ESTREAM: A MOBILE LIVE STREAMING PLATFORM FOR ESPORTS

BY

LIM YU QI

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF COMPUTER SCIENCE (HONOURS)

Faculty of Information and Communication Technology

(Kampar Campus)

JAN 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: ESTREAM: A MOBILE LIVE STREAMING PLATFORM FOR ESPORTS

Academic Session: Jan 2024

I LIM YU QI

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

(Author's signature)

Address:
No. 34, Lorong BLM 3/8
Bandar Laguna Merbok
Sungai Petani, Kedah.

Date: 26/04/2024

Verified by,

(Supervisor's signature)

Jasmina Khaw Yen Min

Supervisor's name

Date: 26/04/2024

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

(KAMPAR CAMPUS)

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 26/04/2024

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that ***Lim Yu Qi*** (ID No: ***20ACB01239***) has completed this final year project entitled " *ESTREAM: A MOBILE LIVE STREAMING PLATFORM FOR ESPORTS* " under the supervision of Dr Jasmina Khaw Yen Min (Supervisor) from the Department of Computer Science, Faculty of Information and Communication Technology.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

_____

(*Lim Yu Qi*)

*Delete whichever not applicable

DECLARATION OF ORIGINALITY

I declare that this report entitled "**ESTREAM: A MOBILE LIVE STREAMING PLATFORM FOR ESPORTS**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : _____ _____

Name : _____LIM YU QI_____

Date : _____26/04/2024_____

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to all those who have contributed to the completion of this project. Their support, guidance, and encouragement have been invaluable.

First and foremost, I extend special thanks and sincere appreciation to my supervisors and moderator, Dr Jasmina Khaw Yen Min, Dr Tan Joi San, and Dr Manoranjitham, for their unwavering support and mentorship throughout this project. Their expertise and insights greatly enriched the quality of this work.

I am also grateful to all my classmates and lecturers, whose support and teaching have contributed to my self-development, knowledge, and attitude towards completing this project.

Not forgetting, my family, who supported me in my studies, gave me good advice, and taught me all the important values in living my life. Your encouragement kept me motivated.

Last but not least, to those whose names may not be mentioned explicitly but whose contributions were nonetheless significant, thank you for being part of this journey.

This project would not have been possible without the collective efforts and support of everyone mentioned above. I appreciate everything that you have all done for me. Thank you again for all of your endless support.

ABSTRACT

In response to the rapid growth of esports and the ever-evolving landscape of live broadcasting platforms, this project endeavors to redefine the user experience within the esports community. To address common challenges in content discovery, community engagement, and profitability, this project aims to introduce a platform for transformative change beyond traditional video streaming methods. By leveraging state-of-the-art artificial intelligence (AI) technology and embracing the mobile-centric preferences of contemporary users, our platform, eStream seeks to provide a dynamic and immersive environment for esports enthusiasts. Through innovative content discovery mechanisms encompassing text, and images, as well as interactive donation features, users can support their favorite streamers, and this mobile app is dedicated to fostering a strong and inclusive sense of community. This project is steadfastly committed to pushing the boundaries of live broadcasting in the digital age, contribute in reshaping the esports landscape and nurturing unparalleled participation and connection among users.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

UI	User Interface
RGB	Red, Green and Blue
CNNs	Convolutional Neural Networks
OViF	Oriented Violent Flows
ViF	Violent Flows
HOOF	Histogram of Oriented Optical Flow
SVM	Support Vector Machine
VOD	Video on Demand
KPL	King Pro League Tournament
KGL	King Growth League Tournament
IDE	Integrated Development Environment
RAD	Rapid Application Development

Chapter 1
Introduction

This chapter discusses the background and motivation of the research, contributions to the field, and the outline of the thesis.

1.1 Problem Statement and Motivation

1.1.1 Problem Statements

As the eSports fan base continues to diversify, accessibility emerges as a central concern. The allure of mobile devices as readily accessible entertainment hubs cannot be overstated. A mobile live streaming platform for eSports possesses the potential to bridge the gap between dedicated eSports enthusiasts and a broader audience, encompassing casual gamers and newcomers to the competitive gaming arena. However, several challenges confront users, specifically audiences and content creators, when employing live streaming platforms for eSports content:

1. **Lacking Mobile-Centric Features in eSports Streaming**
   Interaction and community building are crucial aspects of a mobile app, yet current platforms often restrict viewer participation and interactive features. For instance, existing platforms like GosuGamers only offer live chat as a means of interaction during live streams [1]. Many other existing platforms are primarily focused only on live and video-based streaming, which limits engagement opportunities for end users [2, 3].

2. **Limitations of Sponsorship as Monetization Strategy**
   Content creators faced challenges in monetizing their efforts, with many video streaming platforms primarily relying on revenue from merchandise sales. However, sponsorship deals are prevalent, which create a limitation for content creators without significant resources. Sponsors typically target streamers with large followings, leaving those with smaller fan bases at a disadvantage in terms of income opportunities [4].

3. **Limitations of Esports Content Discovery**

   Live streaming platforms for esports provide a wide array of content, but without proper categorization, users may struggle to find what they're looking for. While users typically search using keyword queries, this method has limitations, as it relies solely on text content. Lengthy keyword queries can lead to extended search times and hinder effective content discovery [5].

4. **Impacts of Negative Content Dissemination in Esports Live Streaming**

   Negative content dissemination in esports live streaming is a concern, as live streaming content should ideally promote moral values, inclusivity, and responsible behavior. However, due to the commonplace nature of typical streaming content, some creators resort to including explicit material to attract attention [6, 7]. This intentional inclusion of explicit content has a detrimental impact on the overall esports live streaming community.

1.1.2 Motivation

The project is inspired by the evolving landscape of social media platforms, which offer diverse content discovery options. While recent live streaming apps primarily focus on video content, social media platforms encompass various content types, including text, images, videos, and live streams. By integrating this multifaceted approach into current esports live streaming platforms, users can explore esports content through pictures, videos, and text, fostering better interactions and enhancing user engagement within the esports community.

With the surge in popularity of live streaming and online shopping due to the COVID-19 pandemic, many individuals have developed habits of consuming video content and making purchases during their free time, by making payment using online banking or e-wallets. While traditional product purchases may not align with the esports world, incorporating e-wallets for donations presents a unique opportunity to enhance the donation feature of existing systems.

In addition, advancements in AI technology, such as image and scene detection, have become increasingly prevalent. These detection methods enable users to discover new knowledge or identify mistakes. Implementing detection methods in esports live streaming apps facilitates easy content discovery for users.

Furthermore, violence detection technology is commonly employed in other sports events and entertainment, such as hockey games and movies. A recent article highlighted an incident involving a gunshot during an esports live stream, which negatively impacted the audience [8]. Integrating such detection methods into current systems can help uphold the integrity of the content being delivered.

1.2 Objectives

The primary objectives of this project are to design, develop, and deploy a dedicated mobile live streaming application tailored to the eSports community. The application aims to enhance the user experience, engagement, and accessibility of eSports content through innovative features and a seamless platform. To achieve these overarching goals, the project is guided by the following specific objectives:

- **To enhance user engagement through mobile centric features.**
  To enhance user engagement for both audiences and streamers, the project implements a feed feature along with live chat and live polls. The feed feature would offer a personalized stream of content, tailored to individual user preferences, ensuring a dynamic and engaging experience. Live chat functionality allows real-time interaction between streamers and viewers, fostering a sense of community and facilitating immediate feedback. Additionally, integrating live polls enables streamers to gauge audience preferences and gather insights, further involving viewers in the content creation process.

- **To improve monetization support on e-sports live streaming platforms.**
  The project enhances the donation mechanisms by introducing virtual rewards, providing viewers with incentives to support their favorite streamers. In

addition, incorporating slider advertisements offers a non-intrusive method of advertising, generating revenue while minimizing disruptions to the viewing experience. Moreover, the project introduces leaderboards based on subscriber count incentivizes streamers to grow their audience, thereby increasing revenue potential through ad revenue and sponsorship opportunities.

- **To improve the content discovery experience.**
  The project implements object detection technology for images in posts and video thumbnails to improve content discovery. This feature enables users to search for specific content within images, enhancing discoverability and enabling more targeted exploration of content. By accurately identifying objects in thumbnails, users can quickly assess the relevance of a video before viewing it, streamlining the content discovery process, and enhancing user satisfaction.

- **To promote and deliver an ethical esports live stream environment.**
  The project integrates violence detection technology in images of posts and video thumbnails to mitigate the presence of explicit content. This technology identifies potentially harmful or inappropriate content, allowing for prompt action to be taken to address or remove such content from public view. By proactively detecting and addressing explicit material, the platform maintains a safe and respectful environment for all users, thereby safeguarding its reputation and fostering trust within the community.

By achieving these objectives, the project can successfully address challenges in the eSports live streaming landscape. In advance, the project enhances accessible and engaging for a broader audience, contributing to the growth and inclusivity of the eSports community.

1.3 Project Scope

The scope of this project encompasses the development of a dedicated Android mobile live streaming application tailored to the eSports community. The platform's core objective is to offer an immersive and seamless mobile experience, catering to players, enthusiasts, and content creators within the eSports ecosystem. It will provide real-time interaction, high-quality video streaming, and user-generated content sharing, all aimed at elevating the overall eSports viewing and engagement encounter. The scope encompasses features such as real-time chat, adaptive streaming, monetization, and community for content sharing, with an appropriate UI design. The platform's target users range from eSports players and avid fans to casual viewers seeking convenient access to competitive gaming content on their mobile devices.

The technical scope will focus on utilizing contemporary technologies like Java programming language for Android app development. Key aspects include addressing challenges in adaptive streaming for varying network conditions, low-latency streaming, and real-time interactive features. The project's timeline spans approximately six months, encompassing design, development, testing, and deployment phases. The project will involve rigorous user testing and iterative refinement to ensure that the developed platform aligns seamlessly with the preferences and expectations of the eSports community.

1.4 Contributions

By focusing on mobile accessibility, the project has made esports content more accessible and engaging for users, catering to the increasing trend of mobile-centric consumption. This strategic approach acknowledges the growing importance of mobile devices in accessing esports content and ensures that users can conveniently engage with esports content anytime and anywhere.

Additionally, the project has implemented effective mechanisms for discovering ideal content, allowing users to easily find and engage with esports-related content that aligns with their interests and preferences. This enhances the overall user experience by providing relevant and engaging content tailored to individual user preferences, fostering increased user engagement and satisfaction.

Apart from that, the project has introduced an innovative donation system that provides users with an interesting and interactive way to support their favorite streamers while simultaneously offering streamers additional income opportunities. By gamifying the donation process or incorporating rewards for donors, the project incentivizes user engagement and fosters a sense of community and support within the esports ecosystem.

Furthermore, the project's positive value delivery is exemplified by the implementation of explicit content detection. Users are promptly notified of any explicit content, empowering them to take further actions as necessary to maintain a secure browsing environment. This proactive approach to content moderation not only safeguards users from potentially harmful or inappropriate content but also fosters trust and confidence in the platform. By prioritizing user safety and well-being, the project enhances the overall user experience, contributing to a positive and supportive environment for both content creators and consumers within the esports community.

1.5 Report Organization

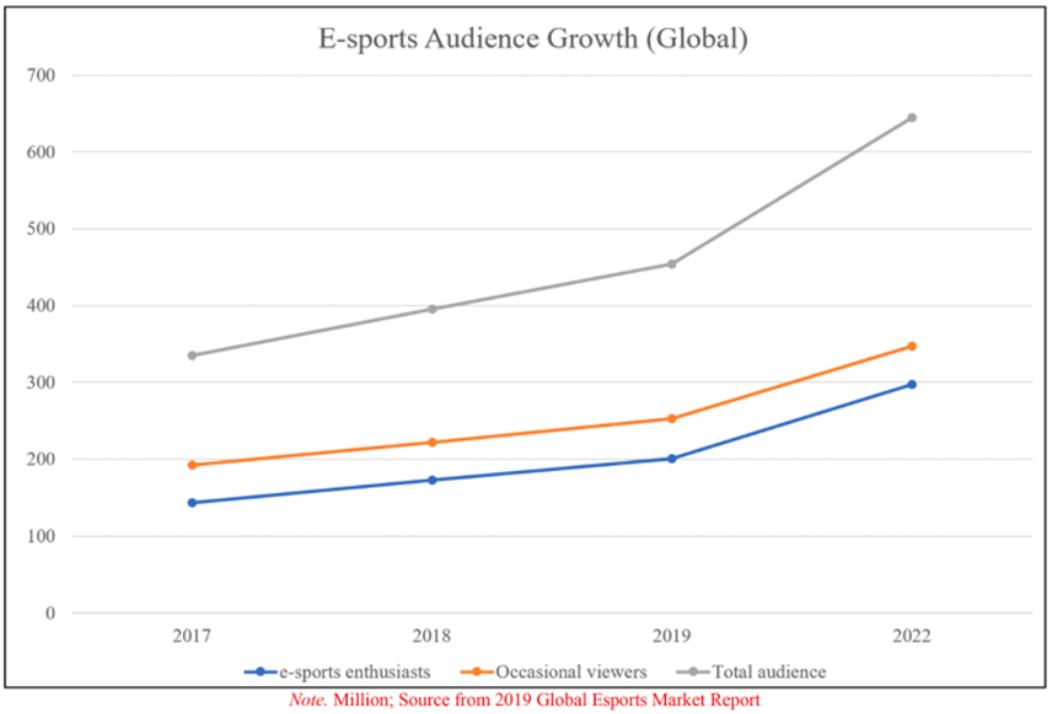
The report is structured across several chapters to systematically present the development of mobile live streaming application for this project. Chapter 1 serves as an introduction, framing the research question, objectives, and delineating the background and significance of the study. Chapter 2 offers a comprehensive literature review on existing live streaming platforms for eSports content and violence detection methods, highlighting current technologies' strengths and limitations while identifying the research gap motivating this study. Chapter 3 introduces the system approach, providing detailed insights into design specifications system use-cases. Chapter 4 presents the system design of the proposed system, specifying the components of the system. Chapter 5 discusses about the implementation of the system, and Chapter 6 presents the test performance matrix. Finally, Chapter 7 offers a comprehensive conclusion, succinctly summarizing the problem, solution, and outlines suggestions for future improvement.

Chapter 2
Literature Review

The development of a mobile live streaming application for the eSports industry represents a fusion of two dynamic domains: live streaming application and the rapidly growing eSports landscape. This literature review explores existing research, studies, and insights that pertain to both live streaming platform development and the eSports industry. By understanding the current state of the live streaming platform, user engagement strategies, and the dynamics of the eSports world, the review aims to explore the development of a robust and engaging mobile platform that caters to the unique needs of eSports enthusiasts.

2.1 eSports Industry Overview

The eSports industry, characterized by competitive video gaming, has witnessed exponential growth and transformation over the past decade. Prior research has shown that the global eSports industry is experiencing exponential growth, with a projected fan base exceeding 646 million by 2023 [9]. The growth of eSports industry comes from diverse sources, including sponsorships, media rights, ticket sales, merchandise, and advertising [10]. As emphasized by Qian et al. [10], eSports competitions blend elements of traditional sports with the digital realm, creating a unique entertainment experience that transcends geographic boundaries. Nowadays, eSports has successfully evolved from small local gatherings to large-scale international events held in arenas and watched by millions of viewers online. The popularity of eSports is further fuelled by the proliferation of smartphones, allowing fans to access content anytime and anywhere. The convergence of eSports and mobile technology presents a compelling opportunity to develop dedicated mobile apps that cater to this growing demand. Moreover, the interaction between eSports and live streaming has been instrumental in this growth, as it allows fans to engage with their favourite players and teams in real time.



Note. Million; Source from 2019 Global Esports Market Report

Figure 2.1.1 eSports audience growth

Source: Adapted from [9]

2.2 User Behavior and Preferences

The evolution of user behavior and preferences towards mobile consumption of digital content is a defining aspect of the digital age. There is a growing body of literature that analyses the demographics of the eSports and live streaming audience. A series of recent studies have revealed that the ratio of male viewers or streamers is significantly higher than that of female viewers or streamers [9, 11]. The demographics are formed by degree graduates, young professionals, and self-employed individuals whose ages range from 20 to 30 years. This group of viewers usually enjoys their leisure time by streaming eSports tournaments, as they have more free time compared to those who are engaged in full-time study or employment. Additionally, live streams by celebrities receive about twice as many views as live streams from non-celebrities, which tends to attract more media attention.

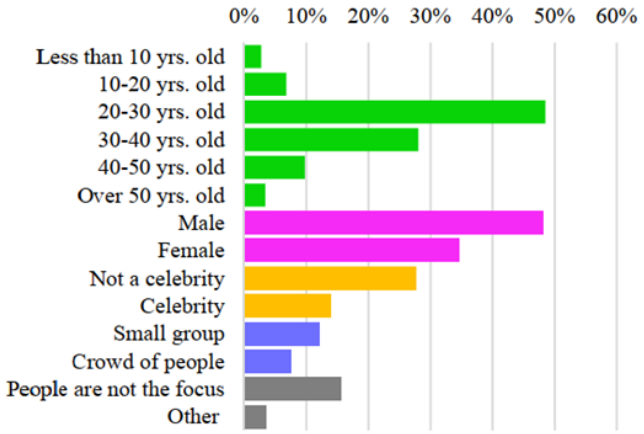


Figure 2.2.1 Demographic analysis of live stream participants

Source: Adapted from [11]

Previous research has established that mobile devices are increasingly becoming the preferred mode of accessing online content, including entertainment [12]. Casual viewers, as well as newcomers to eSports, are more likely to engage via mobile devices due to their accessibility and convenience. Furthermore, users demand seamless experiences that adapt to varying network conditions, reflecting the need for consistent performance even in less-than-optimal connectivity situations [10].

Regarding eSports context, studies by Qian et al. [10] suggest that viewers value diverse content formats, including player characteristics, commentary features, event attractiveness, and schedule convenience — findings consistent with those drawn from traditional sports settings. Additionally, the ability to engage with streamers and other viewers through danmaku is a cornerstone of user preferences, as it establishes a direct connection between streamers and fans [13].

However, research on this issue presents conflicting findings. Chen and Lin [14] stated that the primary channel affecting an audience's intention to watch is the impact of entertainment on attitude, where viewer relieved their stress which consequently influences watching intention. The attitude toward a live stream is directly affected by endorsement and indirectly influenced by social interaction. Celebrity endorsement plays important role in raising attractiveness, trustworthiness, and expertise through streamer's charm. Meanwhile, flow affects viewer through perceived value in terms of customer loyalty, watching intention, and donations. However, attitude exerts more

influence than perceived value, as part of the audience's watching intention remains unaffected by perceived value. Consequently, flow has little effect on watching intention, making it the least influential factor.

2.3 Interactive User Engagement in eSports

User engagement is pivotal in retaining a dedicated audience within a mobile app ecosystem. Extensive research demonstrates the significance of interactive features, such as live chat, stream quality, and streamer traits foster a sense of community and participation among viewers [10]. Viewers like using chat room to communicate with other through online and this find entertaining them while chat room become personal than watching television. Besides that, viewers prefer to enjoy higher stream resolution with good connectivity as this delivers a smooth streaming experience. Plus, viewer prefer the interaction between streamer and crowd as so is the knowledge of streamer. Streamers show their reaction and describe the techniques in games making their audience feel interesting and have better understandable to their stream.

Apart from that, viewers also engage in fan groups and appear to build more trust through communication with each other [12]. Gamification elements, such as knowledge gained, achievements, and player skills, as discussed by Kim and Kim [9], offer potential avenues to enhance user engagement, potentially leading to increased platform loyalty. The immediate feedback loop that live-streaming facilitates enhances the sense of community and fosters a deeper connection between streamers and viewers.

While preliminary, the most unique aspect of esports online spectatorship could be virtual rewards, such as in-game skins, cosmetics, virtual currencies, and customized emotes [10]. As incentives, virtual rewards are often designed to guide active players who may not be passionate viewers toward the competitive esports scene and to similarly motivate occasional viewers to become more involved in the game. This approach is gradually being adopted by game publishers and streaming platforms, rapidly gaining popularity among esports fans.

2.4 Monetization Strategies in Live Streaming

Monetization is a crucial aspect of live streaming, as it allows eSports players and content creators to generate income from their broadcasts and sustain their streaming careers. There are multiple ways of monetization strategies including subscription, donations, sponsorships, tournaments and events, and virtual gift. The choice of strategy depends on the streamer's niche, audience size, and the platform they use, so it is essential to tailor monetization methods to fit their specific circumstances and goals.

Streamers on platforms like Twitch can offer subscription tiers to their viewers. Subscribers typically receive benefits like ad-free viewing, custom emotes, special badges, and access to subscriber-only chats [8]. Streamers earn a portion of the subscription fees, with higher-tier subscriptions offering higher revenue. Apart from that, viewers can donate money directly to streamers during their broadcasts. For instance, Twitch offers a feature called 'bits', which viewers can purchase and use to tip streamers. This can be done through various payment methods, such as PayPal, credit cards, or platform-specific virtual currencies.

Besides, eSports players being the most popular individual in eSports industry, they often enter into sponsorship agreements with brands and companies. These sponsorships can include product placements, shout-outs, or dedicated streams promoting a product or service. Their target audience would be eSports fans who have their admire eSports player. To the extent of this context, eSports players and competitive streamers can participate in tournaments and events with cash prizes. Winning or performing well in these competitions can provide significant monetary rewards. While the audience need to buy ticket for watching the tournaments in the stadium.

Furthermore, live streaming platform in China offers opportunity for viewer and streamer to interact with each other through sending virtual gifts [12]. This sending will be displayed to all viewers and recorded in the leaderboard as appreciation for gift sender and as motivation to encourage other viewers for their engagement in giving. Streamers receive a portion of the revenue generated from these virtual gifts.

2.5 Case Studies and Existing Platforms

The eSports industry has witnessed significant growth and transformation, fuelled by the increasing demand for high-quality and easily accessible content for its global audience. Lu et al. [12] explored that 31% of the live streaming audience watches video game streams to learn game skills and strategies. As eSports continues to expand its reach, the need for dedicated mobile applications that seamlessly integrate gaming and live streaming becomes evident. This literature review explores several prominent live streaming platforms and their unique roles within the eSports ecosystem.

2.5.1 Twitch [16]

Twitch stands as a cornerstone of the eSports world, known for its remarkable capacity to host 200 million viewers and 2 million regular “streamers” [17]. Twitch stands out as a game-centered platform offering a plethora of esports live stream categories, facilitating easy content discovery through effective categorization using labels. It provides a diverse array of content, including live streams and videos, with a unique selling point of hosting specialized esports talk shows and podcasts. Despite its strengths, Twitch lacks robust community features, relying primarily on revenue generation through donations.

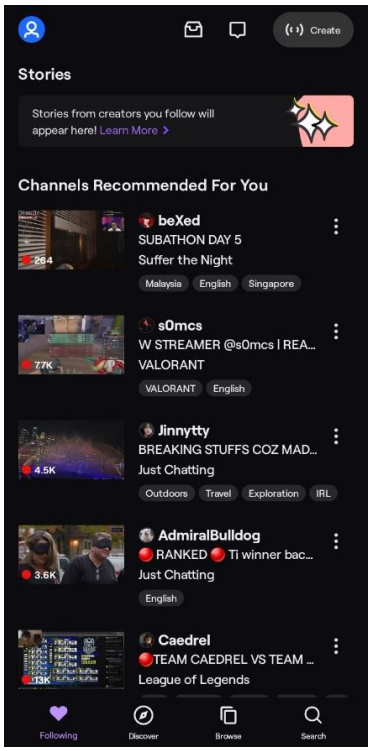


Figure 2.5.1 Twitch mobile application.

2.5.2 DLive [3]

DLive, a relatively new entrant, boasts a simple and excellent UI layout design that is easy for everyone to use. It provides the audience with a great streaming experience by delivering smooth and lag-free performance. Dlive distinguishes itself by offering streamers a higher revenue share and showcasing top earners through its streamer leaderboard. This revenue system attracts eSports content creators seeking DLive as an additional income streaming platform. While it excels as an income streaming platform, its focus remains primarily on live and video streaming content, limiting its scope for diversification.

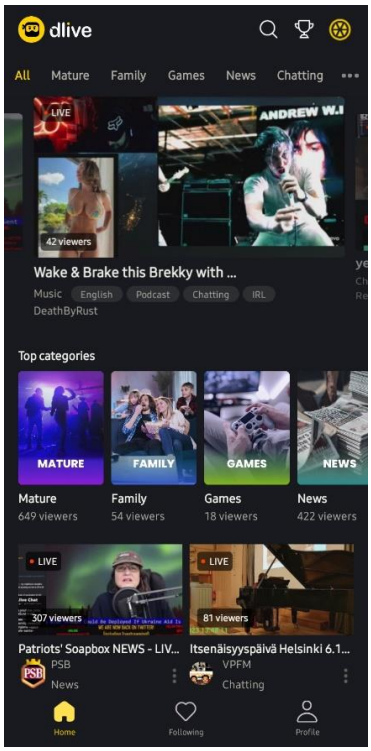


Figure 2.5.2 DLive mobile application.

2.5.3 AfreecaTV [20]

AfreecaTV, also known as Any Free Broadcasting, is a broadcasting platform in South Korea with a focus on a wide range of content, including gaming, entertainment, and real-life broadcasts. Initially, AfreecaTV originated as a TV channel retransmission website [21]. Nowadays, AfreecaTV has gained immense popularity among gamers, making it more famous in the field of game streaming. Afreeca TV specializes in video games and esports streaming, providing users with access to live tournaments and a comprehensive schedule of upcoming events. However, its platform struggles with content discoverability, particularly regarding personal posts on public pages, and the presence of various content types, including explicit material, detracts from its appeal.

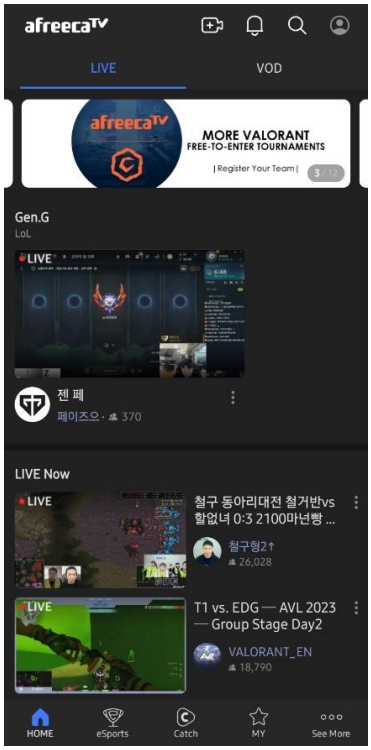


Figure 2.5.3 AfreecaTV mobile application.

2.5.4 GosuGamers [1]

Apart from platforms that allow game streaming, there are platforms that focus on eSports content. GosuGamers is an international site that primarily serves as a competitive gaming news and community platform. It offers tournament live streams covering several video games such as PUBG, DOTA2, Hearthstone, CS: GO, and more. GosuGamers provides eSports enthusiasts with various kinds of detailed information regarding eSports, including up-to-date information, rankings, and tournament coverage. On this live stream site, users can explore the esports tournament events and live streams, receiving all kinds of news and notifications regarding the latest streams. GosuGamers prides itself on covering a wide range of content, including rankings and news, and offers convenient search functionality through filters. Despite its strengths, the platform lacks personalized scheduling options and interactive features such as chat functions and feeds for user engagement.

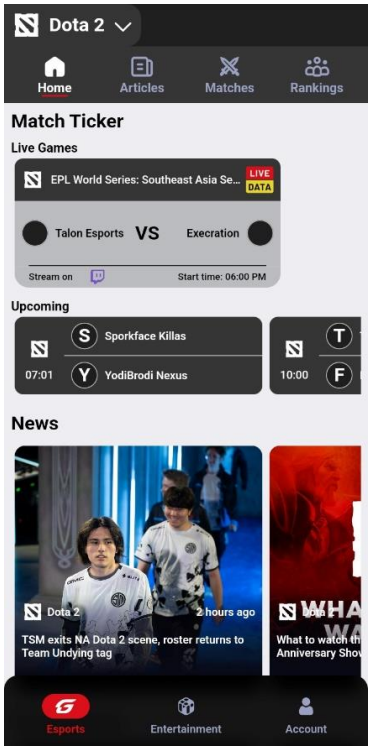


Figure 2.5.4 GosuGamers mobile application.

