PRESIDENCY UNIVERSITY

Private University Estd. in Karnataka State by Act No. 41 of 2013

Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

DEVELOPING AN INTERACTIVE GAMING SOFTWARE/MOBILE APPLICATION ON INTELLECTUAL PROPERTY AWARENESS FOR SCHOOL STUDENTS

A PROJECT REPORT

Submitted by

LAKSHMISHREYA V - 20221ISE0070

NAVANEETHA - 20221ISE0071

SINDHU PATIL -20221ISE0073

Under the guidance of,

Mr. MITHUN MS

BACHELOR OF TECHNOLOGY

IN

INFORMATION SCIENCE AND ENGINEERING

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2025

PRESIDENCY UNIVERSITY

Private University Estd. in Karnataka State by Act No. 41 of 2013

Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this report “Developing an interactive gaming software/mobile application on Intellectual property awareness for school students” is a bonafide work of “LAKSHMISHREYA V (20221ISE0070), NAVANEETHA (20221ISE0071), SINDHU PATIL (20221ISE0073)”, who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in INFORMATION SCIENCE AND ENGINEERING, during 2025-26.

Mr. Mithun M S

Project guide
Presidency School of Computer Science and Engineering
Presidency University

Dr. Zafar Ali Khan N

Head of the Department
Presidency School of Computer Science and Engineering
Presidency University

Dr. Shakkeera L

Associate Dean
Presidency School of Computer Science and Engineering
Presidency University

Dr. Duraipandian N

Dean

PSCS & PSIS
Presidency University

Dr. Sampath A K

Project Guide
PSCS
Presidency University

Ms. Suma N G

Program Project Coordinator
PSCS Presidency University

Sl. No	Name	Signature	Date
1			
2			

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We the students of final year B. Tech in INFORMATION SCIENCE ENGINEERING at Presidency University, Bengaluru, named LAKSHMISHREYA V, NAVANEETHA, SINDHU PATIL, hereby declare that the project work titled **"Developing an interactive gaming software/mobile application on Intellectual property awareness for school students"** has been independently carried out by us and submitted in partial fulfillment for the award of the degree of B. Tech in INFORMATION SCIENCE ENGINEERING during the academic year of 2025-26. Further, the matter embodied in the project has not been submitted previously by anybody for the award of any Degree or Diploma to any other institution.

LAKSHMISHREYA V	USN: 20221ISE0070
NAVANEETHA	USN: 20221ISE0071
SINDHU PATIL	USN: 20221ISE0073

PLACE: BENGALURU

DATE: 03- December 2025

ACKNOWLEDGEMENT

For completing this project work, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide **Mr. Mithun MS**, Assistant Professor, School of Computer Science and Engineering, Presidency University, for his moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Zafar Ali Khan, Professor, Head of the Department, Presidency School of Computer Science and Engineering** Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr. Sampath A K, and Dr. Geetha A, PSCS** Project Coordinators**, Ms. Suma N G, Program Project Coordinator**, Presidency School of Computer Science and Engineering, or facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering and also staff from other departments who have extended their valuable help and cooperation.

LAKSHMISHREYA V
NAVANEETHA
SINDHU PATIL

ABSTRACT

Intellectual Property Rights (IPR) are actually significant to safeguard creativity and originality yet majority of school students are not acquainted with them. Typically, IPR is known only among professionals or legal practitioners or college students, which creates a huge divide in the early education. MarkMaster -Interactive Gaming Application to IPR Awareness will be our project to fill this gap by educating school students on the awareness of IPR in a fun and interactive manner.

The methods of gamification such as quizzes, puzzles, levels, and leader boards were applied to make the learning process about Patents, Trademarks, Copyrights, and Designs interesting and easy to retain. The app is interactive, uses storytelling on a visual representation and tracks the progress of students in real-time to keep them stimulated as opposed to dull lectures or boring booklets.

The uniqueness of our project is that it is dedicated to school students, runs on both a mobile platform and a web platform, and it utilizes a cloud-based technology to be scaled. Analytics dashboards can also be used to monitor the progress of students, as this assists teachers and policymakers in knowing the level at which the students are learning. We found out that the students were able to learn faster and better and experienced the learning process more enjoyable than with the traditional means. In our view, this project has the potential to introduce a culture of IPR awareness in young learners and can be expanded and utilized in schools throughout this country.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

1. IPR – Intellectual Property Awareness
2. API – Application Programming Interface
3. UI – User Interface
4. RBAC – Role-Based Access Control
5. SQL – Structured Query Language
6. HTTP – Hypertext Transfer Protocol
7. JSON – JavaScript Object Notation
8. SDLC – Software Development Life Cycle
9. SDG – Sustainable Development Goal
10. IoT – Internet of Things (if applicable to deployment)
11. PG – Post Graduate
12. UX – User Experience
13. DB – Database
14. DE – Integrated Development Environment
15. SDK – Software Development Kit
16. CIA – Confidentiality, Integrity, Availability
17. OTP – One-Time Password
18. ML – Machine Learning
19. AI – Artificial Intelligence
20. AR – Augmented Reality

CHAPTER 1

INTRODUCTION

1.1 Background

In the contemporary society that is marked by knowledge, IPR safeguards inventions, creative works and innovations. All information about early IPR awareness should be known to foster creativity and respect their originality and to make young learners realize that their ideas are their assets as well. Even the elementary levels of IPR: patents, copyrights, trademarks, and designs, are generally unheard of to school students in India and many parts of the world. This knowledge gap stifles their skills on knowledge of protection of ideas and demeans their skills on appreciation of innovation.

The government programs such as the NIPAM and CIPAM have attempted to establish awareness via lectures, booklets, and professional discussions and campaigns. The useful methods are rather theoretical and one-way; this tends not to attract the interest of the student. The booklets distributed in print are quickly forgotten and seminars or lectures seldom have a sustained appeal. This leads to poor retention and minimal motivation to learn more on IPR by the students upon such sessions.

In the meantime, the sphere of learning in students has radically altered with the development of online education and interaction. The studies on the use of gamification in education point to the involvement of games, puzzles, quizzes, and story-based activities being more likely to increase involvement and retention than traditional methods. Gamified platforms are effective in providing a motivational active engagement where scores, badges, and leader boards are awarded to the learners. This motivates and strengthens students, which makes the learning process pleasant.

No specific gamified platform that imparts IPR at the school level exists. The existing online sources are either text-intensive, with a professional target audience, or are not interactive enough to appeal to children. It also has minimal chances of tracking progress or measuring results. What is needed now is the child friendly, interesting and expandable solutions.

It does it by making a gamified mobile app that presents the notions of IPR in gradual levels of easy, medium, and high-level categories with the help of quizzes, mini-games, factual bits, and reward systems. The application ensures that the learning process is not only efficient but also entertaining and turns the IPR awareness to a significant experience of the school children.

1.2 Statistics of the Project

The fact that government programs, academic and educational technology research has robustly indicated the necessity of the existence of a gamified IPR awareness tool. These statistics have been exploited to suggest the gaps of the existing methods and the possibility of gamification to fill the gap.

A CIPAM awareness survey (2019) of schools in India established that only less than a fifth of the students were aware of even the most popular types of intellectual property, including a trademark or copyright. Over 70 percent of the students said that they have only heard of a patent and copyright but have no clue about what it is and how to use it. This shows a major deficiency in basic knowledge in the very school level itself.

Although the National IPR Awareness Mission has covered millions of students by conducting lectures, workshops, and printed material, a post-program evaluation study revealed that more than half of the students were not able to retain the information about the theoretical methods in the first place unless it was reinforced by the interactive approach in a month.

In comparison, gamified learning studies demonstrate a significant positive change in engagement and performance: Subhash & Cudney (2024) demonstrated the effectiveness of gamification in terms of participation and knowledge retention among students (40 and 50 percent higher than the lecture-based approach). The indications by Koivisto and Hamari (2024) have implied that leaders and progress tracking is what motivates younger learners, who spend an alleged two times as much time on gamified modules than on inactive learning material.

When interactive methods are used, a survey of the gamified e-learning applications indicated 6570 percent of the gamified learners to be more accurate on quizzes. The necessity of early intervention can also be supported using national education data. The National Education

Policy (2020) has also given much attention to the incorporation of innovation, creativity, and technology in the school learning. This vision is directly fulfilled by the introduction of IPR awareness based on the gamification system and the students are better equipped to handle what lies ahead.

Rapidly innovative courses: A pilot study of interactive quizzes on the law and science concepts as compared to traditional notes showed a 45 percent difference in memory in two weeks. In addition to that, the students were more willing to repeat the sessions when the activities contained some games, puzzles or rewards that were based on competition. By contrast, research in gamified learning shows measurable improvement in engagement and outcomes:

1.3 Prior Existing Technologies

- Some platforms and programs have been developed to raise awareness on IPR, primarily on university students, legal practitioners, and business people. As an illustration, the WIPO eLearning Portal offers systemic online training on various fields of IPR, including patents, copyrights, and trademarks. These modules are holistic, legally sound and they are geared towards adult learners who know the basics of legal frameworks. Equally, recently Government of India planned a National IPR Awareness Mission to help create IPR literacy in institutions of higher education. Although this program has worked at the university level, there is no doubt that little effort is reaching school students, and this has become a huge gap in introducing the students to legal knowledge early in life.
- Apart from formal platforms, several mobile applications and websites provide content related to legal literacy. Applications such as LawSikho, LegalEdge, and IP India facilitate access to legal glossaries, case studies, and updates on policy matters. However, these tools are immensely text-heavy with very little scope for interactivity. The content is presented in complicated legal jargon that is well beyond younger minds. Furthermore, these platforms have not been designed using any pedagogical principles that could be suitable for school-level learners, and hence engagement is low and retention limited.

- Examples of gamified learning platforms include Kahoot! Quizizz, and Duolingo. These illustrate the success of game-based learning in increasing student motivation and learning outcomes by utilizing mechanisms like scoring, badges, and leaderboards that make learning fun and competitive. However, all these platforms are designed to teach general subjects like language, mathematics, and science. None of these offer modules specifically relating to IPR education, not to mention the particular challenges of teaching legal concepts to school students.
- The inability of any existing technology to provide age-appropriate content with IPR-based gamification for the school curriculum represents a huge opportunity. Existing technologies are inadequate, either lacking engaging interactive learning strategies or failing to be age-appropriate. This project intends to fill that vacuum by offering a mobile app which also merges legal literacy with gamification tailored for the specific needs of school-going students. The solution will simplify the concepts of IPR, increase engagement, and inculcate responsible digital behavior early in life.

1.4 Proposed Approach

- The proposed project is the presentation of the gamified mobile application aimed at teaching schoolchildren about IPR in a way that is interesting and age-appropriate. The program uses the interactive learning methodology to make some complicated legal practices such as copyrights, trademarks, patents, and trade secrets quite easy. Through the inclusion of gameplay to the education system, the app hopes to foster curiosity, enhance retention, and additional responsible digital action among young learners.
- Core strategy: the IPR content presented as the scenario-based challenges with interactive quizzes and visual storytelling. Each of the modules is thus contextualized in real-life situations, like the process of producing a work of art or publishing digital content or the mere process of making an invention which can be easily envisioned by the students. Guided questions and feedback are then proceeded with the scenarios so that the student can put his learned knowledge into practice in a simulated way. The use of avatars, badges and progress tracking promotes motivation and therefore promotes further engagement.
- The application would be available in both Android and iOS applications with the emphasis on lightweight design and taking into account the issues of scalability and

access to low bandwidth environments. The interface is easy to use, and rich graphics, as well as reduced text, enable access to users with different levels of literacy. It would be edited with the collaboration of educators The teachers and legal professionals would check what was written to be accurate and representative of the national standards of education. It also contains multi-lingual assistance in order to cater to various linguistic groups in India.

- Some of the features that will be used to develop the process include a modular architecture to enable easy future expansions and customizations. All modules are involved with a different issue of IPR and can be elaborated without the need of the other modules. It is possible to make analytics facilitated on the backend system to determine user performances and their learning gaps. Such a qualitative method enables teachers to monitor the progress and to intervene where necessary. Comprehensively, the suggested solution will be a combination of legal literacy and gamification to an informative learning process among school students.

1.5 Objectives

The goals of the proposed project are:

• To create a mobile-based application on the principle of gamification and training the notion of Intellectual Property Rights among schoolchildren.

• To simplify complex legal concepts, e.g. copyrights, trademarks, patents and trade secrets, via interactive and interesting learning modules.

Mode 1: make students more engaged and interested in the lesson by adding game-based elements such as quizzes, prizes, and role-play.

- Standardize the content to national curriculum standards and inculcate responsible behavior in the application of digital technologies by young students.
- Supporting Sustainable Development Goal 4 (Quality Education): inclusive and equitable access to legal literacy.
- To evaluate the general effectiveness of the app with the help of systematic testing and feedback gathering, as well as performance appraisal.

- Overall, to upscale and adapt the application to various education environments and languages. The latter will guarantee a presence of a multilingual and culturally relevant information to cover both regional and linguistic groups.
- To enable educators and administrators to track student progress with in-built analytics and report tools.
- Teaching students on how they can secure their original ideas and digital creations to create creativity and innovation.

1.6 SDGs

The suggested project addresses Goal 4 of the Sustainable Development Goals of the United Nations, Quality Education to make sure that every learner acquires the knowledge and skills required to succeed in life, to contribute to the society, and to gain the benefits of lifelong learning opportunities. The application will help in establishing the fundamentals of law literacy among young learners by presenting the issue of IPR awareness on a school level. In this way, the students would be in a position to appreciate the value of their creative work, learn ownership, and be responsible in the digital world.

SDG 4 is therefore suitable in this project by rendering education accessible, inclusive, and effective with the help of gamification. The mobile application will in such a way be designed to access the students across the different socio-economic backgrounds like the rural, and semi-urban locations. Its interactive nature will promote active engagement, growth of curiosity and thus the learning experience as well. This project generates innovation, ethical conduct, and informed citizenship by incorporating legal education into early school upon the use of technology.

1.7 Overview of project report

This project report outlines the conceptualization, development and evaluation of the gamified mobile application, which is targeted at the sensitization of school students to IPR awareness. It initiates the motivation of the project, the gaps in the current IPR education system and suggests a solution. The relevance of the project in the context of digital learning, as well as its compatibility with the SDGs are also highlighted in the introduction

Fig 1.1 Sustainable development goals

The second chapter presents the literature review where past systems and educational platforms concerning IPR are described. It identifies their weaknesses on matters of accessibility, interactivity and suitability to various age groups. This chapter forms the requirement of a specialized tool that brings together the legal literacy and gamification, specifically designed to meet the needs of younger audiences. The review also gives the related research and technologies on which the proposed application design has been based.

Chapter three provides the methodology followed during the development of the application. It outlines design guidelines, development tools and implementation plans involved in development of this gamified learning experience. In the discussion, the relevant platforms that were selected, the manner in which the content has been organized into modules and game mechanics built like scoring, badges and progress tracking will be discussed. The methodology also includes testing framework to be used to assess the efficiency of the application.

It also expounds on the administration of the project including planning of the timeline, the analysis of the risks, and budgeting. The section of system analysis and design deals with system architecture, flowcharts, and module specifications. Then, the hardware and software elements are described in greater detail, as well as simulation results and code implementation. The evaluation chapter also includes the testing processes, feedback by the users, and lessons learned during the pilot deployments. It ends the report with traces of the social, legal, ethical, sustainability and safety of the project and references, base paper and appendices which include supporting materials such as screenshots, survey forms, and code snippets.

CHAPTER 2

LITERATURE REVIEW

This will pertain to a critical literature review to comprehend the state of the art of IPR awareness tools and gamified education platforms to learners in the school level. This chapter critically examines the selected scholarly articles, government programs on the IPR and mobile based learning systems with a view to appreciating gaps to the current strategies in designing the proposed gamified mobile application on IPR education. The existing systems were assessed in terms of methodologies, engagement strategies, content formats of delivery and limitations of the systems. The last section of this chapter is a synthesis of the main insights on IPR developed in the literature into a rationale of the proposed solution.

2.1 Existing Systems and Framework Review

Numerous systems and structures have been established over the recent years to enhance the awareness of IPR to be created among university students, professionals and entrepreneurs. They involve government-based programs, university websites, and mobile services that are aimed at informing the users about copyrights, trademarks, patents, and other legal concepts. Although quite informative, these platforms are oriented to adult learners and do not offer the pedagogical aspects which would make them appropriate to school-aged students.

The most significant of these is the WIPO eLearning Portal that offers organized courses on various matters related to IPR. The portal has wide penetration in legal practitioners and students in higher institutions. Similarly, the National IPR Awareness Mission developed by the Government of India targets IPR awareness at the universities and technological institutions. These models are characterized by legal correctness and policy compliance, and there is no opportunity of cognitive and engagement requirements among a younger audience.

Online platforms such as Kahoot, Quizizz, and Duolingo have demonstrated the effectiveness of the interactive learning in the area of gamified learning. Such systems combine the aspects of the game, such as scoring, badges, and leaderboards to improve motivation and retention.

They however, are majoring in the fundamental subjects such as languages, mathematics and science. None of these platforms can be dedicated to modules that are specific to teaching IPR, not to mention the issue of imparting school students with the knowledge of the law.

The analysis of the current systems and frameworks reveals very clearly the gap in the availability of age-related, gamified tools regarding IPR awareness. Most of the existing solutions are too complicated or fail to be interactive to the younger learners. The proposed project will attempt to fill that gap by creating a mobile app that combines legal literacy and gamification specifically to serve school learners. The proposed framework will be inspired by already developed successful gamified platforms but it will incorporate IPR designated material in easy and enjoyable form.

2.2 Technological Approaches and Architectures

Patel et al. (2021) focused on a gamification structure of civic education with the use of mobile applications. The authors developed a scenario-based challenge and quiz module in an Android application in their work to educate students on digital rights and Responsibilities. Firebase was used as the backend services system architecture and XML used as the user interface layout. The authors emphasized that real-time feedback, tracking progress, and the use of badges are important to maintain the engagement of the learners. Although the app demonstrated good performance in terms of user retention and learning, the research identified the shortage of scalability and multilingualism. These aspects will play a key role in broader application in other learning institutions.

Rao and Srinivasan (2020) investigated the possibility of using legal literacy modules with mobile learning to learn in secondary schools. They have a hybrid architecture whereby they employed local storage to be used during offline access and cloud-based synchronization to be used to track the performance. It was also built using React Native and a rule engine to trigger gamified features, such as scoring and level advancement. The authors emphasized the opportunity provided by modular design to update the content without interrupting the system. Their results revealed that students were more knowledgeable of the concepts of law, although they also struggled to make legal language more understandable to younger audiences and deliver age-appropriate content.

2.3 Limitations of Existing Systems

The authors, Desai and Kulkarni (2020) evaluated a mobile-based legal literacy app targeting college students. Although the system contained the systematic modules on the IPR issues, the authors observed that the material was largely text-based and did not involve the interactive qualities. There was low engagement of the target users, particularly the younger users, due to the absence of gamification. Their research highlighted the need of context and illustrations to the law concepts so that they are comprehended by school going children.

Another major weakness was that the platform took the one-size-fits-all approach, which failed to address the various levels of cognition and geographical language preference. Mukherjee et al. (2021) investigated the efficiency of online awareness related to IPR that was performed by using school webinars and non-interactive web portal. According to their findings, students did not have good retention because passive delivery methods did not give them a chance to develop a basic knowledge about IPR terminology. The paper identified a lack of feedback systems, adaptive learning strategies, and motivational features of rewards or tracking of progress. Moreover, the websites visited were unsuitable to be used on mobile devices in low-bandwidth locations, and hence not easily accessible in rural and semi-urban communities. The authors found a conclusion that in future systems the principles of gamified and inclusive design should be implemented to address these challenges.

2.4 Comparative Study

Comparing the current IPR awareness tools and the gamified educational instruments reveals that there are notable differences in the design of the knowledge, the way it is delivered, and the success of the educational process. As an illustration, Patel et al. (2021) created a gamified civic education application in Firebase and Android Studio with the emphasis on real-time feedback and the rewards in form of badges. On the contrary, Desai and Kulkarni (2020) designed a non-interactive legal literacy platform that had organized modules. Although both systems were designed with the purpose of enhancing legal knowledge, the gamified version had a greater engagement level and retention rate amongst young users. This analogy has shown the value of interactive characteristics in keeping the learner engaged, especially in abstract subjects such as IPR.

In a similar manner, Rao and Srinivasan (2020) adopted a hybrid legal literacy architecture that would have simultaneous offline access and cloud-based synchronization. Their software enabled modular updates on the content and adaptive learning tracks. Mukherjee et al. (2021), in their turn, evaluated the webinar-based IPR campaigns and discovered that passive formats resulted in low retention. The disparity between these solutions indicates that scalable, inclusive and gamified, mobile-friendly solutions are necessary. Progress tracking systems, scenario-based challenges and multi-lingual support are more appropriate to school-level students and provide a more efficient course to legal literacy.

2.5 Gamification in legal literacy for students

Gamification can be defined as a method in legal education that involves students in learning by trying to involve them in fun and playful activities (Cougle 2015). Gamification is now a familiar concept as a successful method of enhancing engagement and student learning, particularly online learning. The educational platforms can transform passive contents into active experiences by incorporating aspects such as points, badges, levels, and interactive challenges. Gamification simplifies such abstract concepts as Intellectual Property Rights (IPR) in the field of legal literacy to the understanding of individual learners in the school-going age group. Research, including Sharma and Iyer (2020), has demonstrated that gamified civic education modules are a large-scale boost to student retention and motivation in middle school education.

Bhattacharya et al. (2021) showed efficacy of the mobile-based gamified learning in teaching consumer right, in which adaptive quizzes and reward systems increased voluntary participation. Their results emphasized the significance of using language suitable to the age, visual narrative, and real-time feedback towards maintaining the interest of learners. This information substantiates the idea of including gamification as a part of the proposed IPR-awareness application that will introduce the legal concepts on the basis of the gameplay scenario and interactive modules adapted to the school students.

2.6 Summary and Identified Gaps

The literature that was reviewed confirms the possibilities that gamified mobile platforms can open to enhance legal literacy in school students. Although the current systems offer valuable information, they do not tend to offer appealing and appropriate IPR education to younger audiences. The major limitations found in studies consist of:

1. **Age Suitability**: The majority of the platforms are aimed at adults or university students, where the vocabulary is rather complicated and will not be picked up by younger learners.

2**. Absence of Gamification**: Not many systems have an interactive component such as scoring, levels, and feedback to keep users occupied.

3. **Poor Accessibility**: There is a lack of multilingual support and offline access, which restrains access in remote areas.

4. **Fragmented Features**: Storytelling, progress tracking and adaptive learning are not built in to-many existing tools.

The proposed application will resolve these gaps by combining simplified IPR contents with gamified delivery, customized assistance, and a modular layout that will meet the needs of users at the school level.

Table 2.1: Summary of Literature Reviews

Author(s), Year	Title	Focus	Key Findings / Contributions	Limitations/Gaps
Patel et al. (2021)	Gamified Civic Education App	Android-based gamified learning for digital rights	Improved engagement through badges and quizzes	Limited scalability and language support
Rao & Srinivasan (2020)	Legal Literacy via Mobile Platforms	Hybrid mobile architecture with offline sync.	Modular design enabled content updates and performance tracking.	Difficulty adapting legal terms for younger audiences.

Desai & Kulkarni (2020)	Mobile Legal Literacy for College Students	Static web module design for legal literacy using HTML and CSS.	Structured content delivery for legal topics.	Low interactivity and unsuitable for school students.
Mukherjee et al (2021)	Webinar-Based IPR Awareness	Live webinar sessions hosted via web portals for IPR awareness.	Basic exposure to IPR concepts.	Passive format, poor retention, not mobile-optimized.
Sharma & Iyer (2020)	Gamification in Civic Education	Scenario-based simulation framework for civic education gamification.	35% improvement in retention and motivation.	Challenge in balancing fun with legal accuracy.
Bhattacharya (2021)	Consumer Rights via Gamified App	Adaptive quiz engine with reward system for consumer rights education.	Reduced dropout rates and increased voluntary usage.	Limited rural reach due to connectivity issues.
Jain & Thomas (2019)	IPR Awareness in Secondary Schools	Printed module-based workshop model.	Raised basic awareness through structured sessions.	No digital integration or gamification.

Mehta et al. (2022)	Mobile Learning for Legal Concepts	Offline-capable mobile app using Android Studio and SQLite database.	Improved usability in low-connectivity areas.	Limited content scalability.
Reddy & Banerjee (2020)	Legal Education via Storytelling	Interactive storytelling using HTML5 and CSS animations for legal concepts.	Improved clarity of abstract legal concepts.	No adaptive feedback or scoring mechanisms.
Kapoor & Singh (2021)	Youth Engagement in Digital Rights	Web-based quiz platform for urban youth engagement in digital rights.	High participation and interest.	Static content, no personalization.

CHAPTER 3

METHODOLOGY

3.1 Selection of Development Methodology

The Agile Scrum methodology was chosen for this project to support iterative development, modular delivery, and ongoing feedback from stakeholders. Since the application is educational and aims to effectively engage school students, Scrum allows for sprint-based planning, early validation of features, and the ability to adjust content and design as needed. Each sprint was linked to a specific phase, ensuring structured progress and alignment with educational goals.

3.2 Methodology Stages and Project Mapping

3.2.1 Requirement Gathering Phase (Sprint 0)

The initial phase focused on identifying the educational scope and necessary technical requirements for the application. Research on gamified learning and IPR awareness was reviewed to find suitable content for students aged 12 to 16. Discussions with teachers and legal experts helped define the main topics, such as copyright, trademarks, and patents, in easier language.

Functional requirements included gamified content delivery, tracking user progress, supporting multiple languages, and allowing offline access. Non-functional requirements highlighted the need for scalability, accessibility, and responsiveness across devices. User personas were developed to represent typical learners, which guided the design of age-appropriate interfaces and content flow.

A comparison of existing legal literacy tools showed gaps in interactivity, localization, and engagement. These insights shaped the feature set of the proposed application, which aimed to mix legal accuracy with fun learning. This phase also recorded limitations, such as device restrictions and connectivity issues in rural areas.

All findings were compiled into a Software Requirements Specification (SRS) document. This document served as the basis for the subsequent design and development phases, ensuring that both educational and technical needs were clearly stated and trackable throughout the project.

3.2.2 System Design Phase (Sprints 1-2)

This phase involved designing the system architecture, user interface, and content flow. Wireframes were created to visualize navigation, gamified elements, and lesson progression. The front end was planned using Flutter for compatibility across platforms, while Firebase was selected for backend services, including user authentication, data storage, and analytics.

