

# **Upskill App**

## **Major Project I Report**

Submitted in partial fulfillment of the requirements for the degree of

### **Bachelor of Engineering (Computer Engineering)**

by:

MORE NEHA NARSING

TUS3F202123

SUMIT KOLI

TUS3F202167

ROUNAK SINGH

TUS3F202114

RISHABH MUKADAM

TUS3F202116

**Under the Guidance of**

**Prof. Ujwala Gaikwad**



**Department of Computer Engineering**

**TERNA ENGINEERING COLLEGE**

Nerul (W), Navi Mumbai 400706

**(University of Mumbai)**

**(2023-2024)**



**TERNA ENGINEERING COLLEGE, NERUL,  
NAVI MUMBAI**

**Department of Computer Engineering**  
**Academic Year 2023-24**

**CERTIFICATE**

This is to certify that the major project entitled “Upskill App” is a bonafide work of

MORE NEHA NARSING

TUS3F202123

SUMIT KOLI

TUS3F202167

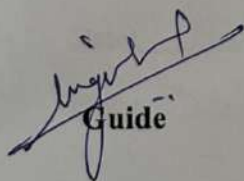
ROUNAK SINGH

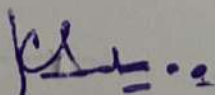
TUS3F202114

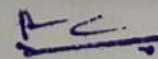
RISHABH MUKADAM

TUS3F202116

submitted to the University of Mumbai in partial fulfillment of the requirement for the award  
of the Bachelor of Engineering (Computer Engineering).

  
Guide

  
Head of Department

  
Principal

Approval Sheet

**Project Report Approval**

This major project report entitled as "Upskill App" by following students is approved for the degree of *B.E. in "Computer Engineering"*.

**Submitted by:**

More Neha Narsing

TUS3F202123

Sumit Koli

TUS3F202167

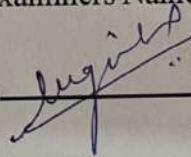
Rounak Singh

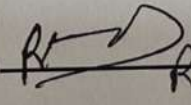
TUS3F202114

Rishabh Mukadam

TUS3F202116

**Examiners Name & Signature:**

1.  Jyoti Gajale

2.  R. P. Shende

Date: 28/11/23

Place: Nerul.

### Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

More Neha Narsing

TUS3F202123-----

*Neha*  
*S.K. Koli*

Sumit Koli

TUS3F202167-----

*Singh*

Rounak Singh

TUS3F202114-----

*Rounak*

Rishabh Mukadam

TUS3F202116-----

Date: 28/11/23

Place: TEC Merul



## Acknowledgement

We would like to express our sincere gratitude towards our guide **Prof. Ujwala Gaikwad**, Project Coordinators **Prof. Minal Chaudhari** for their help, guidance and encouragement, they provided during the project development. This work would have not been possible without their valuable time, patience and motivation. We thank them for making our stints thoroughly pleasant and enriching. It was a great learning experience and an honor being their students.

We are deeply thankful to **Prof. Kishor Sakure (H.O.D Computer Department)** and entire team in the Computer Department. They supported us with scientific guidance, advice and encouragement, they were always helpful and enthusiastic and this inspired us in our work.

We take the privilege to express our sincere thanks to **Dr. L. K. Ragha** our principal for providing the encouragement and much support throughout our work.

More Neha Narsing

TUS3F202123

Sumit Koli

TUS3F202167

Rounak Singh

TUS3F202114

Rishabh Mukadam

TUS3F202116

Date: 28/11/23

Place: TEC, Merul

## Abstract

A good learning platform can reduce the teaching loads and improve the program learning efficiency. Currently, there are many online application systems on the internet. These systems can automatically check the correctness of the question submitted by a user, which is very helpful for program learning.

We have decided to design an online platform that is more suitable for our programming courses and aptitude. The designed platform not only is able to check the correctness of the submitted codes, but also provides some features for teaching, including course management, student management, and problem management.

In a time when technology is developing at an exponential rate, there is an unparalleled need for skilled coders and problem solvers. By presenting a dynamic mobile application created with Flutter, a framework intended for seamless cross-platform experiences, this significant project aims to meet this demand. With its combination of coding and aptitude challenges, the program acts as a comprehensive learning center, guiding users toward technical mastery.

This significant effort advances the idea of all-encompassing technical education by combining the science of logical reasoning with the art of coding. The application is proof of creativity, user-centered design, and the dedication to supporting people in their pursuit of technological excellence.

## **Table of Contents**

<b>Chapter No.</b>	<b>Chapter Name</b>	<b>Pg. No.</b>
	List of Figures	iii
	List of Tables	iv
Chapter 1	Introduction	01
	1.1 Introduction	01
	1.2 Scope	02
	1.3 Organization of the report	02
Chapter 2	Literature Survey	03
	2.1 Existing System	03
	2.2 Comparative Analysis of Literature papers	
	2.3 Problem Statement	04
	2.4 Objective of project	05
Chapter 3	Software Analysis	07
	3.1 Software Model	07
	3.2 Proposed System	09
	3.3 Software Requirement Specification (SRS)	10
	3.4 Software and Hardware used	12
	3.5 Project Timeline Chart	13
Chapter 4	Design	14
	4.1 System Architecture Diagram	14
	4.2 Data Flow Diagram	15
	4.3 Use Case Diagram	16
	4.4 Risk Mitigation Monitoring and Management	17
	4.5 Sequence Diagram	20
Chapter 5	Methodology	21
Chapter 6	Implementation	23
Chapter 7	Conclusion	25
References		26

### **List of Figures**

<b>Fig. No.</b>	<b>Figure Name</b>	<b>Pg. No</b>
3.1	Stages of Software Development	07
3.2	Proposed System	09
3.5	Timeline for Semester 7	13
4.1	System Architecture Diagram	14
4.2	Data Flow Diagram	15
4.3	Use Case Diagram	16
4.5	Sequence Diagram	20
6.1	Initial Landing Page	23



## List of Tables

Table No.	Table Name	Pg. No
2.2	Comparative Analysis of Literature papers	04
4.4	Risk Mitigation Monitoring and Management Plan	17

# Chapter – 1

## Introduction

### 1.1 Introduction

A good learning platform is indeed very helpful for program teaching, which can not only reduce the burden on teachers, but also improve student learning efficiency. Leetcode is well-known website. The user only needs to write a function instead of the entire program.

In the rapidly evolving landscape of technology, where coding prowess and problem solving acumen stand as indispensable skills, our major project embarks on a transformative journey to empower individuals with a cutting-edge learning solution.