2.5.5 DouyuTV (DouYu) [2]

DouYu is one of the largest live streaming platforms in China. It provides a variety of genres of live streams such as life experiences, travel, singing, dancing, etc., but primarily focuses on game-related content such as game streams of League of Legends, Honor of Kings, and DOTA2 [12]. DouYu offers users the opportunity to earn virtual rewards through missions and donations, providing a diverse range of content beyond esports and incorporating chat functionality for interaction. However, its focus on the Chinese audience and its video-centric approach may limit its appeal to a broader demographic.

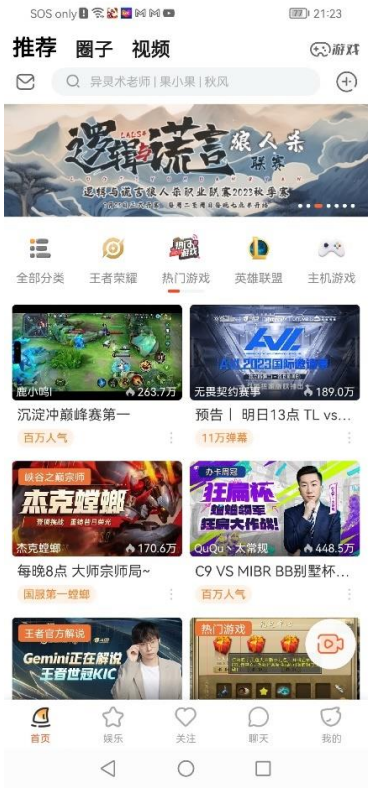


Figure 2.5.5 Douyu TV mobile application.

2.5.6 Critical Remarks of Existing Live Streaming Platforms

Table 2.5.6.1 Comparison of existing live streaming platforms

Features \ Live Streaming Platform	Twitch	DLive	AfreecaTV	GosuGamers	DouYu
User Engagement					
Live streaming	✓	✓	✓	✓	✓
Live chat	✓	✓	✓	✓	✓
Polls					✓
Forum					
Monetization Option					
Advertisements	✓				✓
Sponsorships	✓	✓	✓	✓	✓
Donations and tips	✓	✓	✓	✓	
Virtual gifts			✓	✓	✓
Virtual currency	✓	✓		✓	✓
Subscriptions	✓	✓	✓	✓	✓
Content Management					
Content categorization	✓	✓	✓	✓	✓
Content scheduling	✓	✓	✓		✓
Content Moderation					
Report and block	✓	✓	✓	✓	✓

The results of the comparison reveal that most platforms share common functionalities, including:

Table 2.5.6.2 The common functionalities in each category

Category of Features	Common Features
Video content	Live streaming
Interaction	Live chat, polls
Monetization	Sponsorship, virtual gifts/donations, subscriptions
Content-related	Content categorization, content scheduling, reporting

These functionalities effectively enhance the user experience of using live streaming platforms, allowing audiences to interact better with streamers while accessing content in a clear and accessible manner. With analytical features, streamers can improve content quality while benefiting from the monetization strategies offered by the live streaming platform.

Overall, Twitch, and DouYu are the most popular live streaming platforms among those listed. They encompass majority of the features mentioned in Table 2.5.6.2, providing users with a comprehensive live streaming experience. However, these platforms still lack a crucial polling feature essential for collecting audience feedback. Audience feedback is valuable for streamers to enhance content quality. The addition of a feature for collecting and generating audience feedback statistics could enhance viewer-streamer interaction in the future.

On the other hand, GosuGamers stands out for its unique focus on gamers from around the world. However, it falls short in terms of interactive features, offering only live chat. This limitation may impact the platform's lower user numbers. In terms of monetization features, DLive shines with its distinctive strategies that enable streamers to earn revenue through cryptocurrency. However, DLive does not provide advertisement revenue sharing but sponsorship instead.

2.6 Previous Works on Violence Detection

Violence detection in live eSports streams is of paramount importance as it plays a critical role in ensuring the safety of both players and viewers, while also preserving a positive community image [8]. By actively identifying and addressing violent actions or behaviours in real-time, this technology helps to create a secure and respectful environment within the eSports community. Moreover, it serves as a means to uphold community guidelines and standards, thereby contributing significantly to the overall integrity and quality of eSports events. To enhance the efficacy of violence detection, a comprehensive literature review is conducted to identify and implement methods with the highest accuracy in detecting violent actions.

2.6.1 RWF-2000: An Open Large Scale Video Database for Violence Detection

Cheng, Cai, and Li [22] has introduced a temporal pooling mechanism known as the "flow gate network" to address the issue of motion information becoming less informative due to a coarse pooling mechanism. This proposed model comprises four main components: the RGB Channel, the Optical Flow Channel, the Merging Block, and the Fully Connected layers. It integrates concepts from Convolutional Neural Networks (CNNs) and depth-wise separable convolutions to reduce model parameters without sacrificing performance.

One distinctive aspect of this model is its utilization of the optical flow channel to facilitate the development of a pooling mechanism. In the initial phase, RGB frames are directed into the RGB channel, while optical flows are channelled into the optical flow channel. Both channels undergo 'Relu' activation and sigmoid function operations, respectively, to fuse their outputs. The output of the sigmoid function, which falls within the range of 0 to 1, serves as a scaling factor to adjust the output of the RGB channel. Subsequently, the merging block processes the information after self-learned temporal pooling. Finally, the fully connected layers generate the output.

This model's innovative approach addresses the challenge of preserving motion information while optimizing the model's parameters for improved performance.

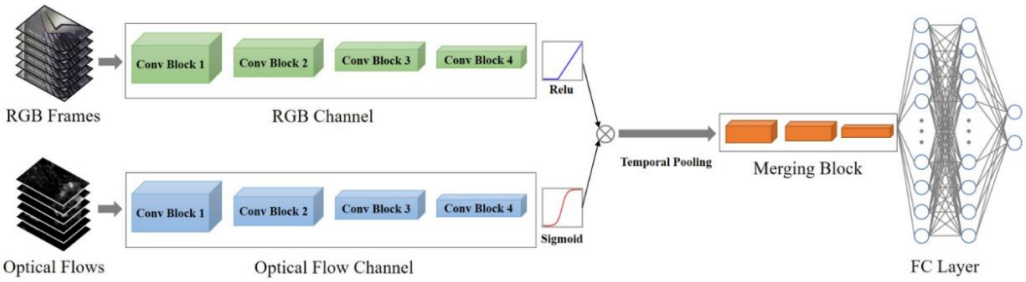


Figure 2.6.1 The strategy of Flow Gate Network

Source: Adapted from [22]

2.6.2 Fast Fight Detection

In reference to Serrano Gracia et al. [23], a method for detecting violent actions has been proposed based on the detection of motion blobs. The process begins by loading and converting the frame from RGB to a grayscale image. Subsequently, the absolute image difference between consecutive frames is computed. This calculation results in an image that is then binarized, leading to the identification of a number of motion blobs.

The next step involves selecting the K largest motion blobs and characterizing them using various distinct features, including area, centroid, parameter, and the distance between blob centroids. Finally, the largest blobs are classified and marked as either part of a fight sequence or a no-fight sequence.

This method offers a systematic approach to violence detection by analysing motion blobs and their characteristics in consecutive frames.

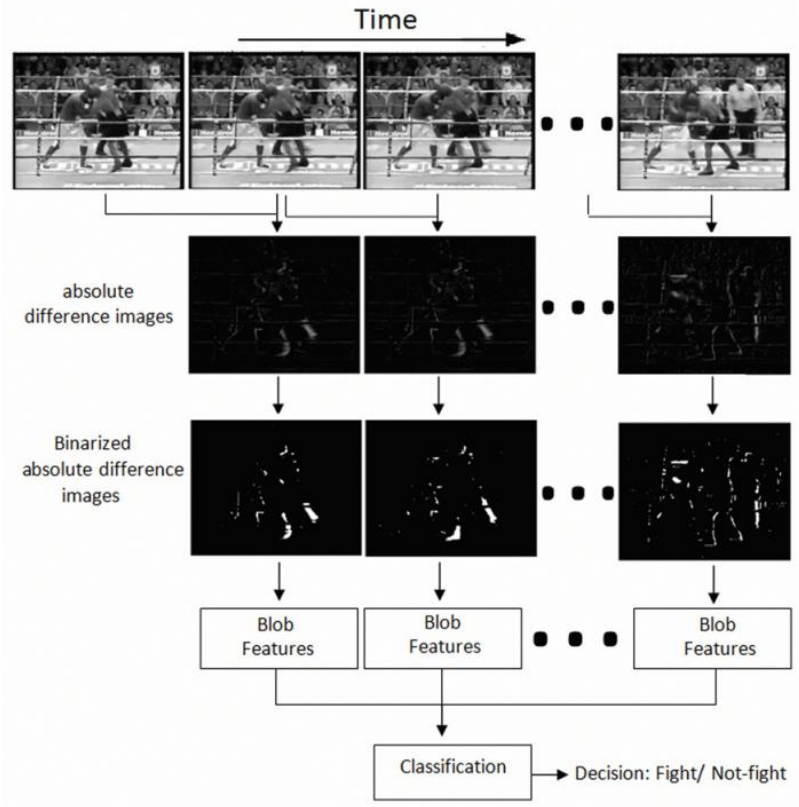


Figure 2.6.2 General diagram of proposed method based on detection motion blobs
Source: Adapted from [23]

2.6.3 Violence detection using Oriented Violent Flows

There is a growing body of literature by Gao et al. [24] that introduced a novel feature known as OViF for violence detection. OViF fully utilizes the orientation information of optical flow, a component that is not considered in the ViF method. Furthermore, experiments incorporate feature combination and multi-classifier combination techniques.

To elaborate, the process begins with the listing of three example frames from an input video. Next, there is a visualization of the computation of optical flow for each frame. Subsequently, the calculation of the Histogram of Oriented Optical Flow (HOOF) is explained using these optical flow images. As a result, the proposed OViF feature is derived by quantifying the change frequencies of HOOFs.

In the final step, the combined features, ViF + OViF, achieve state-of-the-art performance in violence detection when using the AdaBoost + Linear-SVM approach. This outcome underscores the effectiveness of the newly introduced OViF features.

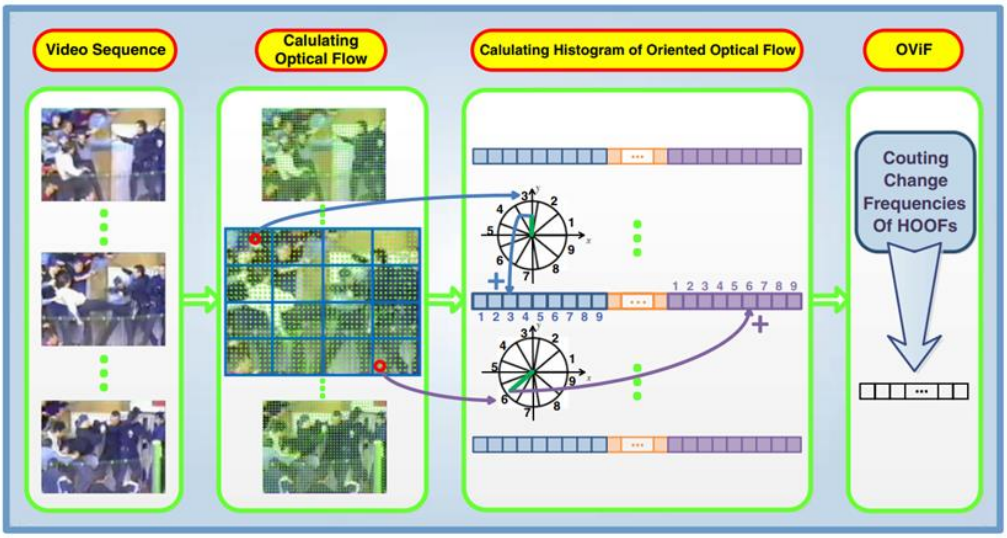


Figure 2.6.3 Flow chart of producing the OViF descriptor

Source: Adapted from [24]

2.6.4 Critical Remarks of Previous Works on Violence Detection

Table 2.6.4 Comparison of previous methods on violence detection

Authors	Dataset	Data Scale	Resolution	Scenario	Accuracy (%)
[22]	RWF-2000	2,000 clips	Variable	Surveillance	87.25
[23]	UCF101, Movies, Hockey	44,121 clips	Variable	Surveillance, Movie, Hockey Games	86.57
[24]	Hockey Fight	1,000 clips	Variable	Hockey Games	87.50

In this comparison, three distinct methods for violence action detection were accessed based on various aspects such as the method employed, the datasets utilized, data scale, content scenarios, and the accuracy of action detection.

Firstly, in reference [22], a technique utilizing a flow-gated network is proposed for violence action detection. This method exhibits strong performance in generic human action recognition, achieving an accuracy rate of approximately 87.25%. However, it is essential to note that video-based violence detection fundamentally hinges on data-

driven solutions. Similarly, in [23], a method relying on motion blob detection is explored, showing poor performance in accurately identifying violent scenes. One limitation of this approach, though, is its suboptimal performance in classifying videos featuring continuous movement, such as those involving moving clouds or tree branches swayed by the wind. While method proposed by [24] to detect violence action using the combination of OiF and OViF is a good choice for non-crowded violence detection applications. However, OViF may not be a decent selection for the violence detection of crowded scenes, hen showing that the performance of OViF is not very satisfy.

In summary, these three methods offer varying degrees of success in violence action detection, depending on factors like data characteristics and the complexity of the content scenario. While the flow-gated network achieves high accuracy in generic human action recognition, the motion blob detection method struggles with videos featuring continuous movement. On the other hand, the combination of OiF and OViF shows promise in non-crowded violence detection scenarios but falls short in crowded scenes. It's essential to consider these nuances when choosing an appropriate method for violence action detection in specific applications.

Chapter 3
System Approach

3.1 System Development Methodology

The project employed Rapid Application Development (RAD) as the development methodology. The development processes can be classified into four distinct phases: requirements planning, user design, construction, and cutover.

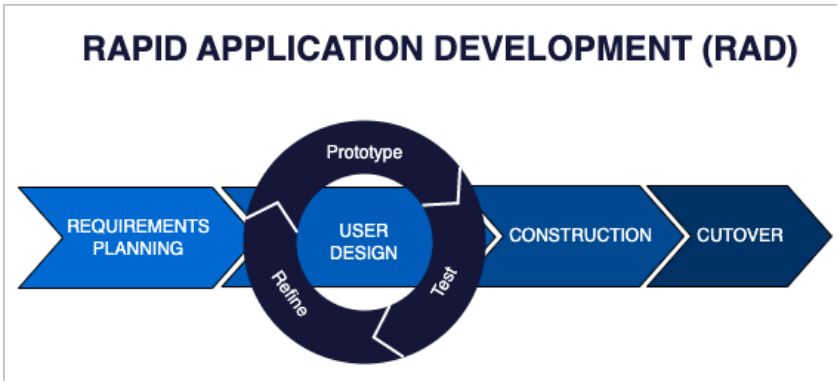


Figure 3.1 The flow of Rapid Application Development (RAD) Methodology
Source: Adapted from [25]

Table 3.1 explain the activities performed in each of the development phases.

Table 3.1 The detailed process of four development phases

Phase	Description
Phase 1: Requirement planning	Firstly, the problem statement was defined to establish clear and measurable objectives. An extensive review of relevant literature was conducted to understand user needs for a mobile live streaming application. Based on the research, the required features for the application were outlined. The scope and objectives of the project were defined, focusing on the development of user-friendly features.
Phase 2: User Design	In user design phase, the application is designed based on the features outlined in the literature review. Wireframes and mock-ups were created to visualize the application's user interface. The visual elements of the application, including icons, colours, and typography, were considered, and worked on. Finally, a clickable prototype was developed to test the application's usability.
Phase 3: Construction	During the construction phase, the application was developed using tools such as Android Studio as the integrated development environment (IDE), Firebase as the database and APIs. Feedback and reviews were collected at this stage and most bugs, issues, and alterations were addressed.
Phase 4: Cutover	In the final stage, the developed application was rigorously tested on an Android mobile device to ensure that all requirements are met and that all functionalities in each module work seamlessly without errors. After testing and maintaining, the finalized application was deployed to the physical device.

RAD methodology is chosen for this project because of its concept that allowed rapid generation of functional prototype. The flexibility and adaptability of the RAD methodology facilitated easy modification of requirements during the development process. Additionally, as the implementation time for this project was approximately six months, the RAD methodology allows the development of the product to be completed faster within such a short period. The RAD methodology also ensures that the project is aligned with its objectives and that the product's functionality is correctly delivered to the end user. In short, RAD practices are beneficial and effective for this mobile application development project.

3.2 System Requirements

3.2.1 Functional Requirements

The functional requirements of the project are specific to:

User Perspective:

- Users and administrators shall be able to access to the application by registering an account.
- Users shall be able to view newest live streams, streamed videos, news, and posts.
- Users shall have the capability to start a live stream by turning on their device's camera.
- Users shall have the capability to upload prerecorded videos in addition to live streams.
- Users shall have the option to add tournament schedules to the calendar.
- Users shall be able to view the leaderboard list sorted by the number of subscribers.
- Users shall be able to receive and view the latest esports news.
- Users shall be able to create posts and engage with the community through actions such as liking, commenting, and sharing.
- Users shall be able to customize content preferences and view relevant information based on their interests.
- Users shall be able to search for videos, news, posts, and users they interested in.
- Users shall be able to view other user's profile and subscribe to them.
- Users shall be able to chat with other users by sending and receiving messages.
- Users shall be able to receive notifications from esports team regarding their feedback, report, and explicit content created.
- Users shall be able to submit feedback regarding application errors for improvement.
- Users shall have access to their accounts where they can view and modify account information, live stream videos, and posted content.

- Administrators shall have the ability to manage user feedback, reports, and explicit content.
- Administrators shall have the ability to manage and add new staffs.
- Users and administrators shall have the ability to log out of their accounts, effectively ending their session with the application.

System Perspective:

- The system shall provide registration options for users to create account.
- The system shall display the newest live streams, streamed videos, news, and posts on user interface.
- The content shall be categorized and displayed in an organized manner for easy access and navigation.
- The system shall allow users to initiate live streams by activating their device camera.
- The system shall require user's device to grant permission for camera, audio and video.
- The system shall support various video formats.
- The system shall display tournament event schedules along with date and time information.
- The system shall sort and display the leaderboard list based on the number of subscribers.
- The system shall deliver the latest esports news to users in a timely manner.
- The system shall facilitate real-time updates and notifications for community interactions.
- The system shall detect characteristics of images and validate both the image and thumbnail to ensure that the content of videos and posts is not explicit.
- The system shall hide explicit content from public view.
- The system shall offer a search feature allowing users to search for specific videos, news, posts, and other users, which search results shall be accurate.
- The system shall provide options for account settings modification.
- The system shall provide channels for users to submit feedback.

- The system shall send notifications to users regarding updates, messages, and feedback.

3.2.2 Non-Functional Requirements

- Usability – The system's user interface must be user-friendly and easy to use.
- Reliability – The system shall consistently deliver reliable functionality to users.
- Security – The system shall verify user information to maintain data integrity and ensure security.

3.3 System Architecture Diagram

The proposed system architecture is based on the client-server model, which the clients (user and admin) interact with the system through the user interface of the mobile application. This client application communicates with various backend components over the Internet. The core backend infrastructure includes a web server, which hosts the web services, serving as the intermediary between the client and other system components. The web server receives HTTP requests from the client and responds with appropriate HTTP responses. It manages user authentication, session handling, and serves dynamic content. Additionally, the system incorporates Ant Media Server, a specialized media server dedicated to handling live video streaming functionalities. The system also includes a database, which stores persistent data such as user profiles, live stream metadata, and application settings. This database facilitates efficient data storage and retrieval for the system. Moreover, the system integrates with external APIs through the web server to access additional functionalities to retrieve tournament events' information. Overall, this architecture enables a distributed and scalable system, ensuring seamless live streaming experiences while maintaining data consistency, reliability, and security.

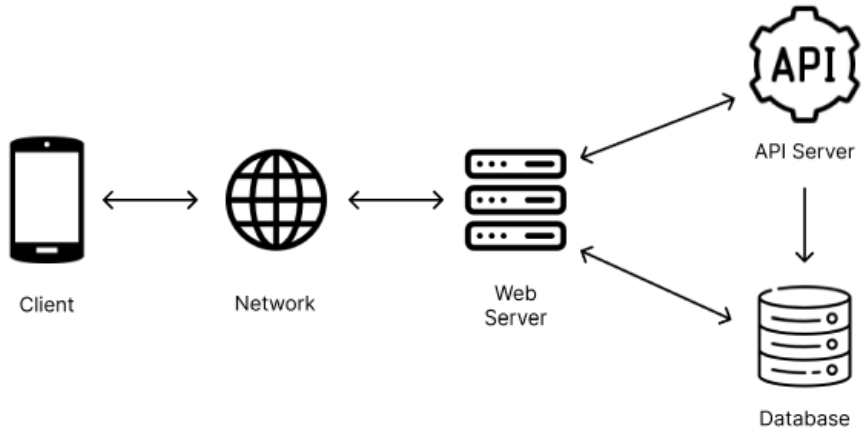


Figure 3.3 The architecture diagram of eStream mobile application system

3.4 Use Case Diagram and Description

3.4.1 Use Case Diagram

This subsection explains the interaction between the end users and the system. Figure 3.2.1 shows the use-case diagram of the eStream mobile application system. There are two actors participating in the system: the user and the admin. The use-case diagram illustrates that the user is allowed to perform use-cases such as creating an account, logging in to the application, streaming video and live stream, starting a live stream and uploading videos, browsing esports news, joining a forum for discussion, chatting with other users, receiving notifications, managing account, submitting feedback, and logging out of the application. Meanwhile, the admin has four main use-cases. The admin is allowed to log in and log out of the system, handle feedback from users, and send notifications to users.

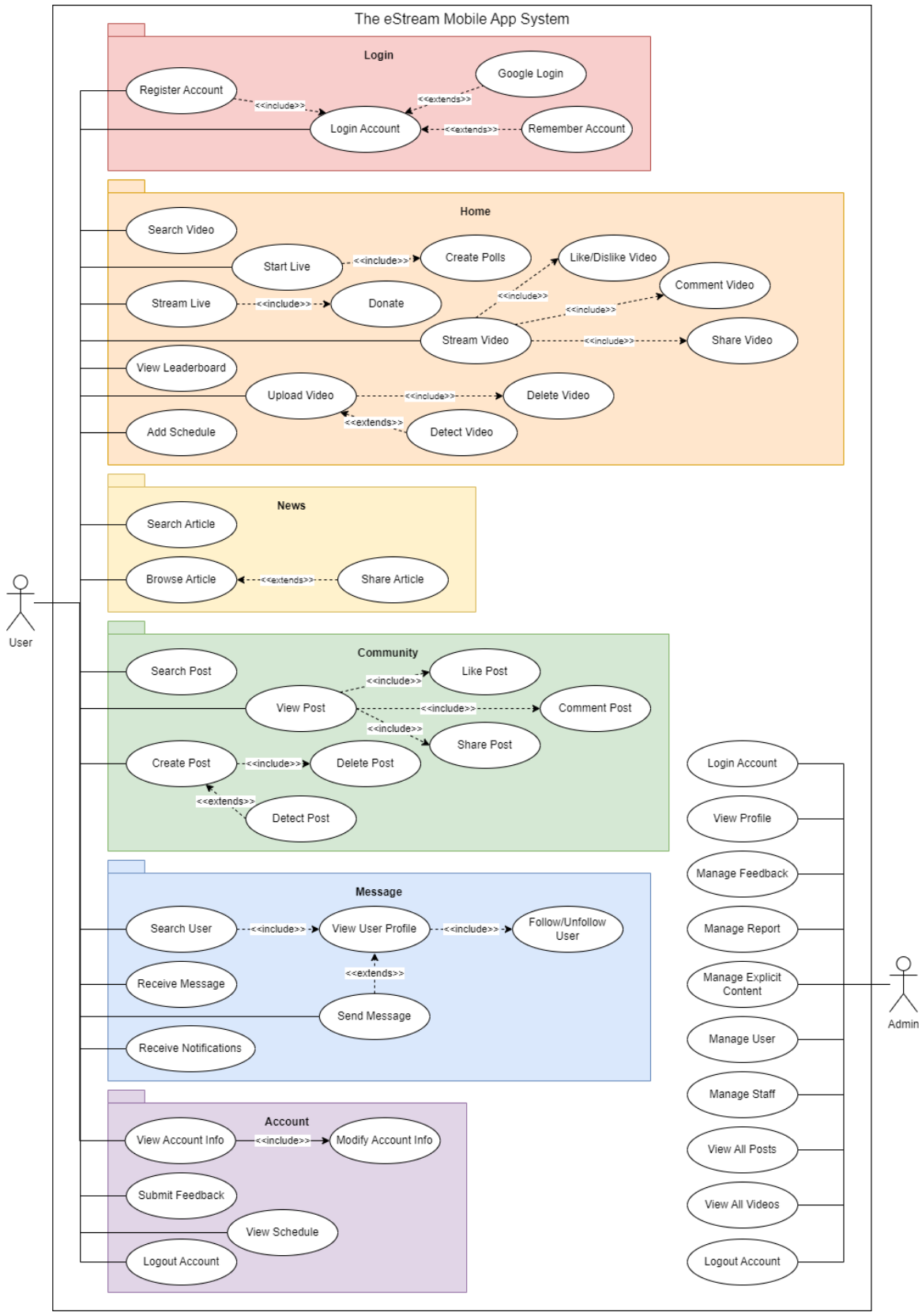


Figure 3.4.1 Use-case diagram of eStream system.

3.4.2 Use Case Description

This subsection presents the use case descriptions of use cases in eStream mobile application system. The use case descriptions explain the complete flow of each use case from the beginning to the end. There is a total of 22 use cases associated with user and 10 use cases associated with admin in eStream mobile application system, as listed below:

User Associated

1. Register Account Use Case
2. Login Account Use Case
3. Search Video Use Case
4. Start Live Use Case
5. Stream Live Use Case
6. Upload Video Use Case
7. Stream Video Use Case
8. View Leaderboard
9. Add Schedule Use Case
10. Search Article Use Case
11. Browse Article Use Case
12. Search Post Use Case
13. Create Post Use Case
14. View Post Use Case
15. Search User Use Case
16. Receive Message Use Case
17. Send Message Use Case
18. Receive Notification Use Case
19. View Account Info Use Case
20. Submit Feedback Use Case
21. View Schedule Use Case
22. Logout Account Use Case

Admin Associated

1. Login Account Use Case
2. View Profile Use Case
3. Manage Feedback Use Case
4. Manage Report Use Case
5. Manage Explicit Content Use Case
6. Manage User Use Case
7. Manage Staff Use Case
8. View All Posts Use Case
9. View All Videos Use Case
10. Logout Account Use Case

User Associated

1. **Register Account Use Case**

Table 3.4.2.1 Use case description of Register Account use case.

Use Case Name: Register Account	**ID:** 1	**Importance Level:** High
Primary Actor: User, admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User, admin – wants to create an account.		
Brief Description: This use case describes how user and admin create an account.		
Trigger: User, admin wants to create a new account. **Type:** External		
Relationship: Association : User, admin Include : Login Account Extend : Not applicable Generalization : Not applicable		
Preconditions: • User or admin is not registered in the system.		
Postconditions: • User or admin account is registered and verified. • User or admin can log in using registered credentials.		
Normal Flow of Events: 1. The user or admin initiates registration. 2. The user or admin enters registration information (username, email, password, and confirmation password). 3. The user or admin selects Create button. 4. The system validates registration information. 5. The system validates email address against user database. 6. The system creates a new user account. 7. The system assigns a unique identifier and store the new user account. 8. The system redirects user or admin to the Login page.		
Sub Flows: Not applicable		
Alternate Flow: 4a. The input information is invalid; the system displays message error messages and prompt the user or admin to enter correct details. 5a. The account already exists; the system displays message “Create Failed” and prompt the user or admin to choose a different email. 6a. The system displays a message “Create Successfully” to the user or admin.		

2. Login Account Use Case

Table 3.4.2.2 Use case description of Login Account use case.

Use Case Name: Login Account	**ID:** 2	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – want to log in to account.		
Brief Description: This use case describes how do user login to their account.		
Trigger: User wants to login to the application to access eSports content. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Google Login, Remember Account Generalization: Not applicable		
Preconditions: • User has registered for an account. • User is not logged in.		
Postconditions: • User is logged in.		
Normal Flow of Events: 1. The user requests login. 2. The system displays options for email/password and Google login. 3. If the user enter email, password, login button is pressed: a. The system checks the entered credentials against the stored user database. 4. If Google login selected: a. The system redirect user to Google login interface. b. The user selects their Google account. c. The system checks the credentials of selected account against the stored user database. 5. If 'Remember Me' selected, the system generates a long-lasting authentication token. 6. The system authenticates the user based on credentials or Google authentication. 7. The system redirects user to the Home page.		
Sub Flows: Not applicable		
Alternative Flow: 3a1.The credentials are invalid; the system display message "Login Failed" and prompt the user to enter valid credentials.		