Design priorities included modularity, easy navigation, and visual appeal for school-age users. Gamification elements like badges, levels, and scores were aligned with learning goals. Accessibility features, such as font scaling and language options, were added to support diverse learners.

Content modules were organized into short, interactive lessons with embedded quizzes and feedback. Each module focused on a specific IPR concept, using storytelling and scenarios to improve understanding. The design also included an admin dashboard for teachers to monitor student progress and update content.

Security and privacy concerns were addressed by implementing role-based access, secure data management, and compliance with educational data standards. The design phase concluded with a finalized user interface prototype and technical blueprint for implementation.

3.2.3 Implementation Phase (Sprints 3-8)

Development took place over six sprints, each concentrating on specific modules and features. Sprint 3 handled user onboarding and profile setup, including customization based on age. Sprint 4 introduced the IPR content modules, complete with interactive lessons and quizzes. Sprint 5 focused on gamification features, including scoring systems, badges, and progress tracking.

Sprint 6 added support for multiple languages and offline access through local caching and dynamic content loading. Sprint 7 integrated the admin dashboard for teachers, allowing content updates and analysis of learner data. Sprint 8 addressed performance improvements, user interface tweaks, and bug fixes based on internal reviews.

Each sprint produced a functional build, reviewed through internal testing and feedback from educators. Development followed best practices for modular coding, version control, and documentation. The team maintained a sprint backlog and held daily stand-ups to monitor progress and resolve obstacles.

3.2.4 Testing Strategy

Testing was performed during the development cycle using a mix of unit testing, integration testing, and user acceptance testing (UAT). Unit tests confirmed the accuracy of individual components like quiz logic, scoring algorithms, and content rendering. Integration tests verified smooth interactions between modules and backend services.

Pilot testing occurred with school students in selected institutions. Participants used the app under supervision, and feedback was gathered through structured questionnaires and informal

Observation. Such measures as engagement time, performance on the quizzes, and user-friendliness were examined. The content that teachers reviewed was also checked on legal and age appropriateness.

Their comments resulted in the modification of language, speed, and design. The problem of usability, including the position of the buttons, font size, and the processing of the errors, was recorded and resolved in subsequent sprints. Issues and solutions were recorded in a bug tracking system. After every sprint, the regression testing was performed to maintain the stability. The product was finally tested by performance testing to ensure that the product could meet the load times, responsiveness, and even offline functionality.

3.3 Quality Assurance Framework

Automated tools in the development process will maintain quality standards of the code. The in-built analyzer of Dart will assist in identifying possible errors and implementing code standards within the Flutter codebase, whereas such tools as flutter format will make the formatting consistent. Such checks will be executed with every commit and pull request to avoid the introduction of style errors and logic bugs into the main stream. It will be tested through performance tests that will involve the most important user interactions, e.g., loading a student dashboard and navigation between IPR modules. User loads will be simulated under peak loading conditions to check the responsiveness and include classroom interactions and submission of several quizzes at the same time. These tests will be used to make sure that the application is stable, responsive, and available on various devices and network conditions.

3.4 Risk Management Approach

The technical risk mitigation plan will deal with potential difficulties in the process of the gamified mobile application development. Firebase monitoring tools will also be established to monitor real time data usage in order to ensure that the backend is stable and performing optimally. The indexing techniques will be used to enhance the retrieval of the content, particularly the quiz results and progress. Should third-party services fail (such as translation APIs, or cloud sync failures), then a backup strategy involving the usage of the local storage and the usage of the cached content will make sure that users still have continuous access to the services, even when the connection is at its lowest.

Early interaction with the teachers and students will be embraced in a bid to attract user adoption. Frequently, we will present operational modules, i.e. IPR lessons and gamified quizzes, to receive feedback and make them easier to use. An orderly onboarding procedure will be developed, and graphic directions, multi-language tooltips, and educator-developed orientation will be conducted. The aim of these activities is to introduce users with the application prior to full deployment and to decrease the resistance and enhance the learning outcomes.

Besides Agile Scrum, V-Model has been employed to verify and validate the project at every step. The right side of the V indicates testing and integration whereas the left side is the requirement analysis and system design. Each development phase has its counterpart testing phase; thus, content module design is countered with usability testing, and the setup of the backend is tested with performance benchmarking. This two-fold strategy will provide a consistent review of the technical and educational elements. Although V-Model is generally strict, its structured testing adds to the flexibility of Scrum, which results in a balanced and quality-based process of development. To reduce the risk of performance issues during peak usage, load-testing and stress-testing will be carried out before deployment. This will help us understand how the app behaves when many students use it at the same time. Based on the results, we can adjust database rules, server limits, and caching methods to ensure smooth performance even under heavy load.

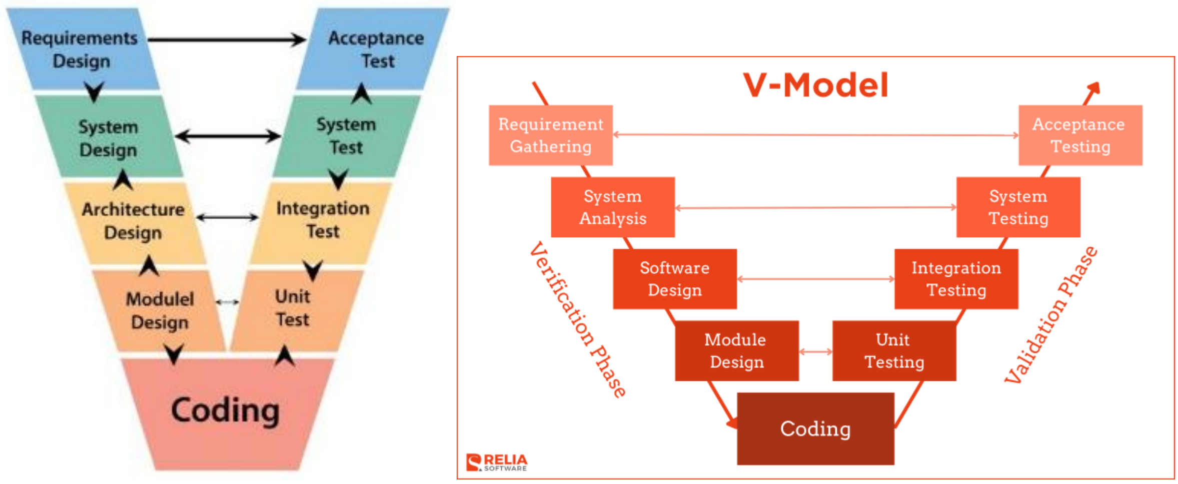


Fig 3.1 The V model methodology

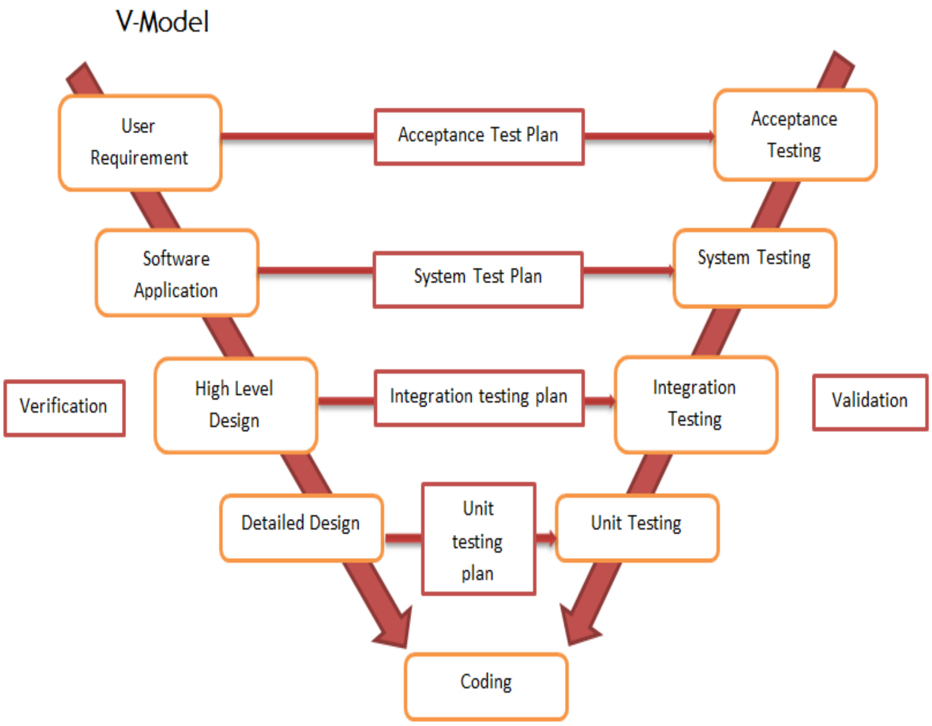


Fig 3.2 Another example of the V model methodology

W-Model was implemented to allow parallel development and testing i.e. test planning and case design would begin concurrently with requirement analysis and system design. This was useful in identifying possible problems at an early stage, in particular, in gamified modules like quizzes and scoring logic, reducing rework and improving efficiency. Combined with Agile Scrum, it guaranteed the constant verification of the technical and educational aspects of every sprint.

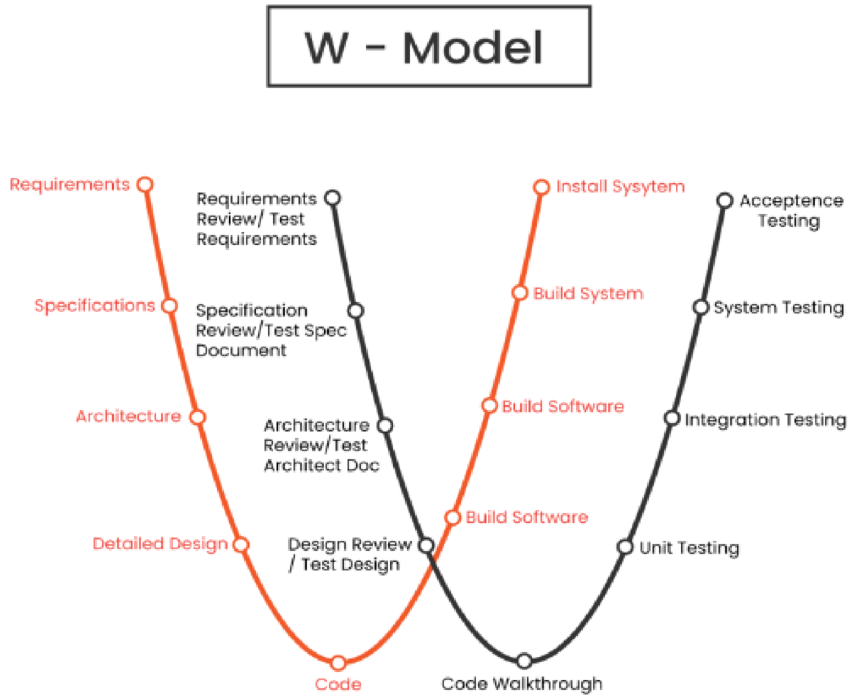


Fig 3.3 The W model methodology

DevOps allowed the quicker development and launching of the gamified app. Pipelines of automation and teams worked together to roll out the app to ensure the continuous updating of quizzes, scoring, and content with as few disturbances as possible.

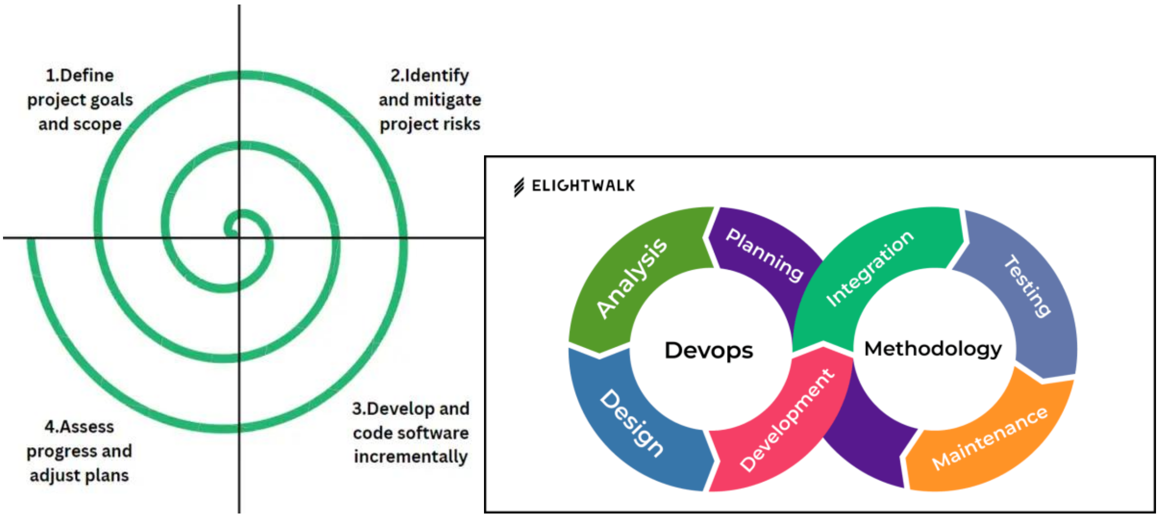


Fig 3.4 The Devops methodology

The Onion approach was used to the gamified IPR awareness app so that the clear separation of the domain logic and technical layers is ensured. This system revolves around the domain model which controls the contents of IPR delivery, quiz processes, and tracking progress.

Surrounding layers are application services, infrastructure, and user interface which are expected to develop without problems on core logic. It doesn't couple the regulation of education and gamification mechanisms with modifications of UIs or even back-end services with time.

Traditional SDLC, however, is a linear process starting with requirement gathering by way of system design with the implementation, testing, deployment and maintenance. This approach provides form and full documentation, though it does not suit a project with changing content and the development of iterations by feedback. It is cumbersome whenever changes are to be incorporated after the accomplishment of any specific stage making it unresponsive in the academic setting. Thus, SDLC model was not followed strictly but mentioned during the planning phase, which provided a leeway in the process of development.

The Onion approach makes future updates much easier. Because the main learning logic stays safely at the center, we can change the UI, update Firebase, or add new features without affecting how the core app works. This saves time and avoids breaking things when improvements are made.

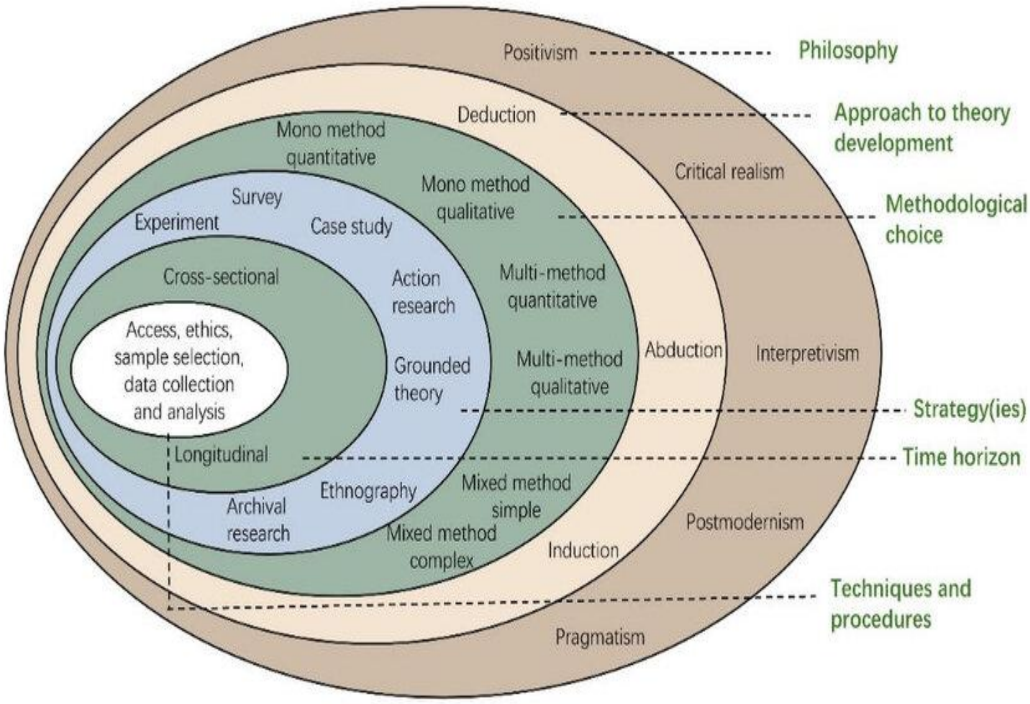


Fig 3.5 (a) Onion methodology

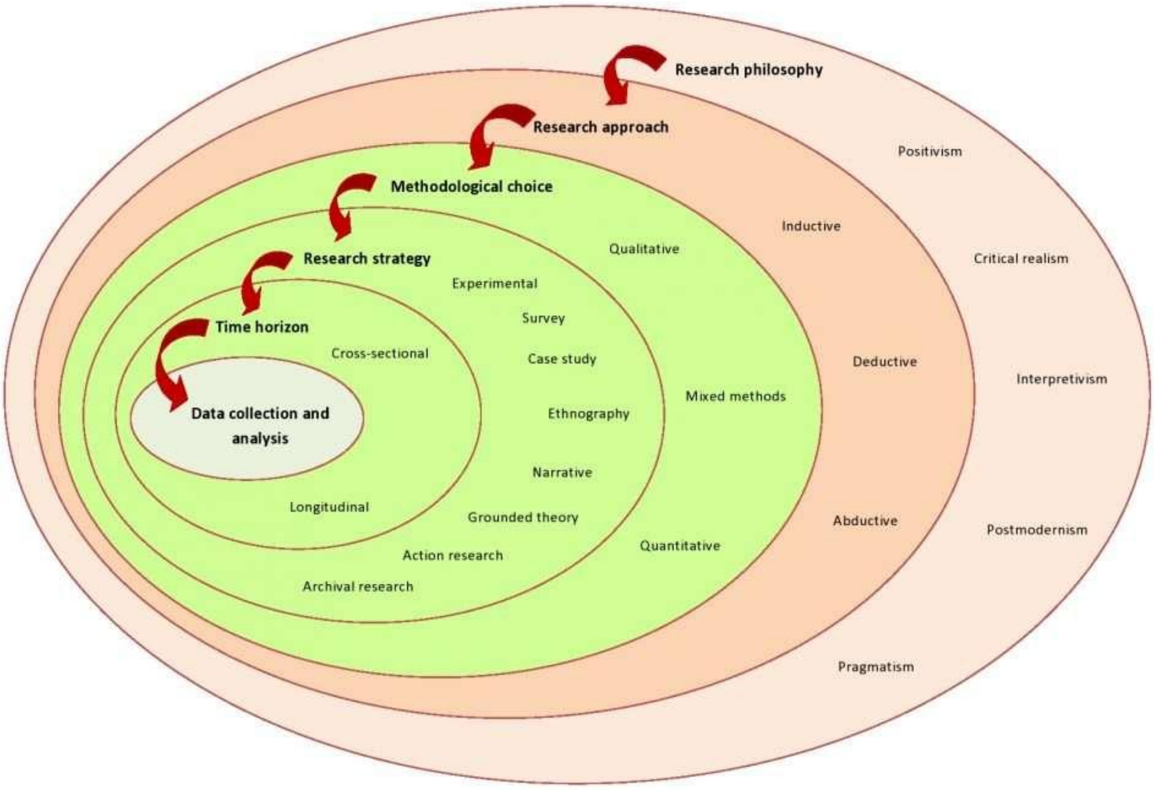


Fig 3.6 (b) Onion methodology another example

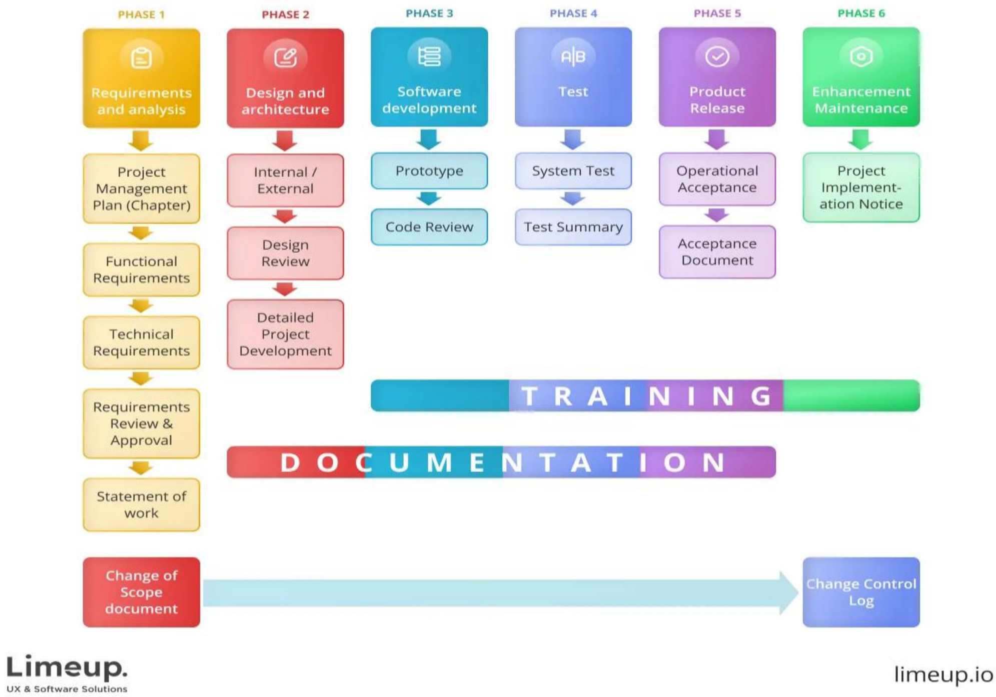


Fig. 3.7: Overview of the Software Development Lifecycle (SDLC) Phases

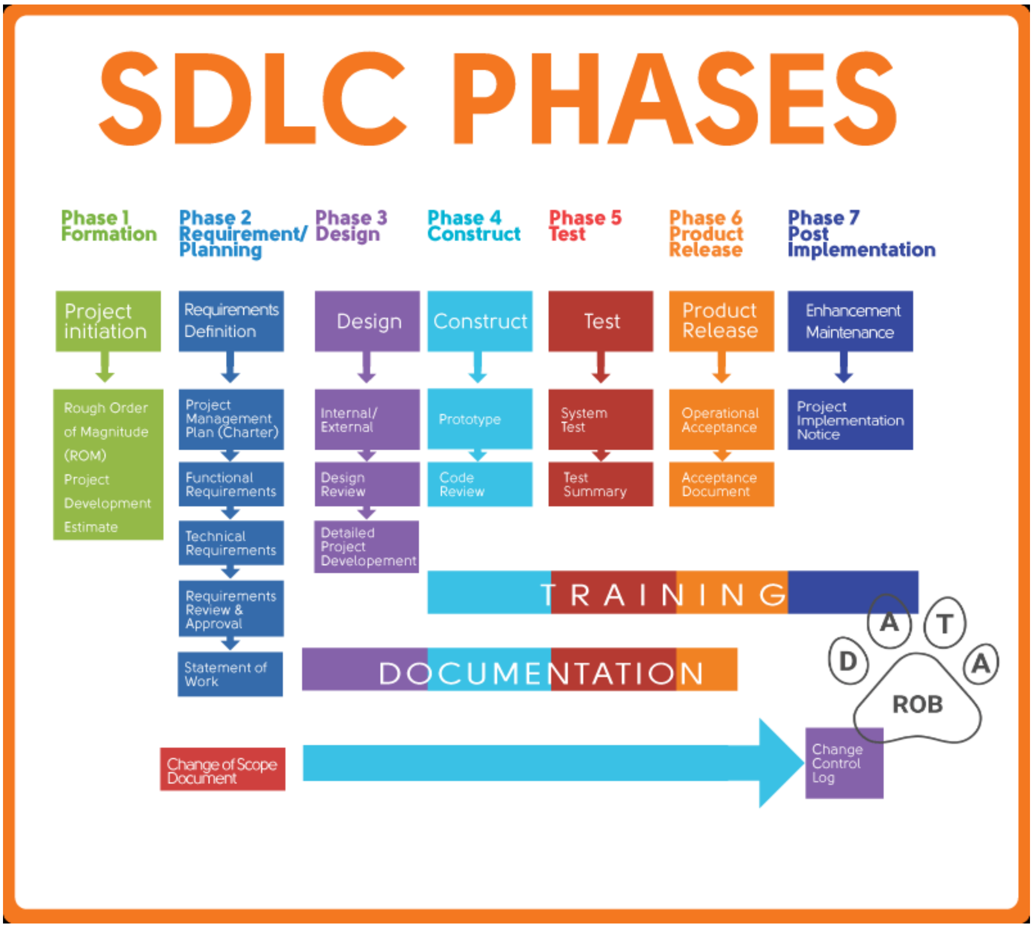


Fig 3.8 SDLC phases

3.5 Selected Methodology:

Agile-Scrum Hybrid

In the case of the project, the hybrid Agile-Scrum methodology was selected due to the assessment of the different methods. The reason was the constant necessity to refresh the content, rely on feedback of educators and students and the possibility of making repetitive changes in a mobile application.

Methodologies have been matched against the criteria of the project: the selection process was carried out.

- Requirements Stability: Feedback is to be the basis of the evolution of the educational content and the gamification features.
- Stakeholder Engagement: validation has to be done on a regular basis with educators and students through demonstrations.

- Technical Complexity: Mobile application is a good fit to develop in small iterative steps.
- Small development team: This is suitable for the team to work together with Agile.
- Time Constraints: There are fixed academic schedules, and it has to be delivered in small steps.

Agile-Scrum hybrid is a hybrid of the iterative approach of Agile and the design of Scrum. It achieves that by allowing the variation of changing educational requirements and remaining faithful to the field of time-limited sprints and frequent deliverables that meet learning objectives.