This project manifests as a dynamic mobile application crafted using Flutter—a versatile, cross-platform framework that promises a seamless and engaging user experience. At its core, this application serves as a comprehensive platform, strategically integrating both aptitude and coding challenges to cultivate a well-rounded skill set in its users.

The necessity of technology:

With technology becoming more and more integrated into our daily lives, there is an increasing need for people who are skilled in coding and have a keen sense of problem solving. When it comes to offering real-world, practical experiences that are in line with the difficulties faced in the workplace, traditional education frequently falls short. Identifying this gap, our initiative aims to close it by launching a mobile application that goes beyond traditional learning parameters and provides an engaging and dynamic platform for skill building.

The Cross-Platform Development Skills of Flutter:

It was purposeful to select Flutter as the development framework because of its track record of providing a unified and aesthetically pleasing user experience across a range of platforms and operating systems. The adaptability and effectiveness of Flutter make

## 1.2 Scope of project

The scope of our major project, an application providing an integrated aptitude and coding platform built using Flutter, extends across various dimensions, catering to the needs of diverse users and addressing the evolving landscape of technological education

## 1.3 Organization of the report

- Chapter 1 gives a brief overview about the aim for developing this project. The introduction and the scope of the project tells us about the expected outcome of the project for the application.
- Chapter 2 of the report includes the literature survey of the existing system, discusses its limitations, identifies the problem statement and establishes the objectives of the project.
- Chapter 3 tells which software model we have used for the project and shows the proposed system of what our software will achieve and gives a brief idea of the various users included and their functionalities. Along with this, the hardware and software requirements and software component are described.
- Chapter 4 shows the system architecture diagram, data flow diagram and some UML diagrams that give us an abstract view of the system.
- Chapter 5 is the conclusion. It describes the tasks that we are going to achieve through this project. Also, this chapter gives a summary of the entire project.



## Chapter – 2

### Literature Survey

#### 2.1 Existing System Survey

##### Paper 1:

**Paper Name:** Programming web designing

**Author Name:** Klement, T. Staubitz

**Explanation:**

The research paper titled "Programming Web Designing" authored by Klement, T. Staubitz delves into the intricate relationship between programming and web design. The paper explores the symbiotic connection between these two domains, emphasizing the significance of proficient programming skills in crafting effective and aesthetically pleasing web designs.

Klement and Staubitz present a comprehensive overview of the evolving landscape of web development, shedding light on the increasing demand for a synergy between programming expertise and design principles. The authors delve into the nuances of how programming languages contribute to the functionality and interactivity of modern websites while also influencing the overall design aesthetics.

The paper highlights case studies and examples to illustrate how specific programming languages, frameworks, and techniques impact the design process. It underscores the importance of a holistic approach, where programmers and web designers collaborate seamlessly to achieve optimal results.

Additionally, Klement and Staubitz discuss emerging trends and best practices in the intersection of programming and web design. They address challenges faced by professionals in mastering both skill sets and provide insights into strategies for overcoming these challenges. In conclusion, "Programming Web Designing" by Klement, T. Staubitz serves as a valuable resource for professionals and enthusiasts seeking a deeper understanding of the interconnected worlds of programming and web design. The paper contributes to the ongoing discourse on the integration of these disciplines, offering insights that are relevant to the dynamic and ever-evolving field of web development.



## 2.2 Comparative Analysis of Literature papers

Sr. No.	Name of author & Year	Paper Title	About
1	Klement, T. Staubitz May 7, 2020	Programming web designing	The designed platform not only be able to check the correctness of the submitted codes, but also provides some features for teaching, including course management, student management, and problem management
2	M. Murata and T. Kakeshita 2016	Analysis Method of Student Achievement Level Utilizing Web-Based Programming Education Support Tool Pgtacer	In programming courses, a good learning platform can reduce the teaching loads and improve the program learning efficiency. Currently, there are many online judge systems on the internet.
3	M. A. Revilla, S. Manzoor 2018	Competitive Learning in Informatics: The UVa Online Judge Experience	A good learning platform is indeed very helpful for program teaching, which can not only reduce the burden on teachers, but also improve student learning efficiency
4	Shivani Surana; Komal Pathak; Mehul Gagnani 2022	Text Extraction and Detection from Images using Machine Learning Techniques: A Research Review	Machine Learning is subset of Artificial Intelligence and there is lots of research growth across the world. It has capability to learn by its own without seeking any help from human beings or without any explicit programming based on its previous experience and knowledge

5	Nataasha Raul and Ruchika Shah 2014	Offline Aptitude Application	Android	The students appearing for placements and various other exams like GRE, GMAT, GATE prepare themselves by appearing for mock tests and searching for the study material from different websites
---	--	------------------------------	---------	--

**Table 2.2**  
**Comparative Analysis of Literature papers**

## 2.3 Problem Statement

In the ever-evolving landscape of technology, the demand for individuals equipped with a robust combination of coding proficiency and logical reasoning skills is on the rise. However, the existing educational frameworks often fall short in providing a comprehensive and interactive platform for users to simultaneously enhance their aptitude and coding abilities. Traditional methods of skill development lack the dynamism required to keep pace with rapidly changing technological requirements.

This Upskill App seeks to address this gap by introducing an innovative mobile application built using Flutter, designed to provide users with a unified platform for refining both their aptitude and coding skills. The absence of such an integrated solution poses challenges for individuals aspiring to excel in competitive programming, technical interviews, and real-world problem-solving scenarios.

Challenges:

Fragmented Learning Resources:

Existing resources predominantly focus on either aptitude or coding skills, creating a fragmented learning experience for users who need to develop both in tandem.

Limited Interactivity:

Many educational platforms lack the interactivity required for users to apply theoretical knowledge in a practical setting, hindering the development of hands-on coding expertise.

Lack of personalization:

The one-size-fits-all approach to conventional educational methods fails to cater to the diverse skill levels and learning preferences of individual users.

Insufficient community engagement:

The absence of a collaborative space where users can discuss challenges, share insights, and learn from each other hampers the development of a supportive and dynamic learning



community.

#### Technological Accessibility:

Some users face challenges in accessing learning resources due to limitations in device compatibility and operating system constraints.

#### Impact:

The identified challenges impede the holistic development of individuals aspiring to excel in the competitive and demanding field of technology. The lack of an integrated platform that addresses these challenges hinders the realization of the full potential of learners, limiting their effectiveness in real-world coding scenarios and technical assessments.

#### Proposed Solution:

This Upskill App aims to overcome the aforementioned challenges by developing an application using Flutter, offering a seamless and comprehensive solution that integrates aptitude and coding learning modules. The proposed platform seeks to provide users with a dynamic and interactive environment, fostering a community of learners who can collaboratively enhance their skills and excel in the multifaceted demands of the technological landscape.

