspect if the design of the application were to include more functionalities, events or promotion that would provide a sense of fulfilment for the users during usage, which was not fully achieved as it was yet to be defined in the project scope.

7.5 Static Code Review

Both web and mobile application code has been submitted to CodeBeat for a static code review. The review attempts to analyse possible vulnerability of the written code and ensure that it adheres to industry standard. CodeBeat's scoring system is GPA based with the best grading being 4.0. Penalty is applied for poor decomposition or large code, duplicated functions, and style violation. The full result can be seen in appendix C. The result of the review is shown as below:

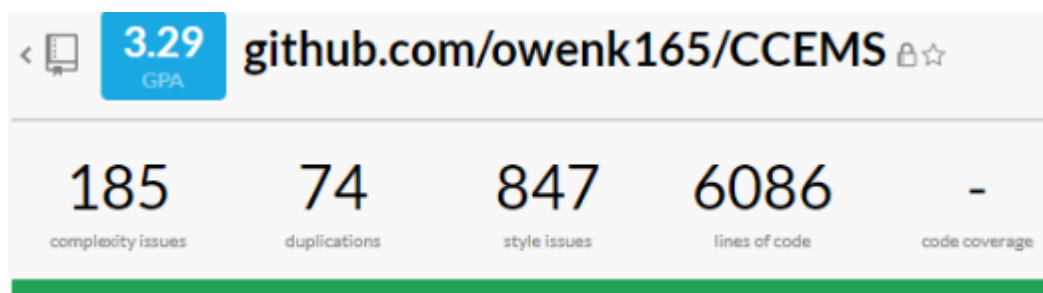


Figure 7.1.1: Review Result For Web Application

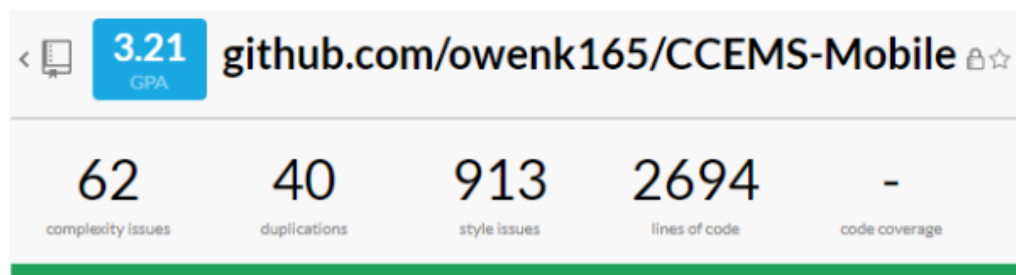


Figure 7.2.2: Review Result For Mobile Application

7.6 Test Summary

Table 7.18.1: Summary Of Manual Testing

Testing Level	Web	Mobile	Total
Unit Testing	33/33	34/34	67/67
Integration Testing	-	-	5/5
System Testing	-	-	3/3

The web and mobile applications developed have passed all the test cases defined for the manual testing process. There is no significant bug elicited during the manual testing process and all the functionality have been confirmed to be working as intended. Integration testing and system testing done to gauge the system's readiness for actual usage have also been passed successfully with no major issue.

The system tests have been converted into test scenarios for the User Acceptance Testing (UAT), of which the result yielded a 100% pass rate. Most participants involved in the UAT are able to complete the assigned tasks based on the steps given without major holdback. Most participants commented that the operation of the system is clear and smooth. Some participants commented that functionalities such as QR scanning and web application's event management page can be improved to be faster, more aesthetically pleasing, and include more functionalities. System Usability Test done by the same group of participants yielded in highest score of 5.964 out of 7 for Ease of Learning aspect due to the simplicity and clear navigation of the system, and lowest score of 5.367 out of 7 for Satisfaction as the application seems to be designed for promoting specific practical usage and is not designed for promoting casual usage.

CHAPTER 8

Conclusion and Discussion

8.1 Conclusion

The system developed is believed to have achieved the goals of facilitating the management and the process flow of a Cross Country Event for secondary school users. By implementing QR scanning technology in the mobile application, the recording of running participants can now be easily logged at any checkpoint. The scanning method will require less effort from the event personnel for data entry of passing participants compared to the traditional manual hand logging method. The utilization of QR scanning using the mobile application during event can contribute to less delay of participants logging at a high traffic checkpoints and therefore improving recording efficiency and effectiveness. The system developed may resolve the issue of high-cost expenditure from employing industry level tracking equipment by utilizing the personnel's personal android devices for the recording of the runners under this system. The cost leverage should reason the difference in tracking accuracy and efficiency between industry level equipment and consumer standard devices. Event result and ranking generation can be triggered upon the ending of any event under this system.

The architecture of the system developed is similar to a flux architecture for the web and mobile application, and a client-server architecture for the communication between various endpoints. In overall, exchange of information via internet can be achieved given that it has connection to the API gateway server. Both mobile and web application are able to display the necessary event management functionalities and dispatch the correct request for data retrieval.

8.2 Limitation

Some limitations are discovered during and after the development of this system:

1. Some parts of the modules and functionalities were written with poor optimization due to poor initial planning and messy structuring. The affected modules are yet to be fully revisited after the completion of the

- system as majority of the efforts at close to the end of the project lifecycle was invested in garnishing the interface and completing the documentation.
2. The API gateway server developed has a weak security measures in filtering unauthorized requests. The API gateway is currently not protected with authorization from the web and mobile system. Therefore, it is susceptible to data leakage if other people were to connect to the gateway through any application other than the intended web and mobile system.
 3. There is difficulty in searching for willing participants that falls under the category of stakeholder from secondary school for the research and system testing due to pandemic movement control. Therefore, the sample size used for the user acceptance testing and system usability may be limited.