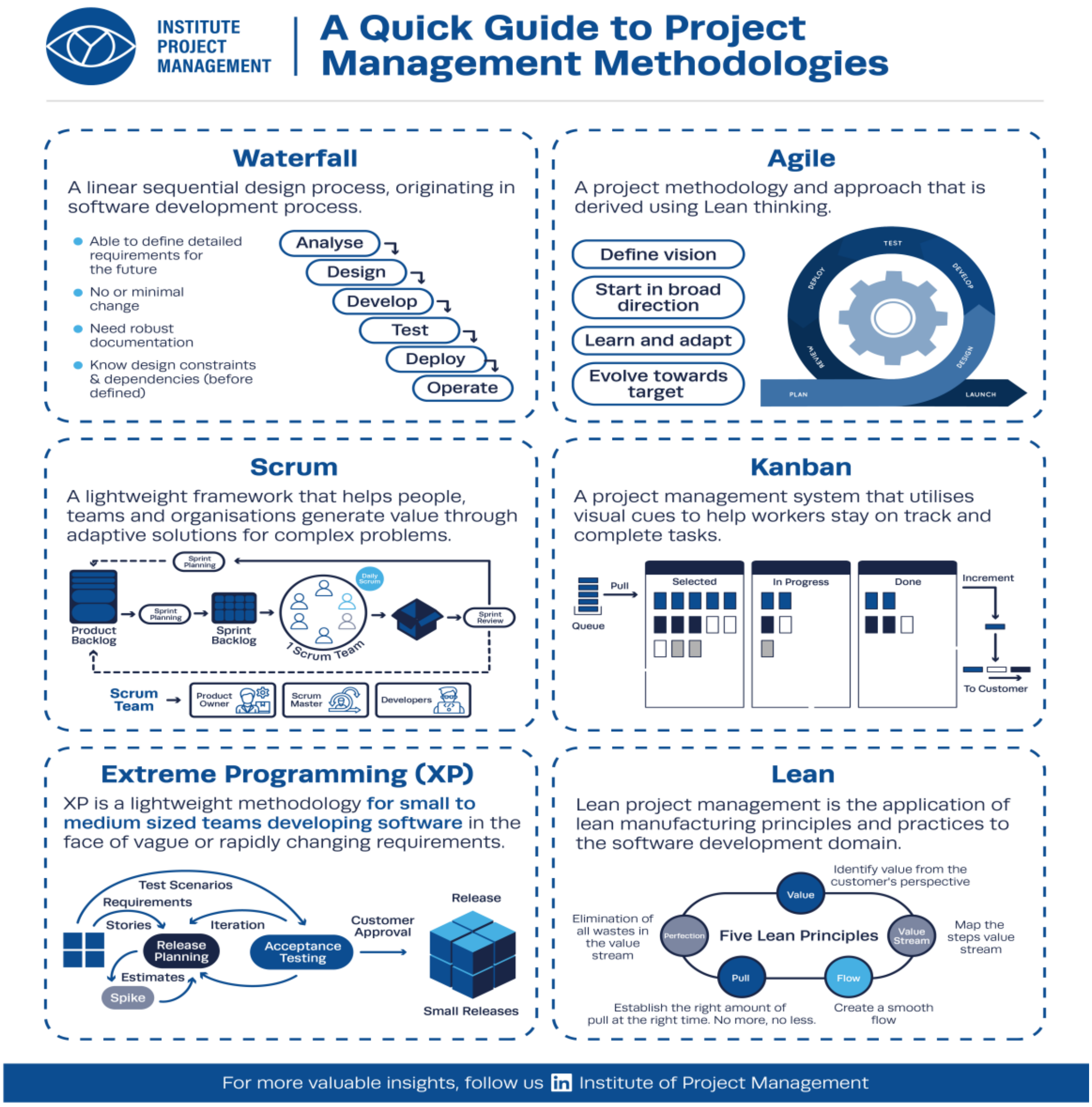


Fig 3.9 Summary of various methodology

Table 3.1: Comparative Analysis of Project Management Methodologies

PROJECT MANAGEMENT METHODOLOGIES	PROS	CONS
Agile Methodology	• Flexibility and adaptability • Collaboration and teamwork • Continuous improvement and feedback	• Lack of predictability • Requires experienced team members • Can be difficult to manage for larger projects
Waterfall Methodology	• Well-defined requirements • Predictable and structured • Easy to manage for larger projects	• Limited flexibility • No room for changes or adjustments • Can be time-consuming and costly
Scrum Methodology	• Collaboration and teamwork • Flexibility and adaptability • Continuous improvement and feedback	• Requires experienced team members • Can be difficult to manage for larger projects • Lack of predictability
Kanban Methodology	• Visual management and continuous flow • Flexibility and adaptability • Focus on quality and customer satisfaction	• Limited structure and predictability • Requires experienced team members • Challenging to manage for larger projects
Lean Methodology	• Focus on efficiency and waste reduction • Continuous improvement and feedback • Structured and predictable	• Limited flexibility • Can be difficult to manage for larger projects • Requires experienced team members
Six Sigma Methodology	• Focus on quality and process improvement • Data-driven decision-making • Structured and predictable	• Can be time-consuming and costly • Requires experienced team members • Limited flexibility
PRINCE2 Methodology	• Structured and predictable • Focus on risk management and quality • Suitable for complex projects	• Time-consuming and expensive • Requires experienced team members • Flexibility constraints

Source: CRM.org

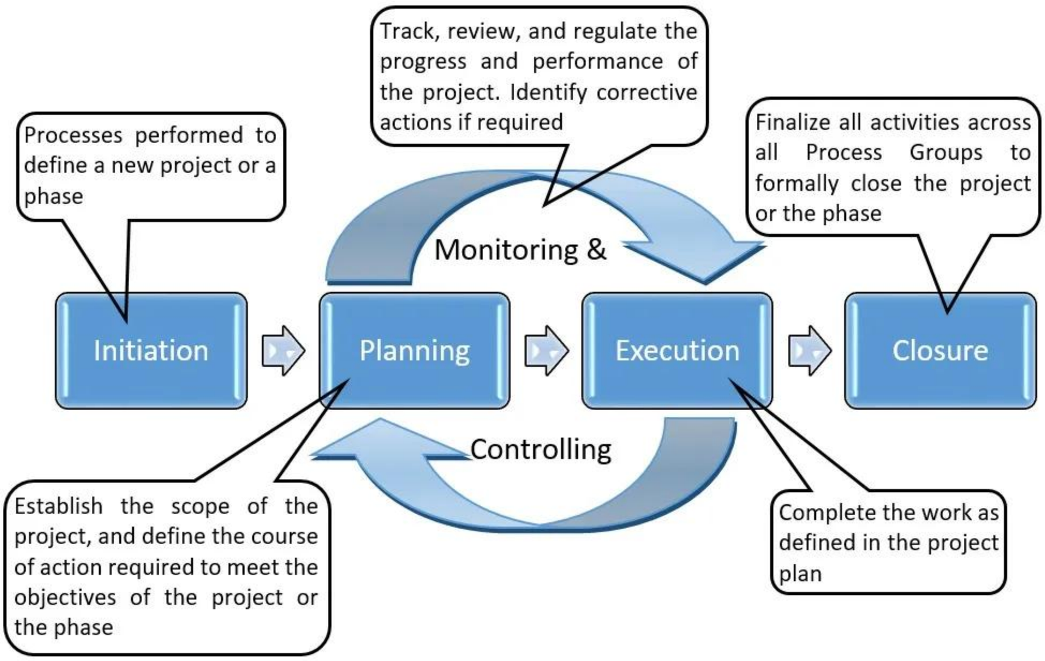


Fig 3.10 Some more methodologies

3.6 Methodology Implementation Framework

3.6.1 Project Breakdown Structure

The entire project was subdivided into well-organized parts to be executed successfully. Its major modules were to create IPR content, design quizzes and scoring logic, multi-lingual support and development of user interface, Firebase integration into backend and testing. All these modules were aligned under certain sprints within the Agile Scrum system enabling them to be developed with a certain level of focus and on-time delivery. This breakdown facilitated parallel progress across technical and educational elements, ensuring alignment with learning objectives and stakeholder expectations.

3.6.2 Sprint-Based Implementation

The project was implemented in five sprints, which focused on certain functional modules. Sprint 1 was concerned with creating the Firebase backend and the simple UI structure. Sprint 2 adopted content IPR modules and multilingual support. Sprint 3 created quiz logics and scoring systems and tracking of progress. The fourth Sprint version incorporated user feedback, fine-tuned user interface, and introduced accessibility. Sprint 5 was concerned with final testing, performance optimization and deployment. All sprints were done in Agile-Scrum, where the planning, stand-ups, reviews, and retrospectives were done daily to deliver on time and quality.

3.6.3 Quality Assurance Integration

To ensure the gamified mobile app is a reliable, usable and educationally effective tool, quality assurance was applied throughout the development process. QA began immediately at the sprint planning phase with the clarification of the acceptance criteria of the individual features which consist of IPR content modules, quiz logic, scoring mechanisms, and multilingual support. The process of writing of test cases was similar to the development of both functional and non-functional requirements validation.

The Firebase was configured with automatic testing tools to execute unit tests, integration tests, and regression checks. Android devices were tested manually in terms of user experience, responsiveness, and accessibility. Validation of quiz flow, scoring accuracy and display across screen sizes and various languages were given extra care. The bugs and inconsistencies were entered into a common tracking system and managed within the same sprint cycle.

Sprint reviews and retrospectives were used as the means to include user feedback provided by educators and student testers. The validation was a series of repetitions ensuring that the app achieved pedagogical objectives without jeopardizing its technical stability. The implementation of QA within every sprint has allowed the team to maintain the stable performance, minimized the number of reworks, and produced the polished version of the application according to the educational goals.

3.7 Alignment with Educational Environment

In the intention to suit the project to the demands of the academic setting, making the gamified mobile application capable of aiding the legal literacy among schoolchildren, the concept of Intellectual Property Rights was simplified and modularized among the learners of middle and high schools. The content was organized in such a way that it focused on relevant examples and interaction. The app was designed to create awareness on IPR by involving interactive and age friendly learning experiences that may be incorporated into the classroom instructions or as an independent activity among the students.

Besides enhancing this pedagogic efficacy, gamification was central to making sure that there was engagement with learning. The quiz modules were meant to instill important IPR concepts with the help of repetition, scoring and feedback. Students could monitor their progress, and educators could rely on performance analytics to get an insight into the understanding of students. The logic behind the scoring was skewed to reward consistency in scoring to ensure that the students repeat the content so as to get a higher score. These attributes aligned with pedagogical goals to ensure the app is not just a digital tool, but an extension that is significant to civic education.

It needed to be made open and participative. The app was supported in multiple languages allowing students to use the app in their native language. The user interface was also optimised in terms of clarity, reactiveness and navigability. This feedback was obtained in sprint reviews, conducted with educators and student testers and was applied to refine the presentation of content, change the difficulty, and improve the general usability. The matching of the technical capabilities and the academic demands placed the app as a practical and scaled solution to develop the IPR consciousness in schools.

• **Curriculum Relevance:** IPR issues were brought closer and were adjusted to the civic education objectives at both high and middle schools.

• **Age-Relevant Design:** The content module design and quiz mechanics were modified to the level of attention and thinking capacity of school students.

• **Engagement through Gamification:** Gamification was utilized through scoring, progress monitoring, and interactive quizzes to motivate the learning process and maintain interest.

• **Multilingual Support:** The application is multilingual to be as accessible as possible in a heterogeneous group of students.

• **Classroom Integration:** The app can be used by the teachers during the legal literacy or the modules could be given to the students as homework.

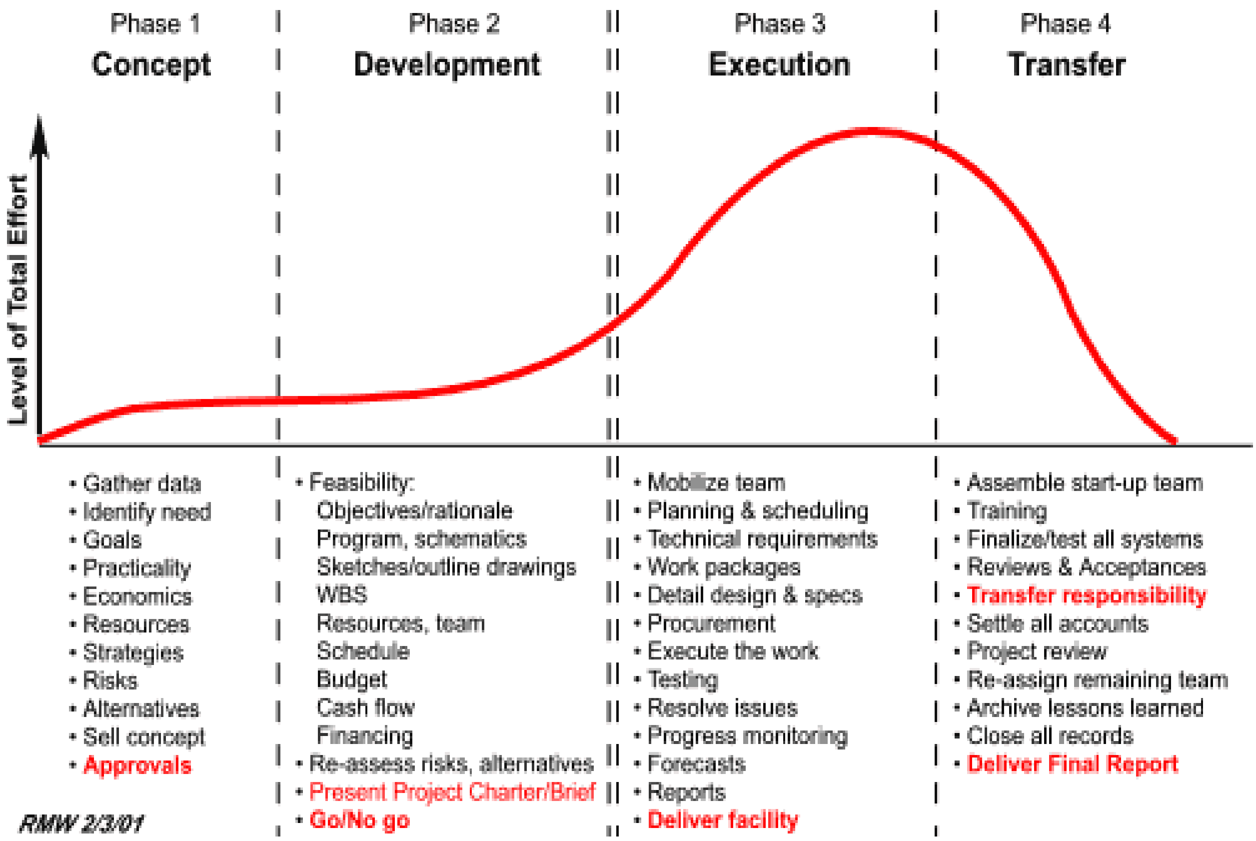


Fig 3.11: Project Life Cycle Phases and Effort Distribution

CHAPTER 4

PROJECT MANAGEMENT

4.1 Project Timeline

The project had a well-organized schedule in line with the academic milestones and sprint development. The overall period was taken to be about 16 weeks, which is broken down into five sprints in addition to the key planning, testing, and deployment phases. The time frame enabled delivery of the core features in distinct phases and approval by the stakeholders, such as, IPR content modules, quiz logic, multi-lingual support, and student tracking.

Week 1-2: Planning and Requirement Analysis This was the initial stage which was concerned with the definition of the project scope, identification of the project stakeholders, and non-functional and functional requirements. The educational objectives as well as the structure of the contents was mapped to deliver age-related information.

Week 3-5: Sprint 1 Backend setup and UI Framework Firebase integration, user authentication and simple UI layout were developed. The premise on the content and quiz delivery in a modular way was set.

Week 6-8: 2 sprint - IPR content and multilingual support The Core IPR modules were made and translated into other languages. The material was organized in the form of interactive lessons, in which there were trigger quizzes.

Week 9-11: Sprint 3 - Quiz Logic and Scoring Mechanism The quiz engine was created and scoring logic was created, which allowed one to track progress. The reinforcing learning was implemented with user feedback loops and retries.

Week 12 to 13: Sprint 4 UI Refinement and Accessibility Enhancements According to educator and student feedback, the UI elements were refined in terms of clarity and responsiveness; new accessibility features to accommodate different learners were included.

Week 14 to 15: Sprint 5 – Testing and Optimization Manual Automated testing of devices occurred. The bugs were removed; the performance was optimized to run in the classroom.

Week 16: Final Deployment and Documentation This was the last time the app was deployed and it was used to conduct a pilot test in an academic environment. Educators were provided with final documentation, user guides and feedback forms.

Table 4.1 Project planning timeline

Major Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15
Project initiation (i)															
Selection of topic															
Background (ii)															
Objectives (iii)															
Methodology (iv)															
Proposal															
Literature review (v)															
Design and Ananlysis															
System Requirement Phase (vi)															
System design phase (vii)															
Functional unit design phase (viii)															
Report															
Final report															

(i) Project initiation - Live Projects, Projects of national importance (Smart - Environment, Mobility, Governance, Building and living, People, Economy, Renewal energy, Water conservation, Waste management, Health, Education, Tourism, Irrigation, Cities), Area for projects (Communication, Embedded systems, Signal and Image processing, VLSI, Controls, Networking, Security and cryptography)
(ii) Background - Background, approach, expected results
(iii) Objectives - Statements that describe the elements to achieve project aim. Writing an objective that is SMART - Specific, Measurable, Attainable/Achievable, Realistic, and Time-bound.
(iv) Methodology - Enlist and briefly describe the different methodology. Briefly describe each stage of the applied methodology , but discuss in details relating the various stages to implement the project.
(v) Literature review - Include a brief description with appropriate illustrations. Discuss the concepts, approach, methods, analysis, and issues adopted in part or full of your approach. Identify inconsistencies, gaps and contradictions, differences. Suggest improvements
(vi) System Requirement Phase - Datasheets, Identifying initial conditions, Identifying input parameters, Identifying system outcomes, Identifying relations, Identifying system constraints
(vii) System design phase - determining functional blocks, Identifying process flow, Identifying inconsistencies, Identifying interfaces, System design and analysis, developing a integrated test plan
(viii) Functional unit design phase - Identifying components, component datasheets, compare components, Unit design and analysis, developing a unit test plan

4.2 Risk Analysis

The risk analysis was conducted in order to describe key challenges that could influence the successful development and implementation of the gamified mobile app. The identified risks were divided into the technical, operational, and academic ones, and evaluated based on the likelihood and impact. The Agile-Scrum had mitigation strategies incorporated within it to ensure that these were identified and addressed early enough.

Examples of technical risks were back-end integration issues, compatibility with several devices, and the error of multilingual render. Firebase configuration and quiz logic were tested on several devices regarding their stability. The auto test and version control did not allow the creep of any regression in the code. Bottlenecks and bugs in the user interface were fixed during sprint retrospectives.

Some operational risks involved timescale slippage, resource constraint, and communication gap. Some of the mitigation plans involved sprint planning, daily stand-ups, and role assignment. Transparency and accountability came about through sharing of a task board and documentation system. A buffer was incorporated into the schedule to support any unforeseen delays without affecting the milestones of deliveries.

Academic Risks centered around the student relevance regarding the content, the student engagement, and adoption by educators. To deal with these, IPR modules were read and tested with groups of students by subject experts. During sprint feedback, the content structure, quiz difficulty and the UI accessibility were refined. The input of educators to the strategies of classroom integration guaranteed that the app aligned itself with pedagogical intentions.

This is because, by early detection and mitigation of risks, the project proceeded at a steady pace, and the application developed was reliable and educationally pertinent. The application of risk analysis during every sprint cycle allowed maintaining both technical and academic standards during the application development.

Table 4.2 Project implementation timeline

Major Task	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	W15
Simulation															
Unit															
Integrated															
Hardware implementation															
Software															
Testing *															
Critical Evaluation **															
Social, Ethical, Legal, and Sustainability															
Report															
Final report															

* Develop test plan, Identifying test points Black box testing (positive, negative, boundary), White box testing (Control flow, Data flow, Branch, Path) Hardware testing - Unit Testing, Integrated testing Software testing System testing - Validation (dynamic, testing user requirements) Tabulating test results	** Identify the Hardware functional units - Sensors, Input devices, Micro controllers, Actuators, Output devices, Interface circuits, Signal conditioning circuits, Driver circuits Identify the Software functional units - Software component, Initializing, Acquiring, Processing, Data Logging, Controlling, Indicating Discuss the properties, issues, constraints of each functional units, Working principle, Signal type (digital or analog), Signal conditioning (signal level, noise, signal conversion), Latency, Linearity, Accuracy Discuss the aspects to improve each functional units, Reliability, Power aware, Interrupt driven, Precise timing (Real time), Indicate output, Meet standards, Safety

A PESTLE analysis was conducted to identify potential risks.

Table 4.3 Example of PESTEL analysis

P Political	E Econimic	S Societal	T Technological	E Environmental	L Legal
- Taxation policies - Trade restrictions - Tariffs - Political stability	- Interest rates - Exchange rates - Inflation rates - Raw material costs - Employment or unemployment rates	- Population growth - Age distribution - Education levels - Cultural needs - Changes in lifestyle	- Technology development - Automation - R&D	- Climate - Weather - Resource consumption - Waste emission	- Discrimination law - Consumer law - Antitrust law - Employment law - Health and safety law

Table 4.4 Another example of PESTEL analysis

P Political	E Economical	S Social	T Technological	L Legal	E Environmental
Explore: • Government stability • Financial stimulus commitment • Pandemic strategic plan • Health service readiness • Pandemic policy factors • Current taxation policy • Future taxation policy • The current and future political support • Grants, funding and initiatives • Trade bodies • Effect of wars or worsening relations with particular countries • Election campaigns • Issues featuring in political agendas	Explore: • National debt levels • Recovery struggle for impacted industry • Strength of consumer spending • Current and future levels of government spending • Ease of access to loans • Current and future level of interest rates, inflation and unemployment • Specific taxation policies and trends • Exchange rates • Overall economic situation • Real estate exodus • Inner city business decline • Supply volatility	Explore: • Pandemic lifestyle trends • demographics • consumer attitudes and opinions • media views • law changes affecting social factors • brand, company, technology image • consumer buying patterns • fashion and role models • major events and influence • Inner city pandemic trends • ethnic/religious factors • ethical issues • Digital relationships	Explore: • Relationship with pandemic • Sector technology demand • Relevant current and future technology innovations • The level of research funding • The ways in which consumers make purchases • Intellectual property rights and copyright infringements • Global communication technological advances • Internet connectivity utility	Explore: • Legislation in areas such as employment, competition and health & safety • Environmental legislation • Future legislation changes • Changes in European law • Trading policies • Regulatory bodies • Pandemic legislation • Working environment • Pandemic legal sensitivities	Explore: • Relationship with global warming • Relationship with recycling and global fight against waste • Relationship with global fight against plastic usage • The level of pollution created by the product or service • Attitudes to the environment from the government, media and consumers • Relationship with renewable energy • Relationship with deforestation

Table 4.5 Example of Project phase risk matrix

Project Phase Risk Matrix

	Regimes	Health and Safety	Environmental Impacts	Financial & Asset Loss	Reputational Damage	Production / Projects	Information Technology	Probability: Almost Impossible (1) Occurs less than once in 10 000 years	Not likely to occur (2) Occurs once in 1 000 to 10 000 years	Could occur (3) Occurs once in 100 to 1 000 years	Known to occur (4) Occurs once in 10 to 100 years	Common occurrence (5) Occurs once in 1 to 10 years
Potential Consequences	Catastrophic (5)	One or more fatalities. Irreversible health problems for employees and/or community.	On or off-site spill causing groundwater pollution, with detrimental long-term effects.	Severe financial loss or asset replacement cost impact. (> US$ 2 million)	International loss of reputation / Damaging International TV exposure with impact	Indefinite cessation of production activity / Extended project schedule slip of > 75% of plan.	Significant failure and operational downtime with permanent loss of critical data integrity.	5	10	15	20	25
	Major (4)	Partial, or medium-term, disabilities or major health problems for employees and/or	Off-site release, contained & medium-term effects on community health and/or groundwater	Major financial loss or asset cost impact. (> US$ 1 million < US$ 2 million)	National loss of reputation / Damaging National TV exposure with impact on customers.	Long-term production cutback / Major project schedule slip of 40 to 75% of plan.	System failure and operational downtime, with loss of critical data integrity and/or confidentiality.	4	8	12	16	20
	Moderate (3)	Lost-time Injuries or potential medium-term health problems for employees and/or community.	On site release, contained & restored, with medium-term effects on employees/ groundwater	Moderate financial loss or asset cost impact. (> US$ 100 000 < US$ 1 million)	Regional loss of reputation / Local radio & newspaper reports impacting suppliers/customers	Medium-term production cutback / Project schedule slip of 20 to 40% of plan.	System downtime with operational impact / restricted loss of data integrity / confidentiality.	3	6	9	12	15
	Minor (2)	Minor, very short-term health concerns or Recordable Injury cases.	On site release, immediately contained & restored, with short-term effects	Tolerable financial loss or asset cost impact. (> US$ 10 000 < US$ 100 000)	Loss of regional reputation by word of mouth re. safety performance & treatment of workers.	Short-term production cutback / Minor project schedule slip of 10 to 20% of plan	Limited downtime, recoverable data loss with limited operational impact, no security breach	2	4	6	8	10
	Insignificant (1)	Inherently safe, Unlikely to cause health problems. First aid injuries.	Minor localised spill with insignificant effects on employees and/or community.	Relatively low financial loss or asset cost impact. (< US$ 10 000)	Unsubstantiated rumours with light to moderate impact on reputation	Very short-term production cutback / schedule slip of up to 10% of plan	Limited downtime, recoverable data loss, workaround possible, no security breach	1	2	3	4	5

Low risk | Medium risk | Significant risk | High risk

4.3 Project Budget

The budget of the project was drawn to represent academic character of the work: development by students, open-source tools, and lack of external dependencies. Through this, the estimate has been broken into 5 broad categories, which include the development costs, content development, testing, deployment, and contingency planning.

1. Development Costs Flutter and Firebase are both open-source, thus, no licensing charges were incurred when developing a mobile app. During the pilot phase, Firebase free had sufficient capacity to accommodate authentication, database, and host.
    - UI/UX prototyping: I did it in the free version of Figma, thus no high-end design software was needed.
    - This involved testing devices that tested 2-3 Android devices to ensure compatibility in screen sizes and operating system.
    - Academic course work required effort by developers who were students and no money had to be provided.
2. Designing and Development of Content. IPR content was found on publicly available scholarly sources and simplified to be understood by school-going students.

- Safe translation into two local languages was carried out manually and proofreading and cultural adaptation budget was made.
- In-house design of gamification logic such as scoring and feedback was based on learning outcomes.
- The icons and illustrations were developed based on free design libraries and the free version of Canva.

3. Quality Assurance and Testing.

The student volunteers and educators conducted manual testing on various devices and user profile.

- Bug tracking and task management: GitHub Issues and Trello were employed; both of them are free of charge. Readability, contrast, and navigation were tested through accessibility, which was performed through the eyes of the younger users; the feedback provided was utilized to make any changes. The collected feedback was refined by analyzing it manually on Google Forms and then analyzed.

