CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

LANGMATES

by

Maria Mathew (2141154)

Sharoan Santhosh (2141160)

Under the Guidance of

Dr. Ramamurthy B

A Project report submitted in partial fulfillment of
the requirements for the award of degree of
Bachelor of Computer Applications of
CHRIST (Deemed to be University)

April - 2024

CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

CERTIFICATE

*This is to certify that the report titled **LangMates** is a bona fide record of work done by **Maria Mathew (2141154) and Sharoan Santhosh (2141160)** of CHRIST (Deemed to be University), Bangalore, in partial fulfillment of the requirements of VI Semester BCA during the year 2024.*

Project In-charge **Head of the Department**

Valued-by:	Name	:	Maria Mathew Sharoan Santhosh
1.	Register Numbers	:	2141154 2141160
2.	Examination center	:	CHRIST (Deemed to be University)
	Date	:	29th April 2024

ACKNOWLEDGMENTS

The project, Langmates – an app to learn languages , was successfully completed with the assistance of many individuals. Firstly, we express our sincere gratitude to our Vice-Chancellor, Dr. Fr. Joseph CC, Pro Vice-Chancellor, Dr. Fr.Viju PD, Head of the Department, Dr. Ashok Immanuel V, Coordinator, Dr. Beaulah S and the faculty for providing us with this invaluable opportunity to put in and improve our technical skills and learn new technologies and work on a practical idea.

We express our gratitude to our project guide Dr. Ramamurthy B , who took a keen interest in helping and guiding us throughout our project work by providing all the necessary information for coming up with a good result.

We would also like to thank other teachers for the suggestions, continuous feedback, and supervision of our project work. We are thankful and fortunate enough to get constant encouragement, support, and guidance from the faculty of BCA, and our fellow batchmates who helped us in successfully completing our project work.

ABSTRACT

Langmates is not just another language learning app – it's your ultimate companion for embarking on a journey to master languages while immersing yourself in diverse cultures. Our innovative platform seamlessly integrates essential language resources, personalized cultural insights, and powerful language tools, all in one place, to make your learning experience effortless and enriching.

With Langmates, bid farewell to the hassles of language learning and embrace a stress-free path towards linguistic proficiency and cultural understanding. Whether you're preparing for travel, enhancing your career prospects, or simply pursuing personal enrichment, Langmates is here to guide you every step of the way.

Key Features of Langmates:

- Language Learning Resources: Access a comprehensive suite of language learning materials tailored to your proficiency level and learning goals.
- Cultural Immersion Module: Immerse yourself in the vibrant tapestry of global cultures through personalized recommendations, cultural insights, and interactive experiences.
- Language Tools: Utilize a range of powerful language tools, including translation, pronunciation guides, vocabulary builders, and more, to enhance your language learning journey.
- Seamless Integration: Enjoy a user-friendly interface that seamlessly integrates all language resources and cultural immersion features, making learning intuitive and enjoyable.
- Community Engagement: Connect with a global community of language learners, exchange cultural experiences, and practice your language skills in real-world contexts.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

1.1 BACKGROUND OF THE PROJECT

Langmates revolutionizes language learning by seamlessly integrating resources, cultural insights, and powerful tools. It offers immersive cultural experiences alongside tailored materials for all proficiency levels and goals. Through community engagement, users practice language skills and broaden their horizons. Langmates goes beyond traditional apps, providing a holistic learning environment that fosters real-world connections and deepens cultural understanding.

1.2 OBJECTIVE

Langmates aims to facilitate seamless language learning with integrated cultural immersion, providing personalized resources tailored to users' proficiency levels and goals. It fosters community engagement for collaborative practice and aims to enhance global understanding by breaking down language barriers and promoting cross-cultural communication.

1.3 PURPOSE, SCOPE AND APPLICABILITY

Purpose

- Langmates aims to provide users with a comprehensive platform for seamless language learning.
- It seeks to foster cultural immersion alongside language acquisition, promoting a deeper understanding of diverse cultures.

Scope

- Langmates offers a wide range of language learning resources, tools, and features suitable for users of all proficiency levels and backgrounds.
- It encompasses cultural immersion experiences and community engagement opportunities, providing a holistic approach to language learning and cultural exploration.

Applicability

- Langmates is applicable to individuals seeking to learn new languages and explore diverse cultures, regardless of their proficiency level or background.
- It is suitable for anyone looking for a comprehensive platform that integrates language learning resources, cultural immersion experiences, and community engagement features.

1.4 MODULES

1.User Management Module:

- User registration
- User login
- Profile management
- Progress tracking

2. Reading Module:

Tailored exercises promoting comprehension and vocabulary acquisition through authentic texts in the target language.

3. Listening Module:

Audio-based exercises offering exposure to native accents and colloquial language, improving listening comprehension skills.

4. Speaking Module:

Interactive speaking exercises facilitating pronunciation practice and real-time conversational skills development.

5. Culture Module:

Details like texts and images regarding the culture of the places where the selected language is used.

1.5 MAJOR OUTCOME

The main outcome of Langmates is to empower users with comprehensive language proficiency and cultural understanding. Through tailored modules encompassing reading, writing, listening, and speaking, users gain fluency in their target language while immersing themselves in diverse cultures. The platform fosters real-world application, ensuring users are prepared for authentic communication scenarios. By integrating practical language use across all modules, Langmates equips users with the skills and confidence to navigate global interactions confidently. Ultimately, Langmates aims to break down barriers to communication, promote cross-cultural understanding, and facilitate meaningful connections in an increasingly interconnected world.

1.6 HIGHLIGHT OF THE PROJECT

- Holistic Learning: Langmates covers reading, writing, listening, and speaking for comprehensive language acquisition.
- Cultural Exploration: Users understand diverse perspectives, enhancing their global awareness.
- Engaging Practice: Interactive exercises encourage active participation and skill application.
- Practical Skills: Users are prepared for real-life communication, boosting confidence in various contexts.
- Supportive Community: Langmates connects learners for collaboration, cultural exchange, and encouragement.

1.7 TOOLS USED

- FlutterFlow
- Firebase
- Assembly AI

2. SYSTEM ANALYSIS AND REQUIREMENTS

This chapter describes the system requirements and its analysis. It includes hardware and software requirements as well as the functional and non-functional requirements.

2.1 EXISTING SYSTEM

Duolingo is a popular language learning platform that offers a gamified approach to learning multiple languages. It provides interactive lessons, quizzes, and exercises covering reading, writing, listening, and speaking skills. Users progress through levels and earn rewards for completing lessons, making learning engaging and fun. Duolingo offers a wide range of languages and adapts to users' proficiency levels, providing personalized learning experiences. Additionally, it offers mobile apps for convenient access and syncs progress across devices. Duolingo's user-friendly interface and effective teaching methods have made it one of the most widely used language learning platforms globally.

2.2 LIMITATIONS OF EXISTING SYSTEM

1. Limited Depth: While Duolingo offers a structured approach to language learning, it may lack the depth required for advanced learners or those aiming for professional proficiency.
2. Lack of Cultural Context: Duolingo's focus on language skills may overlook cultural nuances and real-life application, which are crucial aspects of language fluency.
3. Pronunciation Practice: While Duolingo includes speaking exercises, it may not provide sufficient feedback or opportunities for users to refine their pronunciation skills effectively.
4. Overemphasis on Translation: Duolingo's exercises often involve translating sentences, which may not fully reflect real-life language use or conversational fluency.
5. Limited Language Selection: While Duolingo offers a variety of languages, it may not cover less commonly spoken languages or dialects, limiting options for users seeking to learn specific languages.

2.3 PROPOSED SYSTEM

Langmates is a revolutionary language learning platform designed to provide users with a comprehensive and immersive language acquisition experience. Unlike traditional language learning apps, Langmates goes beyond mere vocabulary acquisition, offering a holistic approach that integrates language skills with cultural understanding.

Key Features:

- Comprehensive Language Learning Modules:

Langmates offers tailored modules covering reading, writing, listening, and speaking skills. Each module is designed to provide users with a well-rounded language learning experience, focusing on both linguistic proficiency and cultural immersion.

- Cultural Immersion Integration:

Users can explore diverse cultures alongside language learning, gaining insights into global perspectives, customs, and traditions. Cultural immersion activities are seamlessly integrated into the language learning modules, providing users with a deeper understanding of the languages they are learning.

- Interactive Engagement:

The platform provides interactive exercises, simulations, and games that foster active participation and practical application of language skills. Users engage in real-life scenarios, such as ordering food or navigating public transportation, to practice their language skills in authentic contexts.

- Real-World Readiness:

Emphasis is placed on preparing users for authentic communication situations, ensuring they can confidently navigate various linguistic environments. Users practice speaking, listening, and writing in real-life scenarios, building the skills and confidence needed for effective communication.

- Community Connection:

Langmates fosters a vibrant community of language learners, providing opportunities for collaboration, cultural exchange, and mutual support. Users can connect with fellow learners from around the world, practice language skills together, and share their experiences and insights.

2.4 BENEFITS OF THE PROPOSED SYSTEM

Langmates offers a holistic approach to language learning, integrating language skills with cultural understanding for a more enriching learning experience. Users gain practical language skills that can be immediately applied in real-life situations, preparing them for authentic communication. The platform enhances users' cultural awareness and understanding, fostering appreciation for diverse perspectives and customs. Additionally, Langmates provides a supportive community where users can connect, collaborate, and encourage each other on their language learning journey. With flexible access on multiple devices, users can learn anytime, anywhere, at their own pace, making language learning accessible and convenient for all.