8.3 Recommendation

There are several recommendations that can be used to further improve the developed system:

1. Modules not included in this report such as the proper Gmail binding, participants list or recording list import or upload functions can be implemented to facilitate further actions.
2. The web hosting and API server can be migrated to a cloud server for better up time, given that the security concerns for the implementation will be properly addressed beforehand. API gateway can be hosted using AWS Lambda in the future for better availability and backend management organization with the RDS used.
3. Participant registration can be implemented in the mobile application to allow full event management operation purely through mobile application.
4. The mobile application can benefit from a periodic update during event progression to update the list of recorded participants when in the recording screen.
5. Improve the mobile and web application styling for better visual clarity.

REFERENCES

- Geambasu, C.V., Jianu, I., Ionel, J. and Gavrilă, A., 2011. Influence Factors for the Choice of a Software Development Methodology. *Contabilitate și Informatică de Gestiune*, [online] 10(4), pp.479–494. Available at: <https://www.researchgate.net/publication/227367934_Influence_Factors_for_the_Choice_of_a_Software_Development_Methodology> [Accessed 10 Jul. 2020]
- Kementerian Pelajaran Malaysia., 2011. *Buku Panduan Pelaksanaan Dasar Satu Murid Satu Sukan (IMIS)*. [online] *Bahagian Sukan Kementerian Pelajaran Malaysia*. Available at: <<https://www.moe.gov.my/dasar/1197-buku-panduan-1murid-1sukan/file>> [Accessed 9 Jul. 2020].
- Lee, D.C., Pate, R.R., Lavie, C.J., Sui, X., Church, T.S., Blair, S.N., 2014. Leisure-time running reduces all-cause and cardiovascular mortality risk. *Journal of American College of Cardiology*, [online]. 64(5), pp 472–481. Available at: <<https://dx.doi.org/10.1016/j.jacc.2014.04.058>> [Accessed 9 Jul. 2020]
- LucidChart.com n.d., 4 Phases of Rapid Application Development Methodology, accessed 14 July 2020, available at: <<https://www.lucidchart.com/blog/rapid-application-development-methodology>>
- Anon 2020. *Difference Between Prototype Model and RAD Model (Tabular Form)*. [online] Available at: <<https://techdifferences.net/difference-between-prototype-model-and-rad-model/>> [Accessed 23 Jul. 2020].
- Axelsson, O. and Carlström, F., 2016. Evaluation Targeting React Native in Comparison to Native Mobile Development. [online] p.62. Available at: <<http://lup.lub.lu.se/luur/download?func=downloadFile&recordId=8886469&fileId=8886473>>.
- Khedkar, S., Thube, S., Estate, W.I., W, N.M. and Naka, C., 2017. Real Time Databases for Applications. *International Research Journal of Engineering and Technology (IRJET)*, [online] 4(6). Available at: <<https://irjet.net/archives/V4/i6/IRJET-V4I6401.pdf>>.
- Mohammadi Kho'i, F. and Jahid, J., 2016. Comparing Native and Hybrid Applications with focus on Features. [online] Available at: <<https://www.diva-portal.org/smash/get/diva2:944058/FULLTEXT02>>.
- Page, V., 2020. *The Economics Behind Marathons*. [online] Available at: <<https://www.investopedia.com/articles/investing/100815/economics-behind-marathons.asp>> [Accessed 9 Jul. 2020].
- Pore, S.S. and Pawar, S.B., 2015. Comparative Study of SQL & NoSQL Databases. *Ijarcet*, 4(5), pp.1747–1753.

Stoica, M., Mircea, M. and Ghilic-micu, B., 2013. Software Development: Agile vs. Traditional. *Informatica Economica*, 17(4/2013), pp.64–76.

Sudiartha, I.K.G., Indrayana, I.N.E., Suasnawa, I.W., Asri, S.A. and Sunu, P.W., 2020. Data Structure Comparison Between MySQL Relational Database and Firebase Database NoSql on Mobile Based Tourist Tracking Application. *Journal of Physics: Conference Series*, 1569, p.032092.

ActiveEndurance.com n.d., Integration For Every Event, accessed 7 August 2020, <<https://www.activeendurance.com/solutions/integration.htm>>

MyLaps.com n.d., Running, accessed 7 August 2020, <<https://www.mylaps.com/active-sports/running/>>

HuskyIntelligence n.d.. Michael "Monty Widenius - Open Source MySQL and MariaDB, accessed 10 March 2021, <<https://huskyintelligence.com/leverage-open-source-code/>>

Labs, O. 2020. What Is The Express Node.JS Framework?, accessed 16 March 2021, <<https://medium.com/@osiolabs/what-is-the-express-node-js-framework-f831d0c36a38>>

Facebook.Github, n.d.. In-Depth Overview, accessed 01 April 2021, <<https://facebook.github.io/flux/docs/in-depth-overview/>>

Arnold, M.L., 2021. Measuring Usability with the USE Questionnaire, retrieved from, Usability Interface, 8(2), pp. 3–6.

MariaDB (2019). Enterprise Database Comparison. Available at https://mariadb.com/wp-content/uploads/2019/04/enterprise-database-comparison_whitepaper_1028.pdf [Accessed 1 April 2021]

APPENDICES

APPENDIX A: Result of survey

1. What are the different tasks assigned to the event personnel of the event ?

FOR QM

1. Pres / VP1

- overall ketua
- Put road sign 1 day before event starts (show direction to student)
- Make sure no student is left outside of school compound after the event.
- Be ready to solve any unaccounted problems/emergencies.