4c1.The selected Google account is invalid; the system display message “Login Failed” and prompt the user to try again. 6a. The system displays a message “Successfully Login” to the user.

3. Search Video Use Case

Table 3.4.2.3 Use case description of Search Video use case.

Use Case Name: Search Video	**ID:** 3	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to search for a video.		
Brief Description: This use case describes how user search for a video.		
Trigger: User wants to search and stream video. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Home page.		
Postconditions: • The user is able to view the search result and stream the video.		
Normal Flow of Events: 1. The user selects the Search icon. 2. The system provides a video search bar, allowing the user to enter search input. 3. The user enters keyword for searching. 4. The user selects Search button. 5. The system validates if the entered search query is empty. 6. The system queries the Video database and Label Detection database to retrieve relevant video results. 7. The system presents a list of video results matching the search query. 8. The user selects a video from the displayed search results. 9. The user initiates stream video use case.		
Sub Flows: Not applicable		
Alternative Flow: 5a. The search input is empty; the system prompt user to enter a valid input. 6a. The search query yields no matching results; the system displays “No result found” message.		

4. Start Live Use Case

Table 3.4.2.4 Use case description of Start Live use case.

Use Case Name: Start Live	**ID:** 4	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to upload a start livestream.		
Brief Description: This use case describes how user start livestream.		
Trigger: User wants to start livestream. **Type:** External		
Relationship: Association : User Include : Not Applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Home page.		
Postconditions: • The user's video is successfully uploaded and visible to public in Home page.		
Normal Flow of Events: 1. The user selects the Start Live button. 2. The system provides a live details interface with fields for text (title and description), media upload (image), and label. 3. The user composes the text content and attach the thumbnail for uploading. 4. The user selects Start button. 5. The system validates the entered video content, ensuring that none of the field is empty. 6. The system saves the live details to database. 7. The system asks user to grant permission for camera, audio, and video. 8. The user grants permission for camera, audio, and video. 9. The system starts broadcasting by turning on the camera and audio feature.		
Sub Flows: Not applicable		
Alternative Flow: 5a. There exists empty field; the system display message error message and prompt user to insert the input.		

5. Stream Live Use Case

Table 3.4.2.5 Use case description of Stream Live use case.

Use Case Name: Stream Live	**ID:** 5	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to join a live stream.		
Brief Description: This use case describes how user can join a live stream.		
Trigger: User wants to join a live stream. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the live streaming platform.		
Postconditions: • The user has successfully joined a live stream.		
Normal Flow of Events: 1. The user selects a live stream from the available content library or search result. 2. The system fetches details of the selected live, including its title, description, and streaming source. 3. The system displays the fetched details. 4. The user views the live stream and interact with streamer by liking, commenting, and sharing.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

6. Upload Video Use Case

Table 3.4.2.6 Use case description of Upload Video use case.

Use Case Name: Upload Video	**ID:** 6	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to upload a pre-recorded video.		
Brief Description: This use case describes how user upload a pre-recorded video.		
Trigger: User wants to upload a pre-recorded video. **Type:** External		
Relationship: Association : User Include : Delete Video Extend : Not applicable Generalization: Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Home page.		
Postconditions: • The user's video is successfully uploaded and visible to public in Home page.		
Normal Flow of Events: 1. The user selects the Upload Video button. 2. The system provides a video upload interface with fields for text (title and description), media upload (image and video), and label. 3. The user composes the text content and attach the thumbnail and video for uploading. 4. The user selects Upload button. 5. The system validates the entered video content, ensuring that none of the field is empty. 6. The system saves the video with its associated details. 7. The system verifies if the thumbnail contains explicit content. 8. The system updates the user's feed to include the newly uploaded video.		
Sub Flows: Not applicable		
Alternative Flow: 5a. There exists empty field; the system display message error message and prompt user to re-enter the input. 6a. The system displays a message "Successfully Uploaded" to the user. 7a. There exists explicit content; the system displays "Explicit content detected" message and sends notifications to inform user.		

7. Stream Video Use Case

Table 3.4.2.7 Use case description of Stream Video use case.

Use Case Name: Stream Video	**ID:** 7	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to stream a video.		
Brief Description: This use case describes how user can stream a video.		
Trigger: User wants to stream eSports related video. **Type:** External		
Relationship: Association : User Include : Like/Dislike Video, Comment Video, Share Video Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the live streaming platform. • User has selected a video to stream.		
Postconditions: • The user has successfully streamed the selected video.		
Normal Flow of Events: 1. The user selects a video from the available content library or search result. 2. The system fetches details of the selected video, including its title, description, and streaming source. 3. The system displays details of video. 4. The system prepares the video stream for playback. 5. The user selects play button. 6. The user watches the streaming video.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

8. View Leaderboard Use Case

Table 3.4.2.8 Use case description of View Leaderboard use case.

Use Case Name: View Leaderboard	**ID:** 8	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to view the leaderboard.		
Brief Description: This use case describes how user can view the sorted leaderboard.		
Trigger: User wants to view the sorted leaderboard. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform.		
Postconditions: • The user has successfully viewed the sorted leaderboard.		
Normal Flow of Events: 1. The user selects Leaderboard slider. 2. The system redirects user to Leaderboard page. 3. The system retrieves and sorts the user list based on number of subscribers. 4. The system displays the list of users. 5. The user selects image of user. 6. The user views the details of the selected user.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

9. Add Schedule Use Case

Table 3.4.2.9 Use case description of Add Schedule use case.

Use Case Name: Add Schedule	**ID:** 9	**Importance Level:** Low
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to add event to schedule.		
Brief Description: This use case describes how user can add an event to schedule.		
Trigger: User wants to add an event to the schedule. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform.		
Postconditions: • The user has successfully added the event to the schedule.		
Normal Flow of Events: 1. The user selects Discover More option. 2. The system redirects user to Event page. 3. The system displays the list of events. 4. The user selects the Add button of the event. 5. The system stores the selected event to database. 6. The system disables the event's Add button.		
Sub Flows: Not applicable		
Alternative Flow: 4a. The system displays Successfully Added message to user.		

10. Search Article Use Case

Table 3.4.2.10 Use case description of Search Article use case.

Use Case Name: Search Article	**ID:** 10	**Importance Level:** Low
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to search for an article.		
Brief Description: This use case describes how user search for an article.		
Trigger: User wants to search and read news. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to News page.		
Postconditions: • The user is able to view the search result and read the article.		
Normal Flow of Events: 1. The user selects the Search icon. 2. The system provides a news search bar, allowing the user to enter search input. 3. The user enters keyword for searching. 4. The user selects Search button. 5. The system validates if the entered search query is empty. 6. The system queries the News database to retrieve relevant article results. 7. The system presents a list of article results matching the search query. 8. The user selects an article from the displayed search results. 9. The user views the content of article.		
Sub Flows: Not applicable		
Alternative Flow: 5a. The search input is empty; the system prompt user to enter a valid input. 6a. The search query yields no matching results; the system displays “No result found” message.		

11. Browse Article Use Case

Table 3.4.2.11 Use case description of Browse Article use case.

Use Case Name: Browse Article	**ID:** 11	**Importance Level:** Low
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to browse an article.		
Brief Description: This use case describes how user browse an article.		
Trigger: User wants to browse news. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Share News Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to News page.		
Postconditions: • The user has successfully browsed and read an article.		
Normal Flow of Events: 1. The system fetches a list of articles from the content database. 2. The system presents the list of articles to the user. 3. The user selects on a specific article from the list to view its full content. 4. The system retrieves and displays the full content of the selected article. 5. The user reads through the article content at their own pace.		
Sub Flows: Not applicable		
Alternative Flow: 4a. There exist technical issues during the retrieval or display of articles; the system displays error message and prompt user to try again.		

12. Search Post Use Case

Table 3.4.2.12 Use case description of Search Post use case.

Use Case Name: Search Post	**ID:** 12	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to search for a post.		
Brief Description: This use case describes how user search for a post.		
Trigger: User wants to search and view post. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Community page.		
Postconditions: • The user is able to view the search result and view posts.		
Normal Flow of Events: 1. The user selects the Search button. 2. The system provides a post search bar, allowing the user to enter search input. 3. The system validates if the entered search query is empty. 4. The system queries the Post database and Labal database to retrieve relevant article results. 5. The system presents a list of post results matching the search query. 6. The user selects a post from the displayed search results. 7. The user initiates View Post use case.		
Sub Flows: Not applicable		
Alternative Flow: 3a. The search input is empty; the system prompt user to enter a valid input. 4a. The search query yields no matching results; the system displays “No result found” message.		

13. Create Post Use Case

Table 3.4.2.13 Use case description of Create Post use case.

Use Case Name: Create Post	**ID:** 13	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to create a new post.		
Brief Description: This use case describes how user create a new post.		
Trigger: User wants to create a new post in Community. **Type:** External		
Relationship: Association : User Include : Delete Post Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Community page.		
Postconditions: • The user's post is successfully created and visible to public in the feeds.		
Normal Flow of Events: 1. The user selects the Create Post button. 2. The system provides a post creation interface with fields for text (content), media upload (image), and label. 3. The user composes the text content and attach the image of the post. 4. The user selects Create button. 5. The system validates the entered post content, ensuring that none of the field is empty. 6. The system saves the post with its associated details. 7. The system verifies if the image contains explicit content. 8. The system updates the user's feed to include the newly created post.		
Sub Flows: Not applicable		
Alternative Flow: 5a. There exists empty field; the system display error message and prompt user to insert the input. 6a. The system displays a message "Successfully Created" to the user. 7a. There exists explicit content; the system displays "Explicit content detected" message and sends notifications to inform user.		

14. View Post Use Case

Table 3.4.2.14 Use case description of View Post use case.

Use Case Name: View Post	**ID:** 14	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to view posts.		
Brief Description: This use case describes how user views posts.		
Trigger: User wants to view posts by other users. **Type:** External		
Relationship: Association : User Include : Comment Post, Like Post, Share Post Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Community page.		
Postconditions: • The user has successfully viewed the posts.		
Normal Flow of Events: 1. The system retrieves the details of all posts from the database. 2. The system displays the list of posts, including the title, content, and image of each of the post. 3. The user views the post. 4. The user may interact with posts by liking, commenting, and sharing.		
Sub Flows: Not applicable		
Alternative Flow: Not Applicable		

15. Search User Use Case

Table 3.4.2.15 Use case description of Search User use case.

Use Case Name: Search User	**ID:** 15	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to search for a user.		
Brief Description: This use case describes how user search for a user.		
Trigger: User wants to search a user. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Community or Message page.		
Postconditions: • The user is able to view the search result and view posts.		
Normal Flow of Events: 1. The user selects the Search button. 2. The system provides a user search bar, allowing the user to enter search input. 3. The user enters the keywords for searching. 4. The user selects Search button. 5. The system validates if the entered search query is empty. 6. The system queries the User database to retrieve relevant user results. 7. The system presents a list of user results matching the search query. 8. The user selects a user from the displayed search results. 9. The user initiates View User use case.		
Sub Flows: Not applicable		
Alternative Flow: 5a. The search input is empty; the system prompt user to enter a valid input. 6a. The search query yields no matching results; the system displays “No result found” message.		

16. Receive Message Use Case

Table 3.4.2.16 Use case description of Receive Message use case.

Use Case Name: Receive Message	**ID:** 16	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to receive message from another user.		
Brief Description: This use case describes how user receive message from another user.		
Trigger: User wants to receive the message from another user. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform.		
Postconditions: • The user has successfully received and read a message from another user.		
Normal Flow of Events: 1. The system notifies user of the arrival of new message through a notification badge. 2. The user selects the chat room. 3. The system retrieves the chat history from database. 4. The system displays all messages in chat room. 5. The user sees the newest arrived message displayed in the chat interface.		
Sub Flows: Not applicable		
Alternative Flow: Not Applicable		

17. Send Message Use Case

Table 3.4.2.17 Use case description of Send Message use case.

Use Case Name: Send Message	**ID:** 17	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to send message to another user.		
Brief Description: This use case describes how user chat with each other.		
Trigger: User wants to chat with another user. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Message Page and select a user.		
Postconditions: • The user has successfully sent a message to another user.		
Normal Flow of Events: 1. The user (sender) enters the text of message in the message composition area. 2. The user (sender) selects the Send button. 3. The system stores the message into database. 4. The system retrieves the updated chat history from database. 5. The system displays the updated chat history in chat interface. 6. The user (receiver) receives the message and execute Receive Message use case.		
Sub Flows: 1a. The user (sender) selects the Image button. 1b. The user (sender) selects an image from the gallery.		
Alternative Flow: Not applicable		

18. Receive Notification Use Case

Table 3.4.2.18 Use case description of Receive Notification use case.

Use Case Name: Receive Notification	**ID:** 18	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to receive notification from eStream team.		
Brief Description: This use case describes how user receive notification from eStream team.		
Trigger: User wants to receive the notification from eStream team. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform.		
Postconditions: • The user has successfully received and read a notification from eStream team.		
Normal Flow of Events: 1. The system notifies user of the arrival of new notification through a notification badge. 2. The user selects the notification subject. 3. The system retrieves the notification content from Feedback, Report, and Explicit Content database. 4. The system displays the list of notification. 5. The user reads the content of notification.		
Sub Flows: Not applicable		
Alternative Flow: Not Applicable		

19. View Account Info Use Case

Table 3.4.2.19 Use case description of View Account Info use case.

Use Case Name: View Account Info	**ID:** 19	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to receive view their account info.		
Brief Description: This use case describes how user can view their account info.		
Trigger: User wants to view their account info. **Type:** External		
Relationship: Association : User Include : Modify Account Info Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Account page.		
Postconditions: • The user has successfully viewed their account information.		
Normal Flow of Events: 1. The user accesses to Settings page. 2. The user selects and accesses to Account Management page. 3. The system fetches the user's account information from the user database. 4. The system displays the user's account information such as username, email address, profile picture and password. 5. The user reviews the display account information to ensure its accuracy.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

20. Submit Feedback Use Case

Table 3.4.2.20 Use case description of Submit Feedback use case.

Use Case Name: Submit Feedback	**ID:** 20	**Importance Level:** Low
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to submit feedback.		
Brief Description: This use case describes how user can submit feedback.		
Trigger: User wants to submit feedback to admin. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Submit page.		
Postconditions: • The user has successfully submitted the feedback.		
Normal Flow of Events: 1. The system displays options for user to choose the type of feedback. 2. The user selects the feedback type. 3. The system navigates to the selected feedback type page. 4. The system displays text field for user to input. 5. The user enters the feedback content in the provided field. 6. The user selects Submit button. 7. The system checks if the content is empty. 8. The system stores the submitted feedback to database. 9. The admin receives feedback at their site.		
Sub Flows: Not applicable		
Alternative Flow: 7a. The user input is empty; the system displays error message and prompt user to enter the input.		

21. View Schedule Use Case

Table 3.4.2.21 Use case description of View Schedule use case.

Use Case Name: Submit Feedback	**ID:** 21	**Importance Level:** Low
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User – wants to view their schedule.		
Brief Description: This use case describes how user can view calendar.		
Trigger: User wants to view their schedule in the calendar. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Account page.		
Postconditions: • The user has successfully viewed the schedule.		
Normal Flow of Events: 1. The user selects the Calendar button. 2. The system retrieves current date events from Tournament Event database. 3. The system displays a list of events on current day.		
Sub Flows: 2a. The user randomly selects a day of the month. 2b. The system retrieves and display the list of events of the selected date.		
Alternative Flow: Not applicable		

22. Logout Account Use Case

Table 3.4.2.22 Use case description of Logout Account use case.

Use Case Name: Logout Account	**ID:** 22	**Importance Level:** High
Primary Actor: User	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • User– wants to log out of the application.		
Brief Description: This use case describes how user can log out of the application.		
Trigger: User wants to log out of the application. **Type:** External		
Relationship: Association : User Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • User is logged into the platform. • User has navigated to Account page.		
Postconditions: • The user has successfully logged out of the application.		
Normal Flow of Events: 1. The user selects Logout button. 2. The system displays a dialog and prompts user to confirm their intention to log out. 3. The user selects Logout button to confirm logout. 4. The system terminates the user's current session and logs them out of the application. 5. The system redirects the user to Login page.		
Sub Flows: Not applicable		
Alternative Flow: 2a. The user selects Cancel button; the system closes the dialog and terminate the process.		

Admin Associated

1. **Login Account**

Table 3.4.2.23 Use case description of Logout Account use case.

Use Case Name: Login Account	**ID:** 23	**Importance Level:** High
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – want to log in to account.		
Brief Description: This use case describes how do admin login to their account.		
Trigger: Admin want to login to the application to access the management system. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • Admin has registered for an account. • Admin is not logged in.		
Postconditions: • Admin is logged in.		
Normal Flow of Events: 1. The admin requests login. 2. The system displays interface for email/password. 3. Admin enters email, password, login button is pressed: 4. The system checks the entered credentials against the stored user database. 5. The system authenticates the admin based on credentials. 6. The system redirects admin to the Admin Home page.		
Sub Flows: Not applicable		
Alternative Flow: 4a1.The credentials are invalid; the system display message “Login Failed” and prompt the admin to enter valid credentials. 5a. The system displays a message “Successfully Login” to the admin.		

2. View Profile Use Case

Table 3.4.2.24 Use case description of View Profile use case.

Use Case Name: View Profile	**ID:** 24	**Importance Level:** Low
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to view their account information.		
Brief Description: This use case describes how admin can view their account information.		
Trigger: Admin wants to view their account information. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully viewed their account information.		
Normal Flow of Events: 1. The admin selects the side navigation panel button. 2. The admin selects the Profile button. 3. The system retrieves user information from User database. 4. The system displays the account information, including profile picture, username, email address and password. 5. The admin views the account details.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

3. Manage Feedback Use Case

Table 3.4.2.25 Use case description of Manage Feedback use case.

Use Case Name: Receive Feedback	**ID:** 25	**Importance Level:** High
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage feedback from user.		
Brief Description: This use case describes how admin can manage feedback from user.		
Trigger: Admin wants to receive and view feedback from user. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully handled the feedback.		
Normal Flow of Events: 1. The admin selects Feedback button. 2. The system retrieves feedback from feedback database. 3. The system displays the feedback list. 4. The admin selects the feedback. 5. The system displays the feedback content. 6. The admin views the feedback details and handle the feedback. 7. The admin selects Completed button. 8. The system displays confirmation dialog. 9. The admin selects Confirm button. 10. The system updates the feedback status to completed.		
Sub Flows: Not applicable		
Alternative Flow: 10a. The system displays “Updated Successful” message to admin.		

4. Manage Report Use Case

Table 3.4.2.26 Use case description of Manage Report use case.

Use Case Name: Manage Report	**ID:** 26	**Importance Level:** High
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage report from user.		
Brief Description: This use case describes how admin can manage report from user.		
Trigger: Admin wants to receive and view report from user. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully handled the report.		
Normal Flow of Events: 1. The admin selects Report button. 2. The system retrieves report from Report database. 3. The system displays the report list. 4. The admin selects the report. 5. The system displays the report content. 6. The admin views the report details and handle the report. 7. The admin selects Completed button. 8. The system displays confirmation dialog. 9. The admin selects Confirm button. 10. The system updates the report status to completed.		
Sub Flows: Not applicable		
Alternative Flow: 10a. The system displays “Updated Successful” message to admin.		

5. Manage Explicit Content Use Case

Table 3.4.2.27 Use case description of Manage Explicit Content use case.

Use Case Name: Manage Explicit Content	**ID:** 27	**Importance Level:** High
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage explicit content.		
Brief Description: This use case describes how admin can manage explicit content.		
Trigger: Admin wants to deal with explicit content. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully handled the explicit content.		
Normal Flow of Events: 1. The admin selects Explicit Content button. 2. The system retrieves explicit content from Explicit Content database. 3. The system displays the explicit content list. 4. The admin selects the explicit content. 5. The system displays the explicit content details. 6. The admin views the explicit content details. 7. The admin selects Completed button. 8. The system displays confirmation dialog. 9. The admin selects Confirm button. 10. The system updates the explicit content status to completed.		
Sub Flows: Not applicable		
Alternative Flow: 10a. The system displays “Updated Successful” message to admin.		

6. Manage User Use Case

Table 3.4.2.28 Use case description of Manage User use case.

Use Case Name: Manage User	**ID:** 28	**Importance Level:** Low
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage users.		
Brief Description: This use case describes how admin can manage users.		
Trigger: Admin wants to manage user account. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully viewed user info.		
Normal Flow of Events: 1. The admin selects Users button. 2. The system retrieves users from User database. 3. The system displays the users list. 4. The admin selects the checkbox. 5. The system display account passwords in numerical format.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

7. Manage Staff Use Case

Table 3.4.2.29 Use case description of Manage Staff use case.

Use Case Name: Manage User	**ID:** 29	**Importance Level:** Low
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage staffs.		
Brief Description: This use case describes how admin can manage staffs.		
Trigger: Admin wants to manage staffs. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization : Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully managed staffs.		
Normal Flow of Events: 1. The admin selects Staffs button. 2. The system retrieves staffs from Staff database. 3. The system displays the list of staffs. 4. The admin selects the Add button. 5. The admin enters user ID, username, or email address. 6. The system matches the input query with user data. 7. The system displays the list of users. 8. The admin selects the user. 9. The system displays the confirmation dialog. 10. The admin selects the Confirm button. 11. The system adds the user as the new staff.		
Sub Flows: Not applicable		
Alternative Flow: Not applicable		

8. View All Posts Use Case

Table 3.4.2.30 Use case description of View All Posts use case.

Use Case Name: View All Posts	**ID:** 30	**Importance Level:** Low
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage view all posts.		
Brief Description: This use case describes how admin can view all posts.		
Trigger: Admin wants to view all posts. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully viewed all posts.		
Normal Flow of Events: 1. The admin selects Post button. 2. The system retrieves post details from Post database. 3. The system displays the list of posts with post details. 4. The admin views the details of posts. 5. The admin selects the image of a post.		
Sub Flows: 5a. The system displays the image in full size.		
Alternative Flow: Not applicable		

9. View All Videos Use Case

Table 3.4.2.31 Use case description of View All Videos use case.

Use Case Name: View All Videos	**ID:** 31	**Importance Level:** Low
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to manage view all videos.		
Brief Description: This use case describes how admin can view all videos.		
Trigger: Admin wants to view all videos. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • Admin is logged into the platform.		
Postconditions: • The admin has successfully viewed all videos.		
Normal Flow of Events: 1. The admin selects Video button. 2. The system retrieves video details from Video database. 3. The system displays the list of videos with video details. 4. The admin views the details of video. 5. The admin selects the thumbnail of a video. 6. The admin selects the link of a video.		
Sub Flows: 5a. The system displays the image in full size. 6a. The system plays the video in full size.		
Alternative Flow: Not applicable		

10. Logout Account Use Case

Table 3.4.2.23 Use case description of Logout Account use case.

Use Case Name: Logout Account	**ID:** 32	**Importance Level:** High
Primary Actor: Admin	**Use Case Type:** Detail, Essential	
Stakeholders and Interest: • Admin – wants to log out of the application.		
Brief Description: This use case describes how admin can log out of the application.		
Trigger: Admin wants to log out of the system. **Type:** External		
Relationship: Association : Admin Include : Not applicable Extend : Not applicable Generalization: Not applicable		
Preconditions: • User or admin is logged into the platform. • User or admin has navigated to Account page.		
Postconditions: • The user has successfully logged out of the application.		
Normal Flow of Events: 1. The admin selects Logout button in the side navigation panel. 2. The system displays a dialog and prompts admin to confirm their intention to log out. 3. The admin selects Logout button to confirm logout. 4. The system terminates the admin's current session and logs them out of the application. 5. The system redirects the admin to Login page.		
Sub Flows: Not applicable		
Alternative Flow: 2a. The admin selects Cancel button; the system closes the dialog and terminate the process.		

3.5 Activity Diagram

User Associated

1. Register Account Use Case

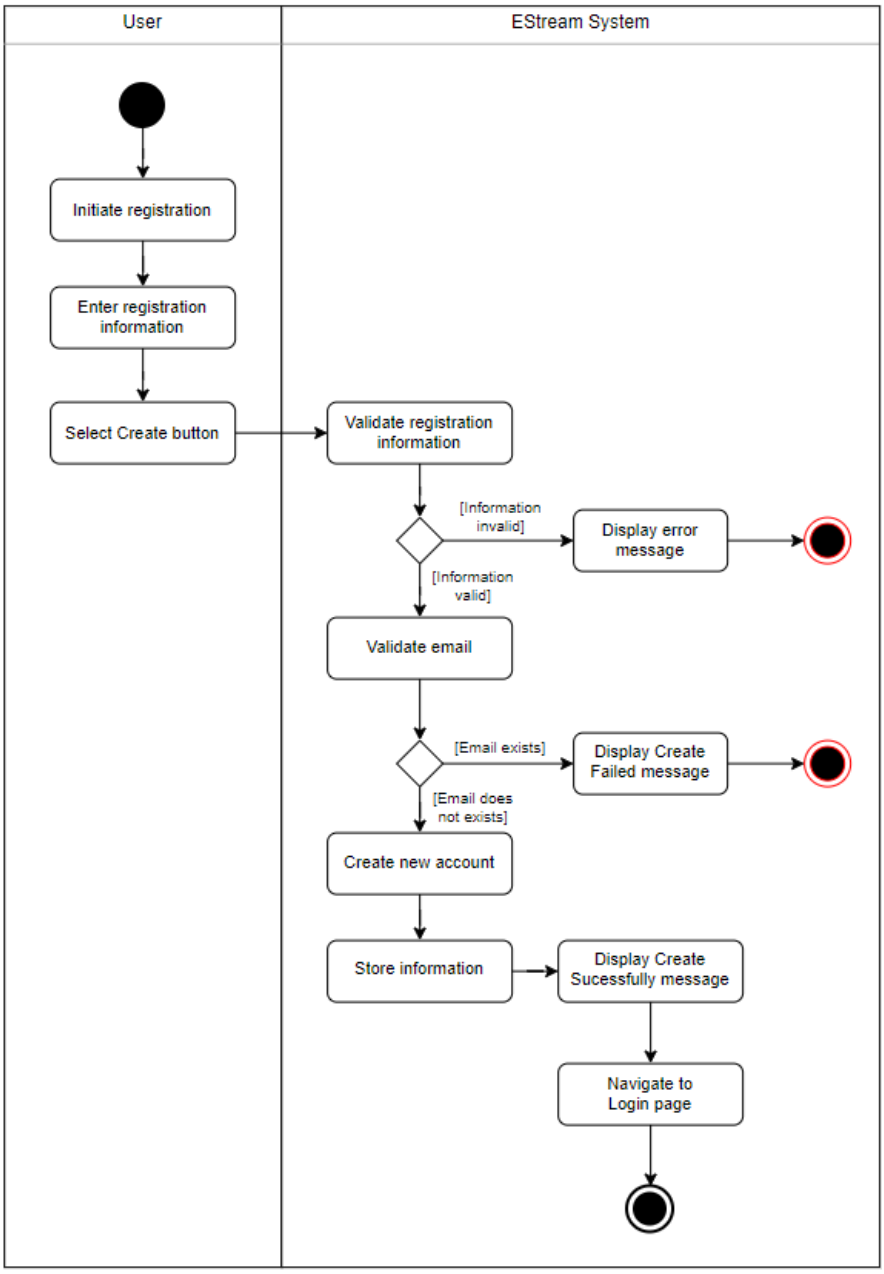


Figure 3.5.1 Activity diagram of Register Account use case.