2.5 FEATURES OF THE PROPOSED SYSTEM

- Tailored Learning Modules: Customized modules covering reading, writing, listening, and speaking skills tailored to each user's proficiency level and learning goals.
- Cultural Immersion Integration: Seamless integration of cultural immersion activities alongside language learning to provide users with a deeper understanding of global cultures.
- Interactive Engagement: Engaging exercises, simulations, and games that foster active participation and practical application of language skills in real-life scenarios.
- Real-World Readiness: Emphasis on preparing users for authentic communication situations, ensuring they can confidently navigate various linguistic environments.
- Community Connection: A vibrant community of language learners where users can connect, collaborate, and support each other on their language learning journey.
- Progress Tracking: Tools for users to monitor their language proficiency, track their learning progress, and set personalized learning goals.
- Cross-Platform Accessibility: Accessible on multiple devices, including smartphones, tablets, and computers, allowing users to learn anytime, anywhere.

2.6 SYSTEM REQUIREMENTS SPECIFICATION

2.6.1 User Characteristics

Langmates users are diverse, ranging from language enthusiasts and cultural explorers to tech-savvy individuals seeking convenient online learning. They value community engagement, collaboration, and flexible study options. Motivated by personal or professional goals, they appreciate the platform's tailored approach to language acquisition and cultural immersion. With a desire to learn and improve, they embrace Langmates as a dynamic and supportive environment where they can connect with like-minded learners, explore diverse cultures, and enhance their language skills at their own pace.

2.6.2 SOFTWARE AND HARDWARE REQUIREMENTS

Software Requirements

- FlutterFlow
- Firebase
- Assembly AI

Hardware Requirements

- Processor - 1.9 gigahertz (GHz) x86- or x64-bit dual core processor with SSE2 instruction set
- Graphic card - Not required
- Disk Capacity - 1-2 Gb of Disk Capacity
- RAM - 2GB of Memory

2.6.3 CONSTRAINTS

Langmates confront constraints such as technological dependencies, requiring internet access and compatible devices. Language options may be limited, potentially omitting less commonly spoken languages. Cultural representation may favor mainstream cultures, overlooking smaller or marginalized communities. Users' varied learning paces necessitate flexible options. Resource availability like textbooks may vary by location. Despite these challenges, Langmates aims to offer an inclusive platform for

language learning and cultural exploration, continually striving to enhance the user experience.

2.6.4 FUNCTIONAL REQUIREMENTS

Table 2.1 Functional Requirements

Requirement Id	Requirement	Description
M1_FR1	User Registration and Authentication	Users should be able to create accounts securely and log in using credentials.
M1_FR2	Language Learning Modules	Each module contains exercises tailored to users' proficiency levels and learning goals.
M2_FR1	Cultural Immersion Integration	Users have access to cultural insights and resources relevant to the languages they are learning.
M2_FR2	Interactive Engagement	Engaging exercises, simulations, and games foster active participation and practical application of language skills.
M2_FR3	Progress Tracking	Tools for users to monitor their language proficiency, track their learning progress, and set personalized learning goals.
M3_FR1	User Profile Management	Users should be able to update their profiles, including language preferences and learning goals.
M4_FR1	Creating	The users can create their own NFT, such as an illustration, photograph, 2D or 3D design, and so on. Anything can be created and posted on your profile. You can set a price for your art and it is ready to be sold.
M4_FR2	Grammar and Syntax Support	Tools and resources to assist users in

		understanding and practicing grammar and syntax rules.
M4_FR3	Progress Reports	Regular reports on users' learning progress, highlighting areas of improvement and achievements.
M5_FR1	Language Variety	Users should have access to languages from different language families and geographic regions.
M6_FR1	Error Correction Mechanism	The system should provide feedback on errors made by users and offer suggestions for correction.
M6_FR2	Content Moderation	In the settings section the user can make settings related to his profile.
M7_FR1	Integration with External Resources	Compatibility with external resources such as language dictionaries, textbooks, and online courses for enhanced learning opportunities.

2.6.5 NON-FUNCTIONAL REQUIREMENTS

Table 2.2 Non-Functional Requirements

Requirement ID	Requirement	Description
NF_R1	Performance	● As it's a mobile application, the network, hardware and other related infrastructure plays a vital role in determining the application performance.

		● Data compression approach has been applied to reduce the network burden. ● Number of navigations and touch are reduced to the minimum. ● Basic and simple color and graphics settings are implemented for high performance.
NF_R2	Safety/Security	● It is the state of being "safe", the condition of being protected against physical, social, spiritual, financial, political, emotional, occupational, psychological, educational or other types of consequences of failure, damage, error, accidents, harm or any other event which could be considered non-desirable. ● This can take the form of being protected from the event or from exposure to something that causes health or economical losses. It can include protection of people or of possessions.
NF_R3	Quality requirements	**1. Reliability** ● The system provides storage of all databases on redundant computers with automatic switchover. ● The reliability of the overall program depends on the reliability of the separate components. ● The main pillar of reliability of the system is the backup of the database

		which is continuously maintained and updated to reflect the most recent changes. **2. Maintenance** ● A commercial database is used for maintaining the database and the application server takes care of the site. ● Also, the software design is being done with modularity in mind so that maintenance can be done efficiently. **3. Usability** ● As the users of the system requirements are different for Everyone, the level of functionalities should require a better understanding of the system. ● The interface for each type of user kept very simple and complete for ease of use. Manuals, demos or the documents made available, simple, clear and definite set of interfaces makes the context easy to understand and use.

3. SYSTEM DESIGN

This document outlines the core components of the Langmates language learning platform, including its modules, system architecture, data flow diagram, ER diagram, UI samples, and database tables. These elements are designed to provide a comprehensive understanding of how the Langmates application operates and how users interact with its features.

3.1 SYSTEM ARCHITECTURE

Langmates, a mobile language learning app developed using FlutterFlow with Firebase backend, employs a client-server architecture. The client side comprises the mobile app running on users' devices, facilitating interactions with the platform. It communicates with Firebase backend services via HTTPS requests, handling user input and content display. Firebase serves as the backend infrastructure, providing Firestore for data storage, Firebase Authentication for user management, and Cloud Functions for server-side logic execution. Communication occurs over HTTPS for secure data transmission, with real-time updates supported through Firebase's capabilities.

3.2 MODULE DESIGN

- **Reading Module -** It aims to improve reading comprehension and vocabulary by engaging users with authentic texts in the target language. Exercises include comprehension questions, vocabulary quizzes, and reading passages.
- **Listening Module -** focuses on improving listening comprehension and pronunciation. Users listen to audio clips, podcasts, or dialogues and engage in exercises like comprehension questions, dictation, and shadowing for pronunciation practice.

- **Speaking Module -** This aims to develop oral communication and conversational skills. Users participate in speaking exercises such as role-plays, dialogues, and recorded conversations. Feedback on pronunciation, intonation, and fluency is provided through speech recognition technology.
- **Writing Module** - users practice writing sentences, paragraphs, and essays to enhance their writing skills and grammar proficiency. Activities may include writing prompts and grammar exercises with feedback provided by the system.
- **Cultural Immersion Module -** The Cultural Immersion Module in Langmates offers insights into diverse global cultures alongside language learning. Through engaging content and activities, users explore traditions, customs, and social norms, fostering cross-cultural understanding. Interactive exercises and discussions enrich cultural awareness, preparing users for meaningful interactions in diverse cultural settings.