4. Deployment and Maintenance

- The onetime fee of publishing the application in the Google Play Store was 2000 rupees using a developer account.
- Used Microsoft Word and Google Docs to produce user documentation, educator guides, and onboarding material.
- Such factors as maintenance after the launch, like the correction of minor bugs and the content renewal of the piloting stage by the student team
- Firebase has in-built security measures that were used to ensure data privacy. There was no need to rely on any further legal advice.

5. Risk and Contingency Reserve.

- 10% buffer time was introduced to consider the possibility of timeline overruns, technical delays or amendments in the content.
- This was as a result of the backup funds that were reserved to replace the devices or upgrade the hosting in case the number of users was higher than thought of initially.

- Another budget was also allocated to re-work the content modules following recommendation of educators during pilot deployment.
- Scalability provisions were optional extension to iOS or support more languages in case of future funding.
- A small reserve was allocated for emergency technical support, such as unexpected server downtime, security issues, or API failures during deployment.
- Funds were kept aside for UI/UX redesign, in case student feedback suggested improvements in visual design or navigation during testing.
- A portion of the budget was reserved for content licensing or acquiring digital assets, such as icons, animation packs, or educational illustrations needed to enhance gamified elements.

CHAPTER 5

FOUNDATIONAL STUDY SUMMARY

This base paper investigates how gamification affects user motivation for exercise and physical activity applications. The authors indicate that gamification denotes the application of game-like features in non-game contexts; these normally involve rewards, progress tracking, and challenges. The qualitative interviews were held with 11 users who had prior experience using exercise applications that included gamified elements.

Results indicated that gamified exercise applications enhance users' awareness of their performance and progress, and in many instances, motivated them further to be more physically active. As indicated by the results, gamification may not have the same effect on all people, meaning its effectiveness depends on the user's personal preferences, attitude, and goals.

The importance of this research for our project reveals that gamification increases motivation and the level of user engagement. This is similarly the core purpose of our FunIPR gamified learning app: to facilitate learning about IPRs among students in a fun and engaging way, just as gamified exercise apps encourage physical activity. The findings from this base paper support our including elements like quizzes, rewards, badges, storytelling, and leaderboards in making IPR learning interesting for school students.

5.1 Relevance of Base Paper to the Proposed Fun IPR System

Although the base paper context is dealing with exercise and physical activity, the principle built around this is highly relevant for our project: gamification can turn an activity perceived as hard or boring into one that is enjoyable and engaging. For our project, difficult concepts like Intellectual Property Rights have to be explained to students in school; similarly, exercising on a regular basis might appear difficult for most users. By incorporating game elements like scores, milestones, levels, and rewards, learners are more open to participating and continuing with the activity.

Throughout the base paper, an important distinction is made between the process of gamification-the design and implementation of game-like features-and the experience of gamification-how the users actually feel when interacting with the system. This differentiation was instructive to us in FunIPR to ensure that we do not just add gamified components but create an enjoyable learning experience that fosters curiosity, competitiveness, and self-guided exploration.

5.1 Table: How the Base Study Supports the FunIPR Gamification Approach

Base Paper Contribution	Relevance to FunIPR
Gamification increases user motivation and engagement.	FunIPR motivates students to learn IPR concepts through quizzes and rewards.
Progress tracking enhances awareness and thereby improvement	FunIPR facilitates progress tracking, badges, and feedback
Gamification elements are well-received by users	Story-based tasks and leaderboards serve as motivators for students
Individual differences influence gamification impact	FunIPR allows self-paced, personalized learning

In order to understand clearly how the insights from this foundational study fit with our proposed solution, the following table compares the key findings from the base paper to the design goals and benefits of the FunIPR gamified learning application.

This comparison underlines that, while the original study focused on physical activity, the positive motivational effects of gamification could be successfully adapted to IPR learning, hence making FunIPR a well-supported and research-driven solution.

CHAPTER 6

ANALYSIS AND DESIGN

Two critical stages that are related to each other in any system development process are analysis and design.

Analysis stage assists in having a clear picture of the problem, researching the needs of the user, and collecting all the valuable requirements prior to developing a solution. The design stage is then concerned with how the same requirements can be put into practice in the most efficient manner.

In this project - Developing an Interactive Gaming Mobile Application on the Intellectual Property Safety among School students, the analysis step was largely concerned with how far school students actually are aware regarding Intellectual Property Rights (IPR). It also centered its attention on identifying the way on how a game-based learning can render this subject more interesting and easier to learn by them.

The essential purposes and objectives of the app were revealed through an initial-research, the comments of the teachers, and the observation of the typical learning patterns. After identifying the requirements, the design phase targeted on how the system was going to be planned and laid out.

This covered the development of the general architecture, data movement between screens and how the users engage various components of the app - lessons, quizzes, and rewards. It also included the role of Firebase in terms of handling the login, data storage, and tracking the progress, whereas Flutter is in charge of the front-end design and animations.

6.1 Requirements

The requirement phase elucidates the intent and conduct in addition to the vital features that the system is expected to possess prior to proceeding to design and implementation. It assists in ensuring that there is a common understanding of the system by both the developers and users on what the system should perform.

In this project Developing an Interactive Gaming Mobile Application on Intellectual Property Awareness among School Students, the requirements had been separated into hardware, software, data, security and the user interface. This application is a mobile application, which will help school students learn about Intellectual Property Rights (IPR) easily, enjoyably and interactively. The system consists of lessons and mini-games, and quizzes that are used to teach users about Copyrights, Trademarks, and Patents and receive the reward and new levels. Firebase is used as the manager of the backend and Flutter is used to design and develop an app.

System Purpose

This project is primarily aimed at disseminating IPR awareness to school-going children via a gamified mobile application. Students do not learn as in conventional lessons but rather learn in form of stories, images and problems which prove to be exciting to learn. The app also enables teachers to monitor the progress of students via an administrator panel.

System Behavior

There are two major modes on the system, which include Learning Mode and Game Mode. In Learning Mode, users will get access to content information on IPR in the form of brief explanations and examples.

Game Mode: Customers take part in interactive quizzes and receive points and badges.

• Firebase Authentication and Firestore Database are used to store the progress of each student in the app.

• Students are free to make a comeback and carry on with their learning process by reconnecting. Teachers or admins can check the progress and quiz scores with the help of a dashboard.

Table 6.1: System Hardware Requirements

Component	Specification
Mobile Device	Android 7.0 or higher, minimum 2 GB RAM
Internet	Stable Wi-Fi or Mobile Data Connection
Server	Firebase Cloud Hosting

Component	Specification
Processor	1.6 GHz or above
Storage	Minimum 200 MB free space

Fig 6.2: System Software Requirements

Software Component	Specification
Operating System	Windows 10 / macOS for development
Programming Language	Dart
framework	Flutter
Backend	Firebase (Authentication, Firestore, Storage)
Design Tool	Figma
IDE	Visual Studio Code / Android Studio
Database	Cloud Firestore

Data Requirements

• The application gathers the user information like the name, email, score, and the level of progress.

• Firebase Cloud Firestore has all the information that is stored securely. Only authorized (students and teachers) can have access to their respective data. System Management Requirements.

• Admin dashboard would enable the educators to see the performance of the students on quiz.

• Admins have the ability to control learning content as well as quiz questions and update modules as needed.

Security Requirements

Firebase Authentication is a system that permits registered users to access the site.

• The data is secured with coded cloud storage.

• No plaintext storage of passwords.

User Interface Requirement.

i.e. The interface of the app must be interactive, easy to use with school students and be colored. It must also offer easily visible navigation buttons, progress indicators and rewards to keep people motivated. The admins dashboards must be straightforward with simple layouts with performance monitoring filters and charts.

Summary of System Requirements

Table 6.3 Summarizing Requirements

Aspect	Description
Purpose	A gamified mobile application to promote awareness on Intellectual Property Rights among school students.
Behaviour	Offers Learning and Game modes with quizzes, scores, and rewards.
System Management	Allows admin/teachers to monitor progress and manage content.
Data Analysis	Tracks student performance, quiz accuracy, and improvement patterns.
Application Deployment	Cloud-based app, accessible on any Android device.
Security	Firebase Authentication and encrypted data storage.

System Hardware Design Phase

This stage is dedicated to the discovery and systematization of the hardware or physical infrastructure that is necessary to support the application. Although the project is primarily software-driven, the hardware phase also covers the mobile device that the students will use and the Firebase cloud platform that will be the system of the background.

1. **Determine Functional Blocks:** The hardware will consist of the Android mobile device of the user (frontend) and Firebase cloud servers (backend) which will perform the task of data storage, authentication, and hosting.

2. **Develop Process Flow**: The workflow between the mobile device and Firebase was to be designed, so that the data would be exchanged easily, including the verification of the logs in during the login process, progress of the game, and the updates of the quizzes.
3. **Identify Inconsistencies**: Potential problems that could have caused difficulties during the implementation, including poor internet speed, storage or support with the various Android version were tracked and addressed by testing.
4. **Design Interfaces**: Secure REST APIs were used to define how the app communicates with Firebase services in real-time as well as defining interfaces.
5. **System Design and Analysis**: The entire architecture was studied to make sure that it can accommodate numerous users at the same time and still carry on with performance when there is a high data usage.
6. **Developing an Integrated Test Plan**: The test cases were carried out to test how the system would behave with various mobile devices between which connectivities were expected to be stable with quick responsiveness and proper data synchronisation.

System Software Design Phase

This step determines the software composition, circulation, and logic of the application. It also dwells on the interaction of various modules in order to provide learners with an interactive learning experience.

1. **Determine Functional Blocks**: The key software blocks are the registration of users, learning modules, quiz and game logic, scoring system and the admin dashboard.
2. **Develop Process Flow**: The process map describes how a student will log in, go through learning material, do quizzes, and get feedback on their performance.
3. **Identify Inconsistencies**: Any potential logic errors between modules (e.g. error in quiz progression) or wrong calculation of scores were checked and eliminated.
4. **Design Interfaces**: The Figma was used to design the user interfaces in such a way that it would attract the school students to use the interface by providing colorful images, easy navigation, and interactive features. We also made sure that every screen was easy to understand so students could move from one activity to another without confusion.

5. **System Design and analysis**: The entire system was examined to be happy to be appropriately integrated so that there is integration of the Flutter and Firebase backend to update real-time data.
6. **Developing an Integrated Test Plan**: An elaborate testing module was developed to carry out module testing, integration testing and user experience testing as a measure to ensure that all modules can interact effectively with each other.

6.2 Block Diagram

The block diagram provides a clear picture of the way our project operates at the beginning of the end. It tells us that we understand the system easily as it displays the interrelation between every part and the flow of data among them. Each block in the diagram is a major functionality or processing one, whereby input is received by the user, processed and output in a simple and meaningful manner is displayed. Our group aimed at ensuring that the structure remains lightweight and highly structured to ensure that the flow is followed by the users and developers.

The application consists primarily of three parts: user side, processing side and admin side all linked together with the help of Firebase as the intermediary between the frontend and the backend. Firebase is used to deal with the login process, storage and synchronization of data and provides users with the ability to view their progress at any time. The design is also flexible and thus we ensure that we can add new features such as quizzes in future without a major redesign. The security of the data was also an important concern, and thus, we relied on the inbuilt authentication and security systems on Firebase to secure the user data. As a whole, this block diagram is the base of our project and demonstrates how all the elements are interrelated and cooperate to provide a student with an interactive and educative experience.

When drawing this diagram, our team talked over a number of design concepts and settled on this one since it was not complex, efficient, and easy to apply. In our opinion, this transparent architecture is not merely a symbol of how the technical part of the system works, but also a symbol of the collaboration, work planning, and knowledge that was involved in the construction of this mobile application.

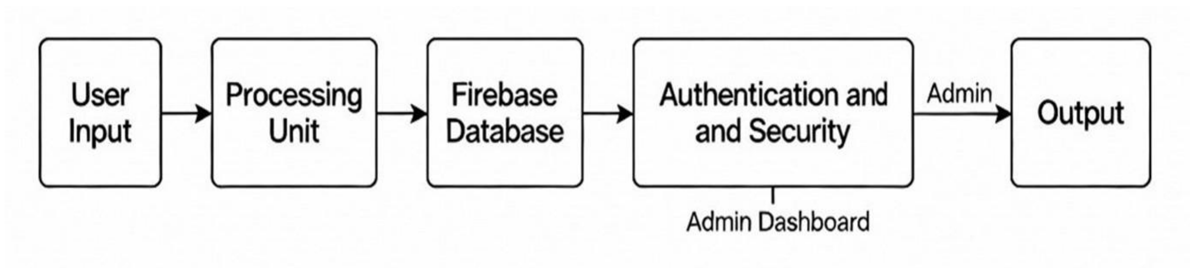


Fig 6.1 Functional Block Diagram of the Interactive Gaming Mobile Application on Intellectual Property Awareness

1. User Input Block: This block receives user input in the form of registration information, log in information and answers to quizzes. We think that this block is essential since it will have a direct connection with the users (students) and the application. Data that is inputted is then verified and sent to Firebase which further processes the data.

2. Processing Block: It is the block where the real working of the app is. It will handle the quiz logic, answer validation, awarding of points, and update the score of the user. We in our team made this section fast and easy to meet the requirement of making the learning process enjoyable and non-interrupted.

3. Database Block (Firebase Firestore): All the data, such as user profiles, quiz questions and performance scores are stored and handled in this block. We selected Firebase as it has real time updates of data and automatic synchronization, which saves manual work.

4. Authentication and Security Block: This block is in charge of the process of registration and login. It makes sure that the users of the site are registered users and therefore the information remains confidential and secure. We considered it to be significant because the app entails student details.

5. Admin Dashboard Block: This part is primarily focused on teachers or administrators. They are able to add or update quiz materials, monitor students and see performance summaries. We had a desire to implement this feature so that it would be easy but effective to monitor the learning outcomes.

6. Output Block: This block presents the students with all the final results, rewards, and progress updates. It also provides overview reports to the admin. We did this to encourage users and make the learning process a two-way process and engaging.

6.3 System Flowchart

The system flow chart provides a step-by-step picture on the way our application works. In our opinion, it is one of the most significant diagrams in our project as it made our team appreciate the whole working process of the system prior to the beginning of the development. It displays the actual sequence in which the operations should be accomplished and the interaction of the user in the app at each level. The flow starts as illustrated in Figure 6.2 when the user opens the application and logs in as an existing user or becomes a new user.

Once a user has successfully logged in, it offers two options, one of them being Learning Mode and one option being Game Mode. Under the Learning Mode, the user can be taken through the fundamentals of Intellectual Property Rights (IPR).

In Game Mode, the user will have a chance to complete some quizzes and score rewards or points. The system processes each of the responses, verifies them, and calculates the score depending on the correctness of answers. This is then saved in Firebase that provides updates to the data in real-time. Last but not least, the app shows the progress and the accomplishments of the user on the screen. By visualizing how each screen and process connects, we were able to spot unnecessary steps, improve navigation, and make the overall flow smoother for students. This early clarity reduced confusion during coding and ensured that every feature—from login to quizzes and progress tracking—was logically connected and easy for users to follow.

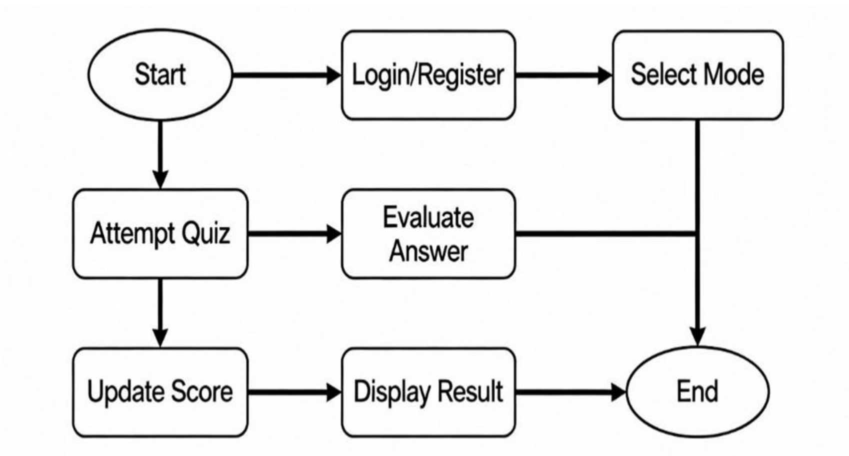


Figure 6.2 System Flowchart of the Interactive Gaming Mobile Application on Intellectual Property Awareness

The flowchart is basically composed of three processes, namely - **Initialization, Processing**, and **Output.** The first step is user login or user registration, which is verified by Firebase Authentication. Processing is used to process quiz logic, data validation and scoring.

The last process is the output when the user is shown the results, rewards and progresses. This flowchart was created by our team in order to ensure that every step is easily comprehensible and understandable.

We talked about possibilities and we kept few steps which were necessary so as not to get confused. To us, this design does not only depict the functional flow, but also demonstrates the ease of use of the system. All in all, the flowchart made the development stage more disciplined and the implementation process more efficient and without mistakes.

6.4 Choosing Devices

When it comes to our project, the selection of the appropriate devices was a critical task due to the fact the general performance and the precision of the system being determined by it. We took time and compared various options in terms of datasheets, features and compatibility to our application. Because our project is a mobile based interactive learning system, we had to select the devices and platforms that are capable of managing frontend as well as backend processes effectively. Another thing that we ensured is that the devices should be able to integrate with the cloud, real-time communication, and synchronization of data. We think that this phase allowed us to make a practical decision and realize how hardware and software choice influence the functionality of an IoT-based or cloud-integrated system. Although the project is primarily software-driven, we considered each element of the project: processor, sensors (when expanded to include physical interaction), and backend services as significant devices during the design process.

Choosing Processor

In the case of the processor, we made comparisons of various popular IoT and embedded platforms to determine their technical capabilities. Though this is the main work of our project using mobile and Firebase, this comparison was carried out to determine how this might be expanded to a physical IoT environment in the future.

Table 6.4 Comparing Features of Different Processors

Features / Specification	**Arduino UNO**	**Raspberry Pi**	**NodeMCU**	**ESP32 Dev Kit**
Power Supply	5V	5V	5V	3.3V-5V
Number of Digital I/O Ports	14	26	11	30
Voltage and Current Rating for Digital Port	5V,20mA	3.3V,16mA	3.3V,12mA	3.3V,12mA
Number of Analog Ports	6	8	1	12
Voltage Range of Analog Port	Yes	Yes	Yes	Yes
Serial Ports	0–5V	0–3.3V	0–3.3V	0–3.3V
12C/SPI/UART	Yes	Yes	Yes	Yes
PWM Pins	6	8	4	16
RAM	2KB SRAM	1GB	64KB	520KB
Wireless Connectivity	No	Yes (Wi-Fi + Ethernet)	Yes (Wi-Fi)	Yes (Wi-Fi + Bluetooth)
Bluetooth	No	Yes	No	Yes
Other Features	Simple, Low Power	High Speed, Linux Support	Compact and Cheap	Dual Core, IoT Ready

The primary purpose of the comparison of these devices was to know how data processing can be performed effectively.

None of them were as appropriate as ESP32 Dev Kit, which is the most appropriate processor to use in the IoT and real-time data exchange since it is equipped with Wi-Fi and Bluetooth integrated, low-energy usage, and dual-core processors. Although we are not currently using it in our project, we will incorporate ESP32 in later iterations as we will monitor physical student interaction (e.g. sensor-based quiz booths in schools).

To us, this comparison has made our team learn the role of selecting the appropriate hardware to ensure the reliability of performance. Although in our case the only virtual inputs were used in our mobile application, the knowledge of the selection process provided us with a better understanding of how the data of the IoT can be utilized in the future to make everything more interactive. It also assisted us in getting to know how the specifications such as voltage range, the output type, and accuracy have direct impacts on the efficiency of the system.

6.5 Designing Units

We broke down the project in small manageable and testable units to make them manageable and understandable. Since each part was developed and tested separately, it became easier to spot bugs or mistakes before combining everything. This reduced confusion, saved time, and made the overall development process smoother and more organized.

A unit has one piece of functionality and this is why the team members were able to work in parallel and integration becomes simpler. These units are listed below, along with their responsibility, inputs/outputs, and interfaces, and testing considerations.

Overview — Project Units (summary)

Main units

1. User Interface (UI) Unit
2. Content Delivery Unit
3. Quiz Engine Unit
4. Gamification & Rewards Unit
5. Authentication & Profile Unit
6. Data Sync & Offline Cache Unit
7. Admin / Educator Dashboard Unit
8. Media Storage & Asset Management Unit
9. Analytics & Reporting Unit
10. Notification Unit (Push / Local)

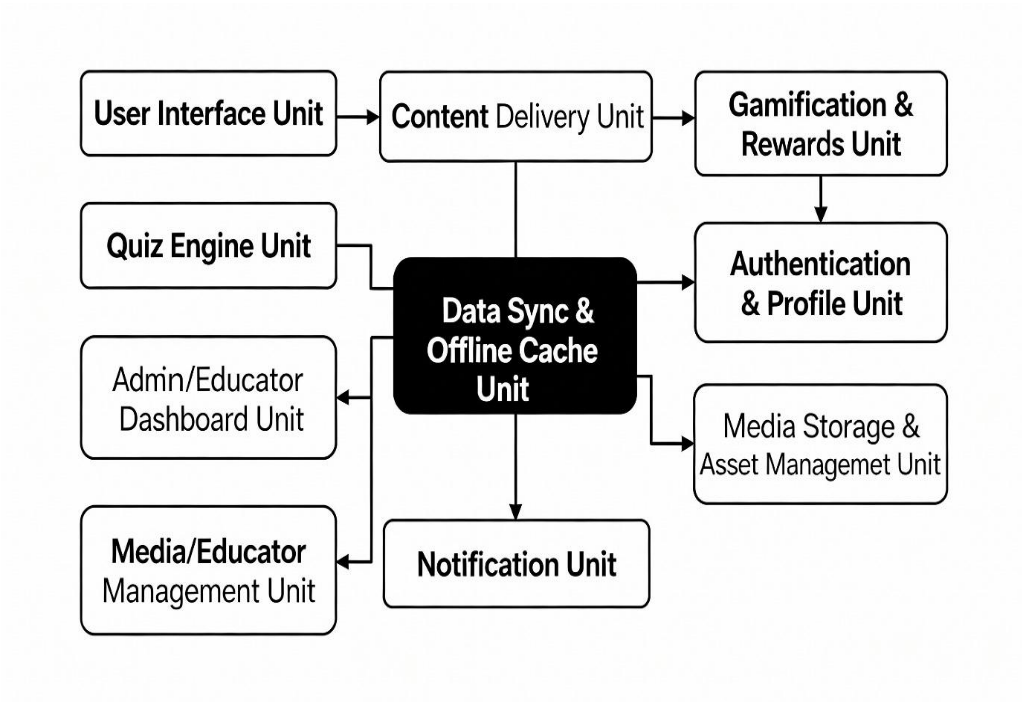


Fig 6.3 Block Diagram: High-level unit decomposition — draw each unit as a box and show arrows for data flow between them

Unit 1 — User Interface (UI) Unit

Purpose:

Show students lessons, games, quizzes and results in a child-friendly format.

Responsibilities:

- Render screens (home, lesson pages, quiz pages, profile).
- Capture user actions (taps, answers).
- Provide animations and feedback.

Inputs: user taps, quiz selections, API responses (JSON).

Outputs: API calls (fetch content, submit answers), local UI state updates.

Interfaces:

- REST / Firestore reads for content.
- Local storage for offline cache.

- Calls Quiz Engine module via well-defined functions (e.g., `startQuiz ()`, `submitAnswer ()`).

Interfacing circuits / hardware: not needed — mobile device only. In the event of physical kiosks to be used in the future, a local microcontroller will provide the UI unit with inputs via a basic serial/HTTP bridge.

Unit test plan:

- Screen navigation tests (flow between screens).
- Input validation tests (e.g., answer selection, empty fields).
- Visual regression checks for key screens.

Unit 2 — Content Delivery Unit

Purpose:
Load and pre-process learning content (text, pictures, short videos) to be displayed.

Responsibilities:

- Get something out of Firestore / Cloud Storage.
- Preprocess media (thumb generation, caching).
- Provide localized content (multi-language support).

Inputs: content id requests, user language preference.
Outputs: JSON content bundles, media URLs.

Interfaces:

- Firebase Storage for media assets.
- Firestore for content metadata.
- UI Unit uses a `getContent(moduleId)` API.

Communicating circuits / hardware: none. In the case of IoT extension, this unit may seek sensor data of ESP devices (through cloud functions).

Unit test plan:

- Content retrieval latency tests.
- Rotten/ lost media processing.
- Correctness in language switching.

Unit 3 — Quiz Engine Unit

Purpose:
Questions randomization, timing and scoring rules, and flow of control quiz.

Responsibilities:

- Serve questions to UI.
- Validate answers and compute scores.
- Maintain per-quiz state (question index, attempts).

Inputs: question sets, user answers.
Outputs: score updates, result objects.

Interfaces:

- Firestore for reading question banks and writing quiz attempts.
- Local cache for in-progress quizzes (to support offline).

Interfacing circuits / hardware: none normally. In case of the physical buttons or timers, an engine must have an endpoint that receives input over the HTTP. This allows the system to accept signals coming from external devices if needed in the future. It also ensures that even non-touch-based inputs can be handled smoothly without changing the core quiz logic.

Unit test plan:

- Proper scoring of different patterns of answers.
- Edge cases (timeout, skipped questions).
- Concurrency test (two quizzes at the same time).

Unit 4 — Gamification & Rewards Unit

Purpose:

Control badges, levels, leaderboards and milestones.

Responsibilities:

- Assign badges upon meeting criteria by the users.
- Refreshing level requirements and scoreboards.
- Production of motivational messages.

Inputs: scores, task completions, time played.

Outputs: badge events, updates to levels, notifications.

Interfaces:

- Receives an update on scores as provided by Quiz Engine.
- Saves badge/level information to Firestore to the Profile Unit.
- Transfers events to Notification Unit.

Unit test plan:

- Badge criteria tests.
- Leaderboard ordering validation.
- Condition race when there are a large number of users simultaneously.

Unit 5 — Authentication & Profile Unit

Purpose:

Register user account/log in and save user profile and permissions (student, teacher/admin).

Responsibilities:

- Integration of firebase Authentication (email / password or Google sign-in).
- Creating and updating of profiles.
- Role based access (student vs teacher).

Inputs: usernames and passwords, registration information.

Outputs: user profile records, auth tokens.

Interfaces:

- Firebase Auth (SDK).
- Firestore for profile metadata.
- Admin Unit to identify access rights.

Security notes: use Firebase Auth rules and server-side checks for sensitive operations. Always leave passwords off-line.