2. Login Account Use Case

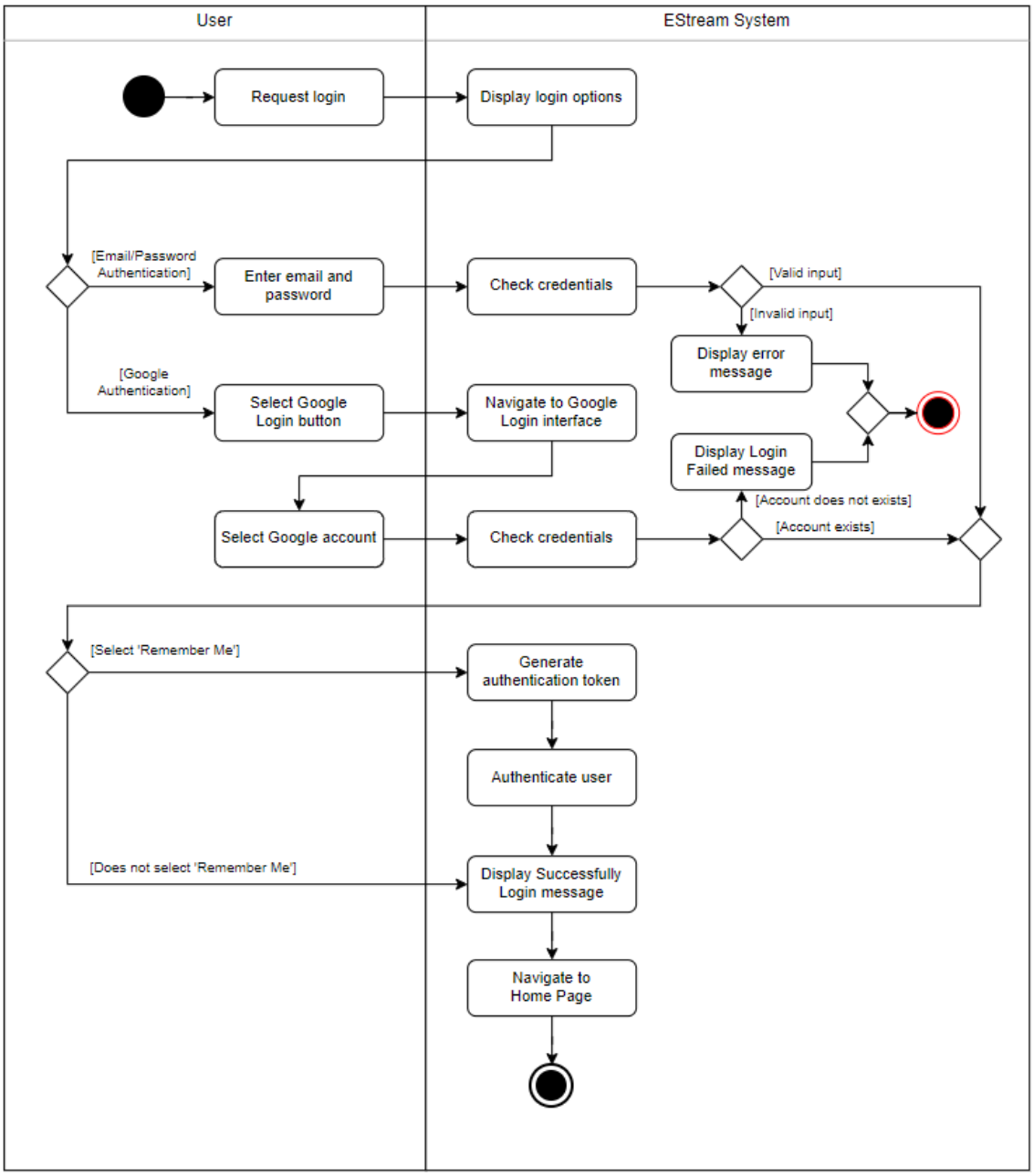


Figure 3.5.2 Activity diagram of Login Account use case.

3. Search Video Use Case

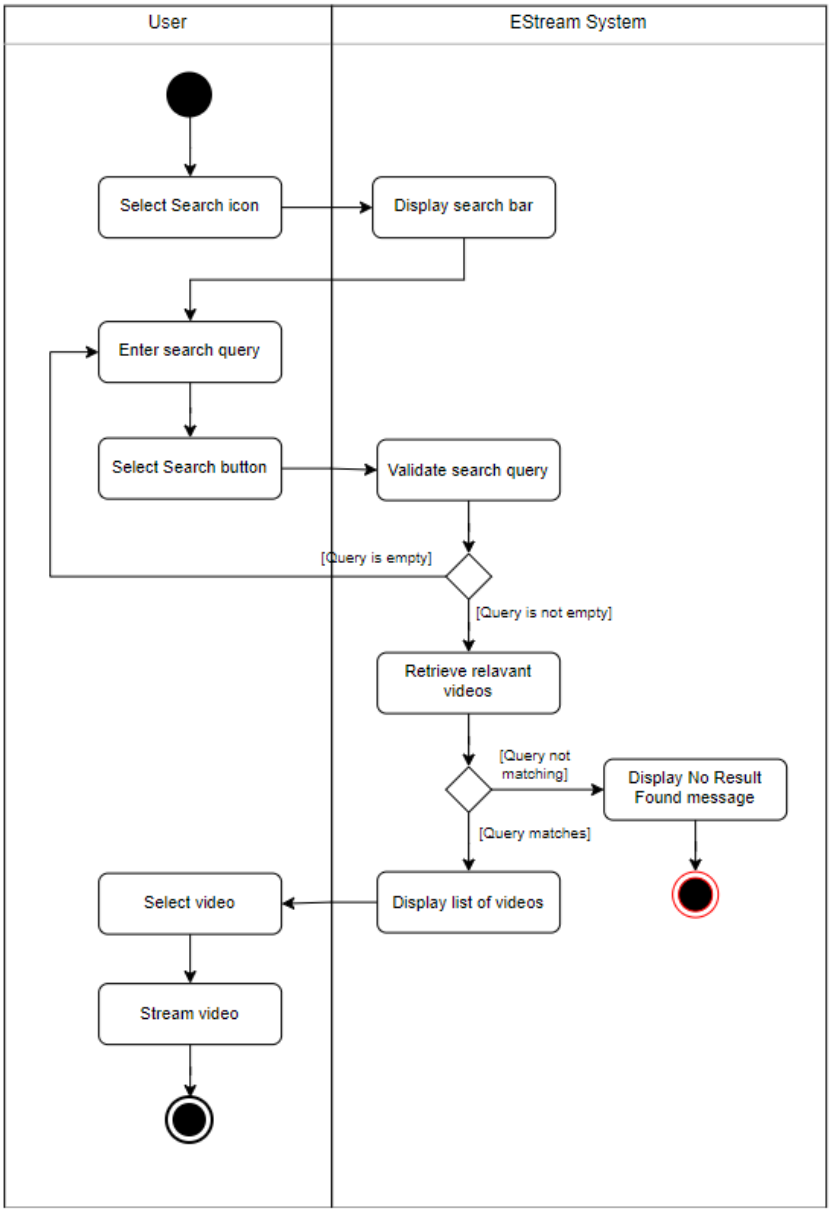


Figure 3.5.3 Activity diagram of Seach Video use case.

4. Start Live Use Case

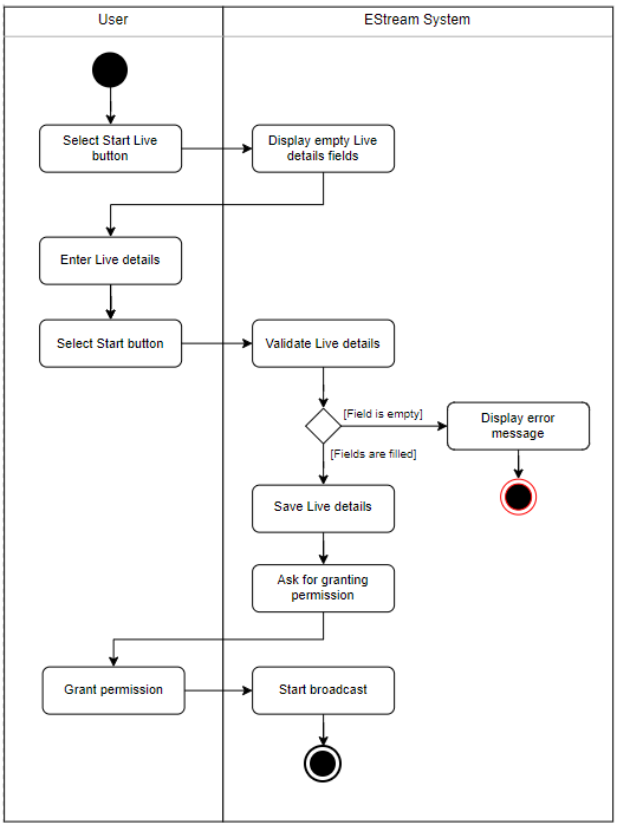


Figure 3.5.4 Activity diagram of Start Live use case.

5. Stream Live Use Case

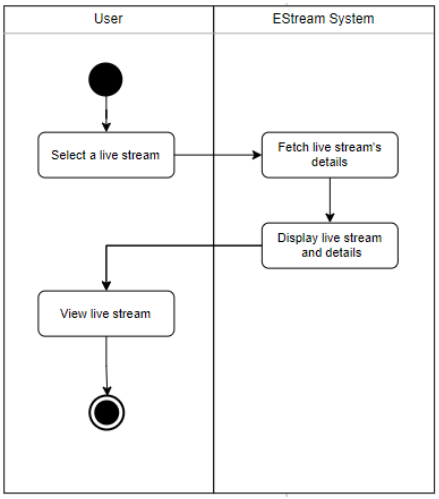


Figure 3.5.5 Activity diagram of Stream Live use case.

6. Upload Video Use Case

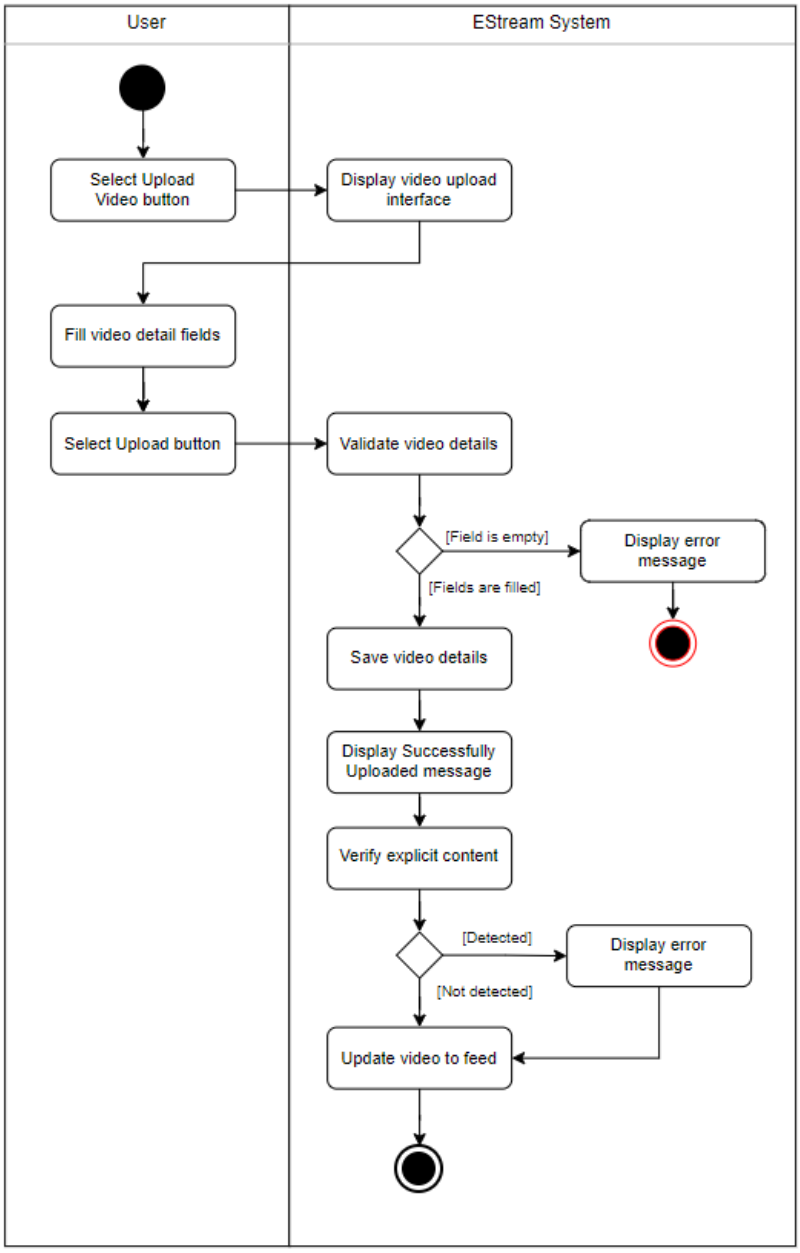


Figure 3.5.6 Activity diagram of Upload Video use case.

7. Stream Video Use Case

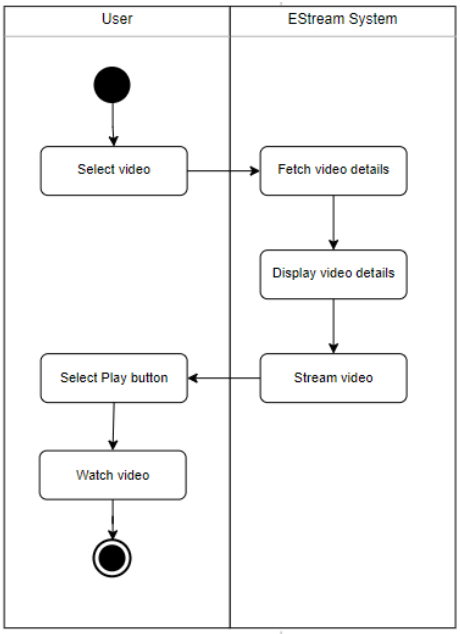


Figure 3.5.7 Activity diagram of Stream Video use case.

8. View Leaderboard Use Case

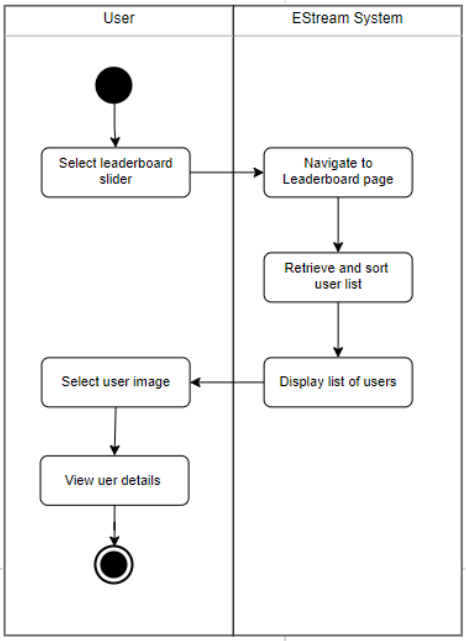


Figure 3.5.8 Activity diagram of View Leaderboard use case.

9. Add Schedule Use Case

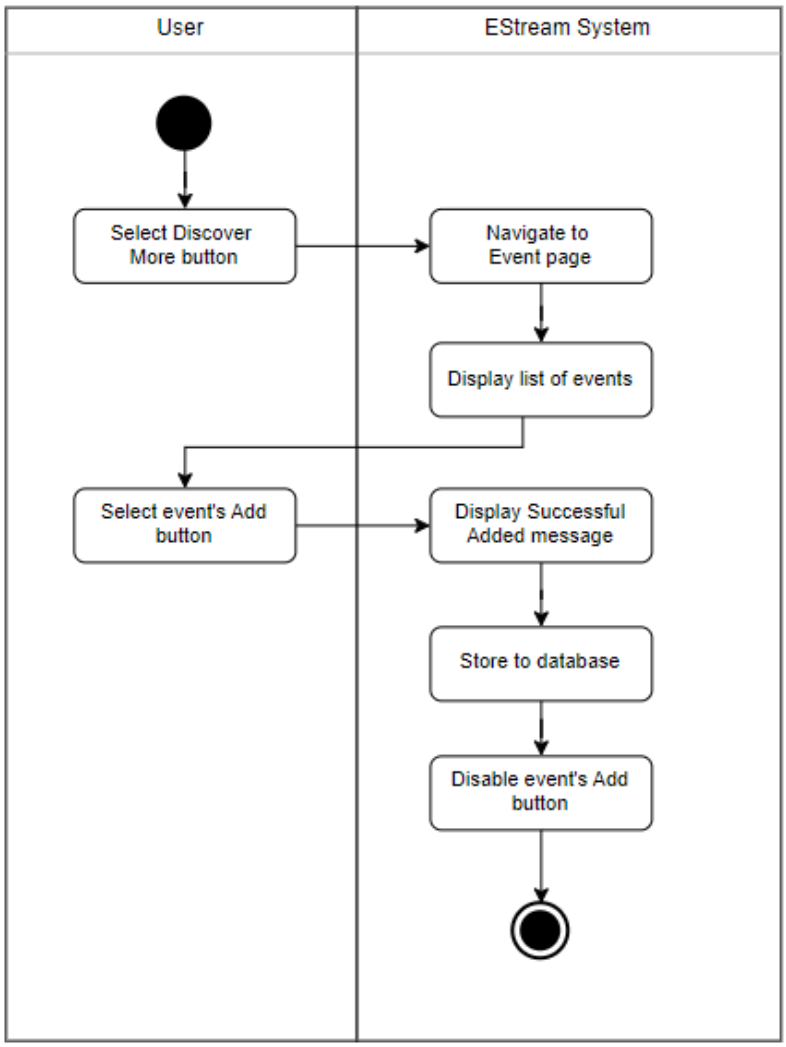


Figure 3.5.9 Activity diagram of Add Schedule use case.

10. Search Article Use Case

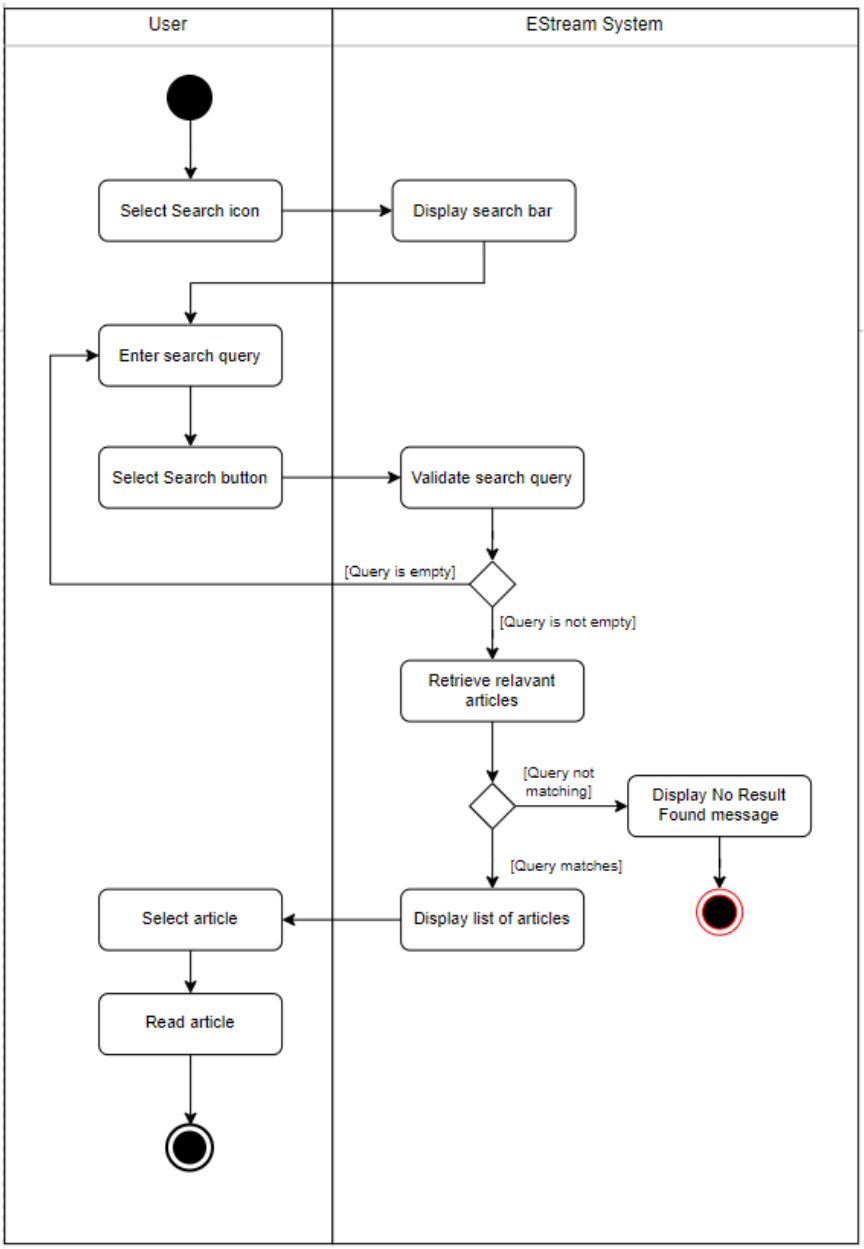


Figure 3.5.10 Activity diagram of Seach Article use case.

11. Browse Article Use Case

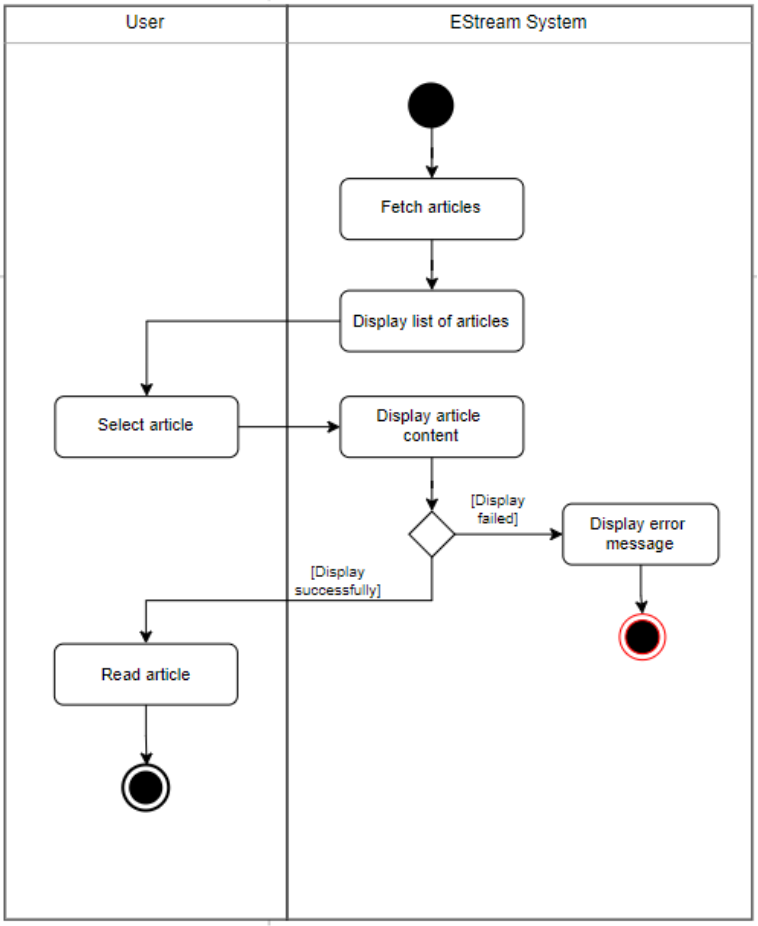


Figure 3.5.11 Activity diagram of Browse Article use case.

12. Search Post Use Case

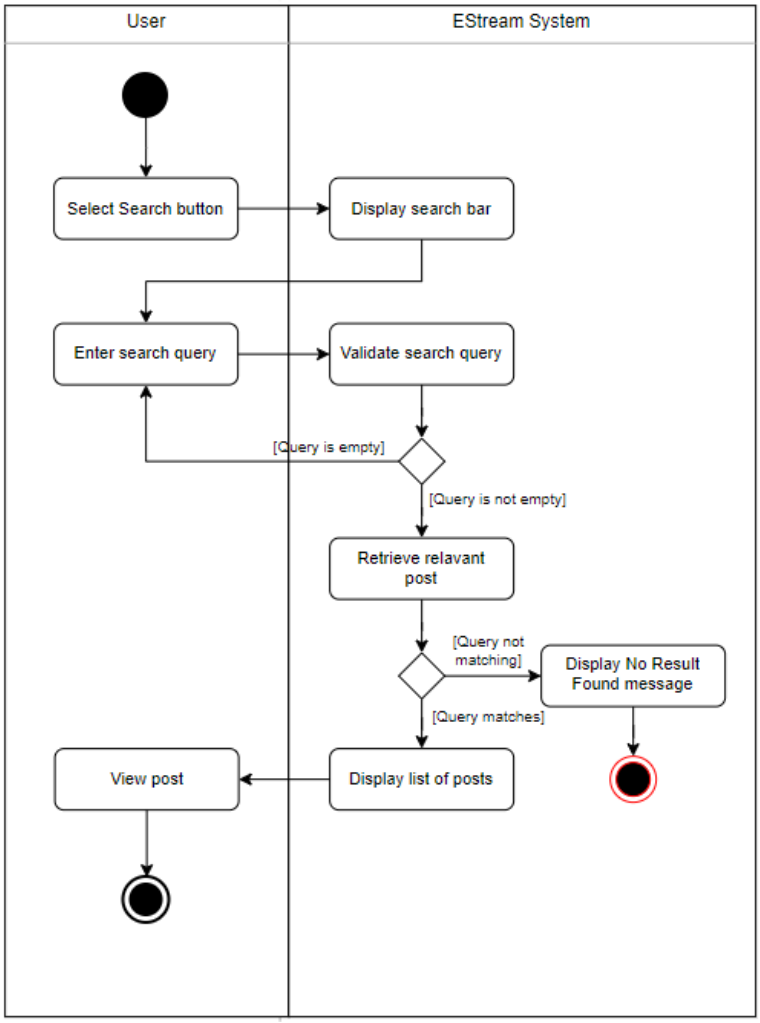


Figure 3.5.12 Activity diagram of Search Post use case.

13. Create Post Use Case

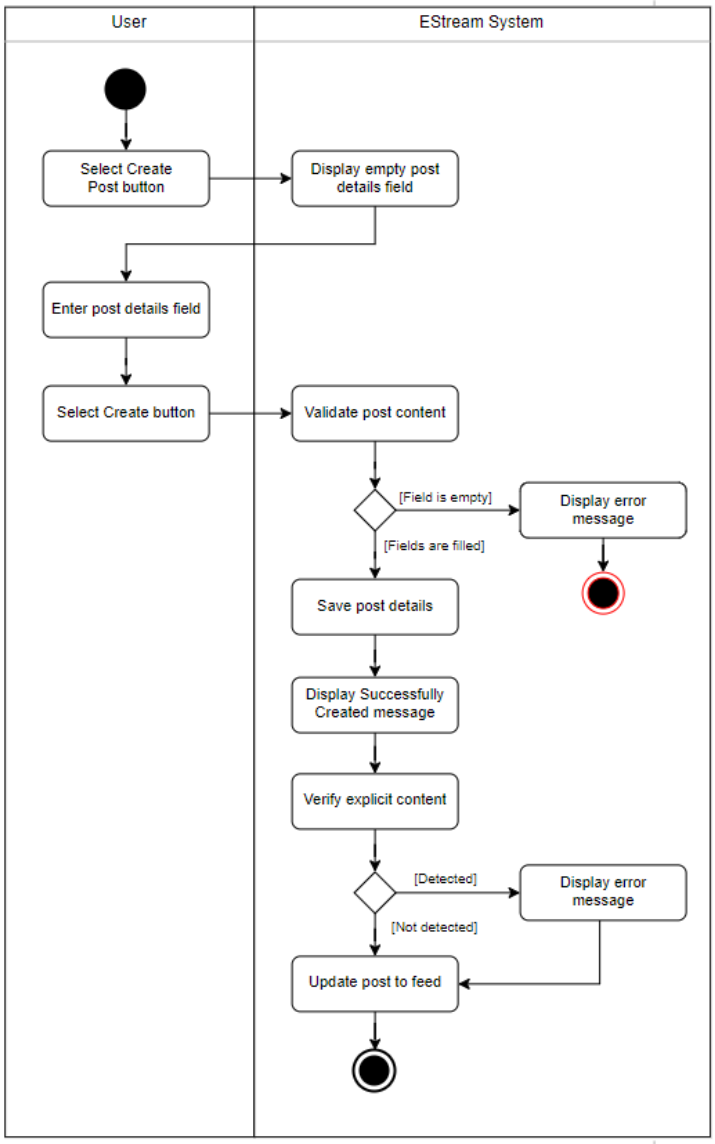


Figure 3.5.13 Activity diagram of Create Post use case.