3.3 DATA FLOW DIAGRAM

3.3.1 DFD Level 0

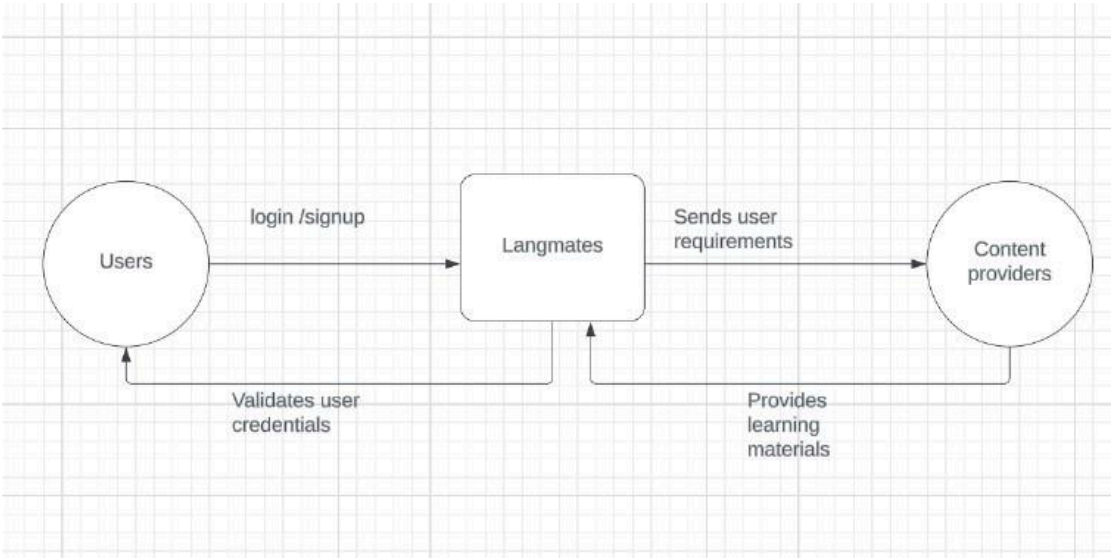

Fig 3.3 DFD level - 0 diagram

3.3.2 DFD Level 1

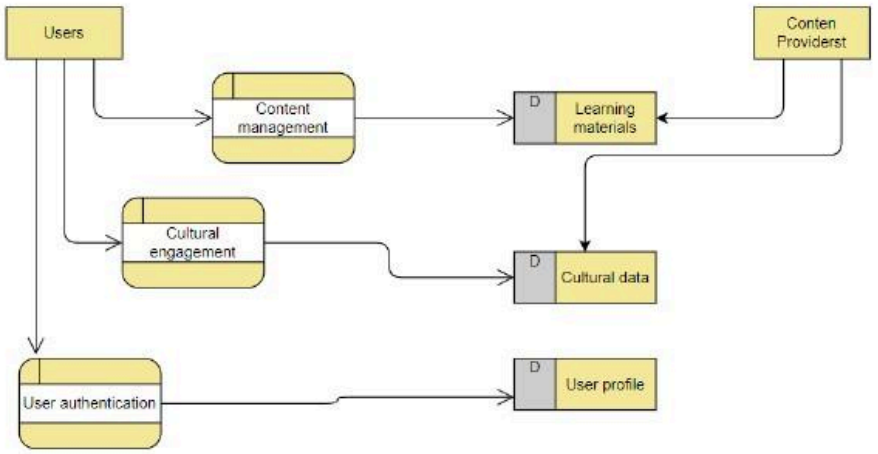

Fig 3.4 DFD level - 1 Diagram

3.4 ER DIAGRAM

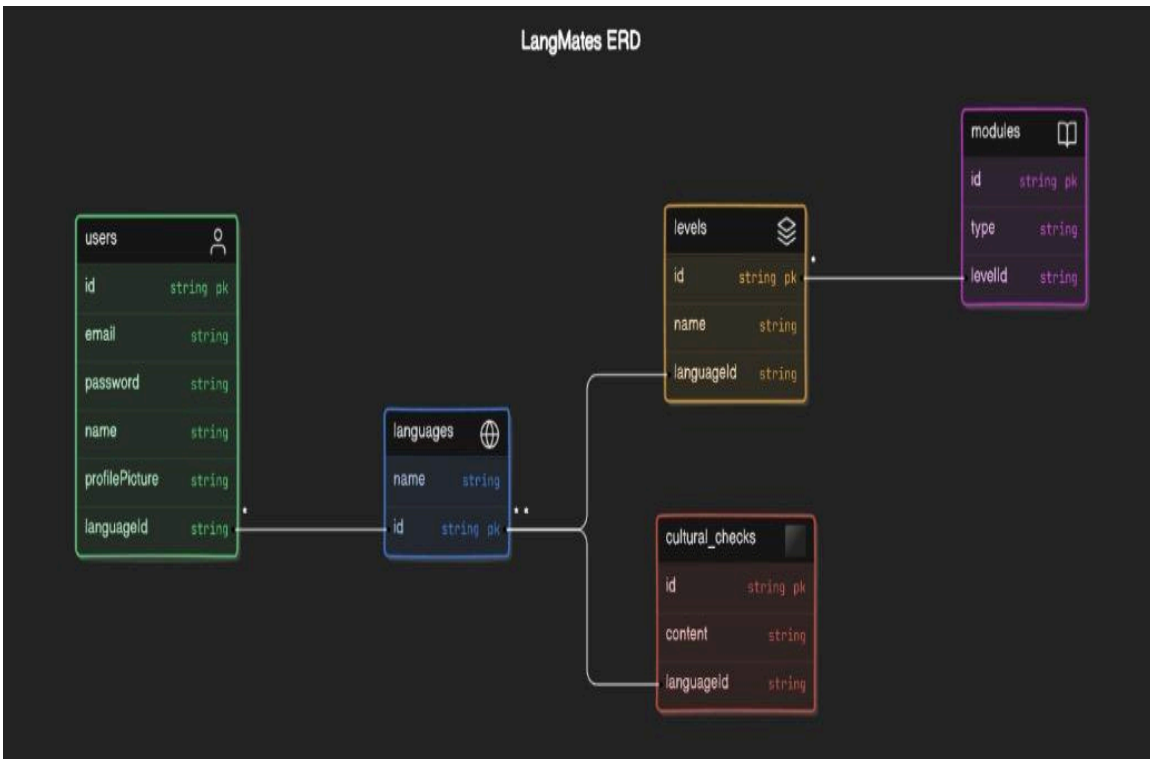

Fig 3.6 ER diagram

3.5 DATABASE DESIGN

3.5.1 Table Design

Table 3.1 User table

Sno.	Attributes	Data type	Description	Constraint
1	UserID	Integer	Unique identifier for each user.	Primary Key, Auto Increment
2	Username	String	Username chosen by the user.	Unique, Not Null
3	Email	String	Email address of the user	Unique, Not Null

4	Password	Varchar(25)	Hashed password for user authentication.	Not null

Table 3.2 Languages Table

Sno.	Attributes	Data type	Description	Constraint
1	LanguageID	Integer	Unique identifier for each language	Primary Key, Auto Increment
2	Language	String	Name of the language	Unique, Not Null
3	NativeName	String	Native name of the language	Not Null

Table 3.3 Modules Table

Sno.	Attributes	Data type	Description	Constraint
1	ModuleId	Integer	Unique identifier for each module	Primary Key, Auto Increment
2	ModuleName	String	Name of the module	Unique, Not Null

Table 3.4 Levels Table

Sno.	Attributes	Data type	Description	Constraint
1	LevelId	Integer	Unique identifier for each level	Primary Key, Auto Increment
2	LevelName	String	Name of the level	Unique, Not Null
3	Description	String	Description of the level	Not Null

Table 3.5 Cultural Check Table

Sno.	Attributes	Data type	Description	Constraint
1	CheckId	Integer	Unique identifier for each cultural check	Primary Key, Auto Increment
2	UserId	Integer	Foreign key referencing the user who performed the check	Foreign Key, Not Null
3	LanguageId	Integer	Foreign key referencing the language checked	Foreign Key, Not Null
4	DateChecked	DateTime	Date and time when the cultural check was performed	Not Null
5	Result	Boolean	Result of the cultural check (true/false)	Not Null

3.5.2 Data Integrity and Constraints

In Langmates, the Users table ensures data integrity by uniquely identifying each user record with the UserID attribute. This attribute serves as the primary key, guaranteeing that each user is uniquely identifiable within the system. Additionally, constraints such as unique usernames and email addresses prevent duplication and ensure the uniqueness of user accounts. Passwords are securely stored and hashed for authentication purposes, while the JoinDate attribute records the date and time of user registration. Optional fields like LastLogin may track additional user activity.

The Languages table maintains data integrity by uniquely identifying each language record with the LanguageID attribute, serving as the primary key. Standardized ISO language codes provide interoperability and facilitate language identification. Constraints such as unique language names and codes ensure the uniqueness of language entries. The NativeName attribute stores the native name of each language, contributing to cultural accuracy and localization efforts.

Modules in Langmates are designed to enhance specific aspects of language learning. Each module is uniquely identified by the ModuleID attribute, ensuring data integrity. Constraints such as unique module names enforce the uniqueness of module entries, preventing duplicates and ensuring clarity in module identification.

Similarly, the Levels table maintains data integrity by uniquely identifying each level record with the LevelID attribute, serving as the primary key. Constraints such as unique level names and mandatory level descriptions contribute to data accuracy and clarity in level representation.

The Cultural Check table ensures data integrity by uniquely identifying each cultural check record with the CheckID attribute, serving as the primary key. Foreign key constraints maintain relationships with the Users and Languages tables, ensuring referential integrity. The DateChecked attribute records the date and time of each cultural check, providing valuable information for analysis and tracking cultural immersion activities.

Users Table:

Data Integrity: Ensures that each user record is uniquely identified by the UserID. User authentication is maintained through secure storage and hashing of passwords.

Constraints:

UserID (Primary Key): Ensures each user record is uniquely identifiable.

Username (Unique): Ensures each username is unique, preventing duplication.

Email (Unique): Ensures each email address is unique, preventing multiple accounts with the same email.

Password (Not Null): Ensures all users have a password for authentication.

JoinDate (Not Null): Ensures the registration date and time are recorded for each user.

LastLogin: Optional field to track the last login date and time.

3.5.3 Data Dictionary

Table 3.7 User Table

Sl. No.	Attribute name	Min	Max	Default
1.	User_ID			auto-generated

Table 3.8 Languages Table

Sl. No.	Attribute name	Min	Max	Default
1.	LanguageId			auto-generated

Table 3.9 Modules Table

Sl. No.	Attribute name	Min	Max	Default
1.	ModuleId			auto-generated

Table 3.10 Levels Table

Sl. No.	Attribute name	Min	Max	Default
1.	LevelId			Not null

Table 3.11 ACultural Check Table

Sl. No.	Attribute name	Min	Max	Default
1.	CheckId			Unique
2.	UserId			Not null
3.	LanguageId			Not null