## 2.4 Objective of project

The following are the main objectives driving this project:

#### Enhancement of Skills:

Enable users to improve their coding skills in a variety of programming languages by providing them with a library of interesting and difficult coding tasks.

#### Growth of Logical Reasoning:

Assist users in honing their logical thinking and problem-solving skills by providing flexible and demanding aptitude tests.

#### Friendly User Interface:

Using Flutter, create a user interface that is both visually beautiful and straightforward to use. This will ensure that users of all technological backgrounds have a seamless and engaging experience.

## Chapter – 3

### Software Analysis

#### 3.1 Software Model

The classical waterfall model is the basic software development life cycle model. It is very simple but idealistic. Earlier this model was very popular but nowadays it is not used. But it is very important because all the other software development life cycle models are based on the classical waterfall model.

The classical waterfall model divides the life cycle into a set of phases. This model considers that one phase can be started after the completion of the previous phase. That is the output of one phase will be the input to the next phase. Thus, the development process can be considered as a sequential flow in the waterfall.

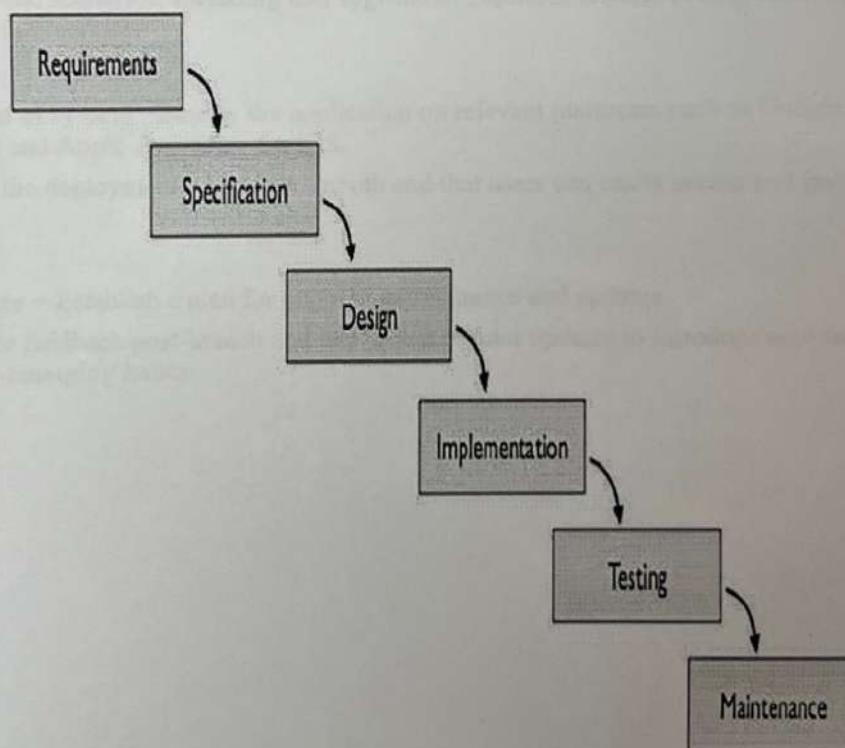


Fig. 3.1 Stages of software development



**The sequential phases in Waterfall model are**

**Requirement Gathering and analysis** – Define and document the specific requirements of the application, including features for aptitude testing, coding challenges, user authentication, and community collaboration.

**System Design** – Create a detailed system design, outlining the architecture, database structure, and user interface components using Flutter.

Define the interactions between the aptitude service, coding service, authentication, and database components.

**Implementation** – Begin the development of the application using Flutter for the user interface.

Implement the authentication system using Firebase Authentication.

Develop the backend services for aptitude testing and coding challenges, integrating with Firebase Fire store for data storage..

**Integration and Testing** – Conduct thorough system tests to validate the overall functionality of the application.

Test end-to-end scenarios, including user registration, aptitude testing, coding challenges, and community interactions.

**Deployment of system** – Deploy the application on relevant platforms such as Google Play Store for Android and Apple App Store for iOS.

Ensure that the deployment process is smooth and that users can easily access and install the application.

**Maintenance** – Establish a plan for ongoing maintenance and updates.

Monitor user feedback post-launch and implement regular updates to introduce new features and address any emerging issues.

