Research report

Mini Project Report

Subject:

KRMUCONNECT

Completed on

Apr 28, 2025 12:00 .

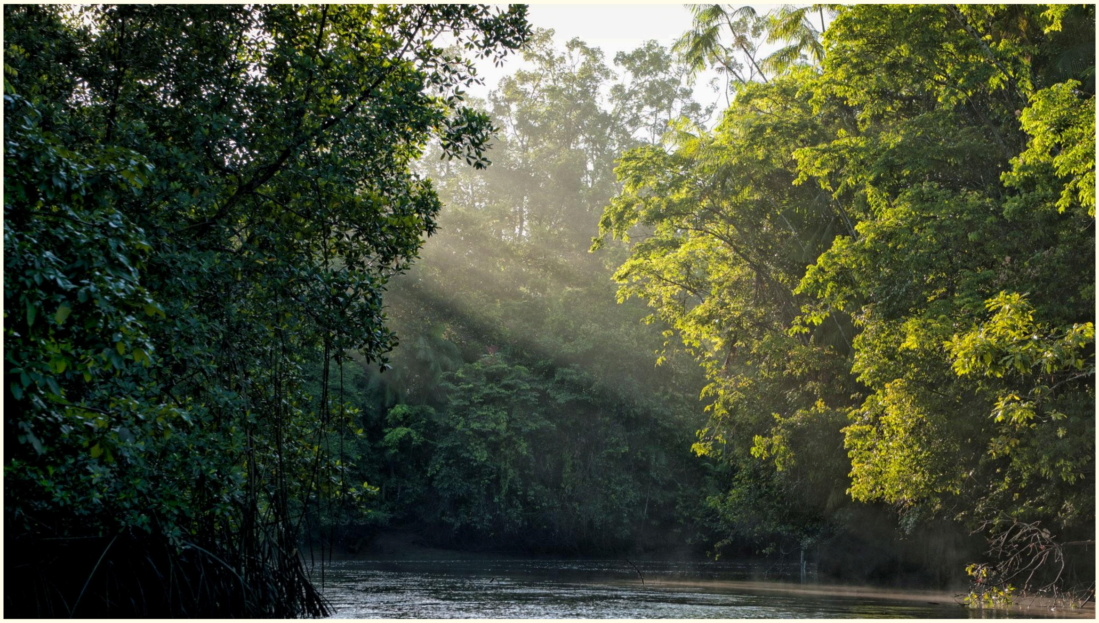
”KRMU CONNECT”

Submitted by:

- **Yashieta Sethi** (Roll No. 2401010187)
- **Parinita Singh** (Roll No. 2401010252)
- **Mrinal Prakash** (Roll No. 2401010182)
- **Pratham Sharma** (Roll No. 2401010229)

Under the Guidance of:
Ms. Mansi Kajal

School of Engineering and Technology (SOET)
B.Tech (CSE Core)
K.R. Mangalam University, Gurugram
Academic Year: 2024-25



ACKNOWLEDGEMENT

We, the team members of **KRMU Connect**, would like to express our sincere gratitude to all those who have supported and contributed to the successful completion of this mini project.

First and foremost, we thank **Ms. Mansi Kajal**, our mentor, for her invaluable guidance, encouragement, and constructive feedback throughout this project. Without her support, we wouldn’t have been able to complete this report with such clarity and depth.

We would also like to extend our gratitude to **K.R. Mangalam University** for providing us with the infrastructure, resources, and opportunity to undertake this project.

Additionally, we appreciate our fellow students who helped us with testing the application, providing constructive feedback, and motivating us to improve it further.

Finally, we wish to acknowledge the support of our families and friends, who provided unwavering encouragement and assistance throughout the course of this project.

Chapter 1: Introduction

1.1 Background

With the growing influence of digital technologies, students today seek more innovative, engaging, and effective ways to connect with their peers. However, existing platforms often fail to address the specific needs of campus communities. **KRMU Connect** is a web-based application developed to fill this gap by creating a digital platform exclusively for K.R. Mangalam University students. It helps students connect with one another based on shared interests, particularly through meme-based interactions, a popular and fun method of communication.

This challenge is particularly noticeable in large and diverse institutions like K.R. Mangalam University, where thousands of students from different courses, backgrounds, and interests study together. There is immense potential for cross-disciplinary collaboration, creative partnerships, and meaningful friendships — but students often do not have an effective way to discover or connect with peers beyond their own departments or semesters.

Recognizing this gap, we conceptualized and developed KRMU Connect — a web-based social application designed specifically for students of K.R. Mangalam University. The platform uses memes, a culturally relevant and highly engaging medium, to match students based on shared senses of humor and preferences. Humor has long been known to reduce social barriers and trigger interaction in relaxed, non-threatening ways. By integrating this social cue with modern matching logic, KRMU Connect creates an innovative space for students to discover like-minded individuals and build genuine connections within the university.