3.6 INTERFACE AND PROCEDURAL DESIGN

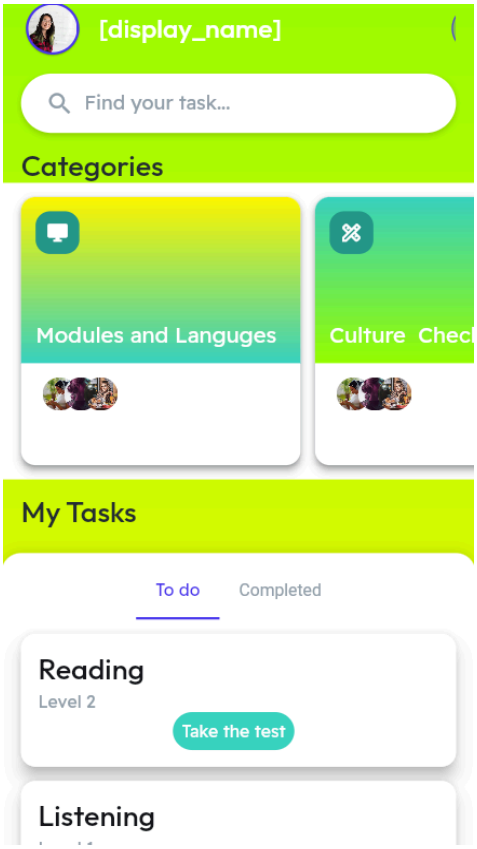

Fig 3.7 Home page

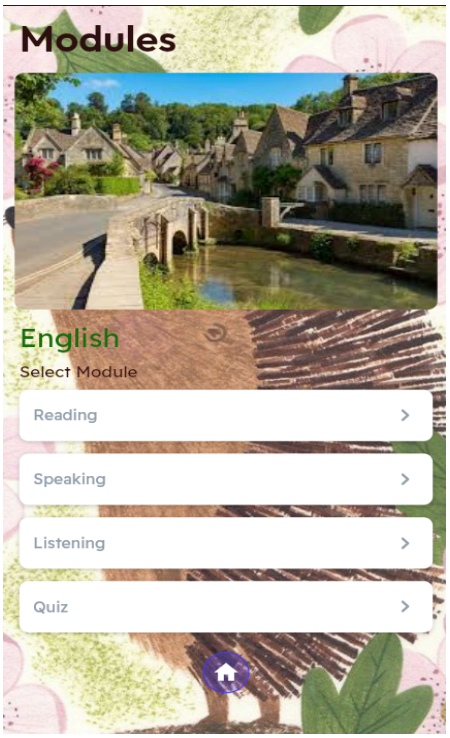

Fig 3.8 Modules page

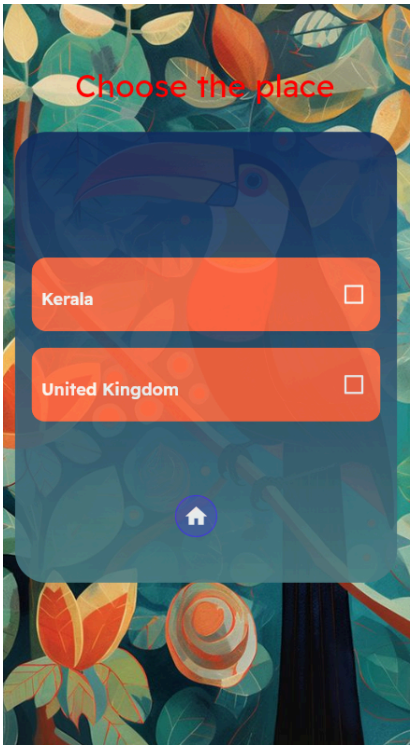

Fig 3.9 Culture module

4. IMPLEMENTATION

This chapter consists of the coding standards used, the coding details, the code as well as the output of the desired implementation.

4.1 CODING STANDARDS

A coding standard gives a uniform appearance to the codes written by different engineers. It improves readability, and maintainability of the code and it reduces complexity also. It helps in code reuse and helps to detect errors easily. It promotes sound programming practices and increases efficiency of the programmers.

The following coding standards are followed:

- **Indenting** - Used an indent of 4 spaces and didn't use any tab because different computers use different setting for tab.
- **Variable Names** - Used all lowercase letters and used '_' as the word separator.
- **Control Structures** - These include if, for, while, switch, etc. Control statements have one space between the control keywords, to distinguish them from function calls.
- **Function Calls** - Functions have been called with no spaces between the function name, the opening parenthesis, and the first parameter; spaces between commas and each parameter, and no space between the last parameter, the closing parenthesis, and the semicolon.
- **Docstrings** - There are both single and multi-line docstrings that can be used in Python. However, the single line comment fits in one-line, triple quotes are used in both cases. These are used to define a particular program or define a particular function.
- **Comments** - There are also various types and conditions that are followed that can be of great help from programs and user's point of view. Comments should form complete sentences. If a comment is a full sentence, its first word should be capitalized, unless it is an identifier that begins with a lowercase

letter. In short comments, the period at the end can be omitted. In block comments, there are more than one paragraphs and each sentence must end with a period. Block comments and inline comments can be written followed by a single '#'.

- **Alignment of Declaration Blocks** - Block of declarations are be aligned.
- **One Statement Per Line** - There is only one statement per line unless the statements are very closely related.

4.2 CODING DETAILS

Audio Recording

```
dependencies:
  flutter:
    sdk: flutter
  audioplayers: ^0.20.1
```

```
import 'package:flutter/material.dart';
import 'package:audioplayers/audioplayers.dart';
import 'package:path_provider/path_provider.dart';
import 'dart:io';

class AudioRecorderScreen extends StatefulWidget {
  @override
  _AudioRecorderScreenState createState() => _AudioRecorderScreenState();
}

class _AudioRecorderScreenState extends State<AudioRecorderScreen> {
  AudioPlayer _audioPlayer = AudioPlayer();
  String _filePath;
```

```
 Recording _recording = Recording();


 Future<void> startRecording() async {
  try {
   if (!await AudioRecorder.hasPermissions) {
    return;
   }
   String path = '${(await
getTemporaryDirectory()).path}/recording_${DateTime.now()}.m4a';
   await AudioRecorder.start(path: path, audioOutputFormat:
AudioOutputFormat.AAC);

   bool isRecording = await AudioRecorder.isRecording;
   setState(() {
    _filePath = path;
    _recording = Recording(path: _filePath, isRecording: isRecording);
   });
  } catch (e) {
   print(e);
  }
 }

 Future<void> stopRecording() async {
  try {
   Recording recording = await AudioRecorder.stop();
   setState(() {
    _recording = recording;
   });
  } catch (e) {
   print(e);
  }
 }
```

```
@override
Widget build(BuildContext context) {
 return Scaffold(
  appBar: AppBar(
   title: Text('Audio Recorder'),
  ),
  body: Center(
   child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: <Widget>[
     if (_recording.isRecording)
      Text('Recording...', style: TextStyle(fontSize: 20.0)),
     SizedBox(height: 20.0),
     RaisedButton(
      onPressed: () {
       if (!_recording.isRecording) {

       startRecording();
       } else {
        stopRecording();
       }
      },
      child: Text(_recording.isRecording ? 'Stop Recording' : 'Start Recording'),
     ),
     if (_filePath != null)
      SizedBox(height: 20.0),
      Text('File Path: $_filePath'),
      SizedBox(height: 20.0),
      RaisedButton(
       onPressed: () {
        _audioPlayer.play(_filePath, isLocal: true);
       },
       child: Text('Play Recording'),
      ),
```

```
          ],
        ),
      ),
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'audio_recorder_screen.dart'; // Import the audio recorder widget

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(

     title: 'Audio Recorder Demo',
      home: AudioRecorderScreen(), // Display the audio recorder screen
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:flutter_sound/flutter_sound.dart';
import 'package:path_provider/path_provider.dart';
import 'dart:io';

class AudioRecorderScreen extends StatefulWidget {
```

```
 @override
 _AudioRecorderScreenState createState() => _AudioRecorderScreenState();
}

class _AudioRecorderScreenState extends State<AudioRecorderScreen> {
 FlutterSoundRecorder _audioRecorder = FlutterSoundRecorder();
 bool _isRecording = false;
 String _filePath;

 @override
 void initState() {
  super.initState();
  _initRecorder();
 }

 Future<void> _initRecorder() async {
  await _audioRecorder.openAudioSession();
 }

 Future<void> _startRecording() async {
  try {
   String path = await _getFilePath();
   await _audioRecorder.startRecorder(
    toFile: path,
    codec: Codec.aacADTS,
   );
   setState(() {
    _isRecording = true;
    _filePath = path;
   });
  } catch (e) {
   print('Error starting recording: $e');
  }
 }
```

```
 Future<String> _getFilePath() async {
   Directory appDir = await getApplicationDocumentsDirectory();
   String filePath =
'${appDir.path}/recording_${DateTime.now().millisecondsSinceEpoch}.aac';
   return filePath;
 }

 Future<void> _stopRecording() async {
   try {
     await _audioRecorder.stopRecorder();
     setState(() {
       _isRecording = false;
     });
   } catch (e) {
     print('Error stopping recording: $e');
   }
 }

 @override
 void dispose() {
   _audioRecorder.closeAudioSession();
   super.dispose();
 }

 @override
 Widget build(BuildContext context) {



   return Scaffold(
     appBar: AppBar(
       title: Text('Audio Recorder'),
     ),
```

```
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            _isRecording
                ? Text('Recording...', style: TextStyle(fontSize: 20.0))
                : Text('Tap to start recording'),
            SizedBox(height: 20.0),
            RaisedButton(
              onPressed: () {
                if (_isRecording) {
                  _stopRecording();
                } else {
                  _startRecording();
                }
              },