### 3.2 Proposed System

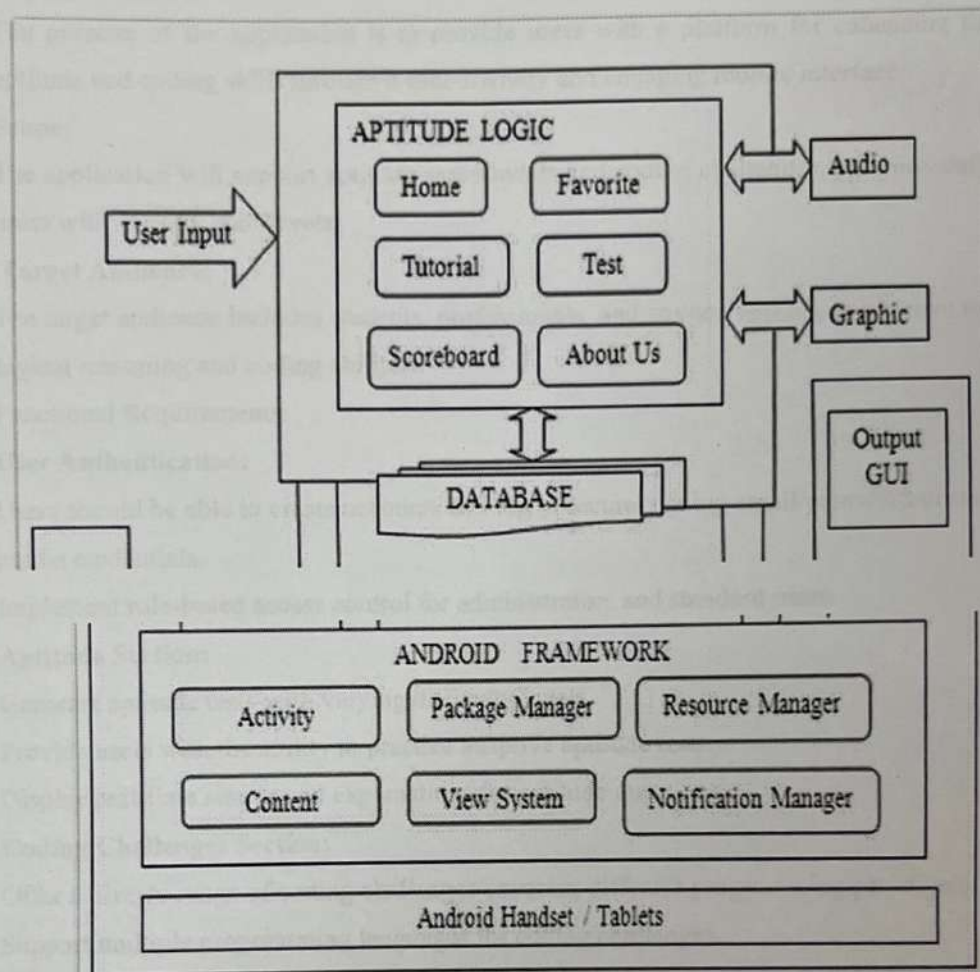


Fig. 3.2 Proposed system

### **3.3 Software Requirement Specification (SRS)**

#### **Introduction:**

##### **Purpose:**

The purpose of the application is to provide users with a platform for enhancing their aptitude and coding skills through a user-friendly and engaging mobile interface.

##### **Scope:**

The application will support aptitude assessments and coding challenges, accommodating users with varying skill levels.

##### **Target Audience:**

The target audience includes students, professionals, and anyone seeking to improve their logical reasoning and coding abilities.

#### **Functional Requirements:**

##### **User Authentication:**

Users should be able to create accounts and log in securely using email/password or social media credentials.

Implement role-based access control for administrators and standard users.

##### **Aptitude Section:**

Generate aptitude tests with varying difficulty levels.

Provide users with the ability to practice adaptive aptitude tests.

Display real-time results and explanations for aptitude questions.

##### **Coding Challenges Section:**

Offer a diverse range of coding challenges covering different programming paradigms.

Support multiple programming languages for coding challenges.

Include an integrated code editor with syntax highlighting and autocompletion.

##### **User Profile:**

Allow users to create and manage profiles with personal details and preferences.

Display user achievements, progress, and history.

#### **Non-functional Requirements:**

##### **Performance:**

Ensure responsive and low-latency interactions.

Support a scalable architecture to accommodate a growing user base.

##### **Security:**

Implement secure user authentication and authorization mechanisms.

Encrypt sensitive user data during transmission and storage.



**Scalability:**

Design the system to handle an increasing number of users and challenges over time.

Utilize cloud services for scalability and performance optimization.

**Compatibility:**

Ensure compatibility with Android and iOS platforms.

Optimize the application for various screen sizes and resolutions

**Usability:**

Design an intuitive and user-friendly interface following Material Design guidelines.

Conduct usability testing to refine the user experience.

**Reliability:**

Minimize system downtime and ensure data integrity.

Implement regular data backups and recovery mechanisms.

**Constraints:****Technology Stack:**

The application will be built using Flutter for cross-platform development, Firebase for authentication and database, and other relevant technologies.

**Internet Connectivity:**

Users must have an active internet connection for accessing challenges, submitting solutions, and receiving updates.

**Programming Language Support:**

The application will support a predefined set of programming languages for coding challenges.

**Assumptions and Dependencies:****Assumptions:**

Users have basic knowledge of the programming languages supported.

The application will rely on Firebase for authentication, database, and cloud functions.

**Dependencies:**

The application is dependent on third-party libraries and tools compatible with the Flutter framework.

These System Requirements Specifications provide a comprehensive outline for the development of the aptitude and coding platform application, ensuring clarity and alignment with the project's goals and objectives.



### **3.4 Software and Hardware requirement**

#### **Hardware Requirement:**

Desktop/Laptop:

Minimum 4GB RAM (8GB or higher recommended for smoother performance).

Multi-core processor (Intel i5 or equivalent recommended).

Sufficient free disk space for development tools and project files.

Mobile Devices:

Android and/or iOS devices for testing and deployment.

#### **Software Requirements:**

Development Environment:

Flutter SDK: The latest version of Flutter SDK for cross-platform mobile application development

Dart SDK: Required for Flutter development, bundled with Flutter SDK.

Integrated Development Environment (IDE):

Android Studio : For Android and iOS development, respectively.

Database:

Firebase: Firebase Authentication for user management and Firestore for real-time cloud database services.

Design Tools:

Figma,: For UI/UX design and prototyping

### 3.5 Gantt charts

A Gantt chart is a type of bar chart that illustrates a project schedule. This chart lists the tasks to be performed on the vertical axis, and time intervals on the horizontal axis. The width of the horizontal bars in the graph shows the duration of each activity. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements constitute the work breakdown structure of the project. Modern Gantt charts also show the dependency (i.e., precedence network) relationships between activities. Gantt charts can be used to show current schedule status using percent-complete shadings and a vertical "TODAY" line. Gantt charts are sometimes equated with bar charts. Gantt charts are usually created initially using an early start time approach, where each task is scheduled to start immediately when its prerequisites are complete. This method maximizes the float time available for all tasks.