14. View Post Use Case

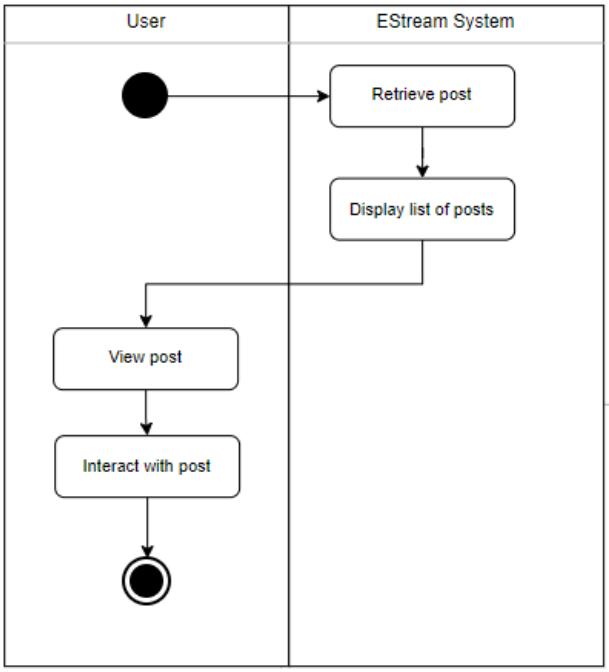


Figure 3.5.14 Activity diagram of View Post use case.

15. Search User Use Case

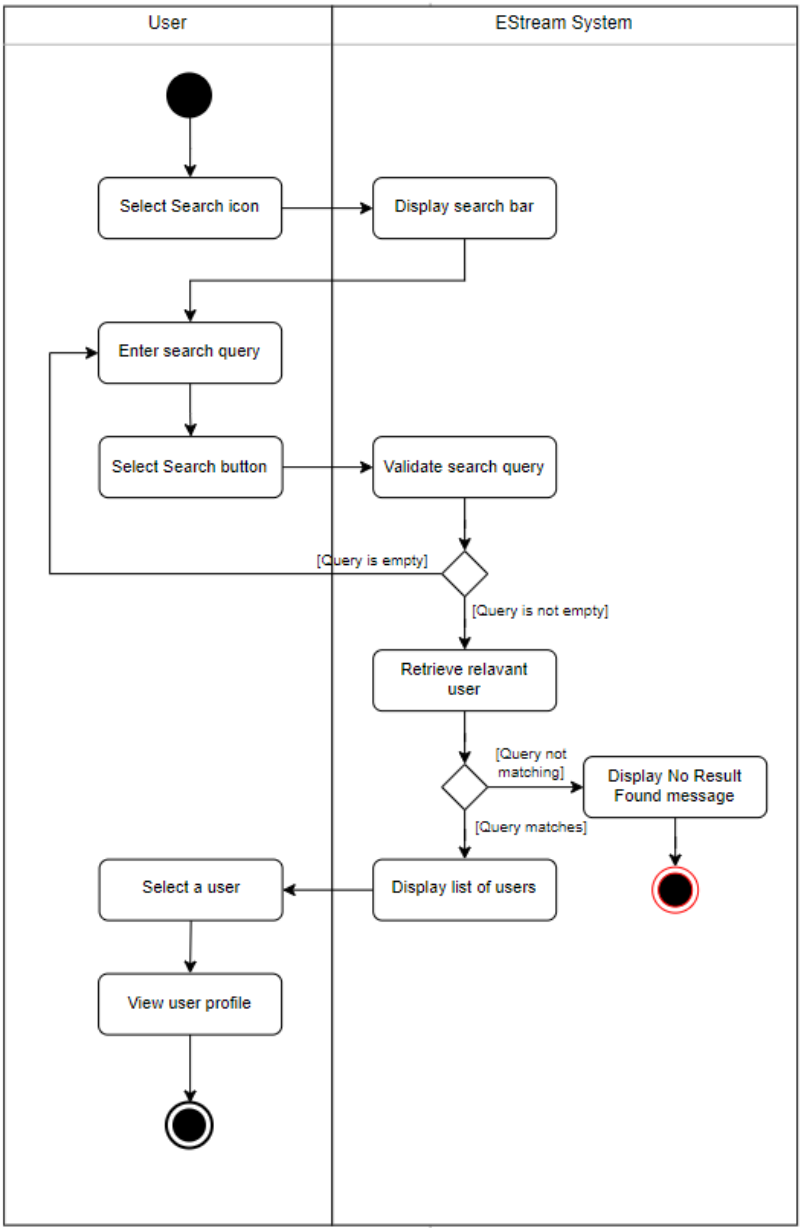


Figure 3.5.15 Activity diagram of Search User use case.

16. Receive Message Use Case

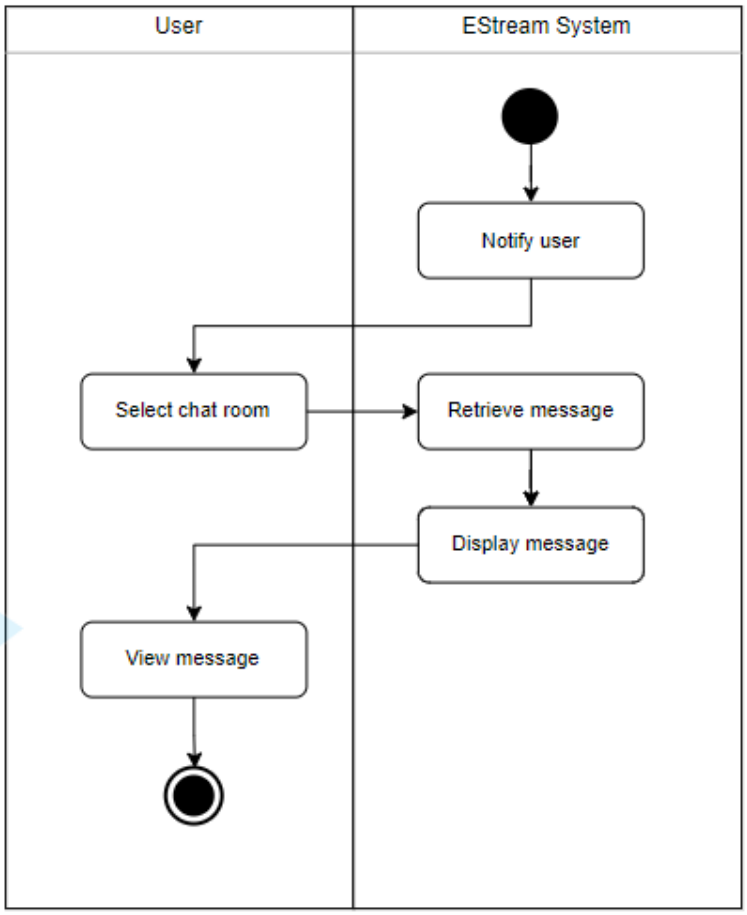


Figure 3.5.16 Activity diagram of Receive Message use case.

17. Send Message Use Case

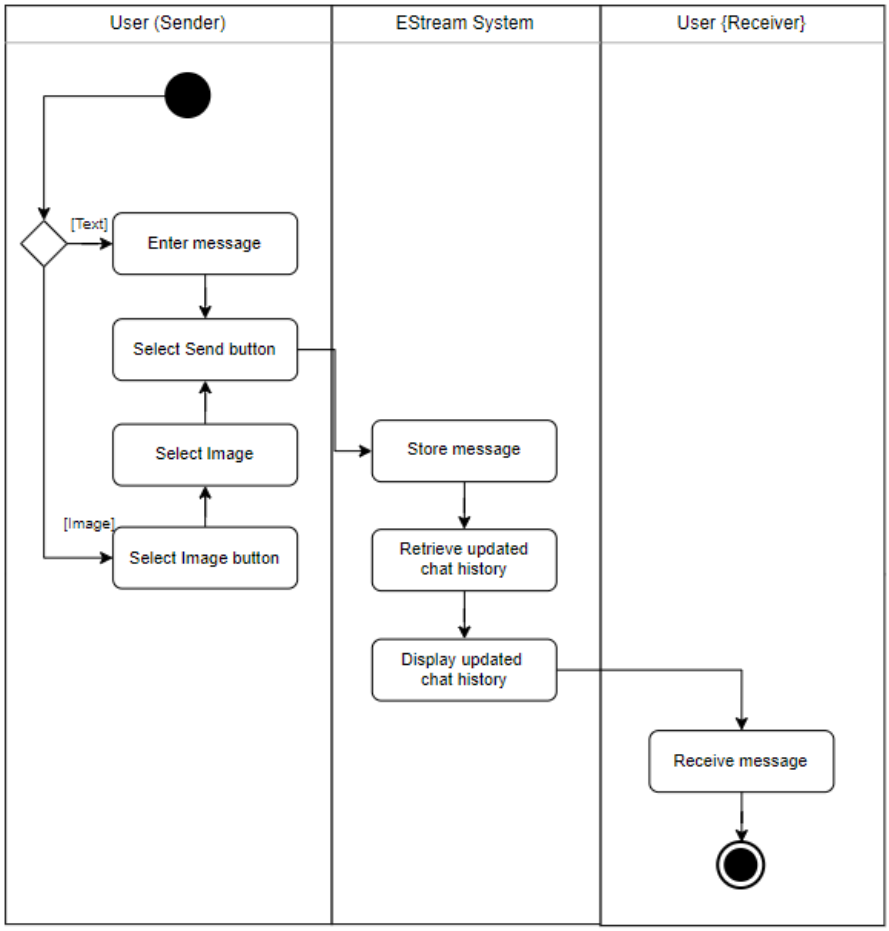


Figure 3.5.17 Activity diagram of Send Message use case.

18. Receive Notification Use Case

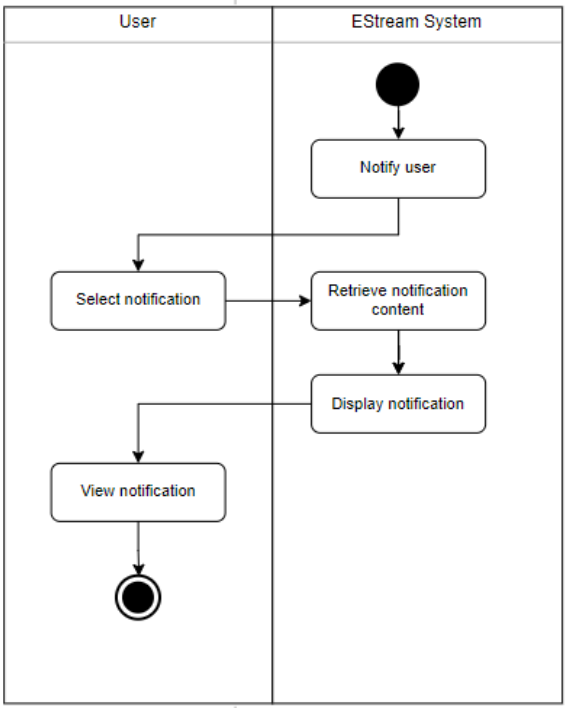


Figure 3.5.18 Activity diagram of Receive Notification use case.

19. View Account Info Use Case

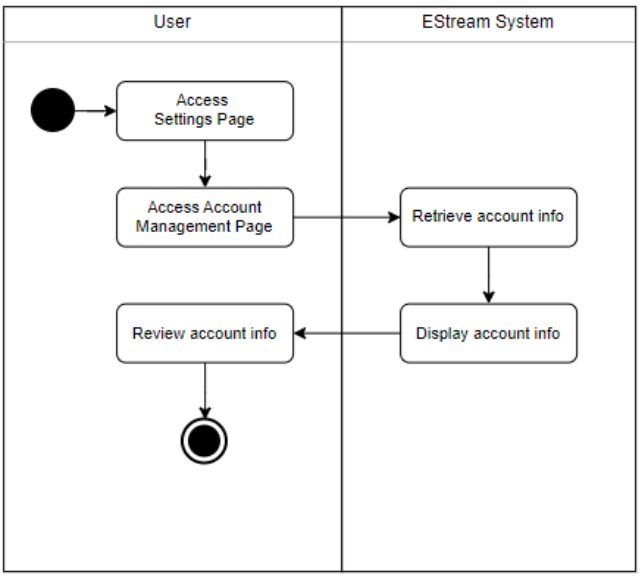


Figure 3.5.19 Activity diagram of View Account Info use case.

20. Submit Feedback Use Case

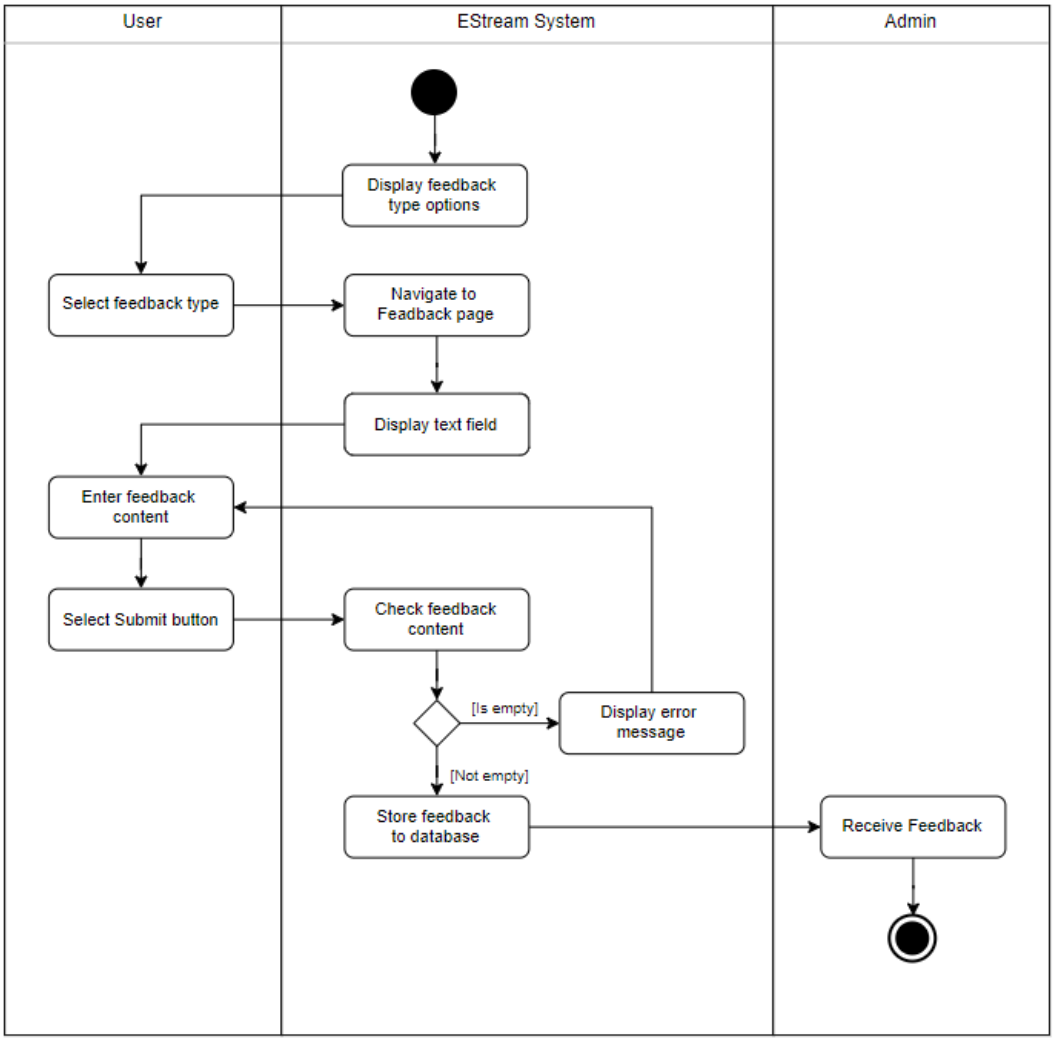


Figure 3.5.20 Activity diagram of Submit Feedback use case.

21. View Schedule Use Case

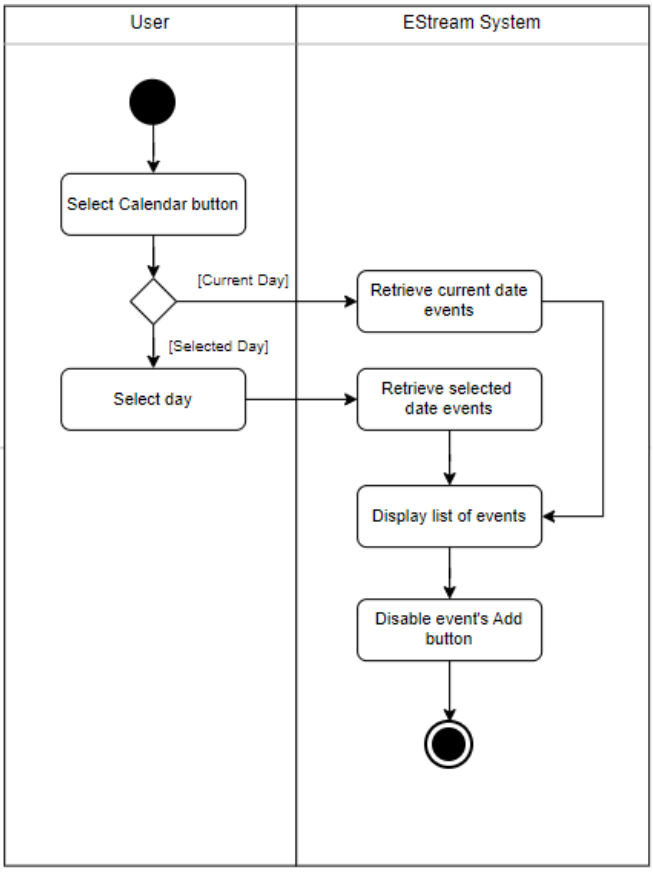


Figure 3.5.21 Activity diagram of View Schedule use case.

22. Logout Account Use Case

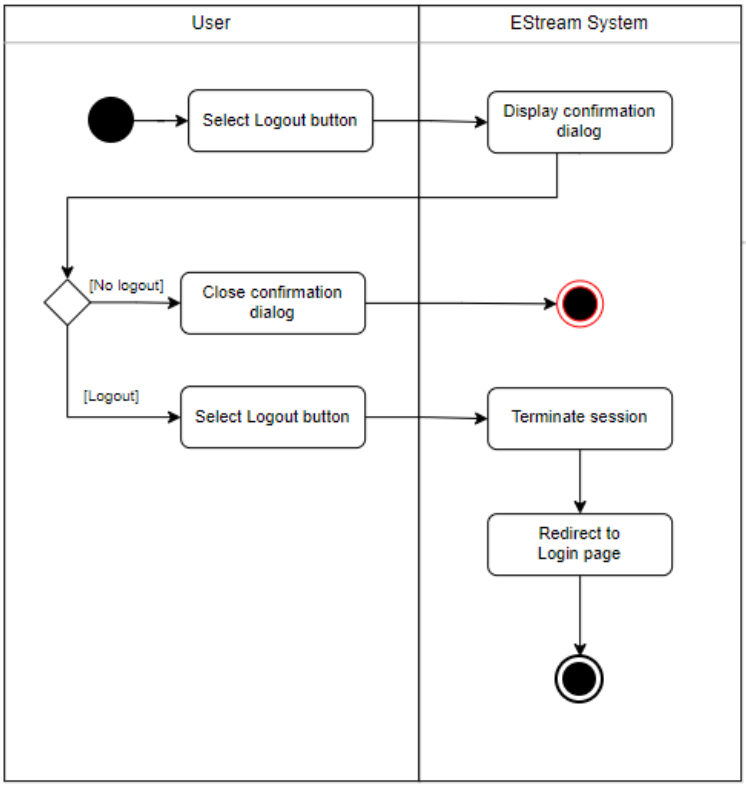


Figure 3.5.22 Activity diagram of Logout Account use case.

Administrator Associated

1. Login Account Use Case

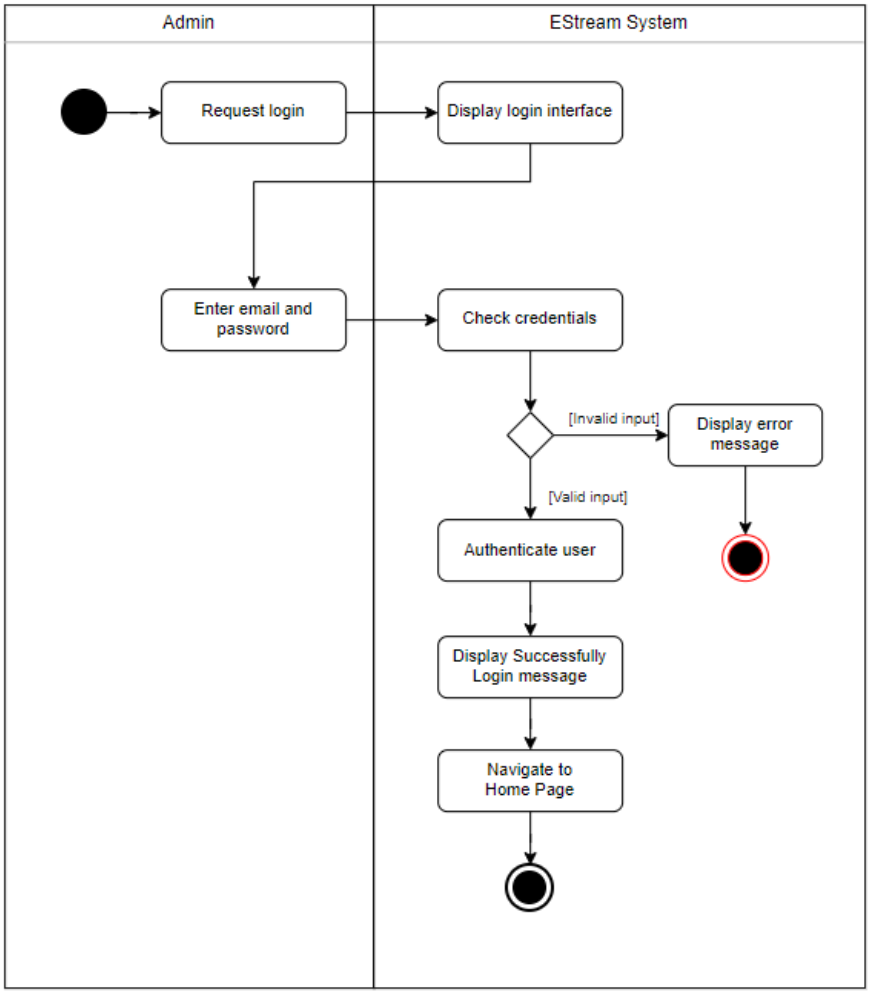


Figure 3.5.23 Activity diagram of Login Account use case.

2. View Profile Use Case

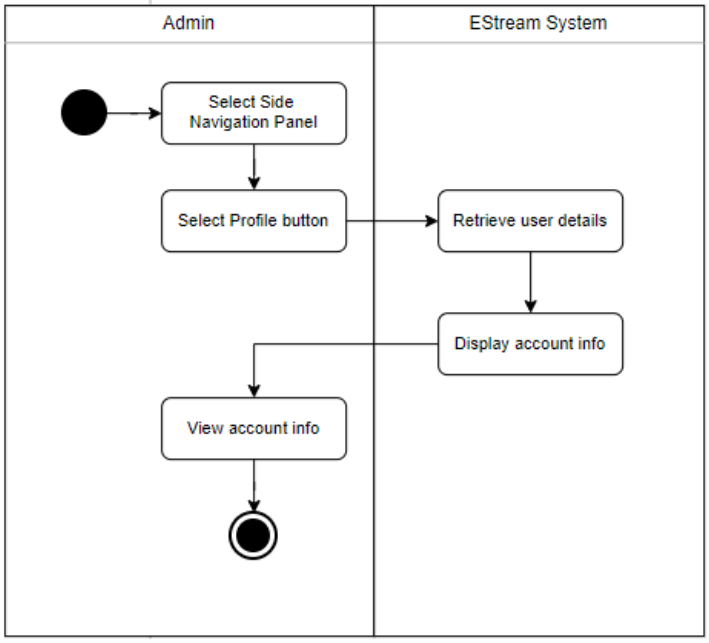


Figure 3.5.24 Activity diagram of View Profile use case.

3. Manage Feedback Use Case

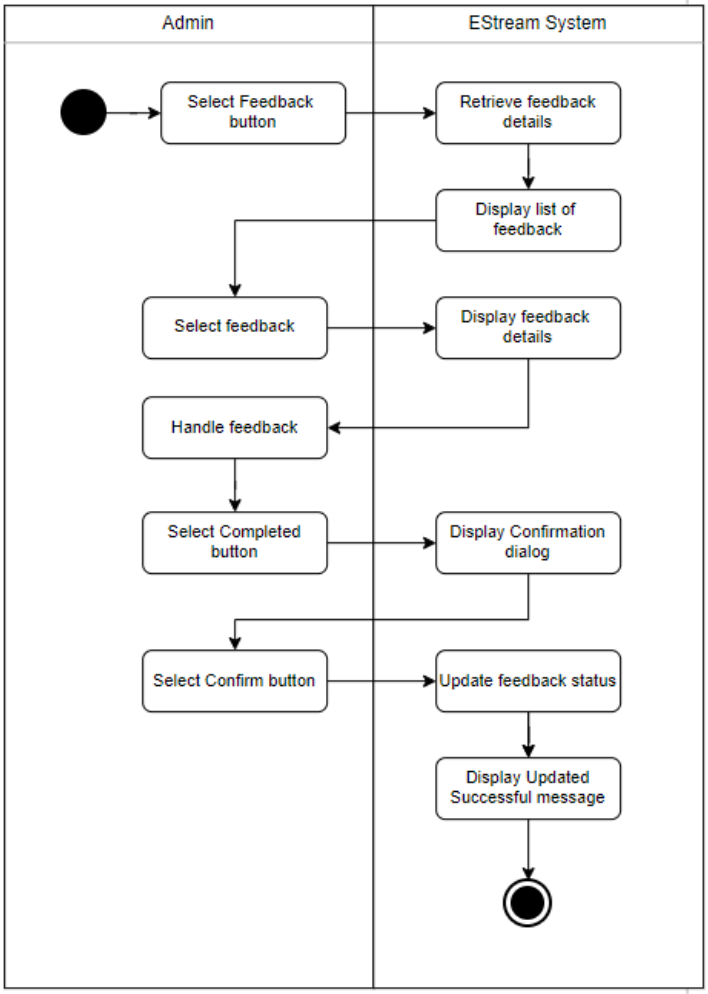


Figure 3.5.25 Activity diagram of Manage Feedback use case.

4. Manage Report Use Case

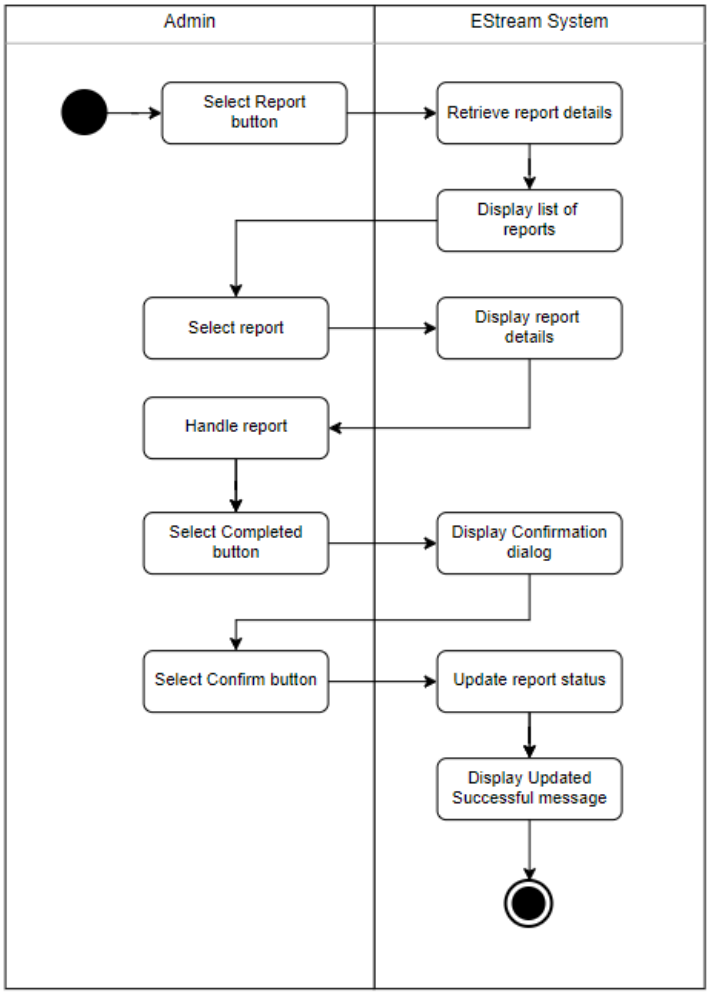


Figure 3.5.26 Activity diagram of Manage Report use case.