              child: Text(_isRecording ? 'Stop Recording' : 'Start Recording'),
            ),
            SizedBox(height: 20.0),
            if (_filePath != null)
              Text('File Path: $_filePath'),
          ],
        ),
      ),
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'audio_recorder.dart'; // Import the audio recorder widget
```

```
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Audio Recorder Demo',
      home: AudioRecorderScreen(), // Display the audio recorder screen
    );
  }
}
```

```
function MyAssets() {
    const collectionCtx = useContext(CollectionContext);
    const marketplaceCtx = useContext(MarketplaceContext);
    const web3Ctx = useContext(Web3Context);
    const [currentAddress, setCurrentAddress] = useState();

    useEffect(() => {
        document.title = 'My assets';
        setCurrentAddress(web3Ctx.account);
    }, [web3Ctx.account]);

    // RETURN ITEMS TEMPLATE
    return (
        <>
            {marketplaceCtx.mktIsLoading ? <FullScreenLoader heading='loading' /> :
null}
            <PageBanner heading={'My Assets'} />

            {/* NFT ITEMS */}
            <section className='py-5'>
```

```
            <div className='container'>
               <header className='mb-5'>
                  <div className='row'>
                     <div className='col-lg-6'>
                        <h2>Your available NFTs</h2>
                        <p className='text-muted text-sm mb-0'>
                           The NFTs you own are listed here!
                        </p>
                     </div>
                  </div>
               </header>

               <div className='row gy-4'>
                  {collectionCtx.collection
                     .filter((item) => item.owner === currentAddress)
                     .map((NFT, key) => {
                        const index = marketplaceCtx.offers
                           ? marketplaceCtx.offers.findIndex((offer) => offer.id ===
NFT.id)
                           : -1;
                        const owner = index === -1 ? NFT.owner :
marketplaceCtx.offers[index].user;
                        const price =
                           index !== -1 ?
formatPrice(marketplaceCtx.offers[index].price).toFixed(2) : null;

                        return (
                           <div className={`col-xl-3 col-lg-4 col-md-6 mix`}
key={key}>
                              <NftItem {...NFT} index={index} owner={owner}
price={price} nftKey={key} />
                           </div>
                        );
                     })}

                  {collectionCtx.collection.filter((item) => item.owner ===
currentAddress).length === 0 ? (
                     <div className='col-9'>
                        <NoDataAlert
                           heading="You don't have any assets that lays under your
ownership, if you put assets for
                              sale you might see them at the section below."
                           subheading='Please note that when you put item for sale the
ownership goes to
                              marketplace and the marketplace will sell it for you.'
                        />
                     </div>
                  ) : null}
               </div>
            </div>
```

```
        </section>



const renderSellers = currentModules
      .sort((a, b) => (a.value < b.value ? 1 : -1))
      .map((seller, index) => {
        return (
          <div className='col-xl-3 col-lg-4 col-md-6' key={index}>
            <div className='card bd-3 card-hover-minimal position-relative'>
              <div className='card-body'>
                <a
                  className='d-flex align-items-center text-reset
text-decoration-none stretched-link'
                  href={configEtherScanUrl(web3Ctx.networkId, seller.address)}
                  rel='noreferrer noopener'
                  target='_blank'
                >

                  <div className='position-relative'>
                    <div className='ms-3' style={{ width: '50px', height: '50px'
}}>
                      <Jazzicon address={seller.address} />
                    </div>
                    <div className='author-img-badge bg-primary text-white'>
                      <i className='las la-check-double la-xs'></i>
                    </div>
                  </div>
                  <div className='ms-3'>
                    <h3 className='h6 mb-1 text-capitalize'>
                      {uniqueNamesGenerator({
                        dictionaries: [starWars],
                      }).replace('_', ' ')}
                      {web3Ctx.account === seller.address ? (
                        <span className='seller-badge ms-2'>You</span>
                      ) : null}
                    </h3>
                    <p className='text-sm text-primary mb-0'>
                      {formatPrice(seller.value).toFixed(2)} <span
className='text-muted'>ETH</span>
                    </p>
                  </div>
                </a>
              </div>
            </div>
          </div>
        );
      });
```

```
function StatelessWidget({ gridWidth }) {
   const [state, handleSubmit] = useForm('xnqwjgvp');
   useEffect(() => {
      // Fetch all the forms we want to apply custom Bootstrap validation styles to
      var forms = document.querySelectorAll('.needs-validation');

      // Loop over them and prevent submission
      Array.prototype.slice.call(forms).forEach(function (form) {
         form.addEventListener(
            'submit',
            function (event) {
               if (!form.checkValidity()) {
                  event.preventDefault();
                  event.stopPropagation();
               }

               form.classList.add('was-validated');
            },
            false
         );
      });
   }, []);
   if (state.succeeded) {
      return (
         <div className={`${gridWidth} text-center`}>
            <p className='mb-0 fw-bold mt-5 mb-0'>
               <i
                  className='las la-grin-beam'
                  style={{ fontSize: '10rem', textShadow: '2px 4px rgba(0, 0, 0, 0.4)' }}
               ></i>
            </p>

            <h1 className='h2'>Thanks for contacting us.</h1>
            <p className='text-muted'>We'll reply back as soon as possible.</p>
            <Link to='/' className='btn btn-gradient-primary'>
               Return Home
            </Link>
         </div>
      );
   }
class AudioRecorderScreen extends StatefulWidget {

  @override

  _AudioRecorderScreenState createState() => _AudioRecorderScreenState();

}
```

```
class _AudioRecorderScreenState extends State<AudioRecorderScreen> {
  FlutterSoundRecorder _audioRecorder = FlutterSoundRecorder();
  bool _isRecording = false;
  String _filePath;
  }
  Future<void> _initRecorder() async {
    await _audioRecorder.openAudioSession();
  }

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Audio Recorder Demo',
      home: AudioRecorderScreen(), // Display the audio recorder screen
    );
  }
}
```