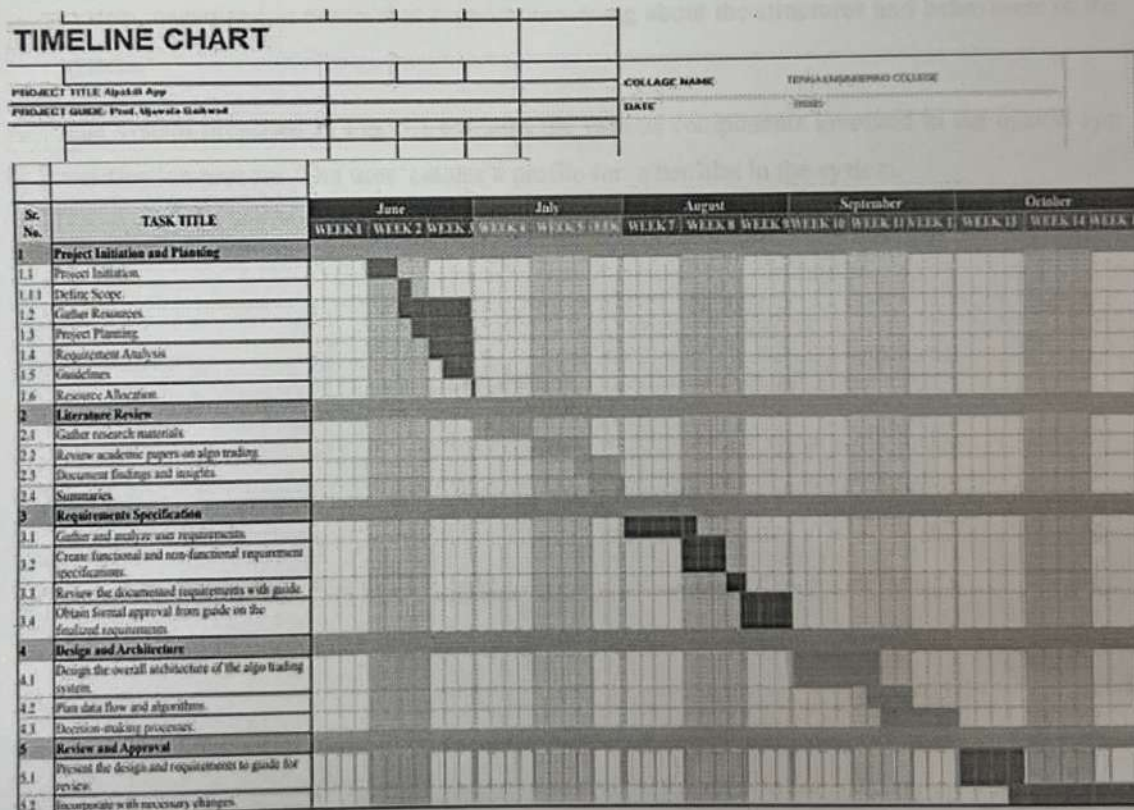


Fig. 3.5 Timeline chart

## Chapter – 4

### Design

#### 4.1 System Architecture

A system architecture is the conceptual model that defines the structure, behaviour, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviours of the system.

The system proposed in Fig. 5.1 explains the various components involved in the upskill app verification process. The user creates a profile for a her/him in the system.

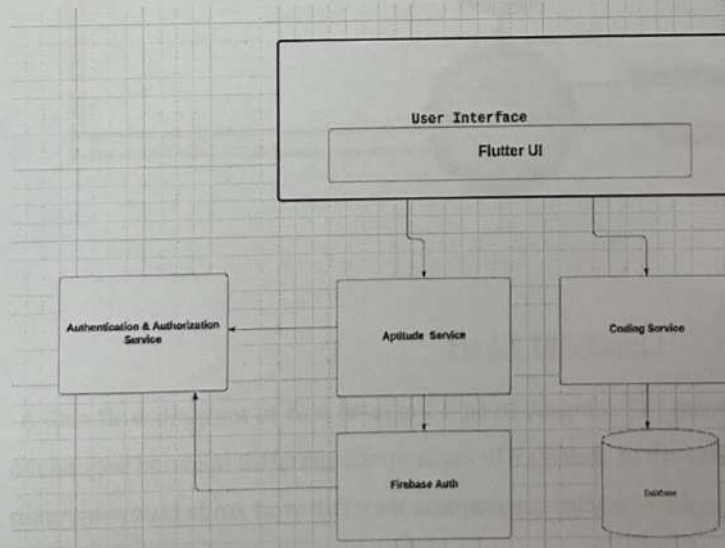


Fig. 4.1 System Architecture



## 4.2 Data Flow Diagram

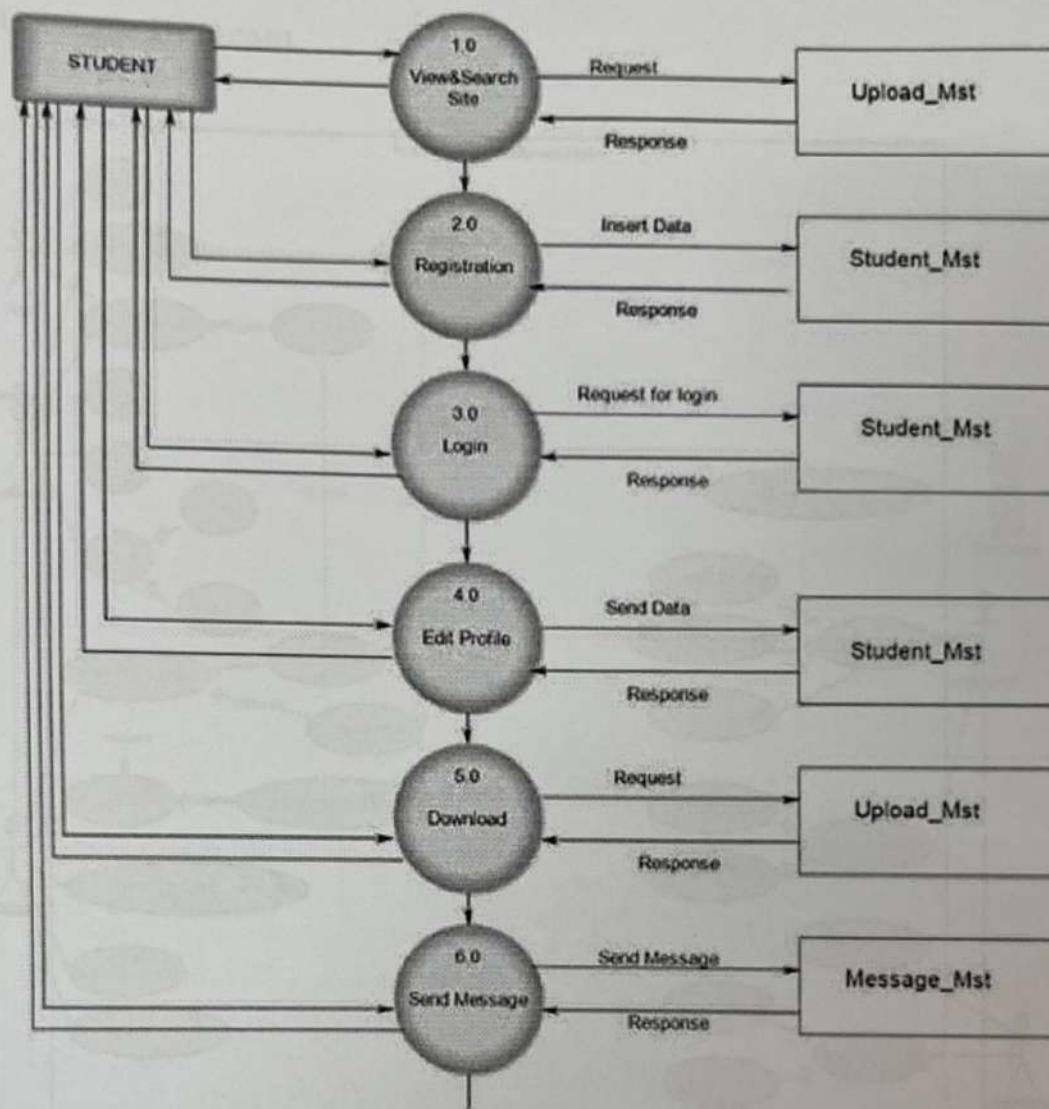


Fig 4.2 DFD Level 1

A data flow diagram of first level is a type of diagram that shows the interactions and messages exchanged between different components of a system. In the case of an Upskill app the sequence diagram would show how different components interact with each other to perform the identity verification process