2. Secretary / Counter

- prepare whole school namelists (according to the house and category)
- apply road check permission slip for QM overall ketua , Badan Beruniform President and Photography club Pres
- arrange student duty lists send give by BADAN BERUNIFORM and CLUBS
- distribute number bibs to house captain & teacher 2 / 3 days before events
- solve problems with bibs *1 or 2 days bfr events* (e.g. : wrong house bibs given , wrong class , wrong gender , didnt get bibs especially for new students)
- on event that day(early in morning) , set a counter outside QM room , fine and do bibs for student who forget to bring their owns ; another group will help to set counter at sport center for later jobs in recording the rank (total 6 counter)
- collect and record results from masa ,tamat and stations (if results is not tally , check with **photographer**, **station ketua** , **tamat** to confirm)
- Print out the results after the event and let teacher to announce the winners , then teacher will announce 15 mins for student to make bataan (work together with masa)

3. Hakim Penamat

- record participant's number bibs on a draft paper
- distribute a special number card for them (as a prove for counter to record)
- after categories time limit reach or 150 person is achieve , rewrite on an official paper and pass it to counter
- work together with PENJAGA MASA

- normally teacher helps to read number bibs and personnel record number bibs

4. Penjaga Masa

- start recording time at **MULA**
- record participants time (when reaching finishing line) on a draft paper
- inform PA when about the time left for each categories (each categories have different timing)
- when time limit is reach , inform PA categories is closed . Write timing on an official paper
- work together with HAKIM PENAMAT
- normally teacher helps to read the timing and personnel record

5. Hakim Mula

- bring participants to starting point.
- countdown and press “ horn “ when start

6. Station

- normally they are separated into a few stations (1A , 1B ,2A , 2B , 3A , 3B ,4A & 5A)
- each station will have 1 ketua
- record the number bibs of participant that passes by them
- distribute “special “ rubber band (as a prove)

7. Rider

- Check and make sure that nothing is obstructing the road before the event.
- 1 day before the event , put road signs (to show participants which way to go.)
- lead the participants to the finishing line by holding up a sign and blowing the whistle regularly.
- Inform MASA and TAMAT when participants are close to the finishing line.

8. Blindspot
 - prevent student from using the shortcut.

9. Penyemak Jalan
 - final checking road sign and road got any problem in the early morning before event start

10. Peronda
 - work together with **BSMM**
 - keep on checking the road to see whether any student is injured

Badan Beruniform

- BSMM , KRS , POLIS , PENGAKAP , GIRL GUIDE , BOMBA
- each badan beruniform need to send 30 people to duty
- each badan will divide 30 ppl into small group (around 3 to 4 ppl) and duty in between 1 main station to another main station
- standby and show direction to student runner
- BSMM will have more student to duty at Base (school dewan) , each main stations and with **Peronda (QM)**
- Girl Guides will helps teacher when giving prizes

Clubs

- PA , LPPM , PHOTOGRAPHY , PREFECT
1. PA :
 - work together with Penjaga Masa
 - Masa need to inform PA to announce time limit / time left for each categories
 2. LPPM :
 - prepare tables and chairs for Counter , Photography clubs , BSMM , stations
 3. PHOTOGRAPHY :
 - standby at each main stations
 - take pictures of participants when they cross each main stations (as a prove)

- sending back photo at their base (work together with counter)
 - when participants come to counter with **not enough 'special 'rubber bands or no card** , write down their number bibs and station they dont have for photography to check
 - if couldn't find his/ her number bibs from every station , the person will be disqualified .
4. PREFECT : - maintain order

2. In your opinion, which task is the most energy exhaustive and why?*

- Counter
- They have to register everyone that finishes the event in time.
- Station
- They have to stand under the sun and record any participants that pass them either by voice or on paper.

3. Is the method employed by your school to record and rank students done in digital or hand written? (If both please specify)

- both
- hand written used in STATIONS , HAKIM PENAMAT , PENJAGA MASA (after that they will pass the results to counter)
- digital method is used in counter (we already have a system by just typing out number bibs of student from HAKIM PENAMAT and it will come out with student name)

4. How do you ensure a student runner has passed all required checkpoints?

- When a participant reaches a station, station members will hand out a rubber band to the participant.
- every station will have a different type of rubber band.

- counter will ask them to show their rubber band , if not enough , write down their number bibs and station that the runner (dont have) and given to the photography club to investigate.
- if there is pics that can prove that the participants did pass by that station , his place will remain
- if no evidence found, we will check with the station members , if his number bibs is recorded , his place is remain
- If either one photographer or station member can prove that the runner has passed by that station , his place will remain .
- if no evidence found from anyone ----- disqualified

5. For all checkpoints, how many instances of student runners (independently of different category/bracket) will be recorded?

- we have 2 different ways(one is for P1, P2, P3,L3 and another one is L2&L1)
- let's say that station is 1A , all the 6 categories will pass by and SO we need to record all 6 categories of student number bibs
- 1 category only needed to record 150 person
- and lets say another longer way for L1 & L2 , because there are only 2 categories will pass by that station , we only need to record 2 categories runner number bibs

6. In average, how long does it take to record one student runner at any checkpoint?

1 second	2 seconds	5 seconds	10 seconds	>10 seconds
----------	-----------	-----------	------------	-------------

- Depending on the situation.
- Sometimes the station members don't have enough time to record their number bibs when a group of participants come together.

- so normally we will record by voice. (backup)(1 person is there to read the runner number bibs and others help to see whether did he left out any)
- after that when the station members come back to school , they will replay it then only write in on an official paper and pass it to counter

7.—Have the checkpoint personnel ever been overwhelmed to record multiple instances of runners crossing the checkpoint at the same time? What can they do when that happens?