4.3 SCREENSHOTS

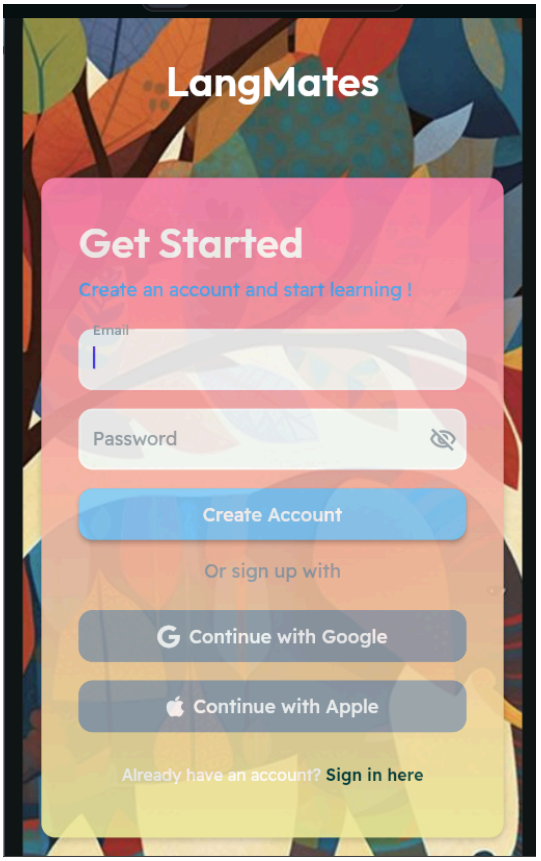

Fig 4.1 Sign in Page

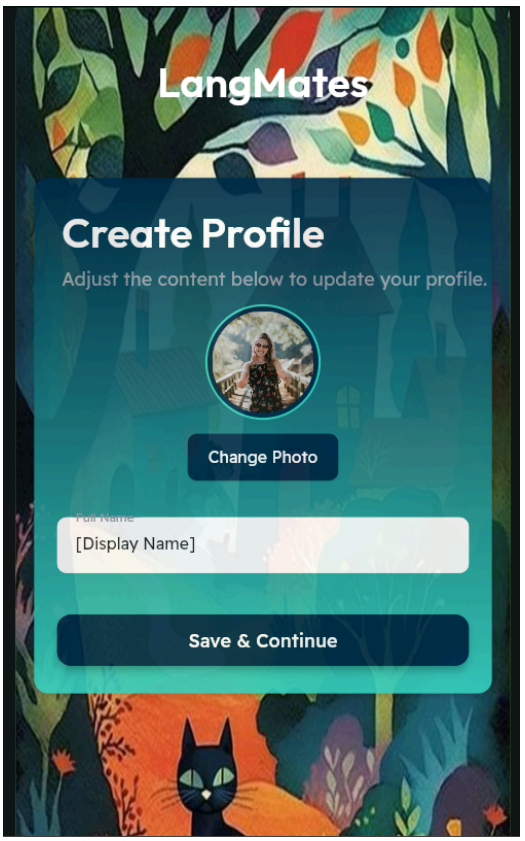

Fig 4.2 Create Profile

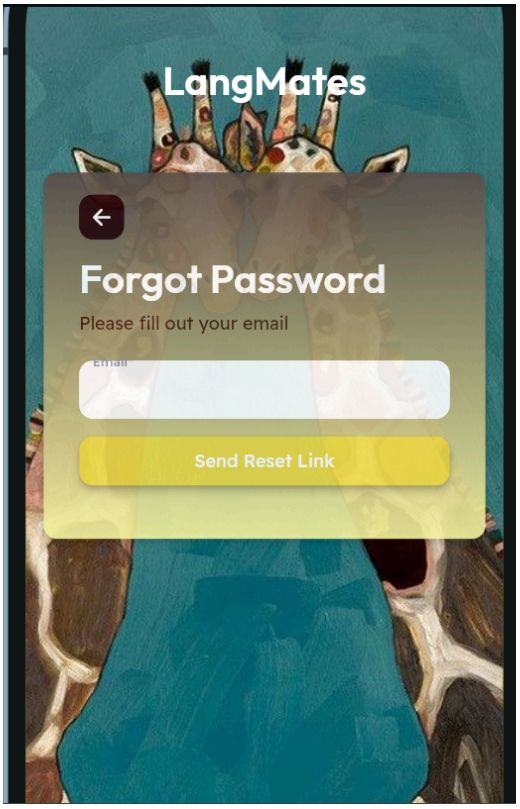

Fig 4.3 Forgot Password

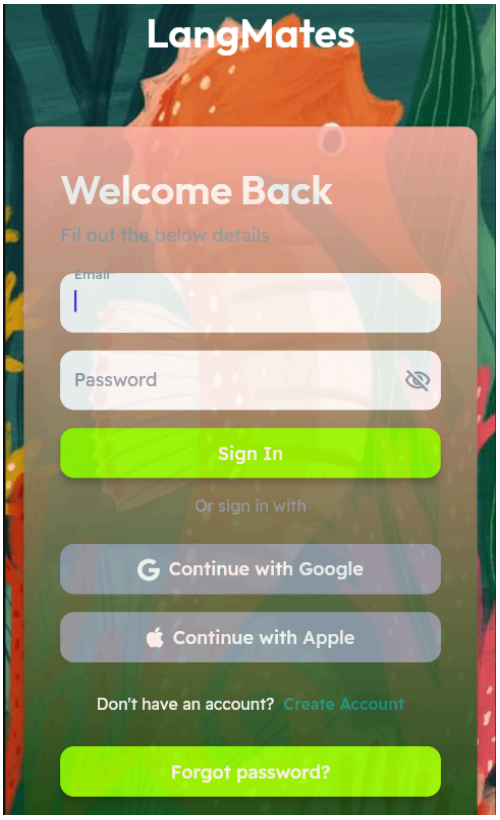

Fig 4.4 User Login

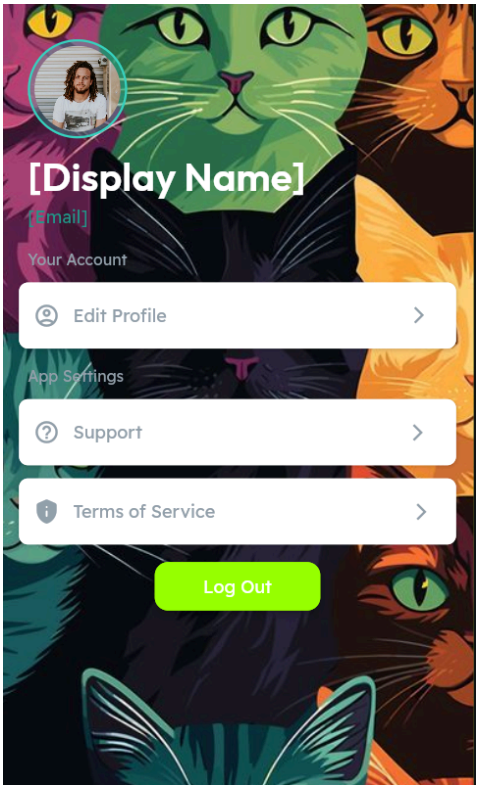

Fig 4.5 User Profile Page

Fig 4.6 Language Selection Page

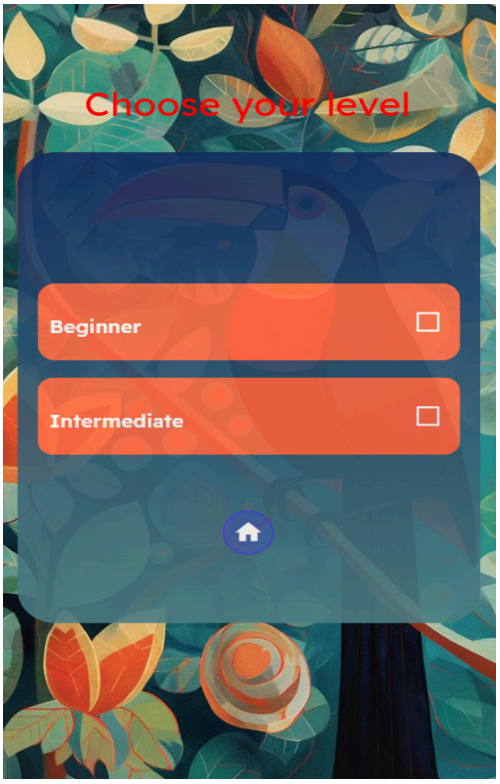

Fig 4.7 Level Selection Page

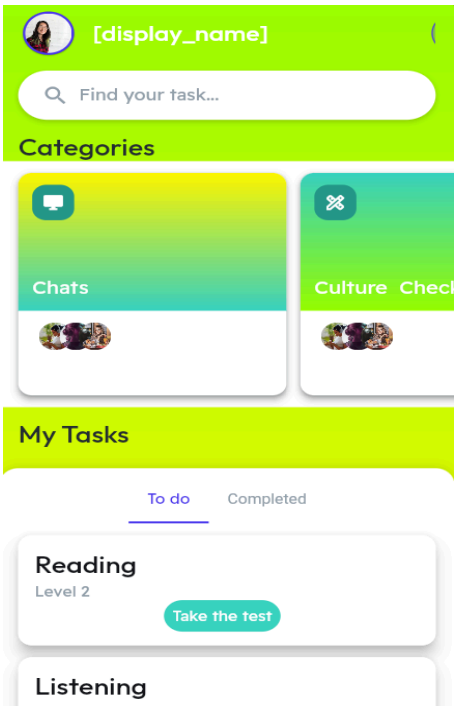

Fig 4.8 User Dashboard

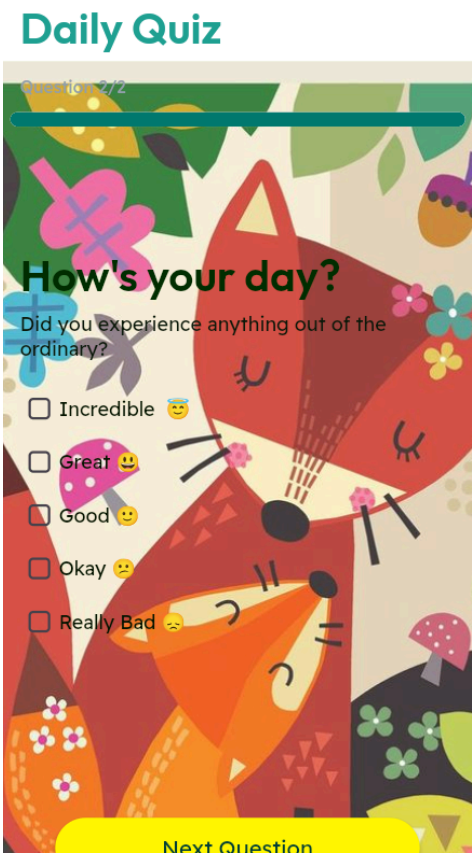

Fig 4.9 Daily Quiz Page

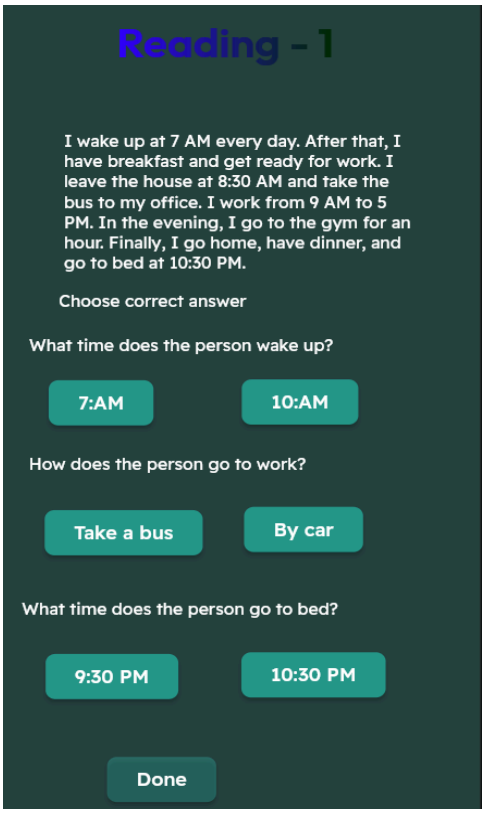

Fig 4.10 English Reading Page

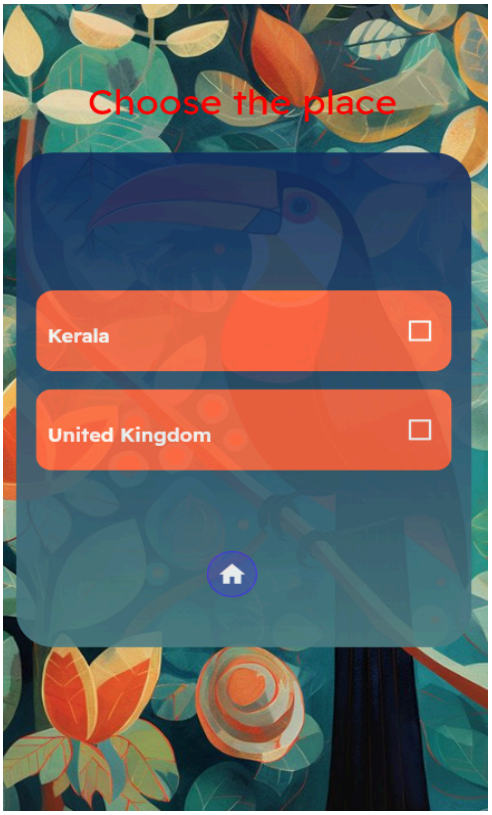

Fig 4.11 Culture Choosing Page

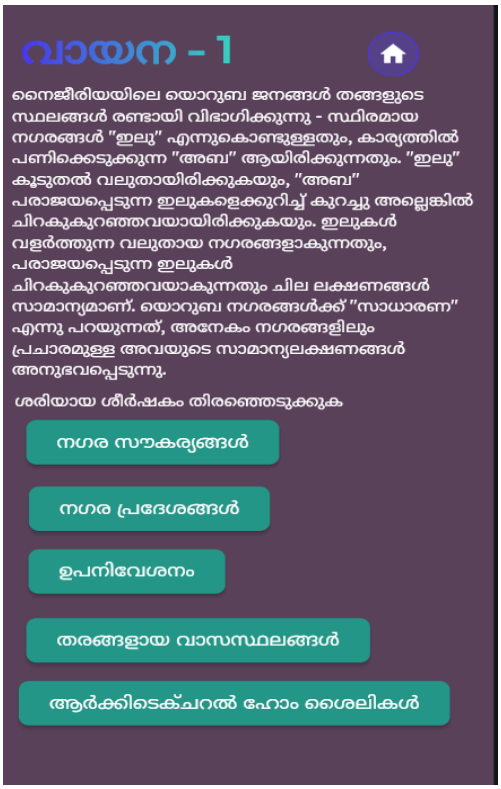

Fig 4.12 Malayalam Reading Page

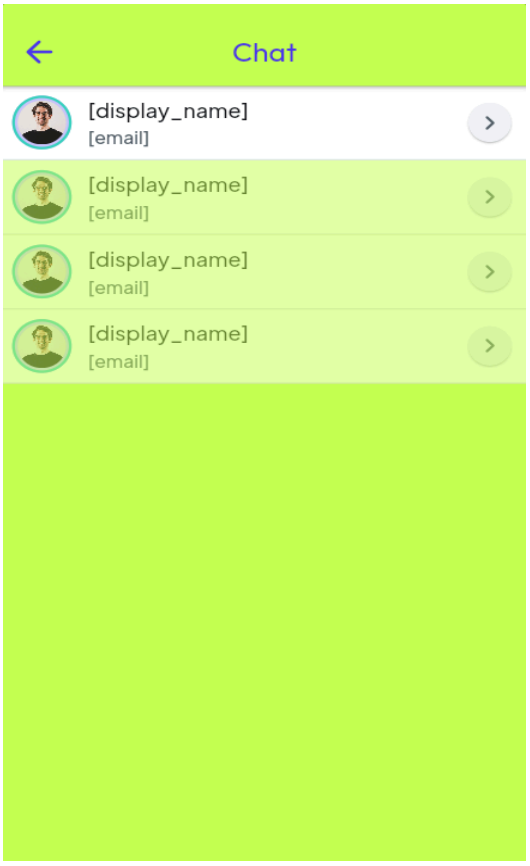

Fig 4.13 Invite User Page

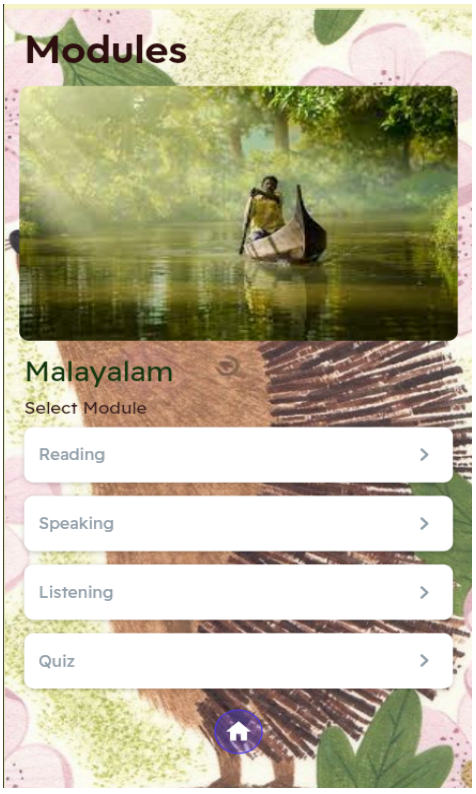

Fig 4.14 Malayalam Language Modules

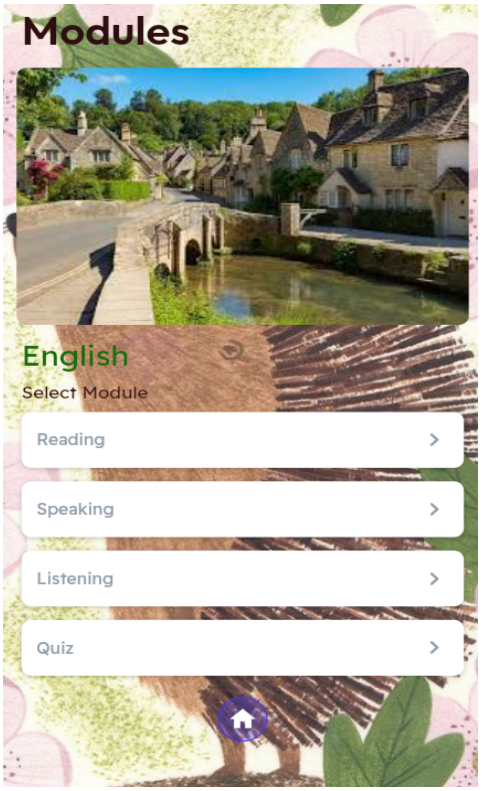

Fig 4.15 English Language Modules

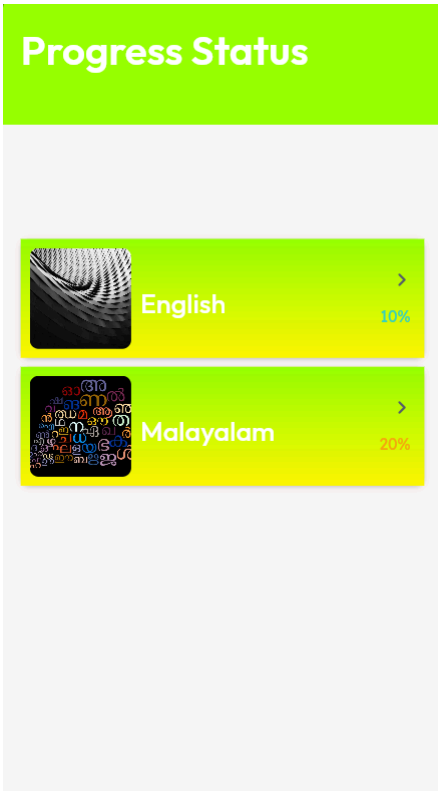

Fig 4.16 Progress Status

Fig 4.17 Kerala Culture Page

5. TESTING

5.1 TEST STRATEGIES

5.1.1 System Testing

System testing is a critical element of quality assurance and represents the ultimate review of analysis, design and coding. Test case design focuses on set of techniques for the creation of test because that meet overall testing objectives. When a system is developed it is hoped that it performs properly. The main purpose of testing an information system Is to find the errors and correct them. The scope of system testing should include both manual and computerized operations. System testing is a comprehensive evaluation of the programs, manual procedures, computer operations and controls.

System testing is the process of checking whether the developed system is working according to the objective and requirement. All testing is to be conducted in accordance to the test conditions specified earlier. This will ensure that the test coverage meets the requirements and that testing is done in a systematic manner.