5. Manage Explicit Content Use Case

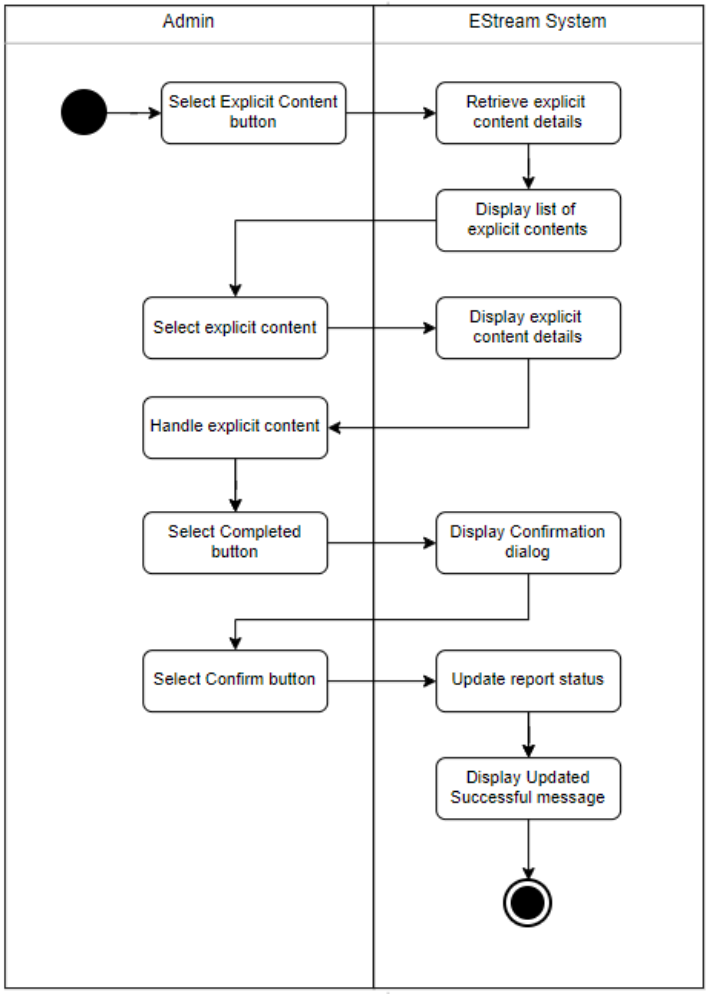


Figure 3.5.27 Activity diagram of Manage Explicit Content use case.

6. Manage User Use Case

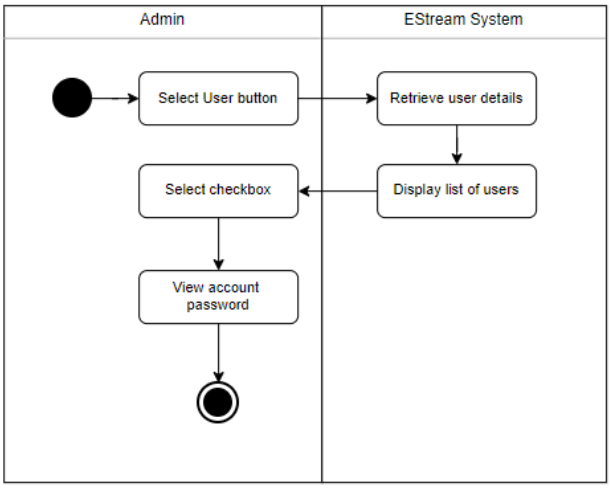


Figure 3.5.28 Activity diagram of Manage User use case.

7. Manage Staff Use Case

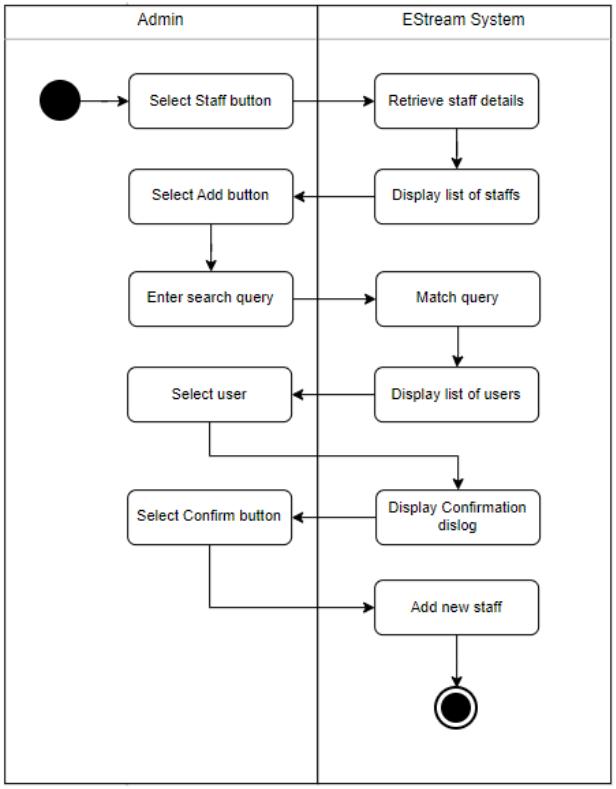


Figure 3.5.29 Activity diagram of Manage Staff use case.

8. View All Posts Use Case

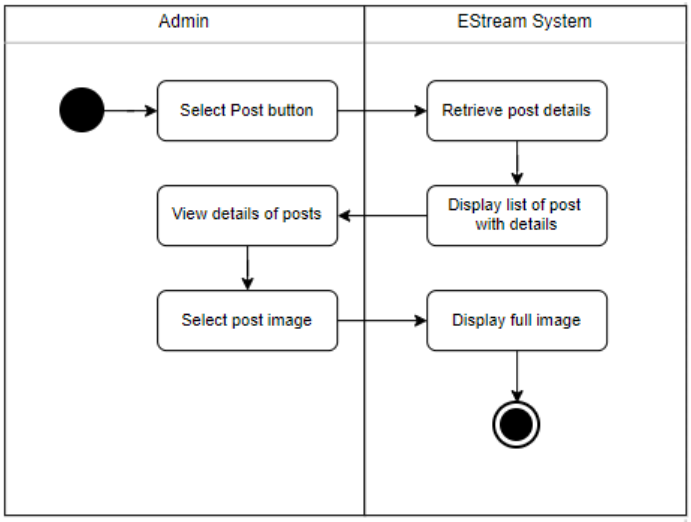


Figure 3.5.30 Activity diagram of View All Posts use case.

9. View All Videos Use Case

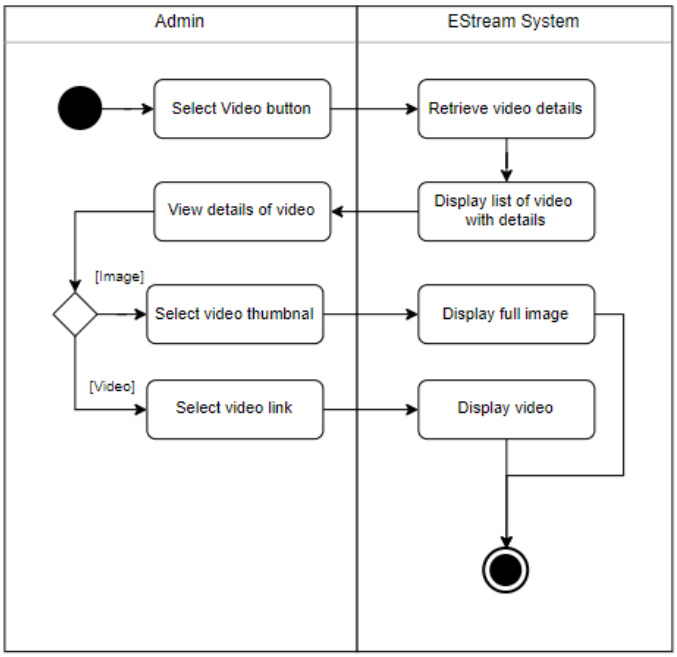


Figure 3.5.31 Activity diagram of View All Videos use case.

10. Logout Account Use Case

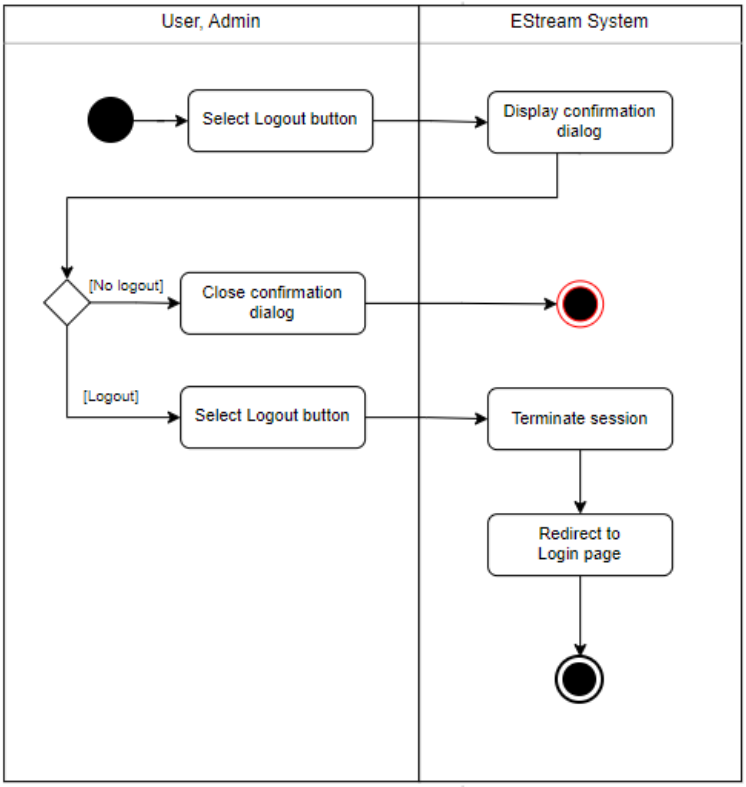


Figure 3.5.32 Activity diagram of Logout Account use case.

Chapter 4
System Design

4.1 System Block Diagram

Figure 4.1.1 and Figure 4.1.2 illustrate the breakdown of the application into its lowest manageable levels from the perspectives of users and administrators, respectively.

User Perspectives

Upon launching the application, users are presented with an inviting splash screen that seamlessly transitions them to the login page. Users have the option to log in with their existing credentials or register for a new account if they are new to the platform. Upon successful authentication, the system redirects users to the home page, serving as the central hub for accessing a wide range of features.

The home page offers users the ability to engage with live and video streaming content, providing them with immersive experiences in the world of esports. Additionally, users can stay updated with the latest esports news through the dedicated News page, ensuring they are always informed about industry developments. Promoting community interaction, the platform includes a Community page where users can connect with each other by sharing images and experiences related to esports. The Messages feature enables users to communicate with friends and receive notifications from esports teams, fostering a sense of belonging and engagement within the community. Furthermore, the Account page empowers users to manage various aspects of their account, including personal information, content creation, wallet transactions, system settings, and feedback submission, providing them with full control over their platform experience.

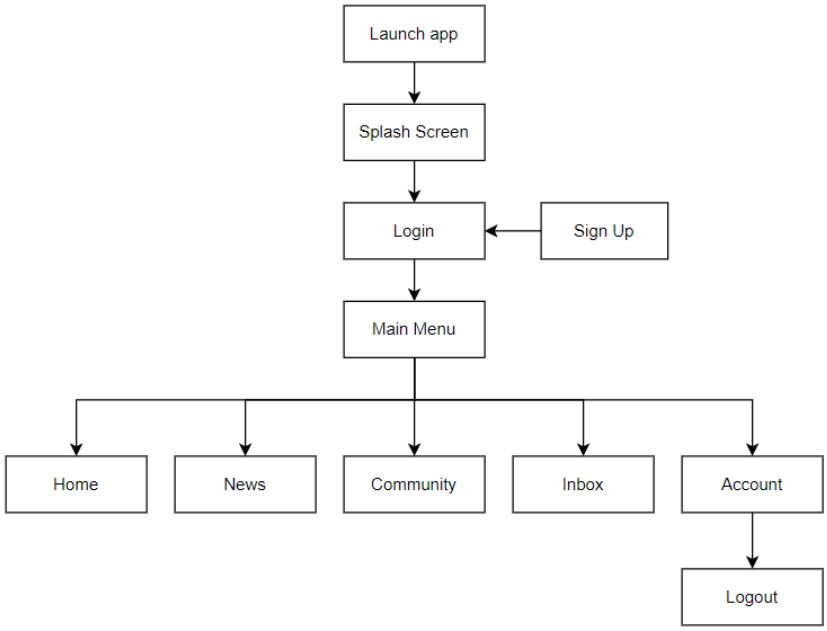


Figure 4.1.1 Top-down system diagram of eStream system (User perspective).

Administrator Perspectives

Administrators access the platform by logging in with their designated credentials, which directs them to the admin home page. In admin home page, administrators have access to a comprehensive suite of tools to manage and oversee various aspects of the platform's operations. Administrators can effectively manage feedback, reports, and explicit content to maintain a safe and enjoyable environment for all users. User management capabilities extend to overseeing user accounts, staff members, posts, and videos, ensuring compliance with community guidelines and standards. Additionally, administrators have the ability to view detailed profile information, enabling informed decision-making and effective communication with users and staff members. The logout feature provides administrators with secure access control and data protection, reinforcing the platform's commitment to security and privacy.

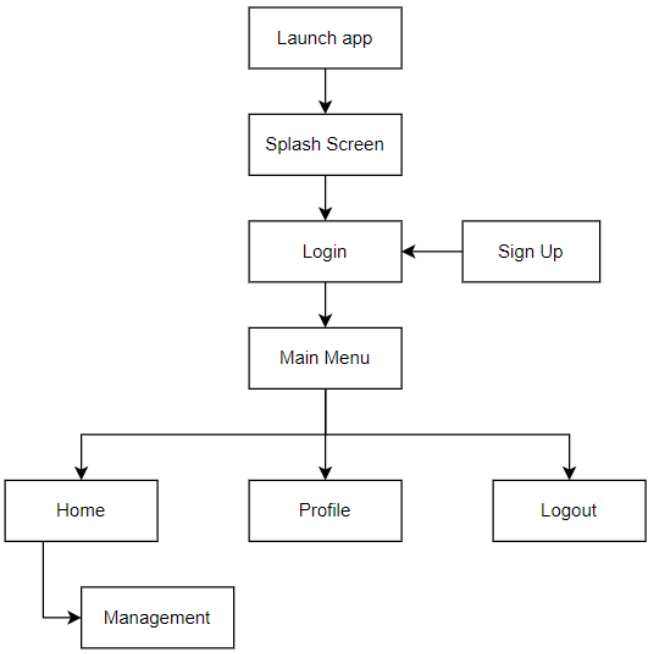


Figure 4.1.2 Top-down system diagram of eStream system (Administrator perspective).

4.2 Overall System Design Illustration

Figure 4.2 shows the proposed framework of this project.

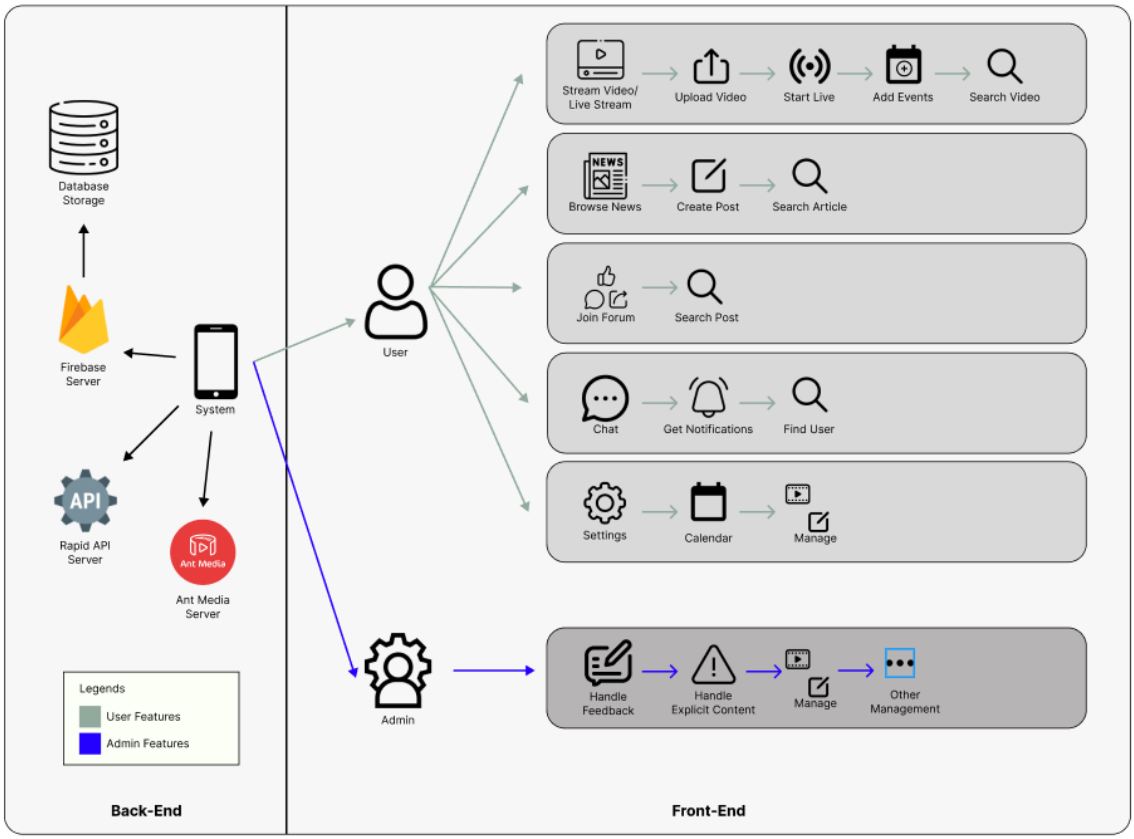


Figure 4.2 Proposed system framework.

The frontend of the system serves as the user interface through which both regular users and administrators interact with the platform. It showcases the features and functionalities available to users and administrators, providing a seamless and intuitive user experience. For regular users, the frontend displays various features such as showcasing live and video streaming content, news updates, community engagement, messaging, and account management. Administrators, on the other hand, have access to additional functionalities such as managing feedback, reports, user accounts, staff members, posts, and videos.

The backend of the system comprises the internal logic and operations that drive the platform's functionality. It handles data processing, storage, and retrieval, ensuring smooth operation and responsiveness of the system. Data storage and management are facilitated through integration with Firebase servers, which provide a reliable and scalable solution for storing user data, video and post content, and other relevant information. Apart from that, the interaction with external APIs was another crucial aspect of the backend functionality. The system interacts with the Rapid API server to send requests and receive responses for data retrieval purposes. This communication enables the platform to access external sources of information such as esports news updates and other relevant data. Additionally, integration with the Ant Media server enables the live streaming feature, allowing users to enjoy real-time esports events and content seamlessly within the platform.

4.3 System Component Specifications

The system components of the proposed project can be specified as follow:

(User Mobile Application)

1. **Login Module**

   Users can either register an account using their email address and password or opt for a quicker login using their Google account. The "Remember Me" feature allows users to remain logged in until they choose to log out.

2. **Home Module**

   The Home section serves as the central hub for accessing various content types. Users can engage in video streaming, enjoying a diverse selection of pre-recorded content, or opt for live streaming to experience real-time esports events or interact with their favourite content creators. Content creators have the option to contribute to the platform by uploading their own videos or initiating live broadcasts to connect with their audience. Users can easily find videos by searching for titles, descriptions, or thumbnail characteristics, and stay informed about upcoming tournament events on the current and next day. A leaderboard based on subscriber numbers, an advertisement slider, and the ability to search for videos are also available.

3. **News Module**

   The News section provides users with the latest updates and news from the esports world. Users can read articles and share them with others.

4. **Community Module**

   Users can actively participate in the platform’s community by creating posts, engaging with content through likes and comments, and sharing interesting posts. The search and filter functionalities make it easy to find specific posts based on their content or characteristics.

5. **Inbox Module**

   The Message feature enables users to communicate with each other via text and image messages. Users can search for other users by UID or email, subscribe to their accounts, and receive notifications about feedback, reports, and explicit content.

6. **Account Module**

   In the Account section, users can view and modify their profiles, check their subscriber and subscription numbers, access a calendar of tournament events, manage their wallet, and handle rewards for live streaming. Users can also manage their created content, including videos and posts, with options to view all content and delete items as needed. Additionally, users can customize their experience by switching between light and dark themes and log out when necessary.

(Admin Mobile Application)

1. **Login Module**

   Admins can register an account and login to the application using their email address and password.

2. **Admin Home Module**

   The Admin Home Module serves as the central hub for administrative tasks and oversight within the platform. Administrators can access various features and functionalities essential for managing and maintaining the platform's operations efficiently. They can view their profile information and manage feedback provided by users, responding to and addressing suggestions and concerns promptly. Additionally, administrators can view and manage reports submitted by users or system regarding inappropriate behavior or violations of community guidelines, taking appropriate actions to maintain a safe and respectful environment. Administrators also have the authority to manage user accounts by reviewing user profiles. They can oversee staff members, assign roles, and add new staff to help manage platform operations effectively. Furthermore, administrators can access lists of all posts and videos on the platform, reviewing content for compliance with guidelines and taking corrective action when necessary. Overall, the Admin Home Module provides administrators with comprehensive tools and capabilities to uphold the platform's standards and objectives while ensuring a positive user experience for all.

Chapter 5
System Implementation

This chapter presents a comprehensive overview of setting up and configuring an Android app, addressing hardware requirements and software installation, followed by detailed discussions on project configuration and system operation with accompanying screenshots. The chapter also explains implementation issues and challenges, providing strategies to overcome common obstacles in app development.

5.1 Technologies and Tools Involved

Hardware

The hardware involved in this project is a laptop and an android mobile device. A laptop integrated with webcam is used for the process of developing and testing the application. A mobile device is used for testing and deploying the product.

Table 5.1.1 Specifications of laptop

Description	Specifications
Model	Acer Aspire 3 A315-55G
Processor	Intel Core i5-10210U
Operating System	Windows 11
Graphic	NVIDIA GeForce MX230 2GB VRAM
Memory	20GB (4+16)
Storage	512GB PCIe NVMe SSD

Table 5.1.2 Specifications of mobile device

Description	Specifications
Model	SM-A136B/DS
Android Version	13
API level	33
Resolution	720 x 1600
Display Picture (dp)	384 x 854

Software

The software involved in this project:

Table 5.1.3 Description of software

No.	Software	Description
1	Android Studio IDE Version Iguana _ 2023.2.1	Android Studio Iguana is used as the integrated development environment for designing, developing, testing, and debugging the application.
2	Figma Version 116.14.9	Figma is employed for creating wireframes to plan the structure and layout of the application.
3	Draw.io	Draw.io is used to generate Unified Modelling Language (UML) diagrams to document the user requirements.

Web Services

The web services included in this project are Firebase Authentication, Firebase Realtime Database, and Firebase Storage.

Table 5.1.4 Description of web services

No.	Web Service	Description
1	Firebase Authentication	This project utilized Firebase Authentication service to enable users sign in using various authentication methods such as email/password, and Google social identity providers.
2	Firebase Realtime Database	Firebase Realtime Database is a NoSQL database being integrated to store and synchronize application data, such as user profiles, chat messages, post details, video details, etc.
3	Firebase Storage	Firebase storage is used to store and serve user-generated content, such as profile images, post images, videos, etc.

APIs

Table 5.1.5 Description of API services

No.	API Service	Description
1	Objects Detection	The Objects Detection API is utilized to detect and identify various objects in images, encompassing a wide range of categories, including people, animals, inanimate objects, color, and more. The resulting keywords extracted from the image will be included as available query for content searching.
2	NSFW Detect	NSFW Detection API is implemented in this project to detect image that is inappropriate, unwanted, or offensive. The categories of detection include nudity, sexual activity, violence, drugs, alcohol, tobacco, hate symbols, rude gestures, etc.
3	EsportsApi	ESportApi offers eSports results for League of Legends, Dota 2, Counter-Strike, providing also eSports standings and tournament live scores. For this project, the system utilizes the API for displaying tournament event to allow user adding tournament event to the calendar in the application.

5.2 Setting and Configuration

Setting up and configuring is the most important process to start up develop an Android app. It involves several key steps to ensure its successful development and deployment.

Firstly, the development environment is established by installing Android Studio, the official IDE for Android app development, and the Java Development Kit (JDK) compatible with Android Studio. Once the environment is set up, a new project is created in Android Studio, and the project configuration process is initiated. This includes selecting project templates, specifying names, package names, and other essential configurations through a guided setup wizard.

In the project configuration phase, Gradle configuration and manifest file settings are focused. Gradle, the build system for Android projects, allows the developer to manage dependencies and build types efficiently. Necessary dependencies are configured and implemented. Meanwhile, the AndroidManifest.xml file contains crucial information about the app, such as package name, permissions, activities, and services. Necessary permissions such as Internet, camera, video, and audio were set up. These elements were configured correctly to ensure the app functions as intended and complies with Android platform requirements.

Moving forward, the app-specific settings and features are set up. This entails designing the user interface (UI), managing resources, defining app themes and styles, supporting localization, handling permissions, and customizing app icons and splash screens. These aspects contribute to creating a visually appealing, user-friendly, and functional app.

Subsequently, the developer focuses on implementing app functionalities and features using Java, the primary programming languages for Android development. They leverage Android APIs to incorporate various features like retrieval of esports tournament events' information and esports news content. Proper handling of app lifecycle events is essential to maintain smooth user experiences throughout the app's usage.

During the testing and debugging phase, the developer rigorously tests the app on different Android devices and emulators to ensure compatibility and identify and

resolve any issues. Android Studio's debugging tools play a crucial role in diagnosing and fixing bugs efficiently.

As the app is completed, the release settings are configured, including generating a signed APK for release, optimizing performance and size, and specifying release-specific parameters like version codes and names. Finally, the app is distributed and uploaded to the Google Play Store, accompanied by comprehensive app store listings, screenshots, and descriptions to attract users.

5.3 System Operation

Login

The splash screen appears briefly upon launching the application, displaying the eStream logo. The login page prompts users to enter their credentials, such as email and password, to access their account. The signup page allows new users to create an account by providing necessary details like username, email address, and password, and agreeing to terms of service.