- answered in Q6

8. How do you feel in the middle of your guarding duty? Will there be a significant drop in energy and accuracy to record student runners?*

- sometimes

9. How often are student runners' information misrecorded or misorganized?

Never	Rarely	Sometimes	Often	Always
-------	--------	-----------	-------	--------

Why do you think the error occurs? (probe)

- system problem
- careless

10. Is there other issue arisen from the currently employed tracking method that affects the personnel or event?

- nope

11. Which aspect of the equipment currently employed to track students is the most frustrating or difficult to endure?

- Jotting down the numbers by hand.

12. How likely do you think the school would collect funds to purchase and use advanced tracking equipment such as 'RFID timing chips' and 'tracking mat'? Why?

Very unlikely	Unlikely	Maybe	Likely	Very likely
---------------	----------	-------	--------	-------------

- high budget
- Not required as the method we are using does not have much problems.

13. How the list of winners is kept after the event and for how long? *

<1 week	<1 Month	<1 Year	<5 Years	Kept forever
---------	----------	---------	----------	--------------

Source: Combined survey answers from teachers and assisting students of SMJK
Katholik PJ

APPENDIX B: Samples of runners' bib implemented by SMJK Katholik PJ





Source: Teacher of SMJK Katholik PJ

APPENDIX C: Code Review From CodeBeat

owenk165/CEEMS ▼ Owen Khew Li Tien ▼

3.29 GPA **github.com/owenk165/CEEMS** ☆ 4seb1d7e@main
Last updated: Today, 6:56 am

185 complexity issues 74 duplications 847 style issues 6086 lines of code - code coverage

Complexity Styles Duplications Security Quick Wins Namespaces Timeline Settings Pull Requests

Code quality breakdown

codebeat computes a global project score, the GPA, and grades individual namespaces to help you measure technical debt and find refactoring opportunities.

Rating ▲	Search for name...	Q	Language	Complexity	Code Issues	Duplication
F	app/containers/UserSetup/index.js		javascript	8	2	2
F	internals/templates/i18n.js		javascript	4	0	1
F	internals/templates/utills/injectReducer.js		javascript	2	0	2
F	app/containers/RegisterParticipants/reducer.js		javascript	1	6	5
F	internals/templates/utills/reducerInjectors.js		javascript	2	0	1
F	app/containers/Event/index.js		javascript	33	8	1
F	app/containers/ParticipantsList/index.js		javascript	40	10	3

owenk165/CEEMS-Mobile ▼ Owen Khew Li Tien ▼

3.21 GPA **github.com/owenk165/CEEMS-Mobile** ☆ 24510d65@master
Last updated: Today, 7:05 am

62 complexity issues 40 duplications 913 style issues 2694 lines of code - code coverage

Complexity Styles Duplications Security Quick Wins Namespaces Timeline Settings Pull Requests

Code quality breakdown

codebeat computes a global project score, the GPA, and grades individual namespaces to help you measure technical debt and find refactoring opportunities.

Rating ▲	Search for name...	Q	Language	Complexity	Code Issues	Duplication
F	src/Containers/Event/RecordParticipants.js		javascript	32	10	6
F	src/Containers/EventList/AssistingEvents/Index.js		javascript	14	2	3
F	src/Containers/Event/Index.js		javascript	47	5	6
F	src/Containers/Event/ResultTable.js		javascript	13	7	2
F	src/Containers/Event/PersonnelList.js		javascript	6	1	3

APPENDIX D: User Acceptance Test and System Usability Test

Informed Consent Form

This study is being conducted by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

1. User Acceptance Test (UAT)
2. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Bahador Kamis

Signature : Bahador Kamis

Date : 14/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	14/4/2021		
Test Start Time	5.00pm		
Test End Time	5.30pm		
Test Name	Bahador Kamis		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	Design is nice, and easy to progress through.
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	Easy to do even for new users.
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	-

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	-
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	-
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	Page moves left and right when changing pages or typing in search box.
View Completed Event	<p>The user wishes to view the result of a past event.</p>	Pass	-

	<ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 3. Confirm if the event result is displayed. 4. Confirm if the sorting functionalities work. 		
--	--	--	--

**User Acceptance Test
Mobile Application**

Test Date	14/4/2021		
Test Start Time	5.30pm		
Test End Time	5.45pm		
Test Name	Bahador Kamis		
Test Module	Test Scenario	Pass / Fail	Comments
Login	<p>The user has downloaded the mobile application for event day.</p> <ol style="list-style-type: none"> 1. Login in the mobile application with the account created in web application. 	Pass	-
Event Start	<p>The user wishes to start the event for the event day.</p> <ol style="list-style-type: none"> 1. Locate the event 2. Click on start event button. 	Pass	-
Event Recording	<p>The user wishes to record participants at different checkpoint.</p> <ol style="list-style-type: none"> 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 	Pass	-

	5. Manually record another participant.		
Event End	The user wishes to end the event. <ol style="list-style-type: none">1. Enter an ongoing event started by the user.2. End the event.3. Visit the event page again to view the result.	Pass	-

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name							
	Strongly Disagree				Strongly Agree		
Factors	1	2	3	4	5	6	7
Usefulness							
1. It helps me be more effective					✓		
2. It helps me be more productive.				✓			
3. It is useful.					✓		
4. It gives me more control over the activities in my life.				✓			
5. It makes the things I want to accomplish easier to get done.				✓			
6. It saves me time when I use it.					✓		
7. It meets my needs.					✓		
8. It does everything I would expect it to do.						✓	
Ease of Use							
9. It is easy to use.						✓	
10. It is simple to use.						✓	
11. It is user friendly.						✓	
12. It requires the fewest steps possible to accomplish what I want to do with it.						✓	
13. It is flexible.					✓		
14. Using it is effortless.						✓	
15. I can use it without written instructions.					✓		
16. I don't notice any inconsistencies as I use it.						✓	
17. Both occasional and regular users would like it.				✓			

18. I can recover from mistakes quickly and easily.						✓	
19. I can use it successfully every time.						✓	
Ease of Learning							
20. I learned to use it quickly.						✓	
21. I easily remember how to use it.							✓
22. It is easy to learn to use it.							✓
23. I quickly became skillful with it.						✓	
Satisfaction							
24. I am satisfied with it.						✓	
25. I would recommend it to a friend.					✓		
26. It is fun to use.					✓		
27. It works the way I want it to work.						✓	
28. It is wonderful.					✓		
29. I feel I need to have it.					✓		
30. It is pleasant to use.					✓		
<i>Optional</i>							
What did you like the best for this system?	Minimalistic design.						
What did you like the least for this system?	-						
Do you have any additional comments on the system?	Page stability can be improved during event searches.						