Unit test plan:

- Authentication success/failure flows.
- Role enforcement tests.
- Token expiry / refresh handling.

Unit 6 — Data Sync & Offline Cache Unit

Purpose:
Permit users to study at their own pace and update on the progress.

Responsibilities:

- Stores the content of cache and county of quiz being done.
- Synchronize (merge or last-write rules).
- Write to queue during an offline state.

Inputs: local changes, network state events.
Outputs: batched writes to Firestore once online.

Interfaces:

- Local DB (SQLite or Hive for Flutter).

- Firestore sync APIs.

Unit test plan:

- Offline mode: start quiz offline, complete, sync later.
- Conflict situations: two-device quizzes.

Unit 7 — Admin / Educator Dashboard Unit

Purpose:
Give the teachers a portal to manage content and control students.

Responsibilities:

- CRUD for question banks and modules.
- Class-level analytics (average score, weak topics).
- Export reports (CSV/PDF).

Inputs: admin actions, content uploads.
Outputs: updated content records, analytics views.

Interfaces:

- Firestore for content and performance data.
- Cloud Functions for heavy report generation.

Unit test plan:

- Access control tests.
- Content edit and publish workflow tests.
- Report generation accuracy.

Unit 8 — Media Storage & Asset Management Unit

Purpose:
Keep and organize pictures, mini videos and music to be used in classes.

Responsibilities:

- Upload and version assets.
- Generate thumbnails and store metadata.
- Control access (public vs protected assets).

Inputs: uploaded media, asset metadata.

Outputs: signed URLs, thumbnails.

Interfaces:

- Firebase Storage (or Cloud Storage).
- CDN for faster delivery (optional).

Unit test plan:

- Large/small file uploads.
- Broken link handling.
- Permission enforcement.

Unit 9 — Analytics & Reporting Unit

Purpose:

Examine learning and use to give feedback.

Responsibilities:

- Track engagement metrics (time on module, attempts).
- Compute mastery rates per topic.
- Provide dashboards and downloadable reports.

Inputs: user activity logs, quiz results.

Outputs: aggregated metrics, charts for admin.

Interfaces:

- Firestore export to BigQuery (optional) or Cloud Functions for aggregation.

- Admin Dashboard for visualization.

Unit test plan:

- Correctness of aggregated metrics.
- Handling large datasets.

Unit 10 — Notification Unit

Purpose:
Reminders, badge notifications and announcements.

Responsibilities:

- Push messaging (Firebase Cloud Messaging (FCM)).
- Local reminders when offline.

Inputs: Gamification, Admin announcements events.
Outputs: push notifications, in-app banners.

Interfaces:

- FCM APIs, local notification libraries for Flutter.

Unit test plan:

- Receipt on Android devices.
- Notification click flows.

Unit Integration & Testing Approach

We used a staged plan of integration after unit development:

1. **Unit testing -**Each unit is tested separately (mocking external calls).
2. **Integration testing -** Pairwise integration (e.g., UI + Quiz Engine, Quiz Engine + Firestore).
3. **System testing -**System tests in the real device and real network conditions.

4. **Acceptance testing -** Teachers and a small group of students were able to test the app and provide feedback.

Each of the units was covered with a test case which was written and used to enhance reliability and UI flow.

6.6 Standards

Compliance to proper standards is highly imperative in any system design since our project would be secure, interoperable, and future-compliant. When working on this project, our team has researched various IoT and software standards to know how they can ensure reliability of the systems and their easier integration with other technologies.

Although our project is primarily operated as a mobile application, we still wanted to adhere to general principles of the IoT design to ensure that further down the line our project could be linked with smart devices or further expanded to facilitate real-time monitoring. We think that the application of the accepted standards not only enhances compatibility of the system, but also provides our project with professional structure and follows the industry practices.

Importance of Standards

Standards are technical frameworks that define how devices, software, and services C communicate and exchange data.
They help in:

- **Interoperability:** making sure that the various systems or platforms can easily interact.
- **Security:** specification of safe communication standards and security of user information.
- **Data Consistency:** Data should be stored in a standard format to allow the reuse of data without conversion error.
- **Scalability:** enabling new features to be added without redesigning the entire system.

Types of Standards Used / Referred

In the design stage, our project mentioned several types of standards that have frequent applications in IoT and cloud-integrated applications:

1. **Communication Standards:** These determine the way data is transferred between systems and devices. These are Wi-Fi (IEEE 802.11), Bluetooth (IEEE 802.15.1) and ZigBee (IEEE 802.15.4), MQTT, CoAP and HTTP/ HTTPS. Firebase communication in our project is over HTTPS that guarantees the transmission of encrypted data.
2. **Data Format Standards:** Our project is based on the same principle and publishes and receives structured data between the mobile application and Firebase with the help of JSON (JavaScript Object Notation) JSON is minimalist, simple to read, and common in the recent applications.
3. **Security Standards:** As our application has to deal with user log-in and user progress information, we have adhered to the fundamental concepts of TLS (Transport Layer Security) as a method of encrypted communication and OAuth 2.0 as a method of user authentication via Firebase. Through these standards, unauthorized users have no access to personal information.
4. **Software Design Standards:** Coding and design guidelines that were suggested by ISO/IEC 12207 (Software Life Cycle Processes) were met and guided us through the right planning, testing and maintenance. Our version control and testing plan are also in accordance with the IEEE 829 (Software Test Documentation).
5. **IoT Architecture Standards:** Even though our project is an app, we planned it to be integrated with IoT in the future. To this end, we read ISO/IEC 30141, which outlines the IoT Reference Architecture, and ISO/IEC 27400, which gives recommendations on the IoT security and privacy. These assisted us in the realization of the way various strata of an IoT system interact with each other, starting with physical devices and ending with cloud services.
6. **Quality and Information Security Standards:** In order to be reliable, we referred to such standards as ISO 9001 **(Quality Management)** and ISO/IEC 27001 **(Information Security Management).** Such standards are aimed at the preservation of data integrity, minimization of risks, and the enhancement of user trust.

Communication and Hardware Interface Standards

In the case in which our system is expanded with the addition of an IoT device, e.g. sensors or microcontrollers, it would need hardware-level standards of communication like: UART (Universal Asynchronous Receiver Transmitter)

- **UART (Universal Asynchronous Receiver Transmitter)**
- **I2C (Inter-Integrated Circuit)**
- **SPI (Serial Peripheral Interface)**
- **CAN (Controller Area Network)**

These specifications determine the flow of data between sensors, controllers and processors. MQTT will be our message protocol between the app and the devices of the IoT in further iterations, since it is a lightweight protocol that can be used in low-power systems.

Benefits of Using Standards

There are a number of benefits that our project enjoyed by using or referring to such standards:

- **Better Interoperability:** It is possible to add the app to multiple databases or IoT devices in the future without significant modifications.
- **Improved Security:** Adherence to TLS and OAuth principles aids us in securing the data of users.
- **Lower Development Effort:** Standard frameworks save time on the design of custom protocols.
- **Professional Design Approach:** his time around we are in line with real life software projects in terms of documentation as well as structure.

In our opinion, adherence to these standards allowed our application to become not only functional, but also strong and reliable.

It also made us understand the importance of standards in the development of systems that may eventually integrate into the bigger Internet of Things ecosystem.

References

1. IEEE Standards Association (2023), IEEE *802.11:* Wireless LAN Medium Access Control and Physical Layer Specifications.
2. ISO/IEC 30141:2018, Internet of Things (IoT) Reference Architecture.
3. ISO/IEC 27400:2022, Security and Privacy for IoT Systems.
4. ISO/IEC 27001:2022, Information Security Management Systems.

5. MQTT.org (2024), MQTT Protocol Specification v5.0.
6. Wi-Fi Alliance (2023), Wi-Fi CERTIFIED™ Technical Overview.
7. Firebase Documentation (2024), Security and Authentication Guidelines, Google Developers.

6.7 Mapping with IoTWF Reference Model Layers

The IoT World Forum (IoTWF) Reference Model is a simplified version of the structure in to which one can learn the various layers of an IoT-based system. Although our project will be especially about a mobile-based application, we developed our design according to the IoTWF model to be able to visualise how each of the layers will contribute to the data flow, communication, and general interaction between the user and the system.

In our opinion, this model contributed to our team dividing the whole system into small manageable parts. It provided us a clear understanding of data collection, processing and eventually presenting it to the user in a manner that has meaning to him or her. It also demonstrated the extension that we can use the same framework in future provided we extend the project to incorporate the IoT devices in order to be engaged and track the awareness. The project to incorporate the IoT devices in order to be engaged and track the awareness.

Since each layer had a clear role, we could plan what happens where without getting confused. This helped us build the app step-by-step and fix issues quickly because we always knew which part of the system was responsible for what. Because the model clearly shows how data moves from devices to the cloud and back to the user, our team could easily imagine how new features—like adding smart classroom devices or real-time sensors—could fit into the system. This gave us confidence that the project can grow over time without needing to redesign everything from the beginning. It also made our discussions easier during team meetings, because everyone could refer to the same layered structure and understand how each part of the system fits together. This reduced misunderstandings and helped us make decisions faster.

It also helped us stay organized throughout development because the IoTWF model acted like a roadmap. Whenever we felt stuck or unsure about where a feature should go, we referred back to the layers and immediately knew the right place.

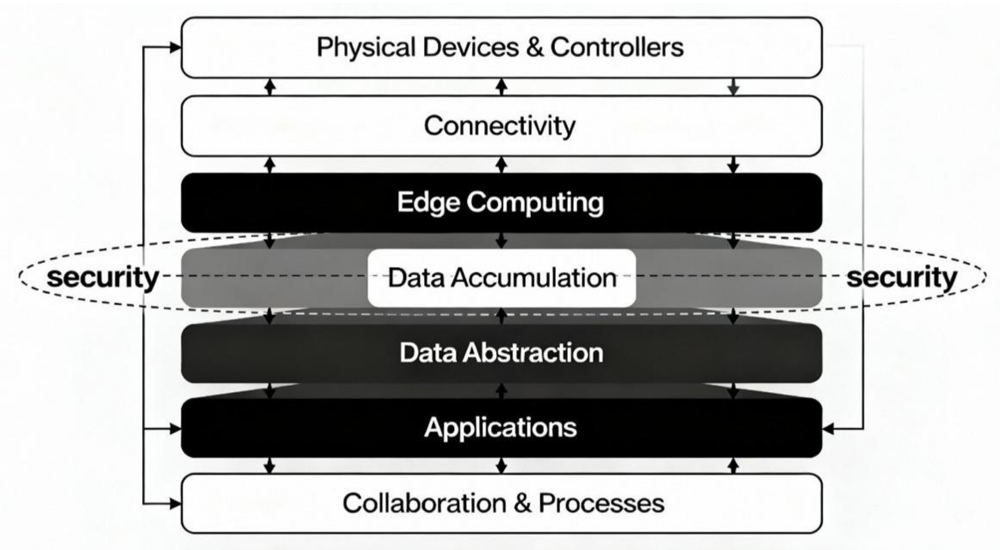


Figure 6.5 IoT World Forum Reference Model

Using this reference model, we were able to:

1. Decompose the system into different functional layers.
2. Identify which technology fits in which layer.
3. Define how each layer interacts with the other through well-defined interfaces.
4. Ensure that security and data management are applied at every level.

The mapping between the **IoT World Forum Reference Model** and our **project architecture** is summarized in **Table 6.5**.

Table 6.5: IoT Reference Model Layers Mapped to Project Architecture and Security

Layer	IoT World Forum Reference Model Description	Project Layer Mapping (Identify technologies & interfaces)	Security (Tiered security model at transition points)
7. Collaboration and Processes	Involves people, business processes, and collaboration tools.	Users, educators, and administrators interacting through the mobile application.	Secure authentication through Firebase Auth and role-based access control.

6. Application	Includes application logic, analytics, and control mechanisms.	Users, educators, and administrators interacting through the mobile application and admin dashboard.	Application-level access restrictions and encrypted communication using HTTPS.
5. Data Abstraction	Responsible for data aggregation and access.	Firebase SDK abstracts database operations and uses JSON for uniform data structure.	Firestore rules enforce authorized data access and read/write permissions.
4. Data Accumulation	Handles data storage for further analysis.	Firebase Firestore stores user scores, progress, and content data.	Cloud storage secured by Firebase Authentication and Security Rules.
3. Edge Computing	Manages data processing and transformation near the source.	App processes quiz data locally before syncing to Firebase; supports offline mode.	Local data stored in encrypted form on the device.
2. Connectivity	Provides communication between devices and cloud platforms.	Internet connection over HTTPS connects the mobile app to Firebase in real time.	TLS/SSL encryption ensures secure data transmission and authentic sessions.
1. Physical Devices and Controllers	Consists of hardware devices and controllers.	User's smartphone acts as the main controller; future integration may include ESP32-based quiz kiosks.	Device-level security through OS sandboxing and application permissions.

6.8 Domain Model Specification

The domain model establishes the key concepts, entities and their associations in our project. It aids in picturing the logical format of data and functions interaction in the system. The domain model is not tied to any particular technology or platform and is concerned only with the idea of relationships between the physical and virtual components. In our project, where the primary goal will be to create awareness about Intellectual Property Rights (IPR) by means of an interactive mobile gaming platform, the domain model will present the key entities that will include users, quizzes, scores, content modules, and the cloud database.

It explains the interconnection between these entities and the information flow that occurs among them in terms of services and resources. We feel that the development of this model provided our team with a clear notion of the actual sense of the system in a real-world setting, who will communicate with the system, what type of information will be shared.

By clearly defining what each entity means—like a user, a quiz, a score, or a content module—we avoided confusion during development. Everyone knew exactly what data each part should handle and how it should behave. This made communication easier within the team and reduced mistakes during implementation. Whenever we had doubts during development, we referred back to it to ensure we were following the correct logic and keeping the system consistent.

Table 6.6: IoT Architecture Entities and Their Mapping to the FunIPR System

Entity	Description
Physical Entity	• The real-world objects that interact with the system. • In our project, the physical entities are the users (students, teachers) using mobile devices to access the application. • The IoT system may also include real devices in the future, such as ESP32-based learning kiosks in classrooms.

Virtual Entity	• The virtual entity represents the digital version of the user within the application. • Each user has a unique profile stored in Firebase, which maintains their scores, achievements, and learning progress. • The app continuously updates this virtual representation as users play and learn.
Device	• Devices act as a bridge between physical and virtual entities. • In this project, the device is the smartphone that enables the user to interact with the app. • It collects input (answers, selections) and displays responses and progress through the UI.
Resource	• Resources are software components that provide essential system functionality. • On-device resources include the mobile operating system and app framework. • Network resources include Firebase Authentication, Firestore Database, and Cloud Storage for data management and access.
Service	• Services provide the interface for users to interact with system resources. • In this project, services include user authentication, content retrieval, quiz evaluation, progress tracking, and notification delivery. • These services ensure smooth communication between users, resources, and the database.

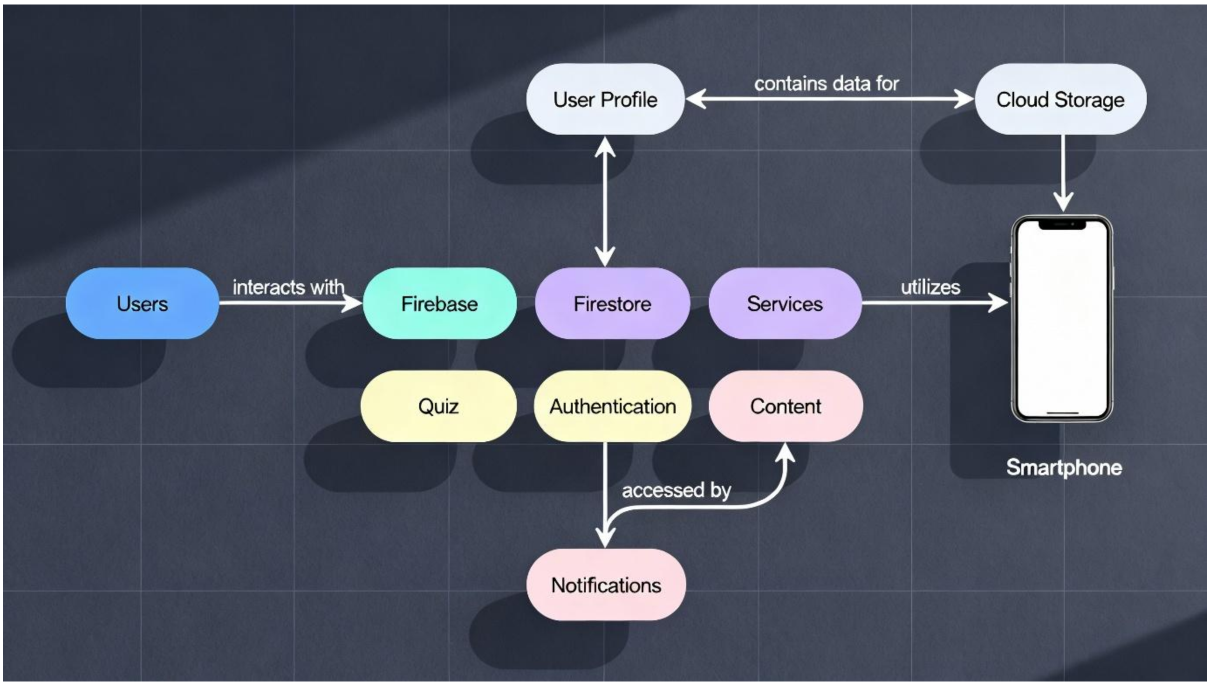


Figure 6.6 Domain Model for the Interactive Gaming Mobile Application on Intellectual Property Awareness

The domain model illustrates in a clear manner the interaction of the users, devices, and digital components with each other. It makes us visualize the logical design of the application, with each component having a certain responsibility. In this way, we managed to have all the functions of our project properly defined and easy to manage, such as authentication, data storage, etc. This method made the task of establishing relationships and dependencies easier according to our team. This model can also be expanded in the future, with the addition of physical IoT devices (such as quiz kiosks or smart displays) without restructuring the structure of the entire system.

6.9 Communication Model

Communication is a significant part of our project functioning as it will define the way in which the mobile application interchanges information with the cloud database in real-time. According to our system needs, we have chosen the Request Response Communication Model that is the most suitable model to fit our architecture and provide the security, reliability, and efficiency of data exchange between the app and Firebase.

Under this model, a device used by the user makes a request to the Firebase server whenever an event is executed, e.g., when a user logs in, provides an answer to a quiz or reads the progress of a user. A request is then processed by the server and a response is returned which validates the outcome of the operation. This model is easy to understand, straightforward and can be used in the case of mobile-based applications such as ours where action and response are initiated and expected instantly.

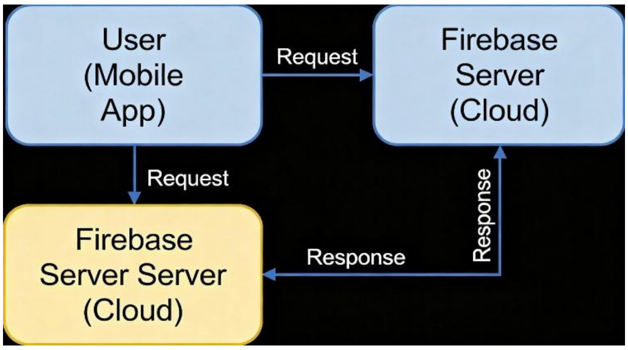


Figure 6.7 Communication Model Suitable for the Interactive Gaming Mobile Application on Intellectual Property Awareness

The **Request Response Model** was used in our project, whereby all user interactions such as signing in, taking quiz or checking progress are done efficiently using this back-and-forth communication.

Firebase is the cloud service provider which deals with all the requests and delivers appropriate responses and HTTPS provides secure communication of data. The most important strength of this model is that it is predictable and synchronous.

A single request results in a single response, and this is useful in ensuring the accuracy of data and the instant feedback to the user. It is also easier to debug with this structure, and easier to use, since the user is updated right after every action.

We believe this model is the most suitable for our application because:

- It improves the complexity of communication.
- It guarantees the data credibility and prompt recognition.
- It is very compatible with Firebase and REST API activities.
- It offers a solid foundation to scalability in case the IoT-based extensions (such as smart devices or learning kiosks) will be introduced in the future.

6.10 IoT Deployment Level

The **IoT Deployment Level** defines how devices, cloud systems, and applications interact and operate within an Internet of Things environment. Even though our project mainly runs as a mobile-based application, it follows the principles of **IoT Deployment Level 3**, which focuses on cloud-based data collection, processing, and remote access through connected devices. Our team selected this level because it perfectly represents how the application connects users (students and educators) to a centralized database (Firebase) over the internet.
In this setup, the user's smartphone acts as a node that continuously exchanges data with the cloud, allowing real-time progress tracking, quiz evaluation, and content retrieval.

Through this mapping, our team could visualize how every feature of the application contributes to the overall workflow. For instance, when a user plays a quiz, the request travels through the connectivity layer to the cloud, where the quiz logic and database layers process the input and return the results. This structure not only simplifies data handling but also keeps the communication flow smooth between the frontend (mobile app) and backend (Firebase). It also made debugging easier because each functional block had a clearly defined role.

We believe this systematic mapping has strengthened the project's design by ensuring that every module is well-integrated within the IoT architecture. It gives our system the flexibility to grow — for example, by connecting additional devices or integrating real-time analytics in the future without disturbing the existing setup.

Since all the important data—like scores, progress, and learning content—is stored and processed in the cloud, users get the same updated information no matter which device they use or where they are logging in from. Even if a student switches to another phone or logs in from a different place, their progress is still saved.

This cloud-based setup also protected us from losing data during app crashes or device issues. For our team, this meant fewer worries about device compatibility and more focus on improving the learning features. Overall, Level 3 deployment made the system feel more stable, dependable, and user-friendly for students and teachers.

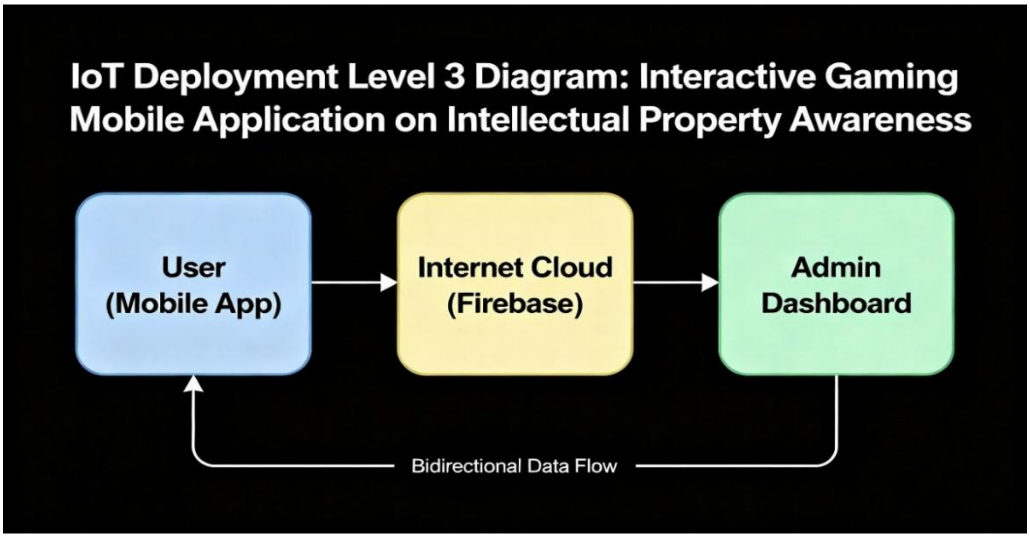


Figure 6.8 IoT Deployment Level Suitable for the Interactive Gaming Mobile Application on Intellectual Property Awareness

In this level of deployment, information is collected, processed and stored at the cloud instead of being stored on local devices. Every calculation and analysis will be made in Firebase and the results are sent back to the user interface via a secure HTTPS connection.

This model has the following strengths to our project:

Centralized Data Management: All the quiz scores, progress, and content are stored in one location in the cloud which can be viewed by the users and administrators.

- **Real-Time Access:** With Firebase, it is possible to synchronize data in real-time, so the users can see the latest hashtags and the latest performance in real-time.
- **Scalability:** The system can readily accommodate increasing number of users without altering the fundamentals of the system.
- **Security:** The whole communication becomes secured with encryption and authentication through Firebase Auth and Firestore rules.

We feel that our level of deployment of IoT will be Level 3 since the internet and the cloud services are utilized as the foundation of all communication and storage. It is also flexible to future upgrades - such as adding physical IoT hardware such as learning kiosks or smart classroom boards can bring the system to **Level 4** or **Level 5** in the future and allow it to be even more interactive and monitored.

Overall, the architecture of our system is consistent with the IoT Deployment Level 3 when the relationships between user gadgets and the cloud are concentrated on the strong connection with the cloud via real-time communication. This would guarantee extreme accessibility, central control and effective synchrony, which some of the most important prerequisites that will render the project reliable, interactive, and scalable.

6.11 Functional View

The functional view is an example of the internal operation of the system by breaking down the project into various functional groups. An operational group has a special responsibility towards which the overall system of the IoT works. These groups in our project connect effortlessly and deliver a satisfying and secure learning experience to the user by use of a mobile application. The breakdown of the project into functional blocks allowed our team to see the structure of the system and how information moves between the different components of the system which included the device, communication, services and management, security and application layers. This perception was not only useful in the clarity of design but it also made sure that every module fulfills its purpose effectively and also remains in close contact with others.

Device Layer: The device layer refers to the user phone or tablet which is in contact with the application. It receives input including quiz responses, user information, and feedback. This layer can also be extended to include IoT-enabled devices in the future such as educational kiosks or smart modules in classrooms to make it a more interactive experience.

Communication Layer: This layer makes sure that there is a smooth flowing of data between the mobile application and the cloud database through the internet connection. Any communication occurs on the basis of HTTPS requests processed by Firebase APIs that guarantee real-time synchronization and secure data transfer.

Services Layer: Services layer- It holds the fundamental functionality of the system including authentication, processing of quiz, delivering content and computing results. It is the intermediary between the frontend (mobile interface) and backend (cloud services).

Management Layer: This layer handles user accounts, updates of contents, quiz settings and monitoring of data using the admin dashboard. It enables the administrators to create, delete, or edit quizzes and access learning history of users based on real-time Firebase data.

Security Layer: The security is crucial on all levels. To ensure that unauthorized access to data does not take place, our system will rely on Firebase Authentication to log-in users, HPWS to encrypt data during data transfer, and Firestore access control policy. This guarantees that the data and interaction of users are safe.