## 4.3 Use-case Diagram

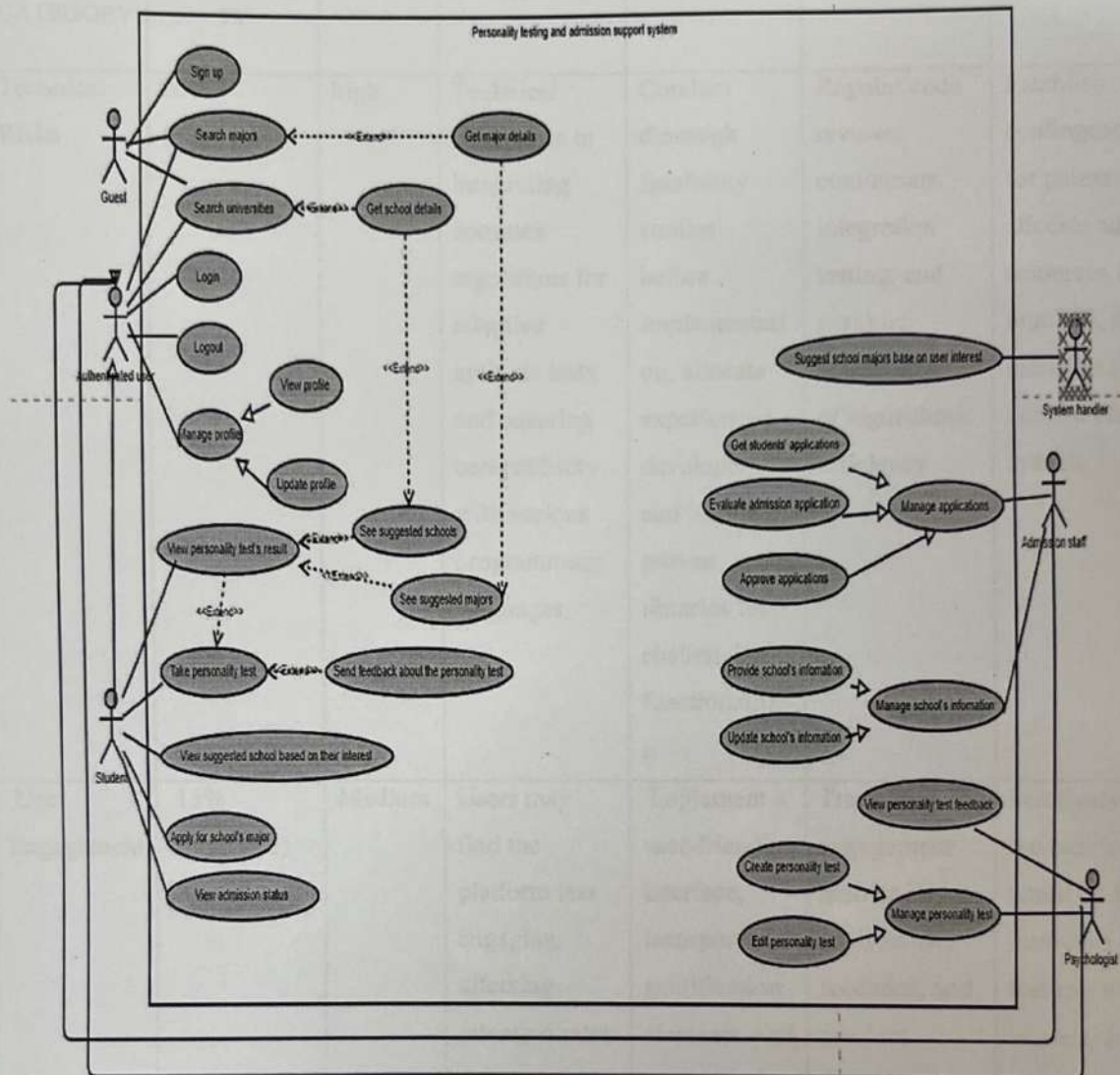


Fig. 4.3 Use Case Diagram

The admin panel is the core component of the upskill app, and it has two use cases: "Submit Identity Information" and "Verify Identity Information". The "Submit Identity Information" use case allows the User to submit their identity information to the system, which is then stored in the database. The "Verify Identity Information" use case allows the External System to request identity verification

#### 4.4 Risk Mitigation Monitoring and Management Plan

Sr. No	RISK CATEGORY	PROBABILITY	IMPACT	DESCRIPTION	MITIGATION	MONITORING	MANAGEMENT
1	Technical Risks	20% (moderate)	high	Technical challenges in integrating complex algorithms for adaptive aptitude tests and ensuring compatibility with various programming languages.	Conduct thorough feasibility studies before implementation, allocate experienced developers, and leverage proven libraries for challenging functionalities.	Regular code reviews, continuous integration testing, and periodic assessments of algorithmic efficiency.	Establish contingency plans for potential delays, allocate additional resources if required, and maintain a robust version control system.
2	User Engagement	15% (moderate)	Medium	Users may find the platform less engaging, affecting retention rates.	Implement a user-friendly interface, incorporate gamification elements, and gather user feedback during beta testing.	Track user engagement metrics, analyze user feedback, and conduct usability testing.	iteratively enhance the user interface based on feedback, introduce new features to maintain interest, and actively engage with the user community.
3	Security and Privacy	15% (moderate)	high	Potential security vulnerabilities and data	Employ secure coding practices, implement	Regular security audits, penetration testing, and	Swift response to security incidents, regular updates to address vulnerabilities, and

				privacy concerns.	robust authentication mechanisms, and adhere to data protection regulations.	continuous monitoring of user data.	transparent communication with users about privacy measures.
4	Firebase Dependency	12% (moderate)	moderate	Unforeseen issues with Firebase services affecting authentication, database operations, or cloud functions.	Stay informed about Firebase updates, maintain backup authentication systems, and implement offline capabilities for essential features	Monitor Firebase status updates, implement error tracking, and conduct periodic performance assessments.	Quickly address any service disruptions, communicate effectively with users about downtime, and explore alternative solutions if required.
5	Community Dynamics	18% (moderate)	high	Potential conflicts or misuse of community features leading to a negative impact on the platform's collaborative environment..	Implement moderation tools, establish clear community guidelines, and foster a positive and inclusive community culture.	Regularly review community interactions, analyze user reports, and conduct community surveys.	Enforce community guidelines consistently, address conflicts promptly, and provide positive reinforcement for constructive contributions.