Informed Consent Form


This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

3. User Acceptance Test (UAT)
4. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Yong Hou Mun
Signature : 
Date : 10/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	10/4/2021		
Test Start Time	11.00am		
Test End Time	11.30am		
Test Name	Yong Hou Mun		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	Home page is clean and offers quick and easy navigation throughout the web application.
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	Process is simple enough for even new users to do it without problems.
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	Good feature because mistakes tend to be

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		made here and there.
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	-
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	-
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	Swapping pages and typing in search box affects the page layout a little.
View Completed Event	<p>The user wishes to view the result of a past event.</p>	Pass	-

	<ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 3. Confirm if the event result is displayed. 4. Confirm if the sorting functionalities work. 		
--	--	--	--

**User Acceptance Test
Mobile Application**

Test Date	10/4/2021		
Test Start Time	11.30am		
Test End Time	11.45am		
Test Name	Yong Hou Mun		
Test Module	Test Scenario	Pass / Fail	Comments
Login	<p>The user has downloaded the mobile application for event day.</p> <ol style="list-style-type: none"> 1. Login in the mobile application with the account created in web application. 	Pass	Text in the home page seemed a little too packed.
Event Start	<p>The user wishes to start the event for the event day.</p> <ol style="list-style-type: none"> 1. Locate the event 2. Click on start event button. 	Pass	-
Event Recording	<p>The user wishes to record participants at different checkpoint.</p> <ol style="list-style-type: none"> 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 	Pass	Application is able to smoothly record runners without issues.

	5. Manually record another participant.		
Event End	The user wishes to end the event. 1. Enter an ongoing event started by the user. 2. End the event. 3. Visit the event page again to view the result.	Pass	-

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name							
	Strongly Disagree				Strongly Agree		
Factors	1	2	3	4	5	6	7
Usefulness							
1. It helps me be more effective					✓		
2. It helps me be more productive.					✓		
3. It is useful.					✓		
4. It gives me more control over the activities in my life.					✓		
5. It makes the things I want to accomplish easier to get done.				✓			
6. It saves me time when I use it.					✓		
7. It meets my needs.						✓	
8. It does everything I would expect it to do.						✓	
Ease of Use							
9. It is easy to use.						✓	
10. It is simple to use.						✓	
11. It is user friendly.						✓	

12. It requires the fewest steps possible to accomplish what I want to do with it.							✓
13. It is flexible.					✓		
14. Using it is effortless.						✓	
15. I can use it without written instructions.						✓	
16. I don't notice any inconsistencies as I use it.						✓	
17. Both occasional and regular users would like it.				✓			
18. I can recover from mistakes quickly and easily.							✓
19. I can use it successfully every time.							✓
Ease of Learning							
20. I learned to use it quickly.							✓
21. I easily remember how to use it.							✓
22. It is easy to learn to use it.							✓
23. I quickly became skillful with it.							✓
Satisfaction							
24. I am satisfied with it.						✓	
25. I would recommend it to a friend.					✓		
26. It is fun to use.					✓		
27. It works the way I want it to work.						✓	
28. It is wonderful.						✓	
29. I feel I need to have it.					✓		
30. It is pleasant to use.						✓	
<i>Optional</i>							
What did you like the best for this system?	Clean overall design and easy navigation.						
What did you like the least for this system?	-						

<p>Do you have any additional comments on the system?</p>	<p>Web application stability can improved as shown when the page is affected when users type in the search box to search for events.</p>
---	--

Informed Consent Form

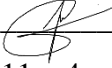
This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

5. User Acceptance Test (UAT)
6. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Khong Joon Kit
Signature : 
Date : 11 - 4 - 2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	11 – 4 – 2021		
Test Start Time	6:05pm		
Test End Time	7:00pm		
Test Name	Khong Joon Kit		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	Register and Login feature have all the normal feature that are usually found on other websites.
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on ‘Add New Event’. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of ‘U10617’ 6. Add some group to the event. 	Pass	I hope that the add a personnel section will remove the input after clicking the add personnel button.
Event Update	The user realized they had more information to add for the event.	Pass	

	<ol style="list-style-type: none"> 1. Find the previously created event. 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Event Deletion	<p>The user accidentally made a duplicate event, they want to remove it.</p> <ol style="list-style-type: none"> 1. Create a new event. 2. Enter the event. 3. Delete the event. 	Pass	
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	

Event Search	The user wishes to search for an event. 1. Use the search bar to search for any events.	Pass	The search function is quick.
View Completed Event	The user wishes to view the result of a past event. 1. Search 'EOJA91' in the search bar. 2. Visit the event page. 3. Confirm if the event result is displayed. 4. Confirm if the sorting functionalities work.	Pass	