The process of analysis of the software item to detect the differences between existing or required condition and evaluate the features of the software items. The thorough testing of the system before release of the software becomes devoid of bugs and uses minimum space requirements as well as minimum time to perform. The test cases were selected beforehand with expected results recorded for comparison. The selection of the test cases is done vide “White Box Testing” technique to check software requirement fulfilment with intension of finding maximum number of errors with minimum effort and time. Although test cases are a design by considering the cyclomatic complexity, conditional test, still the software code is not in its optional form ,as all other possible alternative parts in the software are not considered .At the integration level, the software will be passing to the third party tests which would further enhance the software optimality and efficiency.

5.1.2 Test Data Implementation

The quality and standardization of the software /application package depends truly on the various predefined testing norms and on the performance of the software over

those norms. There are various standards existing in the software industry the engineered end product

Strives to achieve viz. ISO 9002 SEI CMM Level5 etc. These standards are achieved only when the concerned software fulfils the tests as per the respective norms predefined in them vide the various test cases and parameters using the CASE topologies. Generally, Software is tested both on a stand-alone mode as well after integrating all the modules in the system vide deferent available testing methods/norms.

The following Flow Graph methodology was used while testing the software:

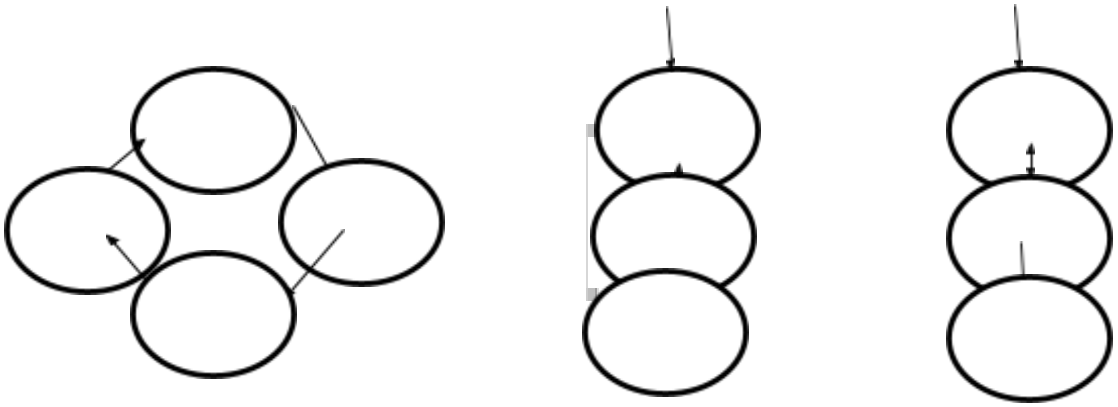

Fig 5.1 Flow graph

Here each circle represents one or more non branching procedural language or source code statements in Flow Graph. While performing Condition Testing Domain Testing methodology was selected. While performing Loop testing simple loops, concatenated loops, nested and unstructured loops were tested thoroughly.

5.1.3 Test Characteristics

1. A good test has a high probability of finding an error.
2. A good test is not redundant.
3. A good test should be “best of breed “.
4. A good test should be neither too simple nor too complex.