6	Regulatory Compliance	8% (low)	high	Changes in data protection or education-related regulations impacting the application.	Stay informed about relevant regulations, conduct legal reviews, and implement necessary adjustments to comply with evolving laws.	Regularly review legal updates, consult legal experts, and participate in industry forums discussing regulatory changes.	Swiftly adapt the platform to comply with new regulations, communicate changes to users, and maintain transparent policies.
---	-----------------------	----------	------	--	--	--	---

**Table 4.4**  
**Risk Mitigation Monitoring and Management Plan**

## 4.5 Sequence Diagram

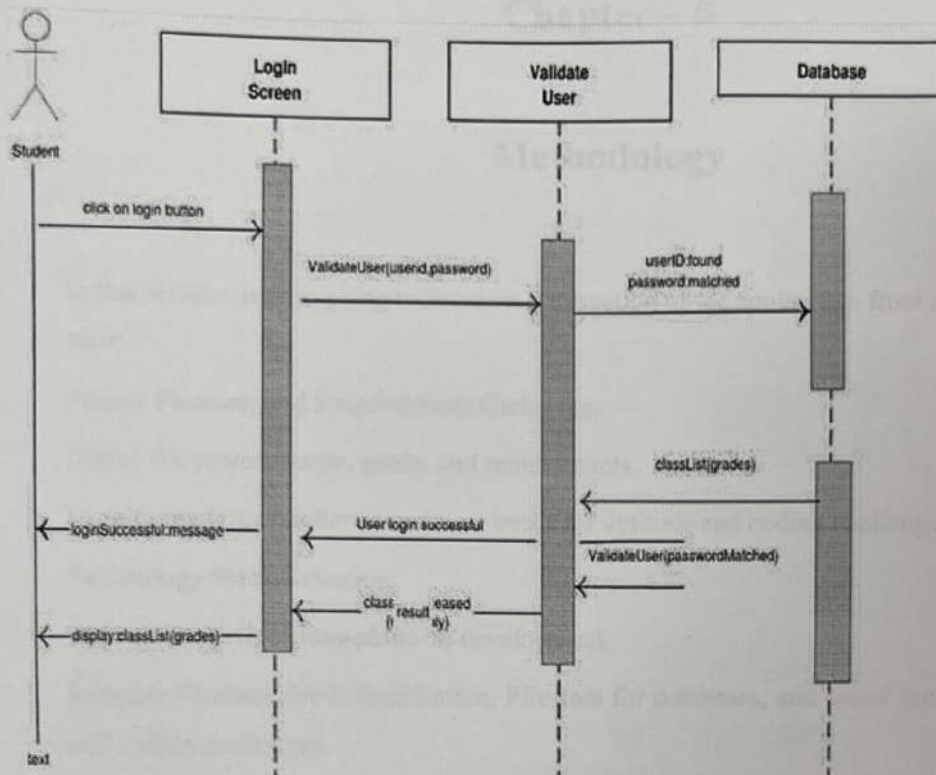


Fig. 4.5 Sequence Diagram

A sequence diagram is a type of UML diagram that shows the interactions and messages exchanged between different components of a system. In the case of an Upskill app the sequence diagram would show how different components interact with each other to perform the identity verification process.

1. Student login profiles.
2. Validate user checking system
3. Database to store the details

The above sequence diagram shows the interactions between the user, upskill app, and iverification process. The user submits their identity information to the upskill app, which then initiates the verification process. The upskill app sends a request to the server to verify the user identity, and the validates the identity using a data stored in database

## Chapter – 5

### Methodology

In this section, we are going to focus on the creation of the application from a technical point of view.

**Project Planning and Requirements Gathering:**

Define the project scope, goals, and requirements.

Identify the target audience and user needs for aptitude and coding challenges.

**Technology Stack Selection:**

Choose Flutter for cross-platform development.

Integrate Firebase for authentication, Firestore for databases, and cloud functions for aptitude and coding challenges.

**UI/UX Design:**

Design user interfaces using Flutter's widgets and adhere to Material Design guidelines. Focus on an intuitive and engaging user experience for both aptitude and coding sections.

**Authentication and Authorization:**

Implement Firebase Authentication for user sign-up and login.

Set up user roles and permissions for accessing aptitude and coding challenges.

**Aptitude Service:**

Develop a service to generate and manage aptitude tests and questions.

Implement algorithms for randomizing questions and assessing user responses.

**Coding Service:**

Build a service for managing coding challenges, storing problem statements, and evaluating user solutions. Support multiple programming languages for coding challenges.

**Database Integration:**

Set up Firestore to store user profiles, aptitude test results, and coding challenge

Design efficient data structures to store and retrieve information.



#### User Interface Implementation:

Develop the Flutter UI for the aptitude and coding sections.

Ensure a responsive and visually appealing design for various screen sizes.

#### Community Features:

Implement a community section for users to discuss challenges, share solutions, and collaborate.

Integrate features for liking, commenting, and following other users.

#### Testing:

Conduct unit tests for individual components and services.

Perform integration testing to ensure seamless interaction between different sections.

#### Deployment:

Deploy the application on both Android and iOS platforms.

Utilize continuous integration tools for automated build and deployment processes.

#### Documentation:

Create comprehensive documentation for code, APIs, and the overall project structure. Provide user guides and tutorials for using the aptitude and coding platforms.

#### Feedback and iteration:

Gather user feedback through beta testing and user surveys.

Iterate on the application based on feedback to improve usability and address issues.

#### Maintenance and Future Development:

Establish a plan for ongoing maintenance and updates.

Consider future enhancements such as new coding challenges, additional features, or integration with educational institutions.