User Acceptance Test

Mobile Application

Test Date	11 – 4 – 2021		
Test Start Time	6:05pm		
Test End Time	7:00pm		
Test Name	Khong Joon Kit		
Test Module	Test Scenario	Pass / Fail	Comments
Login	The user has downloaded the mobile application for event day. 1. Login in the mobile application with the account created in web application.	Pass	The mobile login function is exactly the same as the web version
Event Start	The user wishes to start the event for the event day. 1. Locate the event 2. Click on start event button.	Pass	

Event Recording	<p>The user wishes to record participants at different checkpoint.</p> <ol style="list-style-type: none"> 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 5. Manually record another participant. 	Pass	
End Event	<p>The user wishes to end the event.</p> <ol style="list-style-type: none"> 1. Enter an ongoing event started by the user. 2. End the event. 3. Visit the event page again to view the result. 	Pass	

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name							
	Strongly Disagree				Strongly Agree		
Factors	1	2	3	4	5	6	7
Usefulness							
1. It helps me be more effective					✓		
2. It helps me be more productive.				✓			
3. It is useful.						✓	
4. It gives me more control over the activities in my life.				✓			
5. It makes the things I want to accomplish easier to get done.					✓		
6. It saves me time when I use it.						✓	
7. It meets my needs.					✓		
8. It does everything I would expect it to do.						✓	
Ease of Use							
9. It is easy to use.						✓	
10. It is simple to use.					✓		
11. It is user friendly.					✓		
12. It requires the fewest steps possible to accomplish what I want to do with it.					✓		
13. It is flexible.				✓			
14. Using it is effortless.					✓		
15. I can use it without written instructions.						✓	
16. I don't notice any inconsistencies as I use it.					✓		
17. Both occasional and regular users would like it.					✓		

18. I can recover from mistakes quickly and easily.					✓		
19. I can use it successfully every time.						✓	
Ease of Learning							
20. I learned to use it quickly.						✓	
21. I easily remember how to use it.						✓	
22. It is easy to learn to use it.						✓	
23. I quickly became skillful with it.						✓	
Satisfaction							
24. I am satisfied with it.						✓	
25. I would recommend it to a friend.					✓		
26. It is fun to use.				✓			
27. It works the way I want it to work.					✓		
28. It is wonderful.				✓			
29. I feel I need to have it.				✓			
30. It is pleasant to use.					✓		
<i>Optional</i>							
What did you like the best for this system?	The ease of use and how easy it is to learn it						
What did you like the least for this system?							
Do you have any additional comments on the system?							

Informed Consent Form


This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

7. User Acceptance Test (UAT)
8. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Lee Jet Ying 
Signature : _____
Date : 14/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	14-4-2021		
Test Start Time	2:35pm		
Test End Time	3:12pm		
Test Name	Lee Jet Ying		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	Login is slow
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	-
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	-

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	Need to double click generate on new ID to register
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	Need delete notification
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	-
View Completed Event	<p>The user wishes to view the result of a past event.</p> <ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 	Pass	-

	<ol style="list-style-type: none">3. Confirm if the event result is displayed.4. Confirm if the sorting functionalities work.		
--	--	--	--

**User Acceptance Test
Mobile Application**

Test Date	14-4-2021		
Test Start Time	3:16pm		
Test End Time	3:32pm		
Test Name	Lee Jet Ying		
Test Module	Test Scenario	Pass / Fail	Comments
Login	The user has downloaded the mobile application for event day. 1. Login in the mobile application with the account created in web application.	Pass	Interface is nice
Event Start	The user wishes to start the event for the event day. 1. Enter the event page. 2. Start the event.	Pass	-
Event Recording	The user wishes to record participants at different checkpoint. 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 5. Manually record another participant.	Pass	-
Event End	The user wishes to end the event. 1. Enter an ongoing event started by the user. 2. End the event. 3. Visit the event page again to view the result.	Pass	-

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name	Strongly Disagree							Strongly Agree						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Usefulness														
1. It helps me be more effective				✓										
2. It helps me be more productive.													✓	
3. It is useful.													✓	
4. It gives me more control over the activities in my life.													✓	
5. It makes the things I want to accomplish easier to get done.													✓	
6. It saves me time when I use it.													✓	
7. It meets my needs.												✓		
8. It does everything I would expect it to do.													✓	
Ease of Use														
9. It is easy to use.				✓										
10. It is simple to use.				✓										
11. It is user friendly.				✓										
12. It requires the fewest steps possible to accomplish what I want to do with it.													✓	
13. It is flexible.												✓		
14. Using it is effortless.													✓	
15. I can use it without written instructions.												✓		
16. I don't notice any inconsistencies as I use it.		✓												
17. Both occasional and regular users would like it.				✓										

18. I can recover from mistakes quickly and easily.							✓
19. I can use it successfully every time.			✓				
Ease of Learning							
20. I learned to use it quickly.						✓	
21. I easily remember how to use it.					✓		
22. It is easy to learn to use it.						✓	
23. I quickly became skillful with it.					✓		
Satisfaction							
24. I am satisfied with it.						✓	
25. I would recommend it to a friend.					✓		
26. It is fun to use.			✓				
27. It works the way I want it to work.						✓	
28. It is wonderful.			✓				
29. I feel I need to have it.			✓				
30. It is pleasant to use.					✓		
<i>Optional</i>							
What did you like the best for this system?	The design						
What did you like the least for this system?	The register participant and search page						
Do you have any additional comments on the system?	-						

Informed Consent Form


This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

9. User Acceptance Test (UAT)
10. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Lim Siew Hong
Signature : 
Date : 10/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	10/4/2021		
Test Start Time	1830		
Test End Time	1930		
Test Name	Lim Siew Hong		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	It is easy to find my previous event and edit it.

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Event Deletion	<p>The user accidentally made a duplicate event, they want to remove it.</p> <ol style="list-style-type: none"> 1. Create a new event. 2. Enter the event. 3. Delete the event. 	Pass	The delete button is easily found.
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	The search function is fast.