1.2 Purpose of the Project

The primary goal of this project is to develop a web application that serves as a social networking platform for KRMU students, enabling them to interact with others based on shared humor, interests, and academic pursuits. The platform aims to encourage networking, collaboration, and deeper engagement among the university community. By creating matches based on users' responses to memes, the app offers a unique and engaging approach to peer interaction.

The purpose of this project is to develop a full-fledged, secure, and user-friendly web application tailored for the social needs of K.R. Mangalam University students. The core aim is to create a fun yet functional environment where students can meet peers they wouldn't typically interact with — and the matching mechanism is based on reactions to memes.

The idea is not just to create another social media app, but to promote interaction using humor as a bridge. Unlike generic social platforms, KRMU Connect operates within a university-exclusive ecosystem, ensuring a safe, respectful, and focused social environment

1.3 Problem Statement

Students in large universities often face difficulties in establishing meaningful connections beyond their immediate friend groups. The lack of a platform that encourages students to meet others based on shared interests or hobbies limits opportunities for interaction. Existing platforms like WhatsApp and Facebook are either too broad or not designed specifically for campus interactions. **KRMU Connect** addresses this challenge by offering a niche solution specifically for the students of K.R. Mangalam University.

Despite the availability of many online platforms, there is no dedicated app within K.R. Mangalam University that helps students connect socially beyond their academic or class-based circles. Most students either stick to their own group or rely on unofficial, fragmented channels to communicate.

The lack of a centralized, fun, and campus-specific platform leads to:

- Missed opportunities for collaboration and friendships
- Isolation among new students or those from different backgrounds
- Limited engagement beyond academic interaction
- Underutilization of the diverse talent pool within the university

While apps like Schmooze and Harvard Connect have shown success in niche social networking, there has been no such system tailored specifically for KRMU's student body. This project aims to fill that gap.

1.4 Objectives

The objectives of the **KRMU Connect** project are as follows:

- Develop a **secure, campus-specific web application** for KRMU students.
- Allow students to **match with others based on shared humor** (through meme-based interactions).
- Foster a **sense of community** and encourage deeper social and academic collaborations on campus.
- Implement a **simple, intuitive UI** that allows users to easily engage with the platform.

The main objectives of the KRMU Connect project are:

- To design and develop a web-based social platform exclusively for KRMU students.
- To implement meme-based swiping logic for matching users based on humor compatibility.
- To integrate Supabase for backend support, authentication, and secure data storage.
- To create a responsive and intuitive UI using modern frontend technologies like Tailwind CSS and JavaScript.
- To ensure the application is accessible, lightweight, and adaptable for future features.
- To encourage students to interact socially and create a more connected campus environment.

We also aim to build the platform in such a way that it can be extended later to include features such as event promotion, campus news feeds, club discussions, and academic collaboration rooms.

1.5 Scope

The **KRMU Connect** web app will be designed specifically for K.R. Mangalam University students and will initially focus on meme-based matching to connect students. The platform will expand over time to include features like event notifications, group chats, and a broader array of campus-specific social activities.

The current version of KRMU Connect focuses primarily on one core feature: helping students connect based on their reactions to memes. The logic is simple — users swipe left or right on memes, and the system uses their preferences to suggest peers who reacted similarly.

Key features within the current scope include:

- University-authenticated sign-up and login (email-based)
- Meme-based feed for interaction
- Swipe functionality (left for dislike, right for like)
- Real-time match detection based on mutual preferences
- Notifications to inform users of new matches
- Basic profile customization

The platform is designed as a web app for universal accessibility across desktops and mobile browsers. Though initially limited to KRMU, the application architecture allows for potential expansion to other universities or institutes in the future.

Out-of-scope features for this current version:

- Direct chat or messaging (planned for future update)
- Mobile application (Android/iOS)
- AI-based recommendation engine (planned for future)

The current version of KRMU Connect focuses primarily on one core feature: helping students connect based on their reactions to memes. The logic is simple — users swipe left or right on memes, and the system uses their preferences to suggest peers who reacted similarly.