5.1.4 Black box Testing

The method of Black Box Testing is used by the software engineer to derive the required results of the test cases:

1. Black Box Testing alludes to test that are conducted at the software interface.
2. A Black Box Test examines some fundamental aspect of a system with little regard for the internal logic structure of the software.
3. A limited number of important logical paths can be selected and exercised.
4. Important data structure can be probed for validity.

Black box testing was performed to find errors in the following categories:

- Incorrect or missing functions
- Graphics errors.
- Errors in data in string format.
- Error in data in integer format.
- File error.
- Memory access error
- Variable error.
- Performance error.

5.1.5 White box Testing

White Box Testing is sometimes called as Glass Box Testing. Using White Box Testing methods, the software engineer can derive the following test cases:

1. Guarantee that all independent paths within a module have been exercised at least once.
2. Exercise all logical decisions on their true and false sides.
3. Execute all loops at their boundaries and within their operational bounds.
4. Exercise internal data structures to ensure the validity.

In white Box Testing efforts were made to handle the following:

- Number of input parameters equal to number of arguments.
- Parameters and arguments attributes match.
- Number of arguments transmitted is called modules equal to attributes of parameter.
- Unit system of argument transmitted is called modules equal unit system of parameter.
- Number of attributes and order of arguments to build in functions correct.
- Any references to parameters not associated to build in functions correct.

- Input only arguments altered.
- Global variable definition consistent across module.
- Files attributes correct.
- Format specification matches I/O specification.
- Files opened before use.
- File closed while working.
- I/O errors handled.
- Any textual errors in output information.

Unit Testing

The unit testing is performed to test the validity of the individual units. This is done in the coding phase with the interactive testing. Thus, it itself constitutes a majority of functionality test for each logical unit.

Integrity Testing

When all the development of all the units or modules is completed and integrated, the integrity test phase is started. In this phase, the interface between the modules are tested. This phase basically verifies whether inter module exchange of information and events are as per required system behavior.

Validation Testing

Tests were performed to find conformity with the requirements. Plans and procedures were designed to ensure that all the functional requirements are satisfied. The software was alpha-tested. There are two goals in preparing test plans. Firstly, a properly detailed test plan demonstrates that the program specifications are understood completely. Secondly, the test plan is used during program testing to prove the correctness of the program.

5.2 TEST CASES

Table 5.1 Test Cases

Sl. No.	Module Name	Test Case No	Test Case Description	Expected Result
1	Reading Module	TC1	User opens a reading exercise in the beginner level module.	The exercise contains simple sentences and basic vocabulary suitable for beginner learners.
2	Writing Module	TC2	User completes a writing prompt requiring them to write a short paragraph in the target language.	The system provides feedback on the user's writing accuracy, grammar, and vocabulary usage.
3	Reading Module	TC3	User navigates to a reading exercise on a specific cultural topic.	The exercise provides cultural insights related to the topic, enhancing the user's understanding of the cultural context.
4	Cultural Immersion Module	TC4	Explore cultural insights related to a specific country.	Users gain knowledge about traditions, customs, and societal norms of the selected culture.

5.3 TEST REPORTS

Table 5.2 Test Reports

Sl. No.	Test Case No.	Test Status	Test Result
1.	TC1	Successful	The reading exercise displayed beginner-level content as expected.
2.	TC2	Successful	The system provided accurate feedback on the user's writing accuracy and language usage.
3.	TC3	Successful	The reading exercise provided relevant cultural insights as intended.
4.	TC4	Successful	Users successfully explored cultural insights related to the selected country, gaining knowledge about its traditions, customs, and societal norms.

6. CONCLUSION

The major implementation and design issues along with advantages and disadvantages of the project are properly mentioned. The future scope of the project is also mentioned in this chapter.

6.1 DESIGN AND IMPLEMENTATION ISSUES

Poor design and/or implementation can cause failure or rejection of a software system.

6.1.1 Design Issues

In the Langmates mobile app, design issues may arise that impact user experience. Firstly, the user interface (UI) complexity could overwhelm users, especially newcomers, with an abundance of features, hindering intuitive navigation. Inconsistent design elements across different sections of the app may disrupt visual coherence, leading to confusion. Moreover, the absence of accessibility features tailored to users with disabilities could limit inclusivity and accessibility. Overloading information with excessive options and content may result in cognitive overload and decision fatigue. Lastly, a poor information hierarchy within the app's interface could impede users' ability to locate key features efficiently, leading to frustration and decreased engagement. Addressing these design issues is crucial to enhancing the usability and effectiveness of the Langmates app for language learners.

6.1.2 Implementation Issues

In the Langmates mobile app, implementation details are critical for ensuring seamless functionality and optimal user experience. This involves integrating language learning modules such as reading, writing, listening, and speaking with user-friendly interfaces. Each module must be meticulously designed and implemented to offer tailored exercises and activities suitable for learners at various proficiency levels. Additionally, implementing features for cultural immersion, such as providing insights into diverse cultures alongside language learning, requires thorough research and content curation. Integration with external services like Metamask for wallet functionality and Ethereum blockchain for NFT transactions necessitates robust backend development and secure API integration. Overall,

attention to detail in implementation is essential for delivering a cohesive and effective language learning experience within the Langmates mobile app.

6.2 ADVANTAGES AND LIMITATIONS

6.2.1 Advantages

- Comprehensive Language Learning: Langmates offers a wide range of language learning modules covering reading, writing, listening, and speaking skills, providing users with a holistic learning experience.
- Personalized Learning Experience: The app tailors exercises and activities to users' proficiency levels, ensuring that they receive relevant and engaging content that matches their learning needs.
- Cultural Immersion: Langmates incorporates cultural immersion modules, allowing users to explore diverse cultures alongside language learning, fostering a deeper understanding of global perspectives.
- Flexibility and Convenience: Users can access Langmates anytime, anywhere, making it convenient for busy learners to incorporate language learning into their daily routines.
- Interactive Engagement: The app provides interactive exercises and simulations, fostering active participation and practical application of language skills, enhancing learning effectiveness.
- Progress Tracking and Feedback: Langmates enables users to track their progress and receive feedback on their performance, motivating them to continue their language learning journey.
- Community Support: Langmates fosters a vibrant community of language learners, providing opportunities for collaboration, cultural exchange, and mutual support, enhancing the learning experience.

6.2.2 Limitations

- Potential Overwhelm: The abundance of features and content in Langmates may overwhelm some users, particularly newcomers, leading to confusion and difficulty in navigating the app.

- Accessibility Challenges: The app may lack robust accessibility features, limiting access for users with disabilities and potentially excluding them from fully engaging with the platform.
- Dependency on Technology: Langmates relies on technology for its functionality, meaning that users may face disruptions or limitations in their language learning experience due to technical issues or outages.
- Language Limitations: While Langmates offers a wide range of languages, it may not cover every language or dialect, limiting options for users seeking to learn less common or niche languages.

6.3 FUTURE AND SCOPE OF THE PROJECT

The future of Langmates holds promising opportunities for continued growth and expansion within the realm of language learning and cultural immersion. As technology continues to evolve, the app stands poised to leverage advancements in artificial intelligence (AI) and machine learning to further personalize the learning experience for users. Through sophisticated algorithms, Langmates can analyze user data and behavior to deliver even more targeted and effective learning materials, catering to individual learning styles and preferences.

Moreover, the scope of Langmates extends beyond traditional language learning modules, with the potential to delve deeper into cultural immersion experiences. By forging partnerships with cultural institutions, museums, and local communities, the app can offer users immersive virtual tours, live cultural events, and interactive experiences that bring language and culture to life in dynamic ways.

Additionally, Langmates can explore avenues for gamification and social learning, tapping into the innate human desire for competition, collaboration, and community. By integrating gamified elements such as leaderboards, challenges, and rewards, the app can incentivize learning and foster a sense of camaraderie among users.

Furthermore, the scope of Langmates extends globally, with opportunities to expand its language offerings to include lesser-known languages and dialects, catering to a diverse and multicultural user base. By embracing linguistic diversity and inclusivity,

Langmates can serve as a bridge connecting people from different backgrounds and cultures, fostering understanding, empathy, and mutual respect.

In essence, the future of Langmates is one of innovation, accessibility, and cultural enrichment, with the potential to revolutionize the way people learn languages and engage with the world around them. Through continuous adaptation and evolution, the app is poised to remain at the forefront of language learning technology, empowering users to embark on enriching linguistic and cultural journeys.

REFERENCES

1. Richards, J. C., & Rodgers, T. S. (2014). Approaches and Methods in Language Teaching. Cambridge University Press.
2. Ellis, R. (2008). The Study of Second Language Acquisition. Oxford University Press.
3. Lightbown, P. M., & Spada, N. (2013). How Languages are Learned. Oxford University Press.
4. Brown, H. D. (2000). Principles of Language Learning and Teaching. Pearson Education.
5. Smith, J., & Johnson, M. (2019). "Effectiveness of Language Learning Apps in Vocabulary Acquisition: A Meta-Analysis." Language Learning Journal, 45(3), 289-305.
6. Garcia, A., & Lee, S. (2018). "Mobile Language Learning: A Review of Current Research and Challenges." TESOL Quarterly, 52(3), 701-725.
7. Liu, M., & Pata, K. (2019). "The Impact of Gamification on Language Learning." Educational Technology & Society, 22(2), 291-307.
8. Chen, B., & deNoyelles, A. (2013). "Exploring Students' Mobile Learning Practices in Higher Education." EDUCAUSE Review, 48(2), 74-75.
9. Duolingo. (n.d.). Research and White Papers. Retrieved from https://www.duolingo.com/research
10. Babbel. (n.d.). Insights & Research. Retrieved from https://about.babbel.com/en/insights/
11. Rosetta Stone. (n.d.). Language Learning Research. Retrieved from https://www.rosettastone.com/research/