View Completed Event	<p>The user wishes to view the result of a past event.</p> <ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 3. Confirm if the event result is displayed. 4. Confirm if the sorting functionalities work. 	Pass	
----------------------	---	------	--

User Acceptance Test

Mobile Application

Test Date	10/4/2021		
Test Start Time	1830		
Test End Time	1930		
Test Name	Lim Siew Hong		
Test Module	Test Scenario	Pass / Fail	Comments
Login	<p>The user has downloaded the mobile application for event day.</p> <ol style="list-style-type: none"> 1. Login in the mobile application with the account created in web application. 	Pass	
Event Start	<p>The user wishes to start the event for the event day.</p> <ol style="list-style-type: none"> 1. Locate the event 2. Click on start event button. 	Pass	
Event Recording	<p>The user wishes to record participants at different checkpoint.</p> <ol style="list-style-type: none"> 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 	Pass	The QR scan works but is a bit slow.

	<ol style="list-style-type: none">4. Confirm record the participant.5. Manually record another participant.		
End Event	<p>The user wishes to end the event.</p> <ol style="list-style-type: none">1. Enter an ongoing event started by the user.2. End the event.3. Visit the event page again to view the result.	Pass	

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name							
	Strongly Disagree				Strongly Agree		
Factors	1	2	3	4	5	6	7
Usefulness							
1. It helps me be more effective						✓	
2. It helps me be more productive.					✓		
3. It is useful.						✓	
4. It gives me more control over the activities in my life.			✓				
5. It makes the things I want to accomplish easier to get done.					✓		
6. It saves me time when I use it.						✓	
7. It meets my needs.						✓	
8. It does everything I would expect it to do.						✓	
Ease of Use							
9. It is easy to use.						✓	
10. It is simple to use.						✓	
11. It is user friendly.						✓	
12. It requires the fewest steps possible to accomplish what I want to do with it.					✓		
13. It is flexible.				✓			
14. Using it is effortless.					✓		
15. I can use it without written instructions.					✓		
16. I don't notice any inconsistencies as I use it.					✓		
17. Both occasional and regular users would like it.				✓			

18. I can recover from mistakes quickly and easily.					✓		
19. I can use it successfully every time.					✓		
Ease of Learning							
20. I learned to use it quickly.				✓			
21. I easily remember how to use it.					✓		
22. It is easy to learn to use it.					✓		
23. I quickly became skillful with it.				✓			
24. Satisfaction							
25. I am satisfied with it.					✓		
26. I would recommend it to a friend.				✓			
27. It is fun to use.					✓		
28. It works the way I want it to work.						✓	
29. It is wonderful.					✓		
30. I feel I need to have it.					✓		
31. It is pleasant to use.						✓	
<i>Optional</i>							
What did you like the best for this system?	It is easy to use as all the functions are easily found.						
What did you like the least for this system?	The design is plain.						
Do you have any additional comments on the system?							

Informed Consent Form

This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

11. User Acceptance Test (UAT)
12. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Wong Wei Hao

Signature : *Wong*

Date : 15/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	15/4/2021		
Test Start Time	6.00pm		
Test End Time	7.00pm		
Test Name	Wong Wei Hao		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	The interface can improve aesthetically.
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	
View Completed Event	<p>The user wishes to view the result of a past event.</p> <ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 	Pass	

	<ol style="list-style-type: none">3. Confirm if the event result is displayed.4. Confirm if the sorting functionalities work.		
--	--	--	--

User Acceptance Test
Mobile Application

Test Date	15/4/2021		
Test Start Time	6.00pm		
Test End Time	7.00pm		
Test Name	Wong Wei Hao		
Test Module	Test Scenario	Pass / Fail	Comments
Login	The user has downloaded the mobile application for event day. 1. Login in the mobile application with the account created in web application.	Pass	
Event Start	The user wishes to start the event for the event day. 1. Locate the event 2. Click on start event button.	Pass	
Event Recording	The user wishes to record participants at different checkpoint. 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 5. Manually record another participant.	Pass	
Event End	The user wishes to end the event. 1. Enter an ongoing event started by the user. 2. End the event. 3. Visit the event page again to view the result.	Pass	A search tab can be added to look for a particular participant's

			result quickly.
--	--	--	-----------------

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name							
	Strongly Disagree				Strongly Agree		
Factors	1	2	3	4	5	6	7
Usefulness							
1. It helps me be more effective						✓	
2. It helps me be more productive.							✓
3. It is useful.							✓
4. It gives me more control over the activities in my life.						✓	
5. It makes the things I want to accomplish easier to get done.							✓
6. It saves me time when I use it.						✓	
7. It meets my needs.						✓	
8. It does everything I would expect it to do.						✓	
Ease of Use							
9. It is easy to use.						✓	
10. It is simple to use.						✓	
11. It is user friendly.						✓	
12. It requires the fewest steps possible to accomplish what I want to do with it.						✓	
13. It is flexible.						✓	
14. Using it is effortless.					✓		
15. I can use it without written instructions.						✓	

16. I don't notice any inconsistencies as I use it.						✓	
17. Both occasional and regular users would like it.							✓
18. I can recover from mistakes quickly and easily.							✓
19. I can use it successfully every time.							✓
Ease of Learning							
20. I learned to use it quickly.						✓	
21. I easily remember how to use it.						✓	
22. It is easy to learn to use it.							✓
23. I quickly became skillful with it.						✓	
Satisfaction							
24. I am satisfied with it.							✓
25. I would recommend it to a friend.						✓	
26. It is fun to use.						✓	
27. It works the way I want it to work.						✓	
28. It is wonderful.						✓	
29. I feel I need to have it.						✓	
30. It is pleasant to use.							✓
<i>Optional</i>							
What did you like the best for this system?	The simplicity of the application.						
What did you like the least for this system?	N/A						
Do you have any additional comments on the system?	Application could be further enhanced to cater more events rather than just cross-country events.						