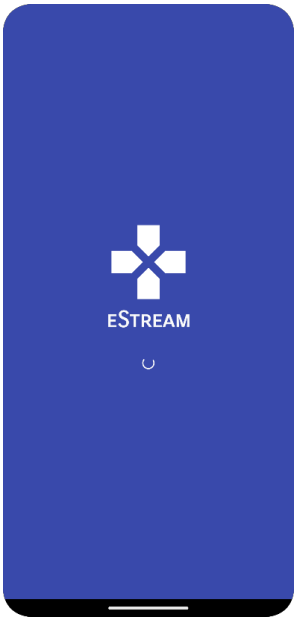


Figure 5.3.1 Splash screen

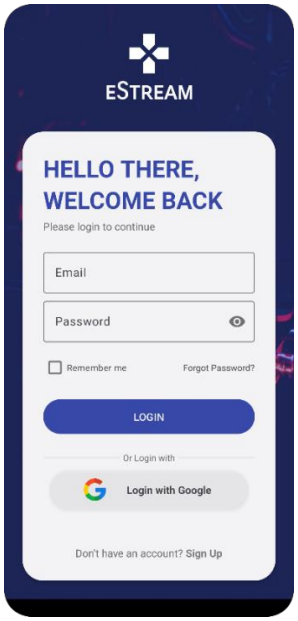


Figure 5.3.2 Login page

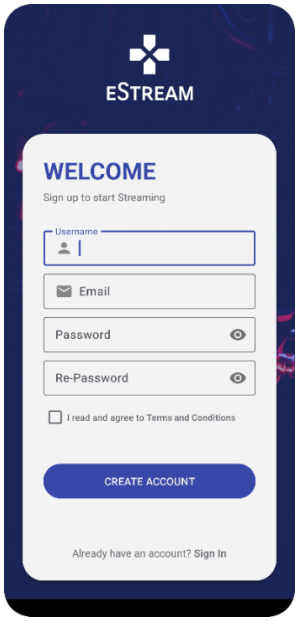


Figure 5.3.3 Signup Page

Home

The Home Page serves as the central interface, offering access to various features including Create Live for initiating live stream sessions, Live Stream for viewing ongoing broadcasts, Upload Video for sharing pre-recorded content, Watch Video for browsing and viewing uploaded videos, and Search Video for finding specific content. The system verifies and notifies user if the video content was explicit. While Leaderboard displays rankings of users based on number of subscribers. Additionally, Tournament Events provides information on upcoming tournaments,

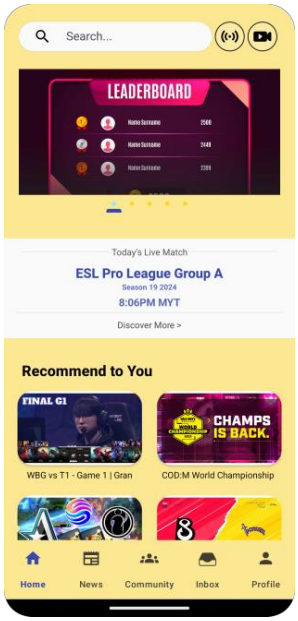


Figure 5.3.4 Home Page

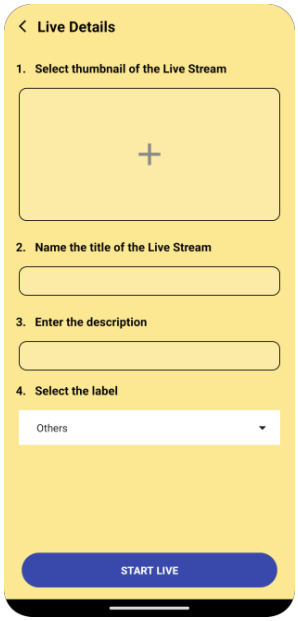


Figure 5.3.5 Create Live

Figure 5.3.6 Live Stream

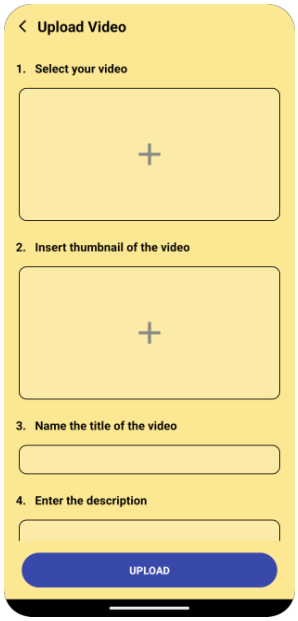


Figure 5.3.7 Upload Video

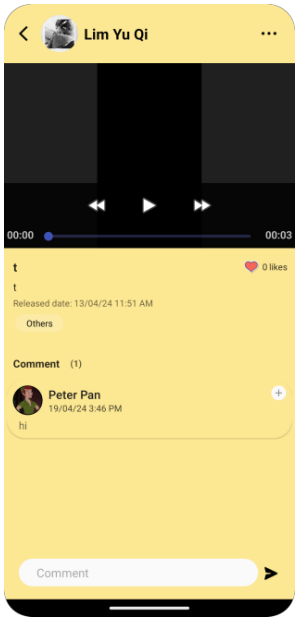


Figure 5.3.8 Watch Video

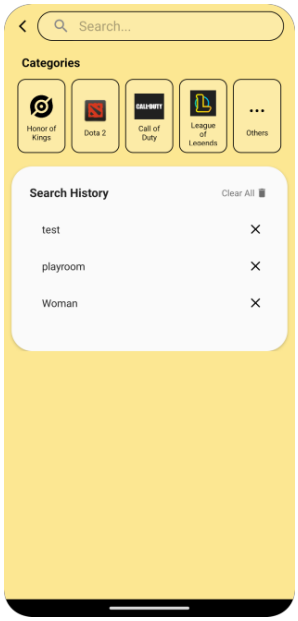


Figure 5.3.9 Search Video

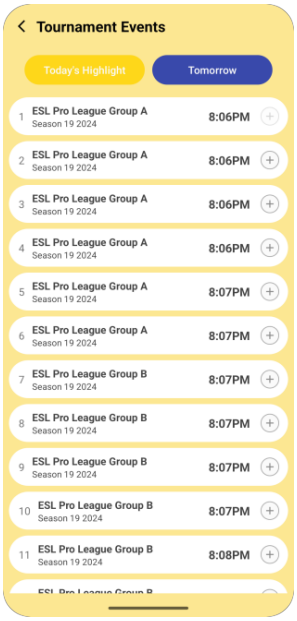


Figure 5.3.10 Tournament Events

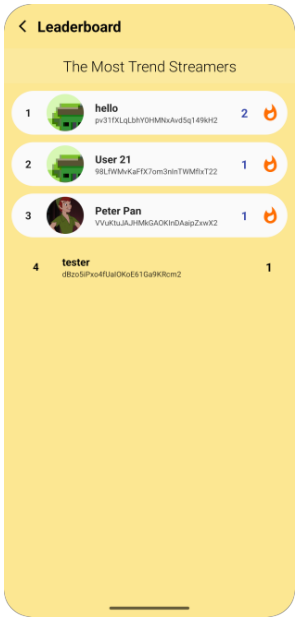


Figure 5.3.11 Leaderboard

News Module

The News Page presents users with a designated section within the application, offering a curated selection of news articles containing the latest updates, announcements, and relevant information. Users can access this section to stay informed about esports related article. Additionally, the Browse Article feature enables users to explore a diverse range of articles available on the platform. By providing a browsing interface, users can navigate through the collection of articles, read in-depth content, engage with topics of interest, and share articles with others.

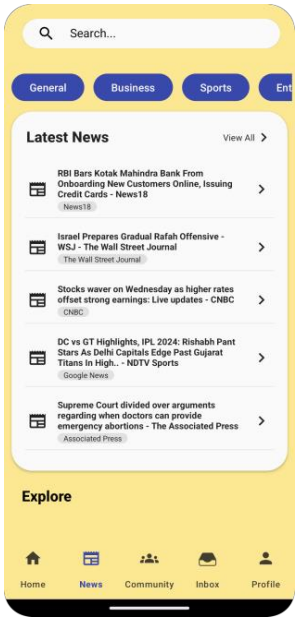


Figure 5.3.12 News Page

Figure 5.3.13 Read Article

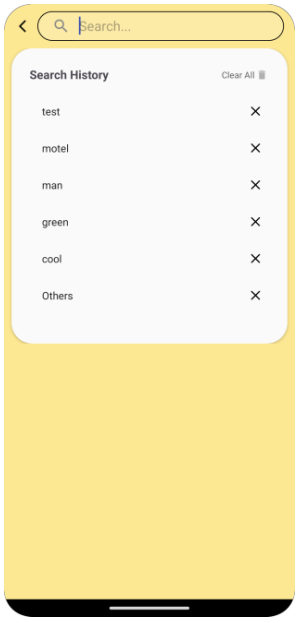


Figure 5.3.14 Search Article

Community

The Community Page provides a dedicated space where users can connect, share, and participate in discussions, creating a vibrant community atmosphere. The Create Post feature empowers users to contribute to the community by sharing their thoughts, ideas, or updates through posts. In addition, the system verifies and notifies user if the post content was explicit. Users can engage themselves through interaction such as liking, commenting, and sharing posts. Additionally, the Search Post feature enables users to easily find specific posts or topics of interest within the community.

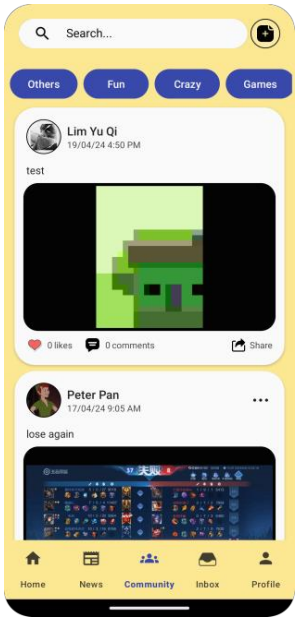


Figure 5.3.15 Community Page

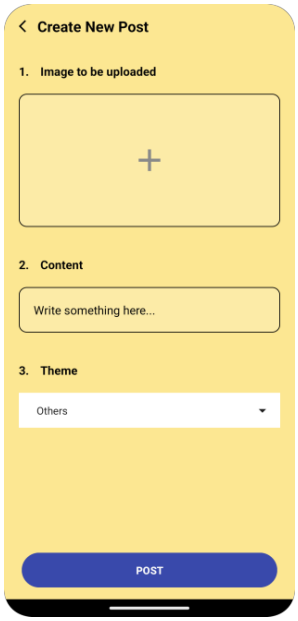


Figure 5.3.16 Create Post

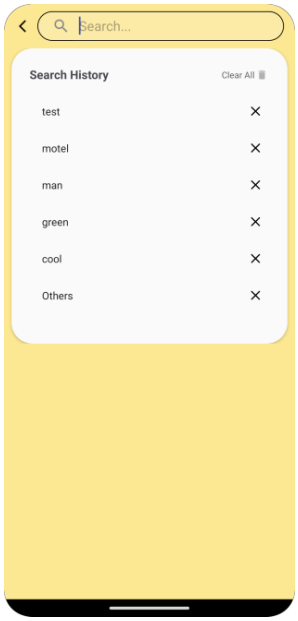


Figure 5.3.17 Search Post

Inbox

The Message feature allows users to send and receive private messages within the application, facilitating direct communication between individuals. Notifications inform users about their feedback, report, and content issue. The Chatroom provides users with a real-time communication platform where they can engage in conversations with other users. Additionally, the Search User feature enables users to find specific individuals within the platform by searching for their username or user ID. Users can then view detailed information about other users using the View User Details feature, which provides insights into their self-created videos and posts.

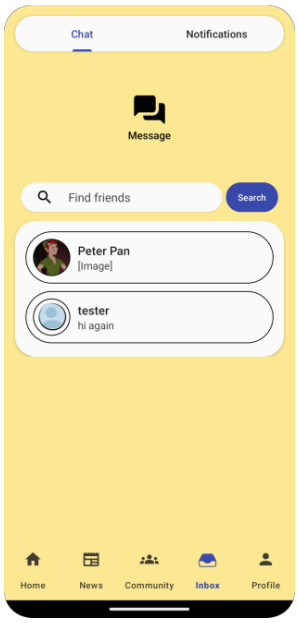


Figure 5.3.18 Message

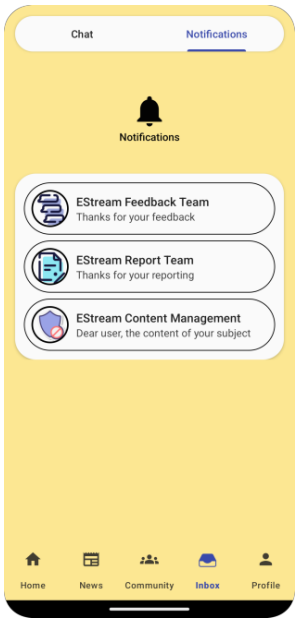


Figure 5.3.19 Notification

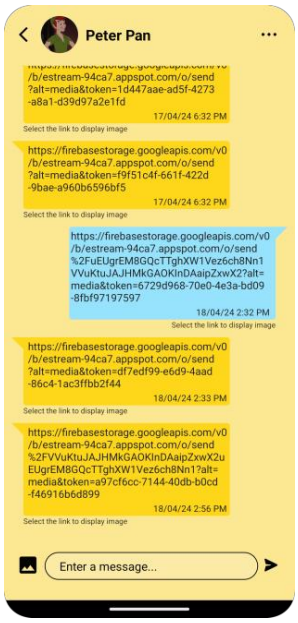


Figure 5.3.20 Chatroom

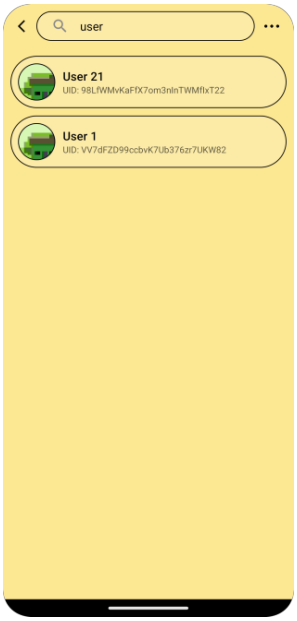


Figure 5.3.21 Search User

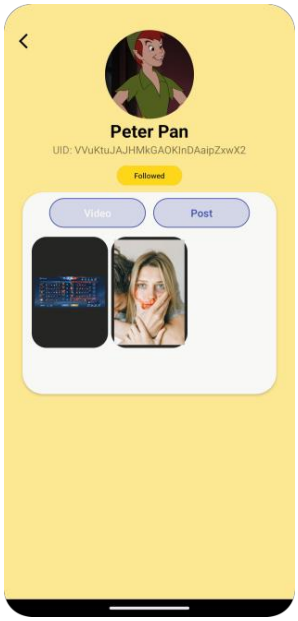


Figure 5.3.22 View User Details

Account

The Account Page enables users to manage their profile settings and personal information. The Calendar feature allows users to view upcoming events. My Wallet facilitates financial management, while My Video and My Post provide access to uploaded videos and posts. Settings allow customization of account and app preferences, and User Profile displays detailed user information. Switch Theme offers light and darker color scheme option. Users can provide report errors or suggest improvements through Feedback features.

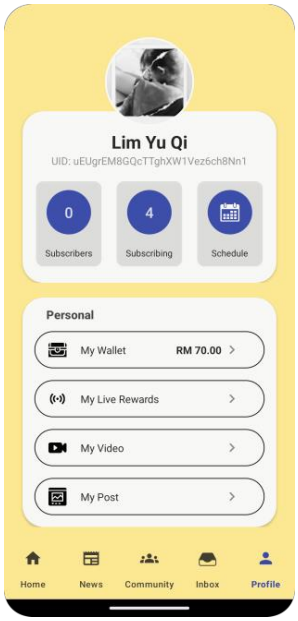


Figure 5.3.23 Account Page

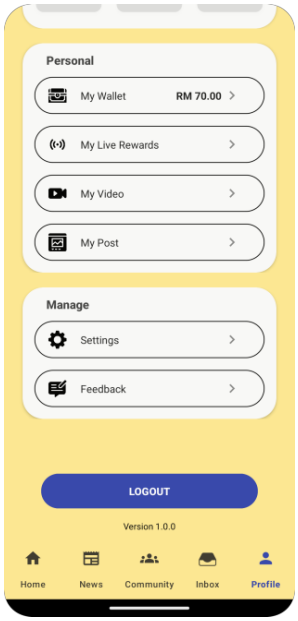


Figure 5.3.24 Account Page (Cont')

Figure 5.3.25 Calendar

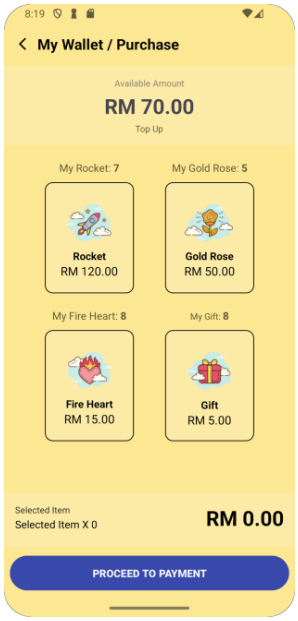


Figure 5.3.26 My Wallet

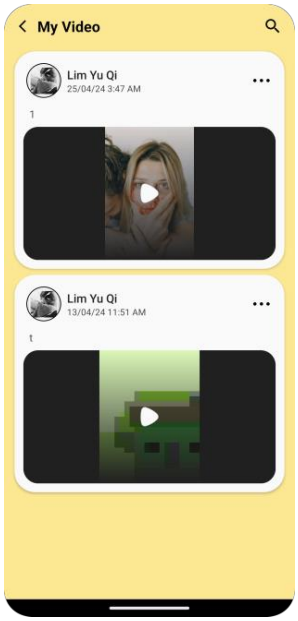


Figure 5.3.27 My Video

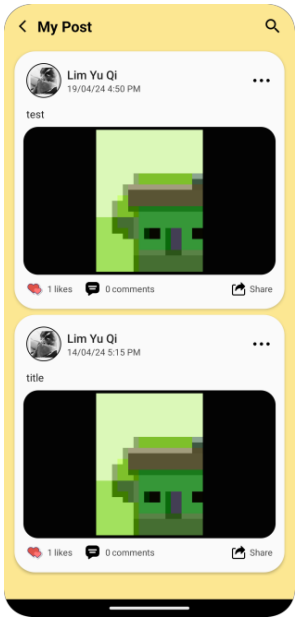


Figure 5.3.28 My Post

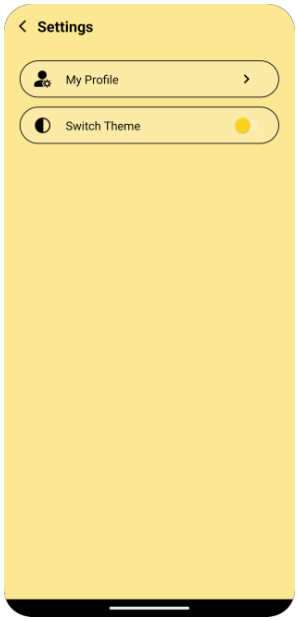


Figure 5.3.29 Settings

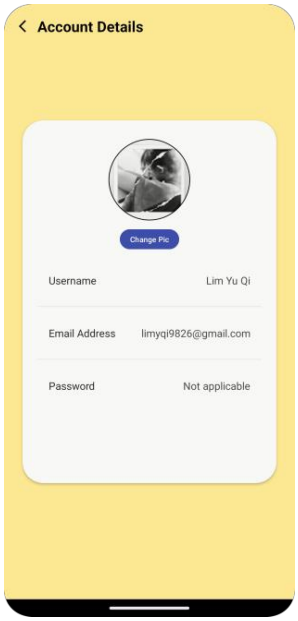


Figure 5.3.30 User Profile

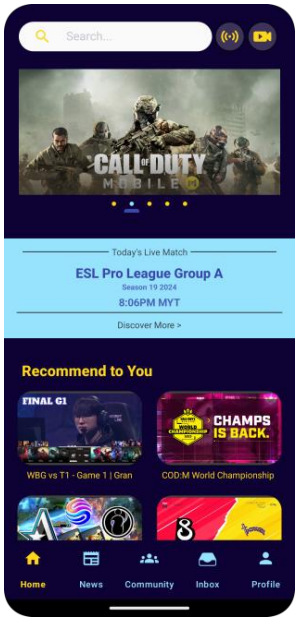


Figure 5.3.31 Dark Mode

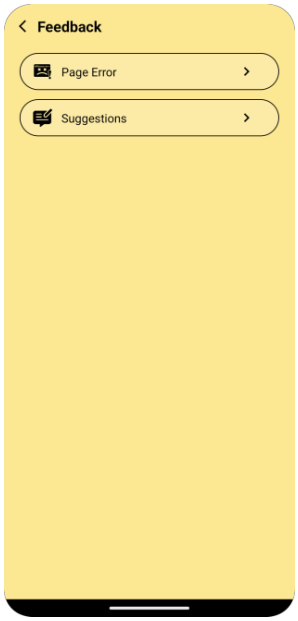


Figure 5.3.32 Feedback Type

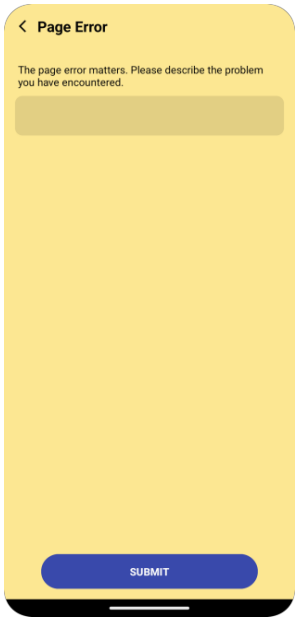


Figure 5.3.33 Page Error

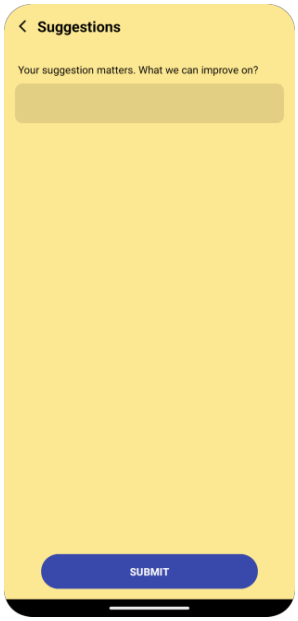


Figure 5.3.34 Suggestions

Admin App

Admin Home

Admin Home Page serves as the central dashboard for administrative tasks, including Admin Profile access. All Feedback lists user feedback, while All Report displays reported issues. Explicit Content manages reported explicit content. Subject Details offers details of cases being handle, and Filter helps refine search results. All Users lists registered users in eStream, and Staff Records displays staff information. All Videos and All Posts list uploaded videos and published posts, respectively.

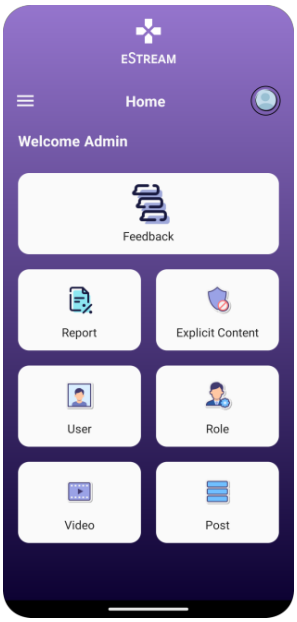


Figure 5.3.35 Admin Home Page

Figure 5.3.36 Admin Profile

Figure 5.3.37 All Feedback

Figure 5.3.38 All Report

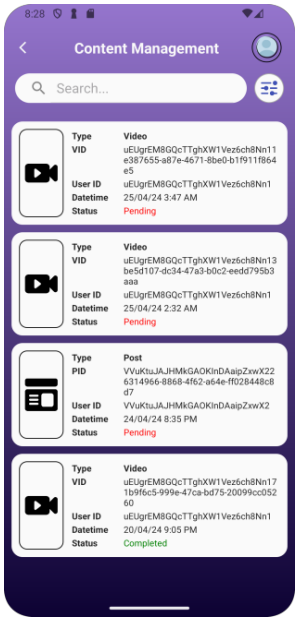


Figure 5.3.39 Explicit Content

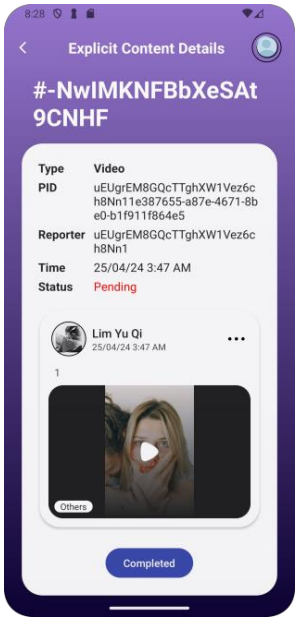


Figure 5.3.40 Subject Details

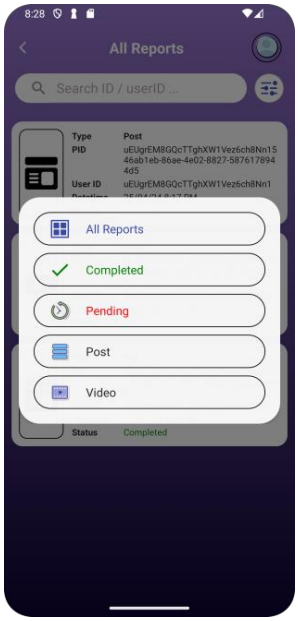


Figure 5.3.41 Filter

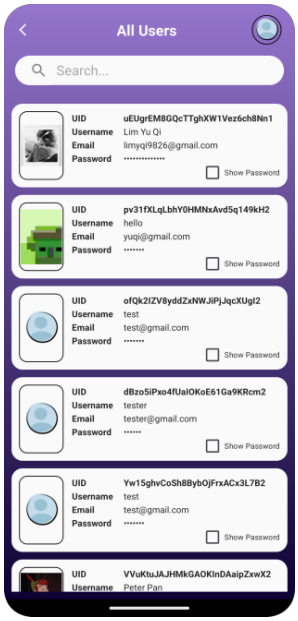


Figure 5.3.42 All Users

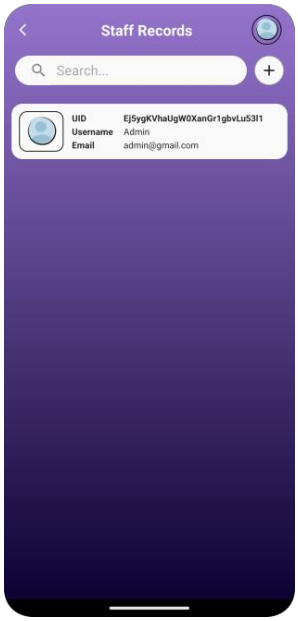


Figure 5.3.43 Staff Records

Figure 5.3.44 All Videos

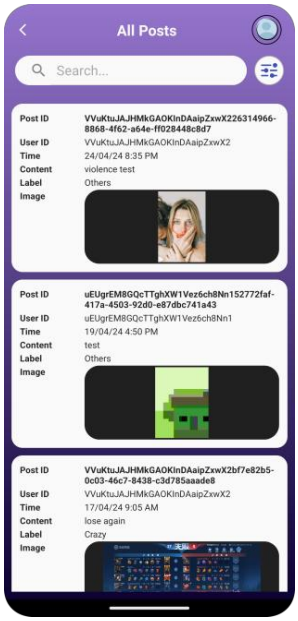


Figure 5.3.45 All Posts

5.4 Implementation Issues and Challenges

During the development process, some deprecated functions were identified in Android Studio. Deprecated functions refer to methods that marked as outdated or obsolete. For instance, as shown in Figure 5.6.1, the usage of the 'startActivityForResult()' function was crossed out when the developer attempt to implement it.

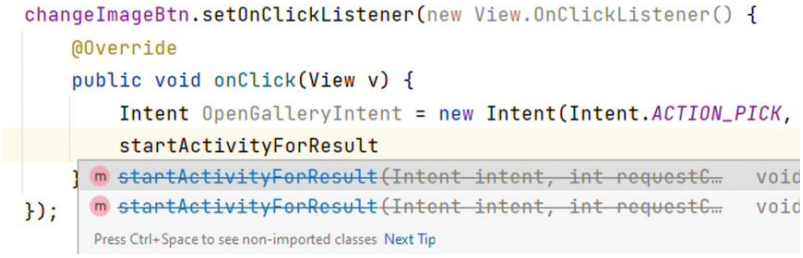


Figure 5.4.1 Deprecated method in Android Studio

Developers are encouraged to avoid using deprecated methods as these methods may be removed in future releases, with newer and more efficient alternatives usually provided. Instead, developers should update the code and use the recommended alternative. This might involve employing a different method, class, or approach. For example, the 'startActivityForResult()' function can be replaced with 'ActivityResultLauncher()' method. Developers can initiate the activity by launching it at where the 'startActivityForResult()' was initially used.

```
changeImageBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { mGetContent.launch( input: "image/*"); }
});
```

Figure 5.4.2 Launch mGetContent activity.

```
1 usage
ActivityResultLauncher<String> mGetContent = registerForActivityResult(new
    @Override
    public void onActivityResult(Uri o) {
        if (o != null){
            profileImg.setImageURI(o);
            imageUri = o;

            uploadImage();
        }
    }
});
```

Figure 5.4.3 Specify the function of mGetContent activity.