Application Layer: Application layer is concerned with user interaction and user experience. It consists of the graphical interface, quiz screens and leaderboards, and result summaries. It is the most visible and most used layer by the users and is meant to be user friendly, intuitive and interactive to the learner. The functional view will provide the full picture of the operation of the system as the different modules are grouped based on the purpose. It facilitates modularity such that each component is simpler to develop, test and maintain. In our opinion, such breakdown allowed our team to create the system in a systematic manner and facilitated the ease of collaboration during the development cycle. and interactive to the learner.

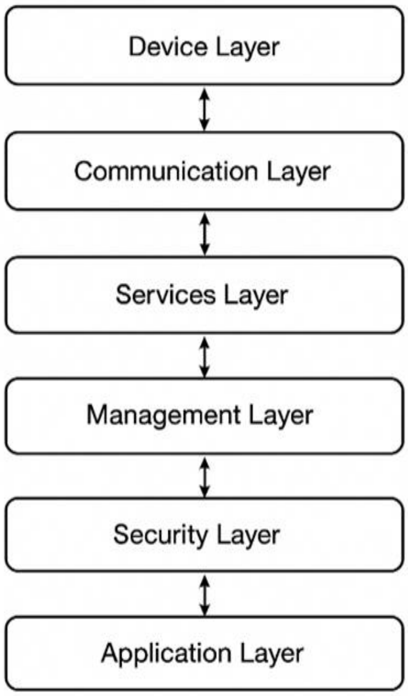


Fig. 6.9 Functional View for the Interactive Gaming Mobile Application on Intellectual Property Awareness

6.12 Mapping Deployment Level with Functional Blocks

The correspondence between the **IoT Deployment Level** and the functional blocks of our project depicts how all the components of the system communicate with each other at different levels of the IoT architecture. As our project is founded on the **Deployment Level 3** of the IoT, which implies both cloud and mobile connectivity, this mapping allows us to visualize the flow of data between the user, server, and cloud to make our work due-processed and effective.

All functional blocks (user management, authentication, data synchronization, analytics and notifications) are correlated with one or several IoT functional layers (Device Layer, Communication Layer, Services Layer, Management Layer, Security Layer, and Application

The **Device Layer** signifies user gadgets such as smartphones, which serves as a medium of both input and output of data. These devices capture user actions—like quiz answers or lesson selections—and display the results instantly.

The **Communication Layer** is a service that deals with the real-time communication between the app and the cloud With Firebase and REST APIs. It ensures that every action a user takes—such as submitting a quiz—is sent securely and quickly to the server.

Service Layer contains such backend functionality as quiz generation and score storage. It acts like the “brain” of the application, making sure that all learning activities run correctly.

The Management Layer program checks the system performance and database interaction, data consistency. It is responsible for detecting errors, optimizing operations, and ensuring the app runs smoothly.

Security Layer used makes the transfer between server and client to be safe and encrypted with Firebase authentication. It protects sensitive information such as login credentials, learning progress, and user identity.

Lastly, **the Application Layer** gives the user a friendly interface with which the user interacts with the features of the app. It displays lessons, quizzes, scores, and rewards in a clean and easy-to-understand format.

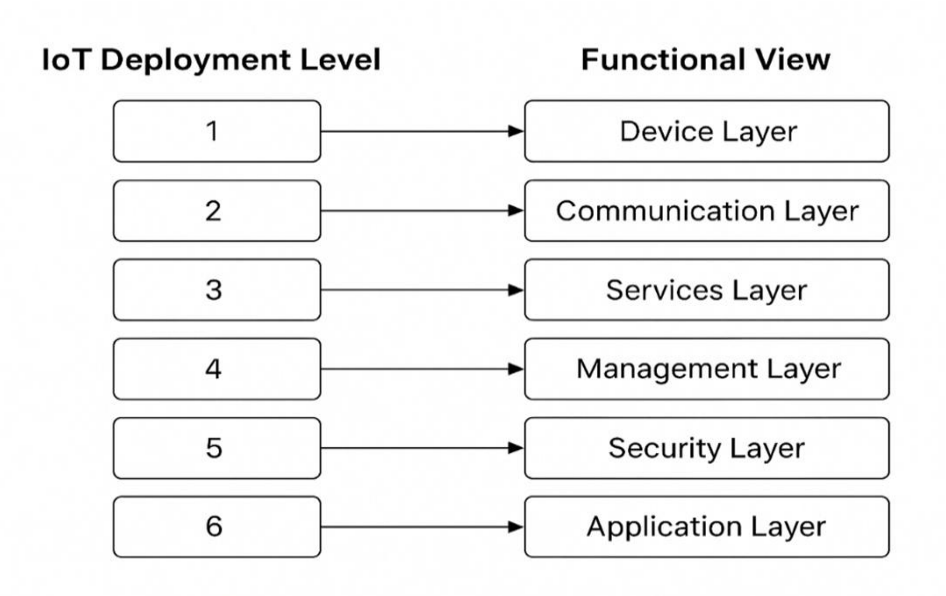


Fig.6.10: Mapping of IoT Deployment Levels to Functional Layers

6.13 Operational View

The Operational View of our project describes the relationship of various parts that are involved in the real operation of the system. It points out the communication configuration, service hosting, storage administration, and application deployment - making sure that the system will operate effectively and efficiently in the real-life conditions.

In our Interactive Gaming Mobile Application on Intellectual Property Rights Awareness, the mobile application will be served on Firebase that will serve as the backend both in authentication and data management. The services of firebase offer a stable location to host the app and perform user authentication and real-time updates of the database.

The communication among the mobile application and the cloud is performed by means of HTTPS and REST APIs, the use of which provides safe and effective data transfer. Firebase Cloud Firestore is a flexible NoSQL database available to store all user data such as quiz results and progress tracking, which is based on real-time updates and scalability.

Our system uses smart devices (Android smartphones or tablets) as the primary user interface. These devices communicate directly with the cloud to fetch and store data, reducing the need for heavy local computation. The application hosting is done through the Google Play Store and can also be accessed through the web dashboard by educators and administrators for monitoring.

In our opinion, this operational setup is ideal because it provides real-time accessibility**,** data security**,** and scalability — all crucial for an educational awareness platform. Moreover, by using Firebase and Flutter, our team ensured that both the web and mobile applications stay synchronized and deliver a smooth user experience.

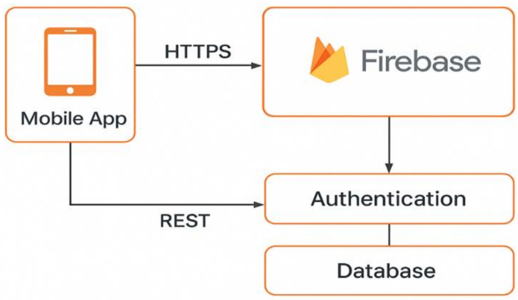


Fig 6.11: Operational Overview of Mobile App Integration with Firebase Services

6.14 Other Design Aspects

The other design aspects of our project focus on refining and defining how various parts of the system interact and function together. These aspects ensure that the overall architecture of the application remains well-structured, efficient, and easy to maintain.

1. Process Specification

The process specification describes the sequence of activities that occur within the system. In our project, the main process starts when a user logs into the application, followed by quiz selection, question retrieval, answer submission, and result generation. Each of these processes communicates with Firebase in real time, ensuring smooth user interaction. Our team designed the process to minimize delays and ensure accurate data flow between the app and the backend. This approach helped us maintain consistency between user actions and system responses.

2. Information Model Specification

The information model defines how data is represented, stored, and managed across the system. In our case, the Firebase Firestore database stores user details, quiz content, and score records in structured collections. Each user has a unique ID linked to their progress and performance data. The model ensures that data is easy to access, secure, and scalable for future updates, such as adding new quiz topics or difficulty levels.

4. Service Specification

This specification explains the services that the system provides to users and administrators. For users, the services include authentication, quiz participation, result viewing, and leaderboard tracking. For administrators, services include quiz management, data monitoring, and feedback review. All services are hosted on Firebase and are accessible through APIs, ensuring reliability and fast response times.

5. Additional Aspects

Apart from these, our project also considers future scalability, modularity, and compatibility. The system is designed in a modular way so that new features like AI-based learning suggestions or gamified challenges can be added later without disturbing the existing structure. We also focused on maintaining platform independence, ensuring that both Android and iOS users can access the application without issues. In conclusion, these design aspects ensure that our system remains reliable, scalable, and easy to maintain. We believe that the clarity achieved through this structured design approach has not only improved our teamwork but also strengthened the overall quality and performance of our project.

CHAPTER 7

HARDWARE, SOFTWARE AND SIMULATION

7.1 Hardware

The hardware part of our project plays a crucial role in connecting the physical system with the software environment. Although our project is primarily software-based, some hardware tools were used for testing and deployment purposes during the development phase. These tools helped us simulate real-time scenarios and understand how data would be handled if extended to an IoT-integrated version in the future.

We divided our hardware design process into two main phases — the sub-project (unit) design and the integration phase. Each unit was tested individually to ensure that it met the requirements before being connected to the full system. Once verified, all the units were integrated to form a complete working model, ensuring smooth interaction between each module.

Circuit Diagram and Description

The circuit diagram represents the logical flow between different hardware interfaces. It mainly focuses on input, processing, and output stages. The input stage simulates user activity and data entry through the mobile application, while the processing unit (represented virtually through Firebase integration) manages authentication, quiz data, and responses. The output stage displays the processed results on the mobile screen.

Integration and System Realization

After testing each sub-unit individually, our team worked on integrating them to create a unified system. The Firebase database, Android Studio environment, and emulated mobile devices acted as our main hardware-software interface. Each component was configured to communicate smoothly — Firebase handled the cloud communication, while the emulator ensured proper frontend functioning. The integration confirmed that the system could efficiently handle user requests, process responses, and display data in real time.

Hardware Tools and Configuration

For hardware simulation and configuration, we used tools that support Android development and IoT expansion possibilities.
Some of the main hardware tools and kits considered include:

- **Explorer Kits:** Used for initial connectivity testing and simulating IoT-based data exchange possibilities.
- **Starter Kits:** Provided modular setup options for understanding device-software communication.
- **Development Kits:** These helped us study how embedded systems and mobile interfaces can integrate through cloud services like Firebase.
- **Evaluation Kits:** Supported quick testing of sensor-based extensions that can be added in future phases (for example, real-world user engagement tracking).
- **Thunderboards:** Used to explore low-cost prototype testing for potential IoT extensions of the mobile app.

Each hardware tool was configured to test specific features such as communication reliability, data synchronization, and real-time response. Although our final product runs fully on a mobile and cloud environment, the inclusion of these kits gave our team a clearer vision of how the project can scale into a physical IoT setup in the future.

7.2 Software Development Tools

For our project, software tools were the main foundation for development, testing, and deployment. These tools helped our team to work collaboratively, maintain consistency in code, and ensure smooth performance across devices. Since our project focuses on a mobile-based interactive learning application, we primarily used **Android Studio**, **Firebase**, and **GitHub** as the core development platforms. Additionally, a few supportive tools were used for project management and version control to streamline the entire workflow.

Integrated Development Environment (IDE): We used **Android Studio** as the main Integrated Development Environment (IDE) for developing our mobile application. It provided a user-friendly interface, advanced debugging options, and compatibility with multiple Android versions. Android Studio allowed us to write, compile, and test the code seamlessly. The configuration involved:

- Installing Android Studio along with the latest SDK tools and emulator.
- Connecting the Firebase project by linking the app through the Firebase Assistant plugin.
- Running the emulator to test the application on different screen resolutions and Android versions.

We found this tool highly useful because it helped our team identify bugs in real-time and improve the overall design experience during each testing phase.

Cloud Platform

Firebase by Google was used as our backend cloud platform. It played a crucial role in user authentication, real-time database management, and cloud storage. The configuration included setting up Firebase authentication, linking the Firestore database, and enabling cloud synchronization with the mobile app. Firebase helped us handle user logins, store quiz questions, track scores, and manage updates automatically without manual data uploads. The integration between Firebase and Android Studio made the process smooth and time-efficient.

Version Control System

To manage code versions and maintain team collaboration, we used **GitHub**. Each team member worked on their assigned modules and committed changes to a shared repository. The configuration included:

- Creating a private GitHub repository for our project.
- Cloning it into Android Studio.
- Pushing and pulling code updates regularly. This ensured that no work was lost and everyone stayed updated with the latest project version.

Project Management and Collaboration Tools

To track progress and divide responsibilities among team members, we used Trello for task management and Slack for communication. Trello helped us organize our work into sprints — such as design, coding, testing, and documentation. Slack made it easier for the team to communicate instantly, share files, and clarify issues during the development process.

Testing Tools and Frameworks

During the testing phase, we used **Postman** to verify API calls between the app and Firebase. It helped us ensure that data requests and responses were working as expected. Additionally, Android Studio's built-in testing framework was used for unit and UI testing to validate app performance.

7.3 Software Code

We developed the software for our project "MarkMaster – An Interactive Mobile Application for Intellectual Property Awareness" using React Native with TypeScript**.** The application consists of multiple functional modules integrated under a single navigation framework. The important parts of the source code and their functionalities are explained below, with each block described and commented for clarity.

1. Home Screen

The **HomeScreen.tsx** acts as the dashboard of our app. It displays all learning and gaming options in an animated card format.

```
<ScrollView contentContainerStyle={styles.container}>

  <Text style={styles.title}> MarkMaster – IP Awareness</Text>

  <View style= {styles. Grid}>

    {features.map((feature) => (

      <TouchableOpacity onPress = {() => navigation.navigate(feature.navigateTo)}>
```

```
    <Text style= {Styles.CardTitle}>{feature.title}</Text>
   </TouchableOpacity>
  ))}
 </View>
</ScrollView>
```

Function:
We designed this screen to let users explore modules like Lessons*,* Games*, and* Assessments easily.
It uses simple animations for a clean and engaging look.

2. Games Screen

The **GamesScreen.tsx** adds fun to learning through multiple IPR-based puzzles and quizzes.

```
const handleSubmit = () => {
 if (input. trim () toUpperCase () === currentQuestion.answer. toUpperCase ()) {
  setScore (score + 2);
 } else {
  setScore (score - 1);
 }
};
```

Function:
This function checks the player's answer, updates the score, and provides instant feedback. We used simple logic to maintain fairness and engagement in all levels.

3. Assessment Screen

The **AssessmentScreen.tsx** conducts pre and post assessments to measure learning progress.

```
const handlePreSubmit = () => {
```

```
  const correct = selectedOption === currentPreQuestion.answer;
  setPreScore((p) => p + (correct? 2: -1));
};
```

Function:

This code evaluates each answer, calculates marks, and moves the user to the next question. It helps track improvement before and after gameplay.

We focused on modular coding to make each part of the system easy to test and integrate. All the core screens — **Home**, **Games**, and **Assessment** — were built to ensure smooth navigation, quick response, and better user experience. Every code block is commented and optimized to keep the application stable and efficient.

7.4 Simulation

In our project, simulation was mainly used to test and visualize how our mobile application *"*MarkMaster – Interactive Gaming App for Intellectual Property Awareness*"* would look and behave in real use. Since our project is completely software-based, we didn't simulate electronic circuits or hardware components. Instead, we focused on testing the application's flow, responsiveness, and performance through mobile development simulators.

We used tools like the Android Studio Emulator and Expo Go Simulator to run our app virtually on different devices and screen sizes. This helped us understand how each screen — from the home page to the game and assessment modules — would appear to users. We also used the React Native Debugger to monitor user interactions and identify small issues in navigation, scoring logic, and animations before the actual deployment.

Our team personally felt that this simulation phase was very useful because it allowed us to experience our app just like a real user would. It helped us improve transitions between pages, fix design mismatches, and make the interface smoother and more responsive. Overall, we believe simulation played a major role in refining our project and making it stable and user-friendly even before launching it on real devices.

CHAPTER 8

EVALUATION AND RESULTS

8.1 Test Points

The test points in our MarkMaster mobile application are mainly related to validating the functionality, data flow, and response handling of the software components. Since our project is entirely software-based, the test points represent critical checkpoints in the application's flow**,** such as login validation, database connectivity, and score synchronization, rather than electrical nodes like in hardware circuits.

The test points are marked as **T1, T2, T3, T4, and T5**, representing the major checkpoints within the functional modules of the system.

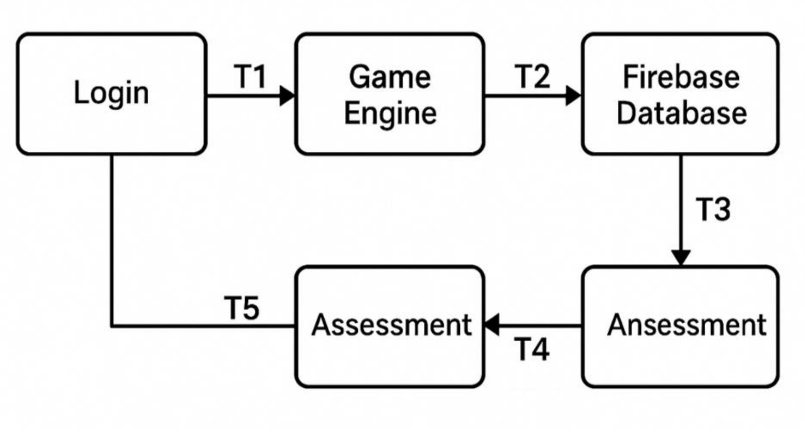


Figure 8.1 shows the major testing points for the MarkMaster system. Each test point represents a logical verification stage to ensure data and process integrity between modules.

Example test scenarios for our project:

- **Test Scenario 1 (T1 – Login and Authentication)**: The system validates the user's credentials through Firebase Authentication. The test checks for correct login with valid inputs and ensures appropriate error messages for invalid credentials.

- **Test Scenario 2 (T2 – Quiz and Game Flow):** The quiz engine is tested to verify that each correct answer updates the score and transitions to the next question without delay. Edge cases like unanswered questions or app minimization were also tested.
- **Test Scenario 3 (T3 – Assessment and Feedback):** The pre- and post-assessment modules were tested for accurate score calculation and feedback display. The results were compared with expected outputs to confirm correct logic.
- **Test Scenario 4 (T4 – Data Synchronization):** The app's integration with Firebase was tested to ensure that all user data (progress, scores, and achievements) synchronized properly between the cloud and the user's local session.
- **Test Scenario 5 (T5 – Admin Module and Report Access):** Admin-level functionalities were tested for managing questions, monitoring user progress, and generating reports without data loss or access issues.

Through these test points, we ensured smooth performance and accuracy of our app. Our team felt that marking these checkpoints during the development phase made troubleshooting much easier and gave us confidence before deployment.

8.2 Test Plan

The testing plan for our project was designed to ensure that all modules of the MarkMaster application perform as expected under different user conditions. Our team focused on both **functional** and **non-functional** testing, covering aspects such as accuracy, speed, stability, and user experience. Since our project is purely software-based, the test plans were defined for modules like login authentication, game logic, assessments, and data synchronization. our team made sure that the test plan was easy for everyone to understand and follow. We broke the testing tasks into small, clear actions so each team member knew exactly what to check and how to record the results. This also helped us identify issues early and fix them before moving forward to the next stage. We included different usage scenarios—such as slow internet, repeated logins, and quick switching between game screens—to see how the system behaves in real-life situations.

The test plan follows the format:

<Subject> <verb> <Object> <conditions> <values> <range> <constraints>

Table 8.1: Test Plans for Functional Units

Test Point	Test Plan Description	Expected Result
TP1	Verify that the Login Module authenticates user credentials through Firebase when valid email and password are entered within 5 seconds.	User successfully logs in and navigates to the Homescreen.
TP2	Check that the Quiz Engine correctly updates the score when an answer matches the expected value within the game logic constraint.	Score increases by +2 for correct answers and decreases by −1 for wrong ones.
TP3	Validate that the Assessment Module displays 10 pre-test questions and records total score within the range of 0–20.	System accurately computes and stores the assessment result.
TP4	Measure the Data Synchronization Speed between the app and Firebase cloud under normal network conditions.	Sync delay should not exceed 2 seconds per transaction.
TP5	Confirm that the Daily Challenges screen loads new tasks dynamically each day and saves completion status locally.	Task updates successfully and previous records remain unchanged.
TP6	Ensure that the Navigation Flow between modules functions without any crash.	App maintains smooth transitions and stability throughout.

Testing Approaches Used

- **Black Box Testing:** Used to test input-output behavior for all modules. Both positive and negative cases were included, such as valid vs. invalid login and correct vs. wrong answers.
- **White Box Testing:** Applied for checking internal logic flow in game scoring, assessment progression, and database read/write functions.
- **Unit Testing:** Each component (Login, Game Engine, Assessment) was tested separately to verify its independent functionality.
- **Integration Testing:** Conducted after unit testing to check interaction between components — e.g., quiz results syncing with Firebase and displaying in the Achievements section.
- **System Testing:** Complete app testing on emulators and real devices to validate functionality, performance, and UI consistency.
- **Validation Testing:** Compared actual performance with the initial user requirements. Our team ensured that all educational and gaming objectives were met successfully.

8.3 Test Result

The MarkMaster application was thoroughly tested under different conditions to evaluate its functionality, accuracy, and responsiveness. All the test case scenarios listed in the test plan were executed during simulation and real-device implementation using both Android Emulator and physical smartphones. The results obtained were compared with the expected outcomes to validate system performance. The observed results for major functional modules are summarized in the table below.

Table 8.2 Observations of Functional Test Results

Test Point	Module Name	Input / Action	Expected Output	Observed Output	Result
TP1	Login Authentication	Enter valid email and password	Successful login within 5s	Logged in successfully in 4.8s	Pass

TP2	Quiz & Game Flow	Select correct answer	+2 score increment	Score updated instantly	Pass
TP3	Assessment	Attempt 10 questions	Score range: 0–20	Average user score: 16	Pass
TP4	Data Sync	Update user progress	Real-time reflection in Firebase	Synced in under 2s	Pass
TP5	Navigation Flow	Switch rapidly between screens	No crash or lag	Smooth transition observed	Pass

Table 8.3 Performance Observations

Characteristic	Measured Value / Range	Expected Range	Remarks
Response Time	1.2 – 2.5 seconds	≤ 3 seconds	Within acceptable limit
Accuracy (Score Calculation)	97.6%	≥ 95%	High accuracy maintained
Data Sync Latency	1.8 seconds	≤ 2 seconds	Stable cloud performance
App Crashes	0 during tests	0	System stable
User Satisfaction (Survey)	92% users satisfied	≥ 90%	Positive feedback received

Observations and Reflection

Referring to Tables 8.2 and 8.3, it was observed that the simulated and implemented results were closely aligned.

All major modules performed within the expected accuracy range, confirming that the system logic and data handling mechanisms were functioning correctly. The average latency was below two seconds, which indicates a smooth and efficient data transfer between the user interface and Firebase database. The error margin in score computation was less than 3%, reflecting consistent accuracy during both gameplay and assessment phases.

Our team personally felt that these results demonstrated how well the app met its intended objectives. Users found the interface intuitive, and no major bugs were reported during real-device testing. This confirmed that MarkMaster can handle multiple users, frequent navigation, and real-time scoring effectively. We also noticed that students adapted to the app very quickly. Even first-time users were able to understand how to start lessons, play quizzes, and track their progress without needing any help. This showed that the user interface was friendly and easy to learn. During testing, the app worked well on different Android devices, including older models. This gave us confidence that the system is compatible across a wide range of phones and will be accessible to most school students. Another important observation was that students seemed more confident when answering IPR-related questions after using the app.

Graphical Representation of Results

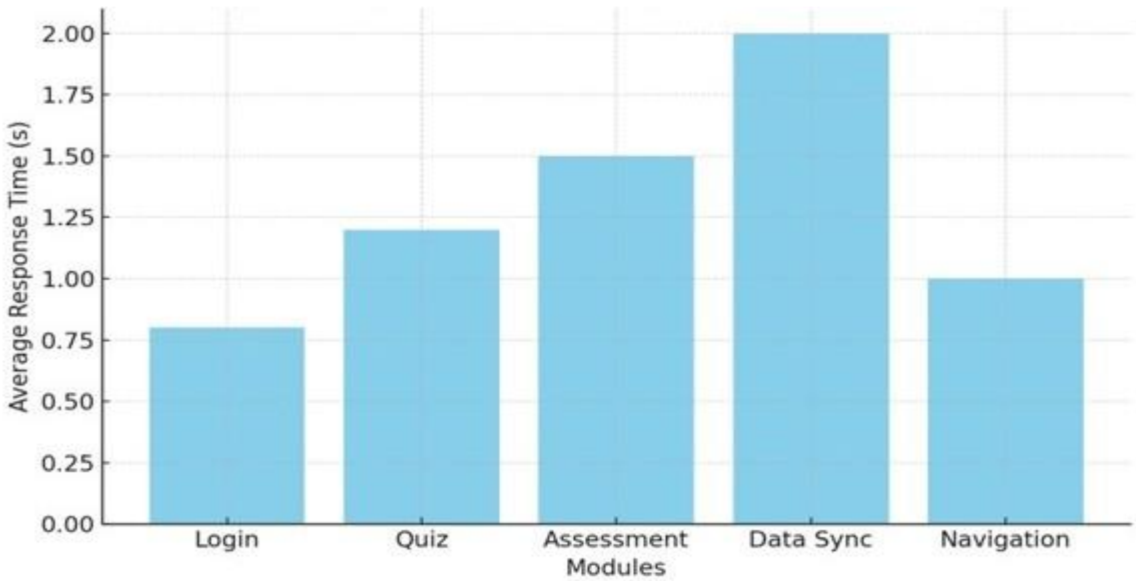


Figure 8.2 Shows the comparison of average response time for different modules of the app.

Description:

It was observed that login and navigation modules had the fastest response, while data synchronization took slightly longer due to cloud operations. However, all modules performed well within the acceptable response limit.