## Chapter – 6

### Implementation

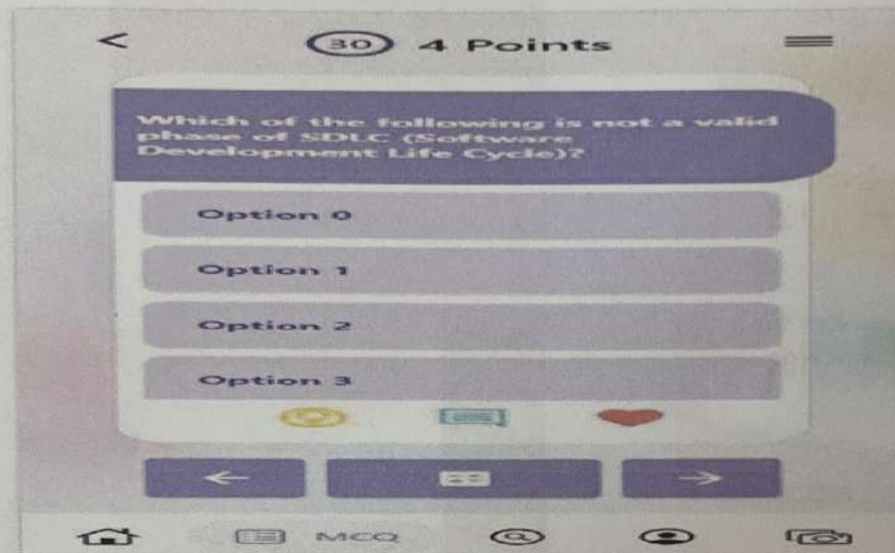
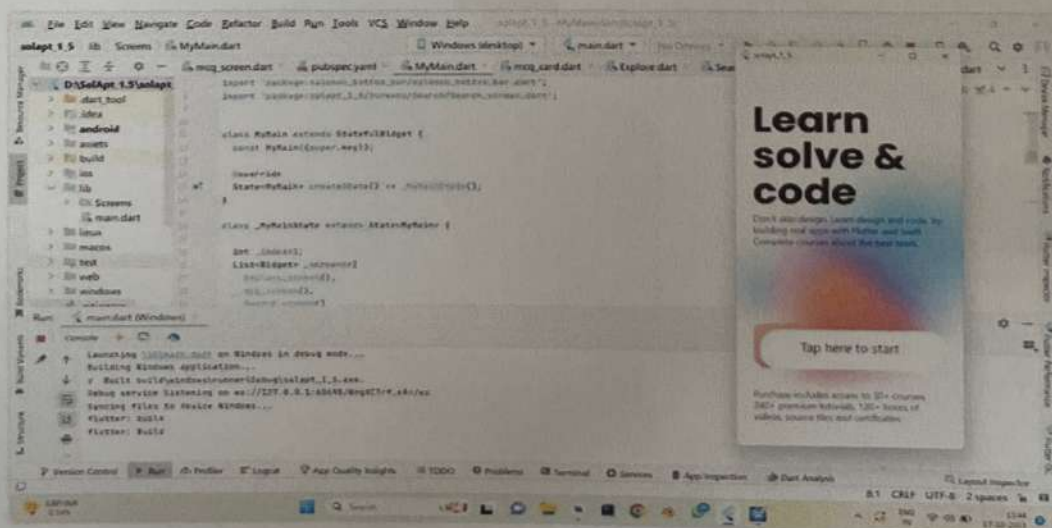
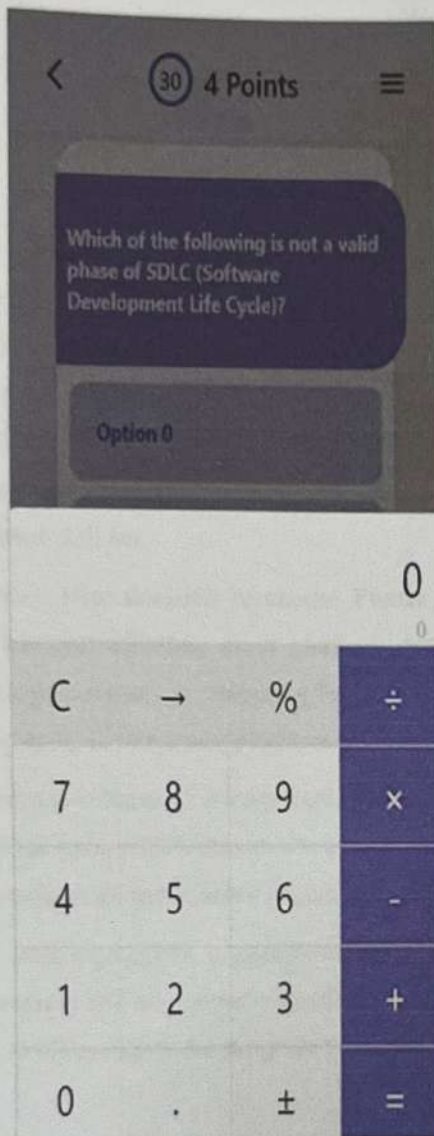


Fig. 6.1 Initial Landing Page

This is the initial landing page while using the platform







## Chapter – 7

### Conclusion

To sum up, the creation and execution of the Flutter-based aptitude and coding platform application mark a noteworthy advancement in meeting the increasing demand for easily navigable and engaging educational resources related to programming and problem-solving abilities. The project's goal was to give users access to an all-inclusive platform that would improve their aptitude abilities, as well as their coding expertise, giving them a more well-rounded skill set.

It was a wise decision to choose Flutter as the application's framework since it provided a flexible and effective cross-platform development environment. This choice expedited the development process, requiring fewer resources and less time, and it also enabled a smooth user experience across many platforms and operating systems.

The main features of the application are the aptitude tests and coding challenges it offers. The aptitude tests, which are intended for a broad audience including novices and seasoned experts, are made to evaluate users' logical reasoning, problem-solving skills, and quantitative ability.

The user experience is greatly enhanced by the user interface, which was created using the expressive and adaptable widgets offered by Flutter. Users with different levels of experience can easily navigate the program thanks to its clear and simple design.

## References

- [1] J. L. Bez, N. A. Tonin and P. R. Rodegheri, "URI Online Judge Academic: A tool for algorithms and programming courses," 2014 9th International Conference on Computer Science & Education, Vancouver, BC, 2014, pp. 149-152, doi: 10.1109/ICCSE.2014.6926445
- [2] M. A. Revilla, S. Manzoor, and R. Liu, "Competitive Learning in Informatics: The UVA Online Judge Experience," Olympiads in Informatics, vol. 2, 2008, pp. 131-14
- [3] T. Staubitz, H. Klement, J. Renz, R. Teusner and C. Meinel, "Towards practical programming exercises and automated assessment in Massive Open Online Courses," 2015 IEEE International Conference on Teaching,
- [4] T. M. Breuel, A. Ul-Hasan, M. A. Al-Azawi and F. Shafait, "Highperformance OCR for printed English and Fraktur using LSTM networks", Proc. 12th Int. Conf. Document Anal. Recognit, pp. 683-687, Aug. 2013.
- [5] A. N. S. Chakravarthy, Penmesta V Krishnaraja and P. S Avahani, "Handwritten Text Image Authentication using Back Propagation", IJNSA, vol. 3, no. 5, 2011.