Chapter 6

System Evaluation and Discussion

6.1 System Testing and Performance Metrics

Table 6.1.0 Launching App Test Case

Test Case Name: Launching App			**ID:** TC00	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system shows the splash screen of the application.	The system redirects user to the Home page of the application.	**PASS**
	2	The system detects user session is not ended.		**PASS**
	3	The system verifies user's credentials.		**PASS**
Alternative Flows	2a	The system detect user session is ended.	The system redirects user to Login page of the application.	**PASS**
	3a	The system fails to verify user's credentials.		**PASS**

Table 6.1.1 Email/Password Sign in Test Case

Test Case Name: Email/Password Sign In			**ID:** TC01	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User enters correct email address and password.	The system redirect user to Home page of the application.	**PASS**
	2	User selects Login button.		**PASS**
	3	The system verifies account's credentials.		**PASS**
Alternative Flows	1a	User enters wrong email address or/and password.	The system displays error message and prompt user to try again.	**PASS**
	3a	The system failed to verify account's credentials.		**PASS**

Table 6.1.2 Google Sign in Test Case

Test Case Name: Google Sign In			**ID:** TC02	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays available google accounts from the device.	The system redirects user to Home page of the application.	**PASS**
	2	User selects a google account.		**PASS**
	3	The system verifies the account's credentials.		**PASS**
Alternative Flows	1a	There is no available google accounts from the device.	The system requests user to sign-in to their Google account.	**PASS**
	2a	User closes the Google sign in interface.	The system displays error message and prompt user to try again.	**PASS**
	3a	The system fails to verify the account credentials.		**PASS**

Table 6.1.3 Sign Up Test Case

Test Case Name: Sign Up			**ID:** TC03	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User enters username, email address, password, and confirmation password.	The system creates new accnt and redirects user to Login page of the application.	**PASS**
	2	User selects Create button		**PASS**
	3	The system verifies the validity of entered details.		**PASS**
Alternative Flows	1a	User does not enter all fields.	The system displays error message and prompt user to enter valid input.	**PASS**
	1b	User inputs wrong email address format.		**PASS**
	1c	User inputs password that has less than 6 characters.		**PASS**
	1d	User enters different inputs for password and confirmation password.		**PASS**

Table 6.1.4 Start Live Test Case

Test Case Name: Start Live			**ID:** TC04	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays empty field for live stream details.	The system starts the live stream.	**PASS**
	2	User enters title, description, selects image from gallery, and selects label for the live stream.		**PASS**
	3	User selects Start button.		**PASS**
Alternative Flows	2a	User does not enter all fields.	The system displays error message and prompt user to enter all fields.	**PASS**
	3a	User selects Return Back button.	The system redirects user to Home page.	**PASS**

Table 6.1.5 Stream Live Test Case

Test Case Name: Stream Live			**ID:** TC05	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the list of available live streams.	The system redirects user to the live stream channel.	**PASS**
	2	User selects and joins the live stream.		**PASS**

Table 6.1.6 Stream Video Test Case

Test Case Name: Stream Video			**ID:** TC06	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the list of available videos.	The system redirects user to the watch video page.	**PASS**
	2	User selects and watch the video.		**PASS**

Table 6.1.7 Upload Video Test Case

Test Case Name: Upload Video			**ID:** TC07	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays empty field for video details.	The system uploads the video.	**PASS**
	2	User enters title, description, and selects image and video from album.		**PASS**
	3	User selects Upload button.		**PASS**
Alternative Flows	2a	User does not enter all fields.	The system displays error message and prompt user to enter all fields.	**PASS**
	3a	User selects Return Back button.	The system redirects user to Home page.	**PASS**

Table 6.1.8 Delete Video Test Case

Test Case Name: Delete Video			**ID:** TC08	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Delete button of the video.	The system deletes he video.	**PASS**
	2	The system asks for user's confirmation delete the video.		**PASS**
	3	User selects Yes button.		**PASS**
Alternative Flows	3a	User selects No button.	The system does not delete the video.	**PASS**
	3b	User dismisses the confirmation dialog.		**PASS**

Table 6.1.9 Like Video Test Case

Test Case Name: Like Video			**ID:** TC09	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the number of likes of the video.	The system changes the presentation of Like button and updates the number of likes (+1).	**PASS**
	2	User selects Like button of the video.		**PASS**
Alternative Flows	2a	User selects Like button of the video again.	The system changes it back the presentation	**PASS**

			of Like button and updates the number of likes (-1).	

Table 6.1.10 Comment Video Test Case

Test Case Name: Comment Video			**ID:** TC10	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Comment button.	The system displays the updated comment list of the video.	**PASS**
	2	User enters the comment in the text field.		**PASS**
	3	User selects Send button.		**PASS**
Alternative Flows	3a	User sends an empty comment.	The system does not send out the comment.	**PASS**

Table 6.1.11 Delete Comment (Video) Test Case

Test Case Name: Delete Comment (Video)			**ID:** TC11	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Delete button of the comment.	The system deletes the selected comment of the video.	**PASS**
	2	The system asks for user's confirmation to delete the comment.		**PASS**
	3	User selects Yes button.		**PASS**
Alternative Flows	3a	User selects No button.	The system does not delete the selected comment of the video.	**PASS**
	3b	User dismisses the confirmation dialog.		**PASS**

Table 6.1.12 Share Video Test Case

Test Case Name: Share Video			**ID:** TC12	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the platforms available for video sharing.	The system sends the link of the video.	**PASS**
	2	User selects a platform.		**PASS**
Alternative Flows	2a	User dismisses the video sharing process.	The system does not send the link of the video.	**PASS**

Table 6.1.13 Search Video Test Case

Test Case Name: Search Video			**ID:** TC13	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Search icon.	The system displays list of videos which the title or description is matching with the search query.	**PASS**
	2	User enters the search query.		**PASS**
	3	The system matches the search query.		**PASS**
Alternative Flows	3a	User searches an empty query.	The system does not perform searching process.	**PASS**
	3b	User selects Return Back button.	The system redirects user to Home page.	**PASS**

Table 6.1.14 Create Polls Test Case

Test Case Name: Create Polls			**ID:** TC14	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Polls icon.	The system creates the polls.	**PASS**
	2	User enters the question.		**PASS**
	3	User enters the first option.		**PASS**
	4	User enters the second option.		**PASS**
	5	User enters the third option.		**PASS**
	6	User enters the fourth option.		**PASS**
	7	User selects Create button.		**PASS**
Alternative Flows	2a	User does not enter the question.	The system displays error message and prompt user to reenter the empty field.	**PASS**
	3a	User does not enter the first option.		**PASS**
	4a	User does not enter the second option.		**PASS**
	5a	User does not enter the third option.		**PASS**
	6a	User does not enter the fourth option.		**PASS**

Table 6.1.15 Donate Virtual Rewards Test Case

Test Case Name: Donate Virtual Rewards			**ID:** TC15	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Rewards icon.	The system deducts the item quantity from audience site and add to streamer site.	**PASS**
	2	User selects a virtual reward.		**PASS**
	3	User selects Send button.		**PASS**
Alternative Flows	3a	User has insufficient quantity of the selected virtual reward item.	The system display error message and notify user.	**PASS**

Table 6.1.16 Send Live Chat Test Case

Test Case Name: Send Live Chat			**ID:** TC16	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User enters live chat.	The system displays the content of live chat.	**PASS**
	2	User selects Send button.		**PASS**
Alternative Flows	1a	User does not enter live chat content.	The system does not display the empty content.	**PASS**

Table 6.1.17 Share News Test Case

Test Case Name: Share News			**ID:** TC17	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the platforms available for news sharing.	The system sends the link of the news.	**PASS**
	2	User selects a platform.		**PASS**
Alternative Flows	2a	User dismisses the news sharing process.	The system does not send the link of the news.	**PASS**

Table 6.1.18 Create Post Test Case

Test Case Name: Create Post			**ID:** TC18	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays empty field for post details.	The system creates a new post.	**PASS**
	2	User enters title, description, and selects image from gallery.		**PASS**
	3	User selects Post button.		**PASS**
Alternative Flows	2a	User does not enter all fields.	The system displays error message and prompt user to enter all fields.	**PASS**
	3a	User selects Return Back button.	The system redirects user to Community page.	**PASS**

Table 6.1.19 Delete Post Test Case

Test Case Name: Delete Post			**ID:** TC19	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Delete button of the post.	The system deletes he post.	**PASS**
	2	The system asks for user's confirmation delete the post.		**PASS**
	3	User selects Yes button.		**PASS**
Alternative Flows	3a	User selects No button.	The system does not delete the post.	**PASS**
	3b	User dismisses the confirmation dialog.		**PASS**

Table 6.1.20 Like Post Test Case

Test Case Name: Like Post			**ID:** TC20	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the number of likes of the post.	The system changes the presentation of Like button and updates the number of likes (+1).	**PASS**
	2	User selects Like button of the post.		**PASS**
Alternative Flows	2a	User selects Like button of the post again.	The system changes it back the presentation of Like button and updates the number of likes (-1).	**PASS**

Table 6.1.21 Comment Post Test Case

Test Case Name: Comment Post			**ID:** TC21	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Comment button.	The system displays the updated comment list of the post.	**PASS**
	2	User enters the comment in the text field.		**PASS**
	3	User selects Send button.		**PASS**
Alternative Flows	3a	User sends an empty comment.	The system does not send out the comment.	**PASS**

Table 6.1.22 Delete Comment (Post) Test Case

Test Case Name: Delete Comment (Post)			**ID:** TC22	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Delete button of the comment.	The system deletes the selected comment of the post.	**PASS**
	2	The system asks for user's confirmation to delete the comment.		**PASS**
	3	User selects Yes button.		**PASS**
Alternative Flows	3a	User selects No button.	The system does not delete the selected comment of the post.	**PASS**
	3b	User dismisses the confirmation dialog.		**PASS**

Table 6.1.23 Share Post Test Case

Test Case Name: Share Post			**ID:** TC23	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	The system displays the platforms available for post sharing.	The system sends the link of the post.	**PASS**
	2	User selects a platform.		**PASS**
Alternative Flows	2a	User dismisses the post sharing process.	The system does not send the link of the post.	**PASS**

Table 6.1.24 Search Post Test Case

Test Case Name: Search Post			**ID:** TC24	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Search icon.	The system displays list of posts which the title or description is matching with the search query.	**PASS**
	2	User enters the search query.		**PASS**
	3	The system matches the search query.		**PASS**
Alternative Flows	3a	User searches an empty query.	The system does not perform searching process.	**PASS**
	3b	User selects Return Back button.	The system redirects user to Community page.	**PASS**

Table 6.1.25 Search Article Test Case

Test Case Name: Search Article			**ID:** TC25	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Search icon.	The system displays list of articles which the title is matching with the search query.	**PASS**
	2	User enters the search query.		**PASS**
	3	The system matches the search query.		**PASS**
Alternative Flows	3a	User searches an empty query.	The system does not perform searching process.	**PASS**
	3b	User selects Return Back button.	The system redirects user to News page.	**PASS**

Table 6.1.26 Search User Test Case

Test Case Name: Search User			**ID:** TC26	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects Search icon.	The system displays list of users which the username or user ID is matching with the search query.	**PASS**
	2	User enters the search query.		**PASS**
	3	The system matches the search query.		**PASS**
Alternative Flows	3a	User searches an empty query.	The system does not perform searching process.	**PASS**
	3b	User selects Return Back button.	The system redirects user to Message page.	**PASS**

Table 6.1.27 Send Message Test Case

Test Case Name: Send Message			**ID:** TC27	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User inputs text content.	The system displays text content in the chat room.	**PASS**
	2	User selects Send Button		**PASS**
Alternative Flows	1a	User selects image from gallery.		**PASS**
	1b	User does not select input.	The system does not display text content in the chat room.	**PASS**

Table 6.1.28 Purchase Virtual Rewards Test Case

Test Case Name: Purchase Virtual Rewards			**ID:** TC28	
	Test Steps	**Action**	**Expected Result**	**Result Status**
Main Flow	1	User selects a reward item.	The system adds reward items to user's wallet and deduct amount from the wallet.	**PASS**
	2	User inputs item quantity.		**PASS**
	3	User selects Check Out button		
Alternative Flows	3b	User does not have available balance in the wallet.	The system does not add reward items to user's wallet and deduct amount from the wallet.	**PASS**

6.2 Testing Setup and Results

In the previous subchapter, test cases were conducted for the proposed system. The test procedures were derived from the use case descriptions, ensuring that all functions performed correctly and followed the proper sequence. Consequently, the actual executions of all 28 test cases met expectations. Overall, the testing results demonstrated the robustness and reliability of the application, ensuring a seamless and satisfactory user experience. By thoroughly testing the application across various scenarios and environments, the quality of the application has been enhanced, and its readiness for deployment to end-users has been improved.

6.3 Project Challenges

The development of the project faces several challenges, including limitations on API responses due to the use of a freemium version, which often imposes restrictions on the volume of data that can be accessed or processed. This challenge hinders the app's functionality by limiting the amount of information the application can retrieve from external sources, potentially impacting features reliant on real-time or extensive data.

Additionally, problems within the Android Studio Integrated Development Environment (IDE) poses significant hurdles, potentially disrupting the development workflow and requiring troubleshooting efforts.

Moreover, time constraints are a common challenge in software development projects, which impacting various aspects such as planning, development, testing, and deployment. It led to trade-offs between implementing desired features and delivering a functional product within the allocated timeframe. However, it is important to recognize that achieving perfection within these constraints is impractical. These challenges provide valuable learning opportunities, guiding the project towards enhanced efficiency and effectiveness over time.

6.4 Objectives Evaluation

The objectives of the project have been successfully achieved by implementing feed features, facilitating donations during live streaming, and integrating content search with content characteristics detection and explicit content notification.

The eStream mobile application includes a dynamic feed feature where users can view and engage with various content updates. Live chat functionality allows real-time communication between users, fostering interaction and community engagement. Additionally, the integration of live polls enhances user engagement by enabling audience participation and feedback gathering during live streams.

In addition, the application facilitates donations during live streaming sessions, providing audiences with a convenient way to support their favourite content creators. This feature enhances the app's monetization strategy while also empowering users to contribute to meaningful initiatives.

Furthermore, the application enhances the usability and content discovery capabilities by allowing users to search for content based on specific characteristics of images. Moreover, the implementation of explicit content detection algorithms ensures a safe and appropriate user experience. Users are promptly notified of any explicit content, allowing them to take further actions as necessary to maintain a secure browsing environment.

6.5 Concluding Remark

This chapter has undergone thorough testing of the proposed system to ensure its usability and reliability. The results indicate that the system performed admirably with no significant errors. However, it's worth noting that the test cases were crafted based on experience rather than by an expert tester. As a result, there may be limitations, as certain aspects could have inadvertently been omitted from the testing process.

Chapter 7
Conclusion and Recommendation

7.1 Conclusion

In summary, the project has successfully addressed the identified problem statement by achieving its objectives of creating a mobile-centric environment for the esports world, facilitating effective content discovery, introducing engaging donation features, and delivering positive value through esports content. Motivated by the evolving landscape of social media platforms and the growing trend of live streaming post-COVID-19, the project has contributed to enhancing user engagement and community cohesion within the esports live streaming ecosystem. By leveraging advancements in AI technology and implementing innovative features, the project has not only addressed existing limitations but also paved the way for a more inclusive, interactive, and responsible platform for users and content creators alike.

7.2 Recommendation

It is recommended that the project's effectiveness can be enhanced by focusing future efforts on refining the detection methods employed, specifically object detection and explicit content detection, through model training to improve accuracy. Object detection can be significantly improved by enriching the dataset used for training, encompassing a diverse array of objects, angles, and lighting conditions. Similarly, explicit content detection can be enhanced by training the model with a comprehensive dataset containing various explicit content types and variations. Leveraging advanced techniques like deep learning and neural network architectures can improve the model's accuracy in identifying object and explicit content while minimizing false positives.

Investing in the improvement of detection methods through model training and optimization help enhance the accuracy and reliability of object and explicit content detection functionalities but also strengthens the platform's safety and security measures. This fosters a safer and more engaging environment for users and content creators, ultimately enhancing the overall user experience by providing more accurate and relevant results.

REFERENCES

[1] *GosuGamers*. (2019), GosuGamers. [Online]. Available: https://www.gosugamers.net/.

[2] *DouYu*. (2019), DouYu. [Online]. Available: https://www.douyu.com/?dyshid=1a1fdc5a-e41f961498be0071dcae796400051601.

[3] *DLive*. (2019), BitTorrent. [Online]. Available: https://dlive.tv/.

[4] M. Mazodier, C. M. Henderson, and J. T. Beck, "The Long Reach of Sponsorship: How Fan Isolation and Identification Jointly Shape Sponsorship Performance," Journal of Marketing, vol. 82, no. 6, pp. 28–48, Oct. 2018, doi: https://doi.org/10.1177/0022242918807673.

[5] Y. Chen, S. Gong, and L. Amazon, "Image Search with Text Feedback by Visiolinguistic Attention Learning." Available: https://openaccess.thecvf.com/content_CVPR_2020/papers/Chen_Image_Search_With_Text_Feedback_by_Visiolinguistic_Attention_Learning_CVPR_2020_paper.pdf

[6] L. W. Smith et al., "Is sexual content in new media linked to sexual risk behaviour in young people? A systematic review and meta-analysis," Sexual Health, vol. 13, no. 6, p. 501, 2016, doi: https://doi.org/10.1071/sh16037.

[7] L. M. Ward, S. E. Erickson, J. R. Lippman, and S. Giaccardi, "Sexual Media Content and Effects," Oxford Research Encyclopedia of Communication, Aug. 2016, doi: https://doi.org/10.1093/acrefore/9780190228613.013.2.

[8] H. R. Knight, K. L. Hartman, and A. Bennett, "Gun Violence, eSports, and Global Crises: A Proposed Model for Sport Crisis Communication Practitioners," *Journal of Global Sport Management*, pp. 1–19, Mar. 2019, doi: https://doi.org/10.1080/24704067.2019.1576144.

[9] J. Kim and M. Kim, "Spectator e-sport and well-being through live streaming services," *Technology in Society*, vol. 63, p. 101401, Nov. 2020, doi: https://doi.org/10.1016/j.techsoc.2020.101401.

[10] T. Y. Qian, J. J. Zhang, J. J. Wang, and J. Hulland, "Beyond the Game: Dimensions of Esports Online Spectator Demand," *Communication & Sport*, vol. 8, no. 6, p. 216747951983943, Apr. 2019, doi: https://doi.org/10.1177/2167479519839436.

[11] J. C. Tang, G. Venolia, and K. M. Inkpen, "Meerkat and Periscope," *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, May 2016, doi: https://doi.org/10.1145/2858036.2858374.

[12] Z. Lu, H. Xia, S. Heo, and D. Wigdor, "You Watch, You Give, and You Engage: A Study of Live Streaming Practices in China," *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*, 2018, doi: https://doi.org/10.1145/3173574.3174040.

[13] J. Zhou, J. Zhou, Y. Ding, and H. Wang, "The magic of danmaku: A social interaction perspective of gift sending on live streaming platforms," *Electronic Commerce Research and Applications*, vol. 34, p. 100815, Mar. 2019, doi: https://doi.org/10.1016/j.elerap.2018.11.002.

[14] C.C. Chen and Y.C. Lin, "What drives live-stream usage intention? The perspectives of flow, entertainment, social interaction, and endorsement," *Telematics and Informatics*, vol. 35, no. 1, pp. 293–303, Apr. 2018, doi: https://doi.org/10.1016/j.tele.2017.12.003.

[15] J. Woodcock and M. R. Johnson, "Live Streamers on Twitch.tv as Social Media Influencers: Chances and Challenges for Strategic Communication," *International Journal of Strategic Communication*, vol. 13, no. 4, pp. 321–335, Aug. 2019, doi: https://doi.org/10.1080/1553118x.2019.1630412.

[16] *Twitch*. (2011), Amazon. [Online]. Available: https://www.twitch.tv/.

[17] J. Woodcock and M. R. Johnson, "Live Streamers on Twitch.tv as Social Media Influencers: Chances and Challenges for Strategic Communication," *International Journal of Strategic Communication*, vol. 13, no. 4, pp. 321–335, Aug. 2019, doi: https://doi.org/10.1080/1553118x.2019.1630412.

[18] K. Pires and G. Simon, “YouTube Live and Twitch,” *Proceedings of the 6th ACM Multimedia Systems Conference on - MMSys ‘15*, 2015, doi: https://doi.org/10.1145/2713168.2713195.

[19] M. Maloney, S. Roberts, and A. Caruso, “‘Mmm … I love it, bro!’: Performances of masculinity in YouTube gaming,” *New Media & Society*, vol. 20, no. 5, pp. 1697–1714, Apr. 2017, doi: https://doi.org/10.1177/1461444817703368.

[20] *AfreecaTV*. (2011), AfreecaTV. [Online]. Available: https://www.afreecatv.com/.

[21] H. Song, “The Making of Microcelebrity: AfreecaTV and the Younger Generation in Neoliberal South Korea,” *Social Media + Society*, vol. 4, no. 4, p. 205630511881490, Oct. 2018, doi: https://doi.org/10.1177/2056305118814906.

[22] M. Cheng, K. Cai, and M. Li, “RWF-2000: An Open Large Scale Video Database for Violence Detection,” *IEEE Xplore*, Jan. 01, 2021. https://ieeexplore.ieee.org/document/9412502.

[23] I. Serrano Gracia, O. Deniz Suarez, G. Bueno Garcia, and T.-K. Kim, “Fast Fight Detection,” *PLOS ONE*, vol. 10, no. 4, p. e0120448, Apr. 2015, doi: https://doi.org/10.1371/journal.pone.0120448.

[24] Y. Gao, H. Liu, X. Sun, C. Wang, and Y. Liu, “Violence detection using Oriented VIolent Flows,” *Image and Vision Computing*, vol. 48–49, pp. 37–41, Apr. 2016, doi: https://doi.org/10.1016/j.imavis.2016.01.006.

[25] “Rapid Application Development (RAD),” *Cost Efficient IT*. https://www.agilelonestar.com/knowledge-base/rapid-application-development (accessed Nov. 20, 2023).

APPENDIX A

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year : Y3T3	**Study week no.:** Week 1
Student Name & ID : Lim Yu Qi 20ACB01239	
Supervisor : Dr. Jasmina Khaw Yen Min	
Project Title : e**S**tream: A Mobile Live Streaming Platform for Esports	

1. WORK DONE

Completed:

- ✓ Functions for Community module. [Create post, view post, filter post, search post, like, comment, share]
- ✓ Functions for Message module. [Chat, search user, view user profile, follow and unfollow]

2. WORK TO BE DONE

Functions for News Module.

3. PROBLEMS ENCOUNTERED

Looking for suitable news API.

4. SELF EVALUATION OF THE PROGRESS

Plan to complete News Module.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year : Y3T3	**Study week no.:** Week 3
Student Name & ID : Lim Yu Qi 20ACB01239	
Supervisor : Dr. Jasmina Khaw Yen Min	
Project Title : e**S**tream: A Mobile Live Streaming Platform for Esports	

1. WORK DONE

Completed:

- ✓ Functions for News module. [Browse news, search news, filter news, share news]
- ✓ Implementation of Esports news API for News module.
- ✓ Use case description and activity diagram of News and Community module.

2. WORK TO BE DONE

Functions for Home Module.

3. PROBLEMS ENCOUNTERED

- Looking for suitable live stream SDK.
- Looking for suitable Esports live stream API.
- Android studio IDE failed to launch.

4. SELF EVALUATION OF THE PROGRESS

- Plan to complete Home Module.
- Try to resolve IDE problem.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year : Y3T3	**Study week no.:** Week 5
Student Name & ID : Lim Yu Qi 20ACB01239	
Supervisor : Dr. Jasmina Khaw Yen Min	
Project Title : e**S**tream: A Mobile Live Streaming Platform for Esports	

1. WORK DONE

Completed:

- ✓ Functions to Start Live and Stream Live.
- ✓ Use case description and activity diagram of Message module.

2. WORK TO BE DONE

Continue to complete functions for Home Module.

3. PROBLEMS ENCOUNTERED

- Delay the progress due to Android Studio IDE problem.

4. SELF EVALUATION OF THE PROGRESS

- Plan to complete Home Module.
- Try to resolve IDE problem.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year : Y3T3	**Study week no.:** Week 7
Student Name & ID : Lim Yu Qi 20ACB01239	
Supervisor : Dr. Jasmina Khaw Yen Min	
Project Title : e**S**tream: A Mobile Live Streaming Platform for Esports	

1. WORK DONE

Completed:

- ✓ Implementation of Live Matches API.
- ✓ Functions to add schedule to calendar.

2. WORK TO BE DONE

- Complete Feedback handling functions.
- Implement violence detection method for image and video.

3. PROBLEMS ENCOUNTERED

- Looking for suitable violence detection method.

4. SELF EVALUATION OF THE PROGRESS

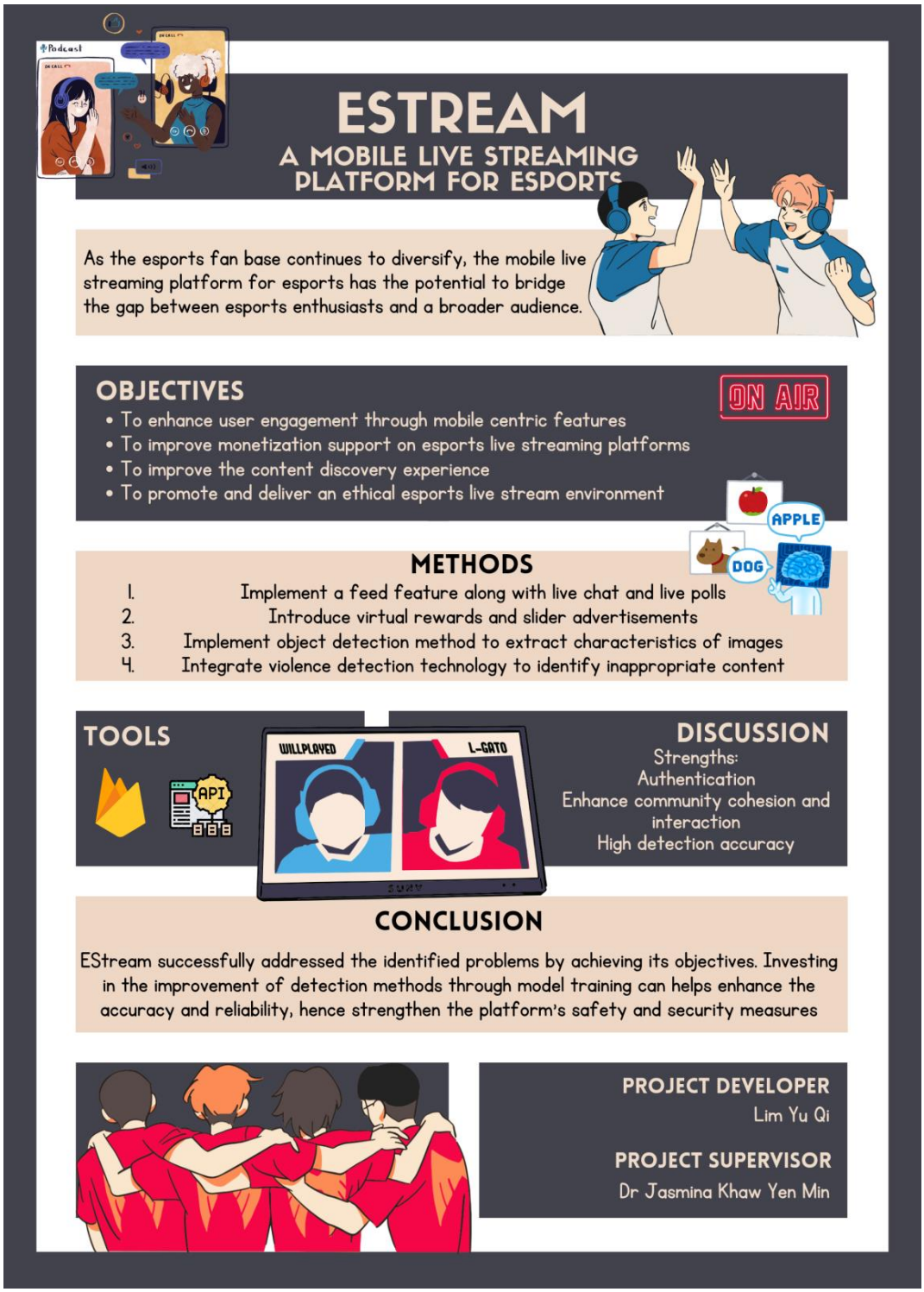
Fair

Supervisor's signature

Student's signature

APPENDIX B

POSTER



PLAGIARISM CHECK RESULT

20ACB01239_FYP2.docx

ORIGINALITY REPORT

17%	11%	2%	13%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Universiti Tunku Abdul Rahman Student Paper	2%
2	doczz.net Internet Source	2%
3	Submitted to Universiti Teknologi Malaysia Student Paper	1%
4	Submitted to University of Greenwich Student Paper	1%
5	Submitted to Sunway Education Group Student Paper	1%
6	Yuan Gao, Hong Liu, Xiaohu Sun, Can Wang, Yi Liu. "Violence detection using Oriented VIolent Flows", Image and Vision Computing, 2016 Publication	1%
7	Submitted to De Montfort University Student Paper	<1%
8	www.coursehero.com Internet Source	<1%

PLAGIARISM CHECK RESULT

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Lim Yu Qi
ID Number(s)	20ACB01239
Programme / Course	Bachelor of Computer Science (Honours)
Title of Final Year Project	eStream: A Mobile Live Streaming Platform for Esports

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 17 **%** **Similarity by source** Internet Sources: 11 % Publications: 2 % Student Papers: 13 %	
Number of individual sources listed of more than 3% similarity: 0	

Parameters of originality required and limits approved by UTAR are as Follows:
(i) Overall similarity index is 20% and below, and
(ii) Matching of individual sources listed must be less than 3% each, and
(iii) Matching texts in continuous block must not exceed 8 words
Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Jasmina Khaw Yen Min

Date: 26/04/2024

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB01239
Student Name	Lim Yu Qi
Supervisor Name	Dr. Jasmina Khaw Yen Min

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
N/A	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

_____

(Signature of Student)

Date: 26/04/2024