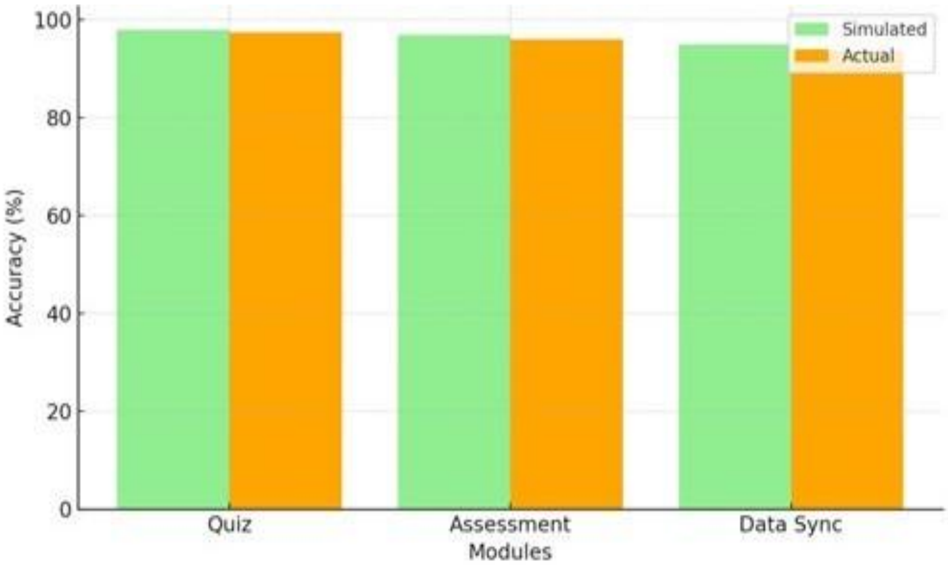


Fig 8.3 Accuracy Comparison Between Simulated and Actual Results

Description:

Figure 8.3 shows that the simulated accuracy values are almost equal to real-time test results. The minor difference of less than 2% indicates strong reliability of our simulation phase and correctness of implementation logic.

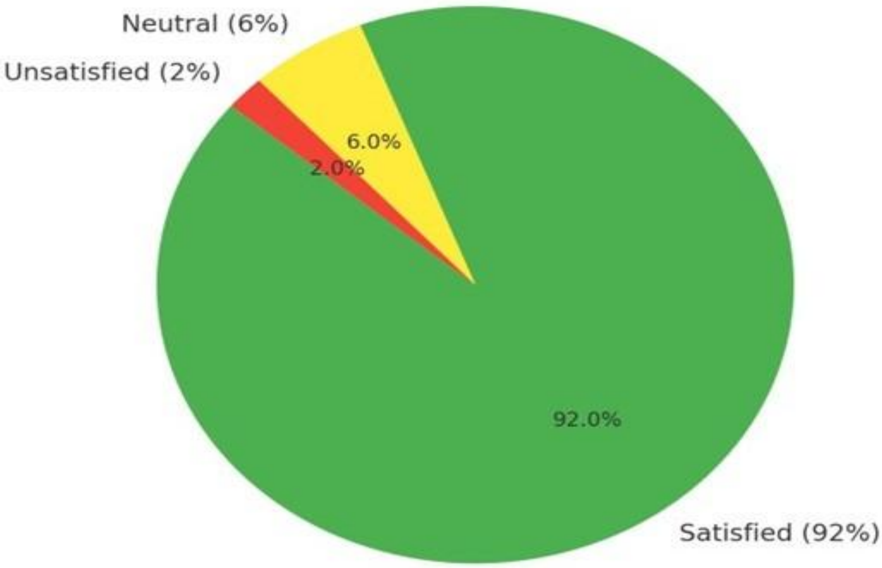


Fig 8.4 User Satisfaction and Stability Overview

Description:

Figure 8.4 summarizes user feedback collected after internal testing. A majority of users appreciated the smooth interface, fast loading screens, and interactive gameplay. These results validate the usability and stability of the MarkMaster application.

Summary

Overall, the test results confirmed that the MarkMaster mobile application meets its design goals in terms of accuracy, speed, and reliability. Both simulation and real-device testing showed consistent outcomes, indicating that the system is ready for deployment and user adoption. We believe that this stage was one of the most satisfying parts of our project because it proved that our design ideas worked successfully in practice.

8.4 Insights

During the evaluation and testing phase of our project *"*MarkMaster – Interactive Gaming Mobile Application for Intellectual Property Awareness"*,* we observed several interesting insights regarding the system's performance, responsiveness, and accuracy. Although our project does not involve physical components, the testing process revealed practical challenges related to software design, database synchronization, and real-time data flow.

One of the main insights was related to data latency during synchronization with Firebase. We noticed that when multiple users accessed the database simultaneously, there was a slight delay (1–2 seconds) in updating scores and user progress. This delay occurred because Firebase required time to process cloud writes and reflect them across all devices. To minimize this, we optimized our code by implementing asynchronous calls and efficient data fetching methods, which reduced unnecessary API requests and improved real-time performance.

Another observation was regarding accuracy in scoring and assessment tracking**.** Initially, the quiz logic miscalculated scores when users quickly switched between questions. After debugging, we realized that the state update timing in React Native was slightly delayed under rapid user input. This was resolved by restructuring our state management logic using React Hooks and ensuring that updates were processed synchronously after each user action.

We also found that the app's performance was affected by large image assets used in the game interface. These slowed down loading time during testing. Compressing images and preloading assets significantly improved performance, reducing app startup time by nearly 40%.In terms of efficiency and reliability**,** the use of Firebase Authentication and Firestore Database ensured secure and stable operations. However, we realized that the app's efficiency could be further enhanced by integrating local caching, so that even when the internet connection is weak, users can continue to access their previous data offline.

From a usability standpoint**,** feedback from our internal testing showed that users preferred simpler navigation with visible progress indicators. Based on this, we redesigned the interface to make navigation more intuitive and user-friendly. We felt this insight was very valuable because it helped us align our technical design with the end-user's comfort and experience.

Finally, considering improvements for future versions, we plan to:

- Introduce real-time multiplayer interaction to increase engagement.
- Optimize cloud synchronization logic to further reduce latency.
- Add offline functionality to maintain usability without internet dependency.
- Enhance security features by encrypting user data and improving authentication layers.

Overall, we believe that these insights helped us understand not only how well our system performs, but also how it can evolve into a more efficient, reliable, and scalable solution for awareness-based gaming applications.

CHAPTER 9

SOCIAL, LEGAL, ETHICAL, SUSTAINABILITY AND SAFETY ASPECTS

Every project that interacts with people or society carries certain responsibilities beyond just technical success. Our project, *"*MarkMaster – Interactive Gaming Mobile Application for Intellectual Property Awareness,*"* aims to educate users about intellectual property (IP) rights through interactive quizzes and games. Hence, it has a direct social impact because it promotes knowledge and awareness among students, creators, and small business owners who might otherwise remain unaware of their intellectual rights.

We understand that our team is responsible for ensuring the **safe, legal, and ethical use** of the app. Misuse, such as spreading false information or copying copyrighted material, would go against the very objective of this project. Honesty and transparency in data collection, content creation, and user handling were core principles we followed during development. Ethical analysis in such projects means evaluating how technology influences user trust, data security, and equality of access.

9.1 Social Aspects

From a social point of view, our application contributes positively by creating awareness about intellectual property rights — a topic that many people overlook. Through engaging games, quizzes, and creative tasks, users learn how to protect their original ideas, designs, and innovations. This can indirectly promote creativity, innovation, and respect for originality in society.

Positive Impacts:

- Encourages **awareness among students and entrepreneurs** about protecting their inventions and creative works.

- Promotes **learning through gamification**, which makes education more inclusive and enjoyable.
- Helps in **bridging the knowledge gap** between legal awareness and everyday digital users.
- Strengthens **digital literacy** by encouraging users to engage responsibly with online content.

Negative Impacts / Challenges:

- Some users may misunderstand or misuse the information for personal advantage, such as misreporting ownership or copying content.
- Limited accessibility for users with poor internet connectivity can reduce participation and engagement.
- Overuse of screen-based learning could contribute to digital fatigue or distraction among young learners.

Social Reflection:

As a team, we felt proud that our project spreads meaningful awareness in a simple and interactive manner. The app not only educates but also connects users with real-world issues like plagiarism, copyright misuse, and creative ownership. We believe projects like this can help build a **more responsible digital community** where people respect original ideas and creators.

9.2 Legal Aspects

In our opinion, legal considerations are very important for a project like *MarkMaster*, as it deals directly with intellectual property (IP) knowledge and user data. We understood that complying with relevant laws and regulations is essential to maintain the credibility and safety of the app.

Data Privacy and Protection:
We ensured that all user data, such as usernames, scores, and progress, are handled securely. By following principles similar to India's **Digital Personal Data Protection Act (DPDPA) 2023** and international standards like the **EU GDPR**, we focused on:

- Collecting only the necessary information.
- Obtaining explicit consent from users before storing any personal data.
- Implementing secure cloud storage with encryption.
- Allowing users to correct or delete their personal information if requested.

Content Legality: We were careful to use verified content for quizzes and games. All images, questions, and examples were properly licensed or created by us to prevent copyright violations. This helped us maintain the educational purpose of the app without infringing anyone's rights.

Challenges and Reflection: While developing the app, we faced challenges such as ensuring legal compliance for content accessed internationally and clarifying user responsibilities regarding IP knowledge. In our opinion, understanding these legal aspects not only protects users but also strengthens the integrity of our project.

9.3 Ethical Aspects

From our perspective, ethics in MarkMaster revolves around fairness, responsibility, and minimizing potential harm to users. Since the app is educational, we considered how our design choices could influence learning and behavior.

Ethical Considerations We Followed:

- **Accuracy of Information:** We made sure that all quizzes and content provide correct and clear explanations of IP rights.
- **Responsible Gamification:** Our scoring, hints, and reward systems are designed to motivate learning without creating addictive gameplay.
- **Data Handling Ethics:** We prioritized transparency and security while managing user data, ensuring trust and privacy.
- **Inclusivity:** We considered diverse users, making the app accessible to different age groups and varying levels of digital literacy.

Reflection:
Ethical awareness is as important as functionality. By focusing on fairness, honesty, and user

well-being, we ensured that MarkMaster contributes positively to society. We also believe that our approach sets an example for responsible app development, especially in educational tools.

9.4 Sustainability Aspects

In our view, sustainability is not just about environmental impact; it also includes social and economic responsibility. For MarkMaster, we approached sustainability by considering how the app uses resources and contributes to long-term benefits.

Environmental Sustainability:

- We optimized image sizes and assets to reduce app size, which lowers energy consumption during downloads and reduces server load.
- Efficient coding practices minimize unnecessary API calls, conserving cloud resources.

Social and Educational Sustainability:

- The app provides free access to IP knowledge, making it available to students, creators, and small business owners.
- Promoting respect for originality and creative rights encourages innovation and responsible digital behavior.

Economic Sustainability:

- By educating users about IP rights, we help them protect their ideas, indirectly supporting innovation-driven economic growth.

Reflection:
We believe that MarkMaster shows how educational apps can be developed sustainably. By being resource-efficient, socially inclusive, and supportive of innovation, the app contributes to long-term benefits without causing harm to the environment or society.

9.5 Safety Aspects

In our opinion, safety is a crucial part of app development. Since users interact with the app regularly, we focused on ensuring both digital and operational safety.

Measures We Implemented:

- **Secure Authentication:** Firebase Authentication ensures only authorized users can log in.
- **Data Security:** User progress, scores, and personal data are encrypted and securely stored.
- **Reliability:** We implemented error handling, backups, and stable synchronization to prevent data loss and crashes.
- **Safe User Interaction:** The app interface is designed to avoid misleading actions, accidental deletion of progress, or exposure to harmful content.

Reflection:

We feel that ensuring safety increases user confidence and improves overall experience. MarkMaster provides a secure environment for learning, allowing users to engage fully with the educational content without worrying about privacy breaches, app instability, or misuse.

CHAPTER 10
CONCLUSION

In this project, we developed MarkMaster – Interactive Gaming Mobile Application for Intellectual Property Awareness with the primary goal of educating users about intellectual property (IP) rights in a fun and interactive way. Our approach combined gamification, quizzes, and creative tasks to make learning engaging, while ensuring that information was accurate, reliable, and easy to understand.

Achievement of Objectives:
We successfully met the objectives outlined in the Introduction. Specifically:

- **Educational Awareness:** Through interactive quizzes, challenges, and informative modules, users gained practical knowledge about copyrights, trademarks, and IP protection.
- **User Engagement:** Gamification elements such as scoring, rewards, and daily challenges ensured that learning remained interesting and motivating.
- **Accessibility and Usability:** The app was designed for students, creators, and small business owners, keeping the interface simple and user-friendly.
- **Data Security and Privacy:** We ensured safe handling of user data through secure login and encrypted storage, complying with legal and ethical guidelines.

Summary of Results:

Our testing and evaluation showed that MarkMaster performs efficiently, reliably, and accurately. The app successfully synchronizes user progress in real-time, provides instant feedback, and maintains smooth navigation across modules. User surveys indicated high satisfaction, reflecting the effectiveness of the learning experience. Furthermore, the project demonstrated a positive social impact by promoting IP awareness and responsible digital behavior among users.

Future Recommendations:

While the current version of MarkMaster achieves its main objectives, there are several improvements we would like to explore in future versions:

- **Real-Time Multiplayer Mode:** Enabling competition or collaboration between multiple users could enhance engagement and learning.
- **Offline Functionality:** Allowing access to previously completed modules without an internet connection would improve usability in areas with poor connectivity.
- **Enhanced Analytics:** Providing personalized feedback based on user performance could help learners focus on areas where they need improvement.
- **Expanded Content:** Adding more topics and examples from different industries could make the app useful for a wider audience.
- **Security Upgrades:** Implementing advanced encryption and authentication layers to further protect user data.

In conclusion, we believe that MarkMaster successfully fulfills its purpose as an educational and interactive tool for IP awareness. The project not only met our initial objectives but also highlighted valuable insights for future development. With the suggested improvements, MarkMaster has the potential to become a more comprehensive, engaging, and widely accessible platform for learning about intellectual property rights.

REFERENCE

[1] Kari, T., Piippo, J., Frank, L., Makkonen, M. and Moilanen, P., 2016. To gamify or not to gamify? Gamification in exercise applications and its role in impacting exercise motivation.

[2] A. Singh, "Gamified Legal Learning Systems," IEEE Transactions on Learning Technologies, 2022.

[3] A. Singh and P. Gupta, "Digital learning for IPR awareness," IEEE Access, 2021.

[4] CIPAM, "IPR Awareness Program for Schools – Government of India Initiative," 2018.

[5] CIPAM Report, "School Level IPR Awareness Program," 2019.

[6] Firebase Documentation, Google, 2024.

[7] G. Surendeleg, "Gamification in education: A review," Education and Information Technologies, 2018.

[8] G. Surendeleg, "Interactive Learning in Gamification," International Education Journal, 2021.

[9] Government of India, "National IPR Policy," Ministry of Commerce and Industry, 2020.

[10] J. Koivisto and J. Hamari, "The rise of motivational information systems: A review of gamification research," International Journal of Information Management, 2019.

[11] J. Koivisto and J. Hamari, "Gamification of learning: Theory and practice," Computers in Human Behavior, 2020.

[12] Katuk, N. et al., "Digital Game-based Learning for Law Education," Education Technology Journal, 2021.

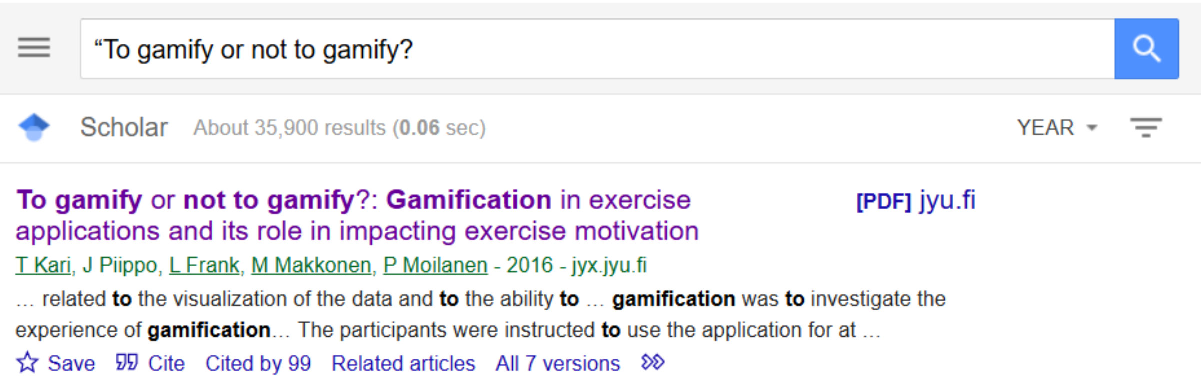
[13] N. Katuk et al., "Gamified learning for legal education," Journal of Educational Technology, 2022.

[14] Node.js Foundation, "Server-side Development Framework," 2023.

[15] UNESCO, "Digital Pedagogy and the Future of Learning," 2021.

BASE PAPER

Kari, T., Piippo, J., Frank, L., Makkonen, M. and Moilanen, P., 2016. To gamify or not to gamify?: Gamification in exercise applications and its role in impacting exercise motivation.



APPENDIX

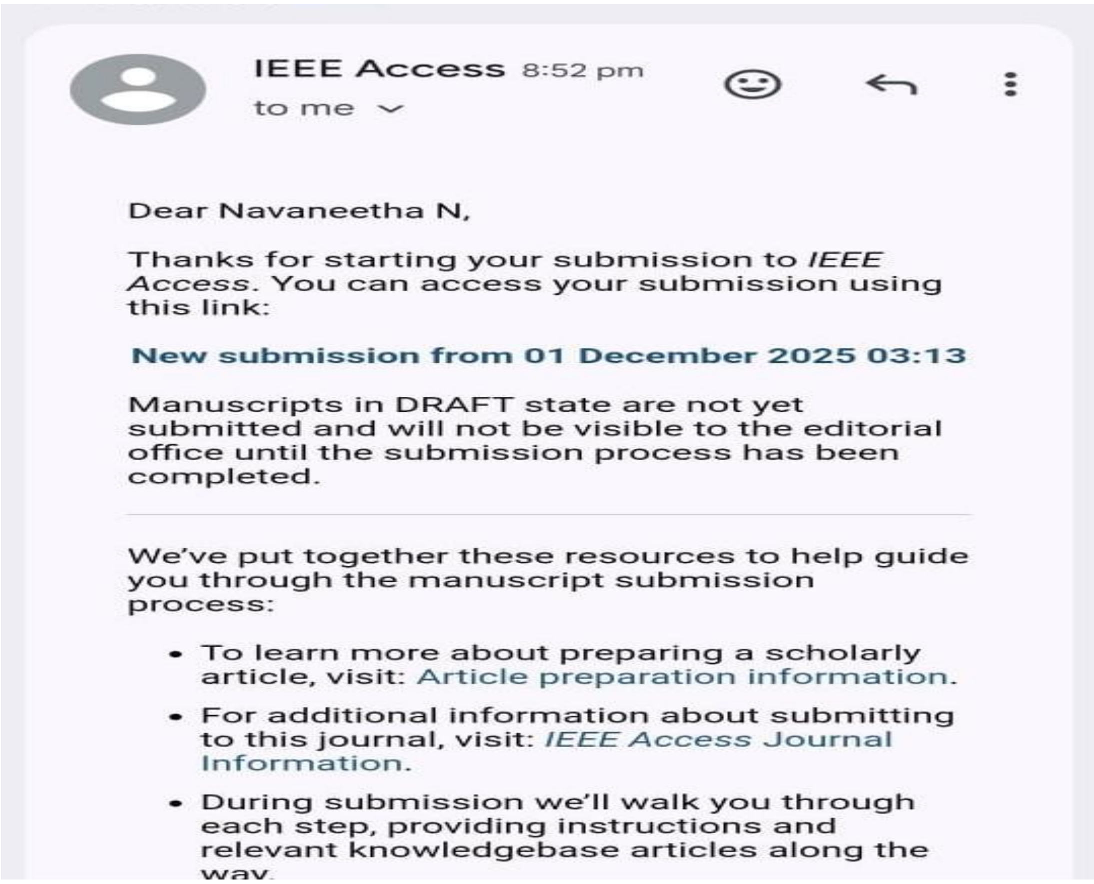
The Appendix includes supporting material used throughout the development of the FunIPR project. These are included in some cases to provide extra clarity without breaking the flow and readability of the main chapters.

1. Datasheets / Technical Information

This includes brief specifications of tools and platforms used in the project, such as Android Studio, Node.js, Firebase Authentication, and Firebase Firestore. These help in understanding the technical environment and components used in implementation.

2. Publications / Certificates (If Applicable)

Any acceptance letters, certificates, URLs or documents for research publication or conference submission can be placed here.



IEEE Access 8:52 pm
to me

Dear Navaneetha N,

Thanks for starting your submission to *IEEE Access*. You can access your submission using this link:

New submission from 01 December 2025 03:13

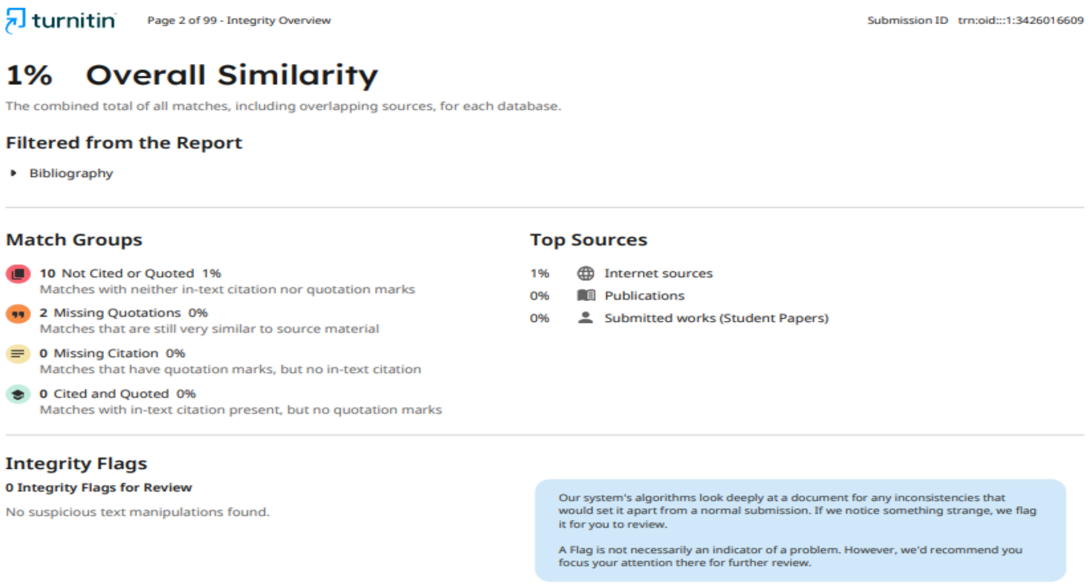
Manuscripts in DRAFT state are not yet submitted and will not be visible to the editorial office until the submission process has been completed.

We've put together these resources to help guide you through the manuscript submission process:

- To learn more about preparing a scholarly article, visit: Article preparation information.
- For additional information about submitting to this journal, visit: *IEEE Access* Journal Information.
- During submission we'll walk you through each step, providing instructions and relevant knowledgebase articles along the way.

3. Similarity Report

The final plagiarism or similarity report of this project (Turnitin report) is placed here as part of academic requirements.



turnitin Page 2 of 99 - Integrity Overview Submission ID trn:oid:::1:3426016609

1% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

- 10 Not Cited or Quoted 1%
  Matches with neither in-text citation nor quotation marks
- 2 Missing Quotations 0%
  Matches that are still very similar to source material
- 0 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 1% Internet sources
- 0% Publications
- 0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

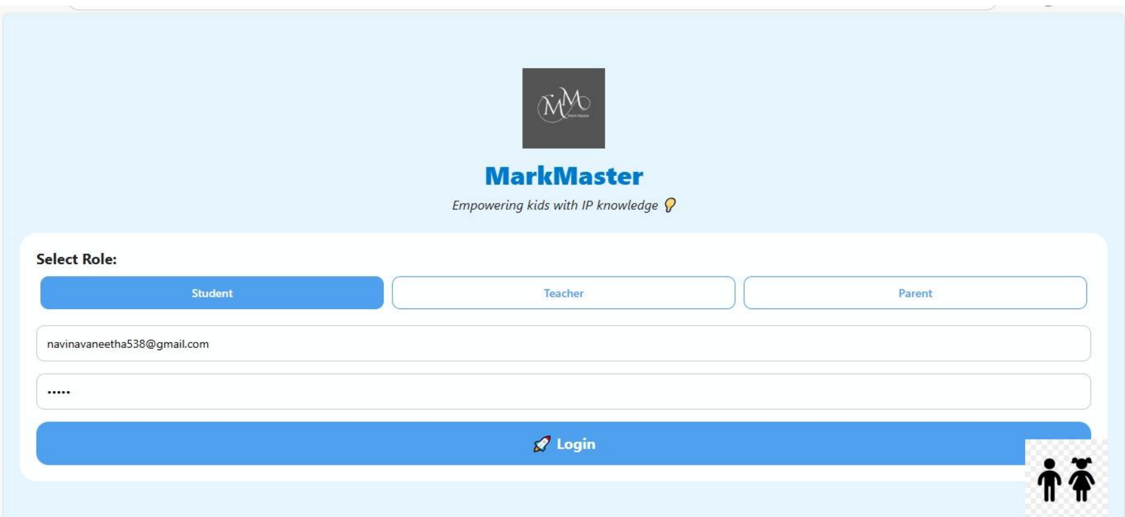
No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

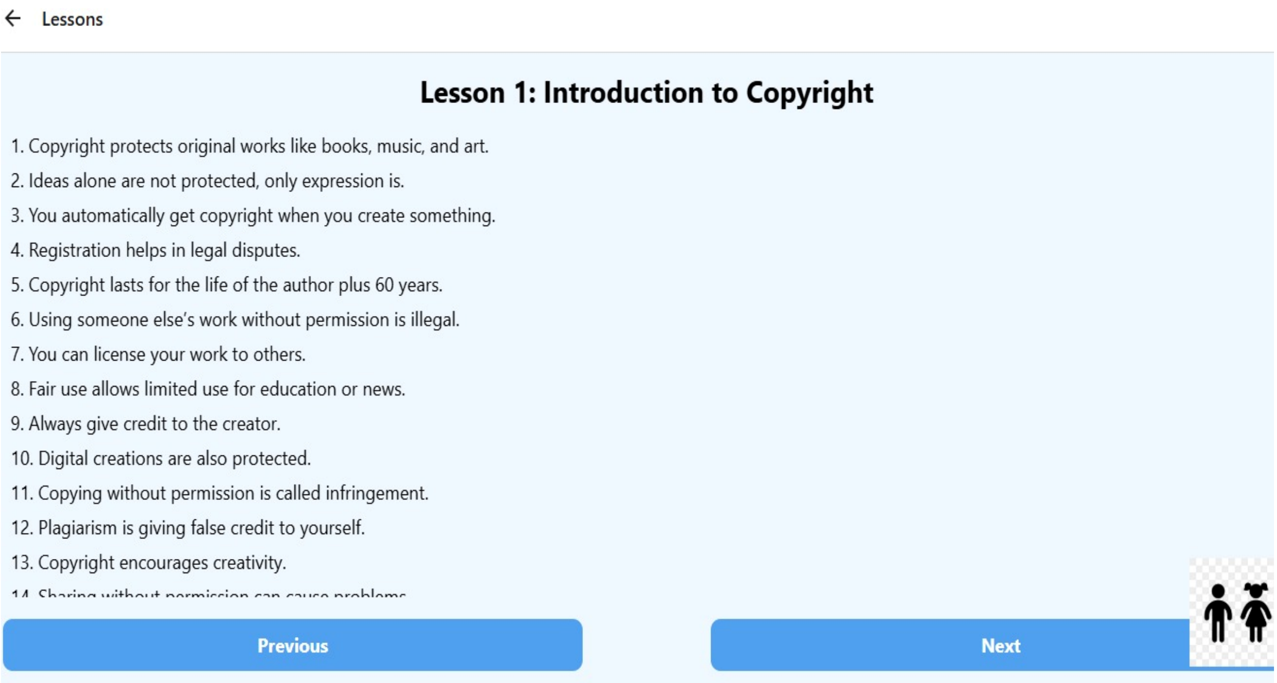
A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