Informed Consent Form

This study is being conduct by Khew Li Tien, UTAR Software Engineering student to evaluate the usability and feasibility of the developed system under the title of “Cross Country Event Management System for Malaysian Secondary School”.

You have been invited to participate in this research study which will consist of the:

13. User Acceptance Test (UAT)

14. System Usability Test

By signing this form, you acknowledged that:

- You have agreed to participate in this study.
- You have understood that this usability study is voluntary and are free to raise any concerns or discomfort during the study session, and to discontinue the participation at any time.
- You have understood that the User Acceptance Test conducted will require you to navigate the developed system as per given the instruction sheets.

Name : Lee Yong Ching

Signature : Dash

Date : 15/4/2021

Thank you for your participation.

All information submitted will not be disclosed for any purpose other than the evaluation of the developed system.

User Acceptance Test
Web Application

Test Date	15/4/2021		
Test Start Time	20:44		
Test End Time	21:00		
Test Name	Lee Yong Ching		
Test Module	Test Scenario	Pass / Fail	Comments
Register and Login	<p>A new user wants to register an account.</p> <ol style="list-style-type: none"> 1. Register at home page. 2. Log out. 3. Login again with the same credential. <p><i>The credential will be used again for next few steps.</i></p>	Pass	Smooth operation.
Event Creation	<p>The user wishes to create a new upcoming event.</p> <ol style="list-style-type: none"> 1. Go to home page. 2. Click on 'Add New Event'. 3. Complete the first page of event creation process. 4. Add some personnel to the event. 5. Add a personnel with the ID of 'U10617' 6. Add some group to the event. 	Pass	Finish should redirect to event page.
Event Update	<p>The user realized they had more information to add for the event.</p> <ol style="list-style-type: none"> 1. Find the previously created event. 	Pass	

	<ol style="list-style-type: none"> 2. Modify the fields of the event. 3. Add or remove some personnel. 4. Add or remove some group. 5. Confirm update. 		
Participants Registration	<p>The user wishes to add participants or student runners for the event.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Click on register participant. 3. Register a / some new participant(s). 4. Download some of the runner's QR code. 	Pass	QR generated properly.
Participants Management	<p>The user wishes to manage the added participants.</p> <ol style="list-style-type: none"> 1. Find the relevant event. 2. Enter the participants list page. 3. Check the QR code of the added participants. 4. Remove 1 participants. 	Pass	
Event Search	<p>The user wishes to search for an event.</p> <ol style="list-style-type: none"> 1. Use the search bar to search for any events. 	Pass	
View Completed Event	<p>The user wishes to view the result of a past event.</p> <ol style="list-style-type: none"> 1. Search 'E0JA91' in the search bar. 2. Visit the event page. 	Pass	

	<ol style="list-style-type: none">3. Confirm if the event result is displayed.4. Confirm if the sorting functionalities work.		
--	--	--	--

**User Acceptance Test
Mobile Application**

Test Date	15/4/2021		
Test Start Time	20:44		
Test End Time	21:00		
Test Name	Lee Yong Ching		
Test Module	Test Scenario	Pass / Fail	Comments
Login	The user has downloaded the mobile application for event day. 1. Login in the mobile application with the account created in web application.	Pass	
Event Start	The user wishes to start the event for the event day. 1. Enter the event page. 2. Start the event.	Pass	
Event Recording	The user wishes to record participants at different checkpoint. 1. Enter an ongoing event. 2. Enter the recording screen. 3. Scan the QR of participants. 4. Confirm record the participant. 5. Manually record another participant.	Pass	Smooth operation.
Event End	The user wishes to end the event. 1. Enter an ongoing event started by the user. 2. End the event. 3. Visit the event page again to view the result.	Pass	

System Usability Test

(✓) on the scale that best suit your comments on the system. Try to respond all the items.

Respondent name	Strongly Disagree							Strongly Agree						
	1	2	3	4	5	6	7	1	2	3	4	5	6	7
Usefulness														
1. It helps me be more effective					✓									
2. It helps me be more productive.						✓								
3. It is useful.							✓							
4. It gives me more control over the activities in my life.						✓								
5. It makes the things I want to accomplish easier to get done.							✓							
6. It saves me time when I use it.					✓									
7. It meets my needs.						✓								
8. It does everything I would expect it to do.							✓							
Ease of Use														
9. It is easy to use.						✓								
10. It is simple to use.					✓									
11. It is user friendly.							✓							
12. It requires the fewest steps possible to accomplish what I want to do with it.							✓							
13. It is flexible.							✓							
14. Using it is effortless.						✓								
15. I can use it without written instructions.					✓									
16. I don't notice any inconsistencies as I use it.						✓								
17. Both occasional and regular users would like it.					✓									

18. I can recover from mistakes quickly and easily.							✓
19. I can use it successfully every time.							✓
Ease of Learning							
20. I learned to use it quickly.							✓
21. I easily remember how to use it.					✓		
22. It is easy to learn to use it.							✓
23. I quickly became skillful with it.							✓
Satisfaction							
24. I am satisfied with it.							✓
25. I would recommend it to a friend.					✓		
26. It is fun to use.							✓
27. It works the way I want it to work.							✓
28. It is wonderful.							✓
29. I feel I need to have it.					✓		
30. It is pleasant to use.							✓
<i>Optional</i>							
What did you like the best for this system?	QR code generator						
What did you like the least for this system?	None						
Do you have any additional comments on the system?	Nope						