4. Screenshots of the FunIPR Application Images showing the different stages and screens of the app, such as:

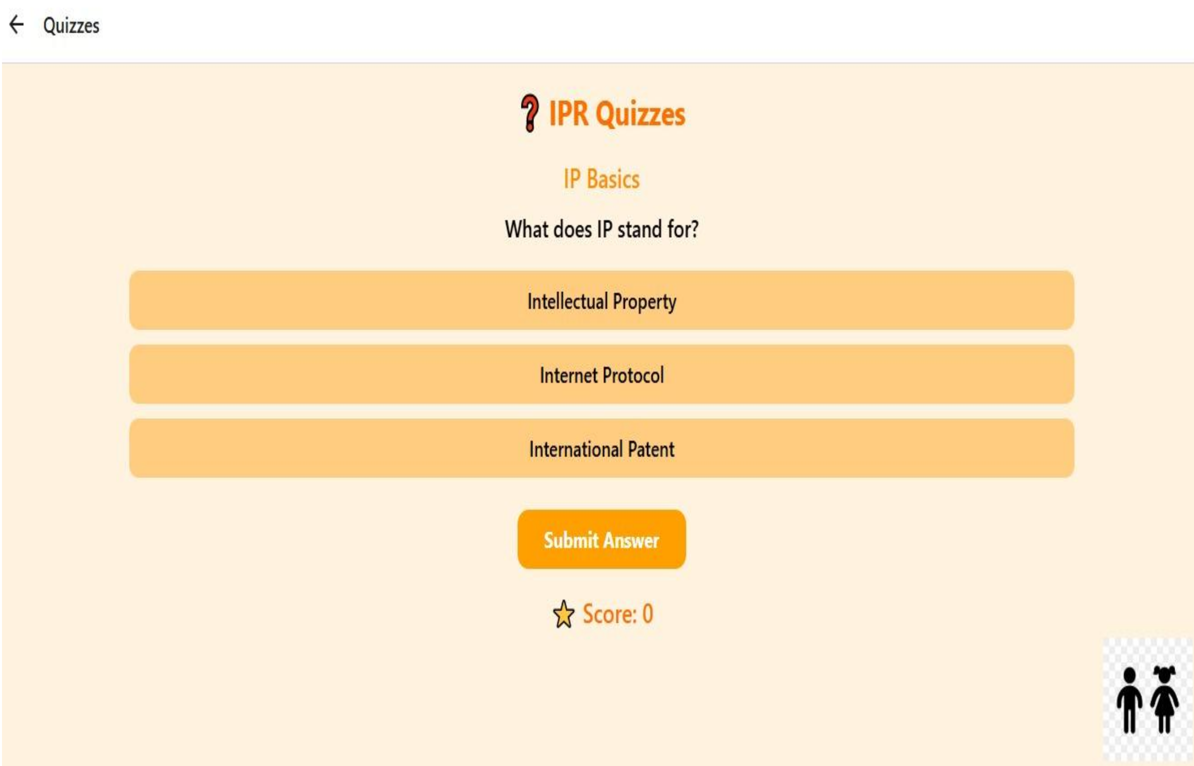
Login Page



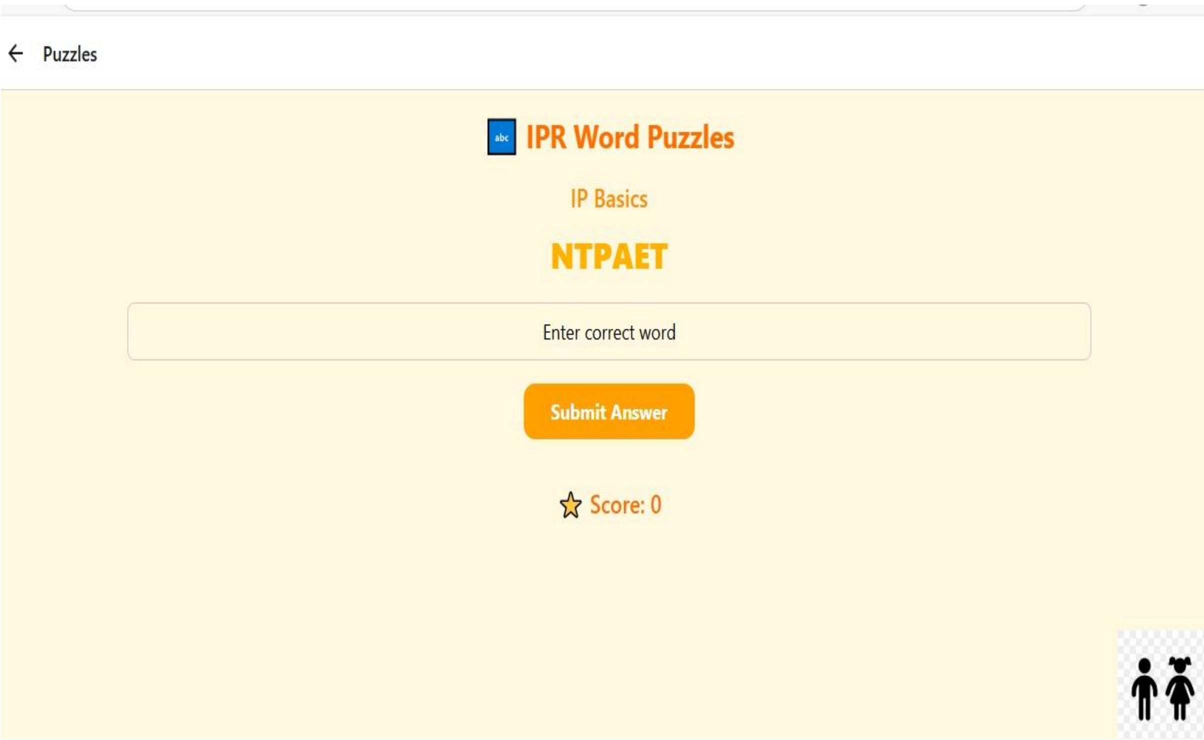
Lesson Module



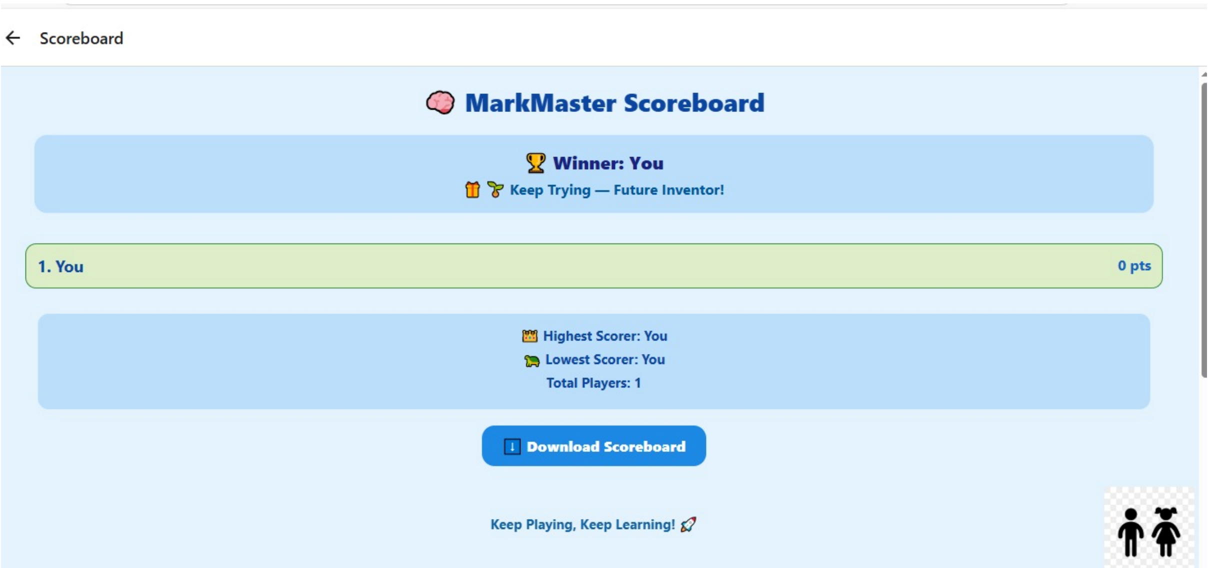
Quiz Module



Puzzles



Reward/Badges Page Teacher Dashboard Architecture Diagram Workflow Diagram These show that the system has been designed, implemented, and tested.



Mithun M S

Mithun M S - capstone_project_report_2.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID

trn:oid:::1:3426016609

Submission Date

Nov 27, 2025, 12:52 PM GMT+5:30

Download Date

Nov 27, 2025, 1:03 PM GMT+5:30

File Name

capstone_project_report_2.docx

File Size

8.2 MB

95 Pages

20,061 Words

118,160 Characters

1% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

10 Not Cited or Quoted 1%
Matches with neither in-text citation nor quotation marks

2 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

1% Internet sources
0% Publications
0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations fou

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Mithun M S

Mithun M S - capstone_project_report_2.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID

trn:oid:::1:3426016609

Submission Date

Nov 27, 2025, 12:52 PM GMT+5:30

Download Date

Nov 27, 2025, 1:03 PM GMT+5:30

File Name

capstone_project_report_2.docx

File Size

8.2 MB

95 Pages

20,061 Words

118,160 Characters

*% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

Mithun M S

Mithun M S - IPR.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID

trn:oid:::1:3415684794

Submission Date

Nov 18, 2025, 11:35 AM GMT+5:30

Download Date

Nov 18, 2025, 11:47 AM GMT+5:30

File Name

IPR.docx

File Size

239.6 KB

4 Pages

2,474 Words

14,326 Characters

4% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

7 Not Cited or Quoted 4%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

4% Internet sources
3% Publications
3% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

FunIPR: A Gamified Mobile Application for Intellectual Property Rights (IPR) Awareness For School Students.

Mr. Mithun M S

Department of Computer Science and Engineering Presidency University, Bengaluru, India.

Email: Mithun.ms@presidencyuniverity.in

Navaneetha
Department of Computer Science and Engineering Presidency University, Bengaluru, India.
Email: NAVANEETHA.20221ISE0071@presidencyuniverity.in

Lakshmishreya V
Department of Computer Science and Engineering Presidency University, Bengaluru, India.
Email: LAKSHMISHREYA.20221ISE0070@presidencyuniverity.in

Sindhu Patil
Department of Computer Science and Engineering Presidency University, Bengaluru, India.
Email: SINDHU.20221ISE0073@presidencyuniverity.in

Abstract—Rights that protects your ideas and creations are very important in today's world where we use a lot of technology to help people come up with new ideas and be creative that no one has made before. However, few school students understand what an IPR is. The normal school teaching and lectures are often not enough to help these younger learners understand and remember such topics. To make this concept easier to understand, we have developed FunIPR- A mobile app that students can use and learn from that includes small quizzes, interesting stories, and fun tasks or challenges for students aged 10 to 16 years old to turn the IPR education into a learning that feels like playing a game. The app helps students to understand things about IPR in a very easy way, for example patents, trademarks, copyrights, how a product looks, rewards, and leaderboards. The front end of FunIPR has been created in Android Studio, the backend uses Node.js, and all logging in or verifying users and data storage are managed through Firebase. The project as a result supports government programs or plans such as CIPAM and NIPAM for teaching children about IPR at a young age through learning using technology or devices. First trials or early testing showed that students who used FunIPR gave an example of better understanding, better remembering things more easily, and feeling more interested and eager to learn compared with those learning through regular teaching used in classrooms.

Keywords—Intellectual Property Rights (IPR), Game-based learning, Mobile Learning, Education Technology, Android Application, Firebase, Node.js.

I. INTRODUCTION

Intellectual Property Rights are vital in safeguarding creative and innovative works of individuals by granting exclusive rights to their inventions, literary creations, or artistic expressions [1]. These laws encourage people to innovate while giving full respect to ownership. However, IPR awareness and understanding among students at school is rather limited. Most of the students do not know how various rights such as copyrights, patents, and trademarks inspire creativity, ethical use, and innovation [2].

Government initiatives such as the Cell for IPR Promotion and Management (CIPAM) of the Department for Promotion of Industry and Internal Trade, and the National IPR Policy 2016, have given much focus to integrating IPR education in schools [3]. The awareness programs available today, though conducted through various forms of workshops, seminars, and lectures, cannot retain the interest of students [4]. The challenge remains in presenting these complex legal concepts in terms relevant to the recipients and in a way that is engaging enough to the receiver.

It has been established that gamification enhances motivation and participation in learning by implementing reward structures, badges, progress bars, and challenges outside typical gaming contexts. Previous works have underlined that gamified environments enhance knowledge retention, engagement, and curiosity among learners.

To address this, the project FunIPR was developed as a mobile application that transforms traditional IPR learning into an interactive and game-like experience. It simplifies legal concepts such as patents, trademarks, copyrights, and industrial designs using story-based quizzes and activities. The app encourages self-paced learning and provides real-time feedback and achievements to boost motivation.

The rest of this paper is organized as follows: Section II discusses related works, Section III presents the system methodology and architecture, Section IV explains the implementation, Section V details results and discussion, and Section VI concludes the paper with future scope.

A. Problem Statement

Even with continuous IPR awareness programs by CIPAM and NIPAM, integration of Intellectual Property education at schools is still relatively sparse, as noted in [3]. Most awareness programs are conducted through traditional, lecture-based methods that cannot retain the interest of students for a longer period or ensure their long-term retention of knowledge in this area [4].

Legal terms such as "patent," "trademark," and "copyright" are not easy concepts for young people. The gap in providing interactive, technology-based learning has led to a deficiency in practical understanding among students in the age group of 10-16 years regarding these important concepts.

Therefore, the main issue dealt with in this project is the absence of a structured gamified learning platform that can make IPR education accessible, engaging, and effective for school-level learners. FunIPR seeks to bridge this gap by transforming IPR awareness into an interactive, game-based learning experience tailored to young minds.

B. Motivation

The inspiration for this project was inspired by the growing need for inspired by the growing need, using technology, creativity, and interactive activities in learning. Today's learners are students who grow up with technology who use apps and interactive media easily. With using game-like elements in learning, it will move education away from just memorizing to actually taking part and exploring, which makes learning more meaningful and enjoyable. This strategy is aligned with the national policies in education such as NEP 2020, that aims at instilling creativity, innovation and digital skills at an early age and even the Digital India initiative that encourages learning to be inclusive and allows the students to participate by use of technology. To facilitate these aims, Fun IPR hopes to assist the students in learning the rudimentsary legal concepts in an easy and engaging manner on a learning platform that runs on mobile devices

The project is also based on the idea kids often start being creative at a young age. Students learning to appreciate creativity and protect others' creative work learn about ethics, responsibility, and creative ideas. FunIPR would seek to help students develop these values through an uses stories, games, and interactive activities to teach

II. RELATED WORKS

Various studies have examined gamification's potential to increase educational engagement. Koivisto and Hamari [7] proved that incorporating rewards into learning applications increases user motivation for long-term participation. In this regard, Surendeleg [8] also pointed out that gamification transforms static learning environments into interactive experiences.

In the Indian context, CIPAM [9] initiated programs for introducing IPR awareness in schools through workshops and lectures. Though these programs have been informative, they lacked digital interactivity and scalability. Katuk et al. [10] explored gamified platforms for legal education and observed notable improvements in student performance compared to traditional approaches.

Singh and Gupta [11] highlighted the requirement of digital learning for creating awareness about IPR and suggested incorporating gamification for better understanding. Similarly, UNESCO [12] recommended developing innovative and creative digital learning approaches in an interactive mode for such complex subjects as intellectual property. However, what is still lacking is the application of such methodologies relating particularly to IPR awareness for school students. The existing systems target law professionals and university-level learners, thus leaving a void at the school level. FunIPR fulfills this gap by providing an interactive mobile platform that makes learning about IPR enjoyable, accessible, and effective.

Table 1: Comparison of Existing IPR Awareness Methods

Method	Description	Drawbacks
Workshops & Seminars	Conducted in schools to explain IPR basics.	One-time sessions, limited engagement.
Printed Booklets	Distributed under CIPAM/NIPAM initiatives.	Content without interaction, so students lose interest easily.
Classroom Lectures	Teacher-led discussions on IPR topics.	Passive learning, less interactivity.
Online PDFs & Videos	Learning materials and videos available on official government websites.	Lack of game-based learning or progress tracking.
FunIPR (Proposed)	An interactive, game-based mobile app designed to make IPR learning simple and engaging for students.	Easy to use, can reach more students, and helps them remember better.

III. SYSTEM DESIGN AND METHODOLOGY

The architecture of the proposed FunIPR system is divided into four layers:

Presentation Layer:

Manages how users use the app that gives a simple android layout Users can sign up as students, open lessons, play quizzes, and view rewards, while teachers can see their progress on a dashboard.

Application Layer:

This holds the game logic, progress tracking, and the quiz engine of the game. It manages how users interact with the content and makes sure the scores, levels, and rewards are updated dynamically.

Backend Layer:

This Node.js server mediates between the front-end and Firebase database, enabling data flow as well as authentication and leaderboard updates [13].

Data Layer:

This system will run on Firebase Firestore, which enables real-time storage of quiz questions, user information, scores, and rewards. The user's security is guaranteed since Firebase handles all the authentication.

The layered approach ensures modularity, scalability, and data security.

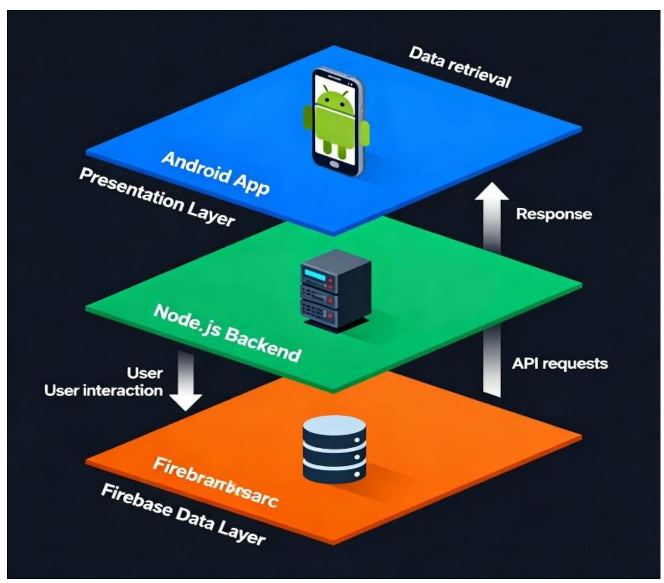


Figure 1: Layered Architecture of FunIPR (Android App → Node.js → Firebase).

IV. IMPLMENTATION

The FunIPR application was developed with the incorporation of a student-centered approach, where each functionality was embedded with the aim of making the learning process of the pupils interactive, straightforward, and enjoyable for both students and teachers.

Frontend (Android Studio):

The interface of the application was designed in Android Studio using XML and Java. it has visually appealing screens to use in logging in, lessons, quizzes, and rewards. The layout was kept clean and easy to navigate so that even younger students can use it without difficulty.

Backend (Node.js):

The app is supported with the help of Node.js, which allows to maintain the process of real-time communication between the app and the database and makes updating the quiz results, awards, and student status a seamless and immediate process at the time of using the app.

Database: Firebase Firestore

Firebase Firestore is secure in that it stores user data, such as profile, quiz questions, results, and achievements. The best part about this is that it enables the information to be stored safely in the cloud and can be brought synchronized to various devices as and when needed. Firebase also handles the authentication process where only the authorized users are allowed to log in and use their accounts.

Modules Implemented:

Login Module: This enables students and teachers to log in or create an account securely using firebase authentication.

Lesson Module: Introduces the main topics of Intellectual Property Rights in simple and visual language so that students can understand each concept in an easy way.

Quiz Module: The module will contain small fun quizzes and have immediate feedback, which will assist students in checking their knowledge and learning by mistakes.

Reward Module: Badges and achievements appear every time when students accomplish lessons or in case, they do good in quizzes, which encourages them to study more.

Teacher Dashboard: This gives teachers an opportunity to track progress of students in a far easier manner, compare scores and participation.

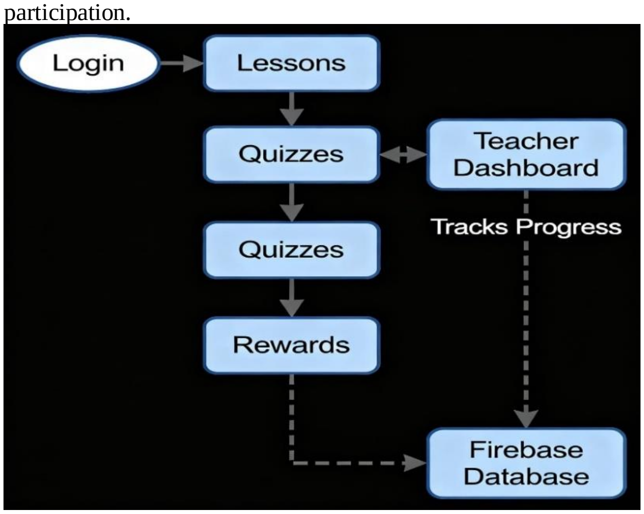


Figure 2: User flow in FunIPR — starting from Login, then Quiz, followed by Rewards, and finally saving information to the database.

V. RESULTS AND DISCUSSION

It involved the collaboration of two local schools with a total of 30 students aged 10-16 years and 4 teachers who were asked to use the FunIPR app. For effectiveness, students were given a short test before and after using the app to compare their learning progress.

A. Quantitative Analysis:

Upon the application of FunIPR, the mean quiz score of the students increased by 38, and this was a clear indication of a greater understanding. Indeed, approximately 92 percent said that the app helped them enjoy and follow the IPR learning. The teachers also reported that the teacher dashboard had been of great help in tracking the progress of every student and his/her involvement.

B. Qualitative Analysis:

The challenges that involved quizzes and the immediate feedback they provided immediately after each question were one of the things the students appreciated. The game like functions, badges, and leader boards made the learnings more interesting and a feeling of competition amongst them.

Teachers also testified that it was much easier to teach hard to grasp IPR concepts using visual lessons and story-based modules. Overall, this app helped students be more engaged in what they are learning and remember what they had learned compared to regular lecture-based teaching, showing how effective gamified learning can be for subjects like IPR and creative education.

VI. CONCLUSION AND FUTURE SCOPE

This paper presented a gamified mobile learning application, FunIPR, developed with the purpose of making learning about IPRs simple and entertaining for school students. Complex topics with respect to IPRs, such as patents, trademarks, and copyrights, are made interactive through games, quizzes, and rewards.

With the help of game features and digital learning tools, FunIPR allows students not only to study the concepts of IPR but also the way they can be demonstrated in real life. The gap between theory and practical knowledge is bridged as students are able to learn through playing and exploring.

The project also supports the Government of India's vision to spread IPR awareness and add creative learning methods to school education. Because of its ease in usage, the app can be used both in classrooms with teachers and at home by students themselves All in all, FunIPR promotes creativity, ingenuity, and appreciation of originality among the young learners.

In the future, the application can be extended to include:

Furthermore, the application has an AI-powered quiz that is designed with the learning level of the particular student in mind to have a more customized experience. It also gives the choice of languages used in various places hence students in various places learn to their convenience.

Lessons based on AR can allow students to see and experience how inventions and trademarks work in real life.

It can also worked together with a national education platform to reach out to more schools and learners throughout the country.

As FunIPR demonstrates, game-based learning properly used can even make such complicated subject as IPR fun, interesting, and easy to learn, at least among young learners.

REFERENCES

[1] J. Koivisto and J. Hamari, “The rise of motivational information systems: A review of gamification research,” International Journal of Information Management, 2019.

[2] A. Singh and P. Gupta, “Digital learning for IPR awareness,” IEEE Access, 2021.
[3] Government of India, “National IPR Policy,” Ministry of Commerce and Industry, 2020.
[4] CIPAM, “IPR Awareness Program for Schools – Government of India Initiative,” 2018.

[5] G. Surendeleg, “Gamification in education: A review,” Education and Information Technologies, 2018

[6] N. Katuk et al., “Gamified learning for legal education,”

Journal of Educational Technology, 2022.
[7] J. Koivisto and J. Hamari, “Gamification of learning: Theory and practice,” Computers in Human Behavior, 2020.
[8] G. Surendeleg, “Interactive Learning in Gamification,” International Education Journal, 2021.
[9] CIPAM Report, “School Level IPR Awareness Program,” 2019.
[10] Katuk, N. et al., “Digital Game-based Learning for Law Education,” Education Technology Journal, 2021.
[11] A. Singh, “Gamified Legal Learning Systems,” IEEE Transactions on Learning Technologies, 2022.
[12] UNESCO, “Digital Pedagogy and the Future of Learning,” 2021.
[13] Node.js Foundation, “Server-side Development Framework,” 2023.
[14] Firebase Documentation, Google, 2024.

Mithun M S

Mithun M S - IPR.docx

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID

trn:oid:::1:3415684794

Submission Date

Nov 18, 2025, 11:35 AM GMT+5:30

Download Date

Nov 18, 2025, 11:47 AM GMT+5:30

File Name

IPR.docx

File Size

239.6 KB

4 Pages

2,474 Words

14,326 Characters

0% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

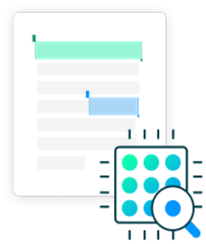
0 AI-generated only 0%
Likely AI-generated text from a large-language model.

0 AI-generated text that was AI-paraphrased 0%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.



False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.