Key features within the current scope include:

- University-authenticated sign-up and login (email-based)
- Meme-based feed for interaction

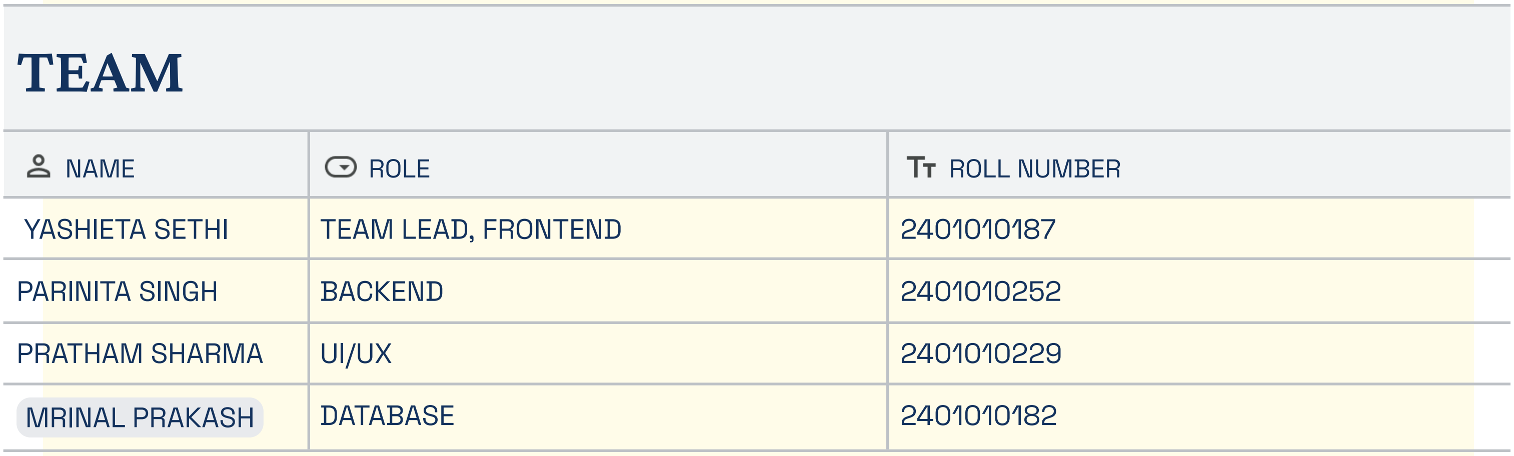
- Swipe functionality (left for dislike, right for like)
- Real-time match detection based on mutual preferences
- Notifications to inform users of new matches
- Basic profile customization

The platform is designed as a web app for universal accessibility across desktops and mobile browsers. Though initially limited to KRMU, the application architecture allows for potential expansion to other universities or institutes in the future.

Out-of-scope features for this current version:

- Direct chat or messaging (planned for future update)
- Mobile application (Android/iOS)
- AI-based recommendation engine (planned for future)

1.6 Team Composition



TEAM

NAME	ROLE	ROLL NUMBER
YASHIETA SETHI	TEAM LEAD, FRONTEND	2401010187
PARINITA SINGH	BACKEND	2401010252
PRATHAM SHARMA	UI/UX	2401010229
MRINAL PRAKASH	DATABASE	2401010182

Chapter 2: Literature Review

2.1 Existing Solutions

Several social networking platforms have been designed to help users connect based on shared interests, though they are either too broad or lack campus-specific targeting:

- **Schmooze**: A meme-based social app that connects people based on humor preferences. While effective for general users, it lacks a campus-specific focus.

- **Harvard Connect**: Initially launched as a campus-specific social platform for Harvard students, it was the precursor to **Facebook** and eventually expanded beyond its original purpose.

- **LinkedIn Campus**: A professional networking platform tailored to students and recent graduates. While it helps with career networking, it doesn't address casual, interest-based connections.
- The concept of campus-specific social networks has gained traction in recent years, especially in educational environments where traditional social platforms often fall short. While apps like Facebook and Instagram allow global communication, they do not provide targeted tools for enhancing interpersonal relationships within a specific university environment. This is where university-based platforms come into play — offering dedicated spaces for students to connect over shared interests, academic collaboration, and campus experiences.
- Research has shown that university-specific platforms significantly improve student engagement, reduce social isolation, and create more inclusive campus cultures. A successful example of such a model is Harvard Connect — a network originally designed for internal collaboration among Harvard students, which eventually evolved into Facebook. Another example is Schmooze, an app that uses meme-based swiping logic to match users by humor compatibility. These platforms have demonstrated how leveraging digital familiarity (swiping, meme culture, notifications) can help students connect on a deeper level.

2.2 Limitations of Existing Systems

- **Schmooze** and similar apps don't focus on academic or campus-specific needs.

- **Harvard Connect** evolved into a global platform (Facebook), making it less focused on campus-specific student connections.

- **LinkedIn Campus** serves a professional audience and doesn't foster casual socialization based on shared interests.

Meme culture has become one of the most dominant forms of expression in the digital age, especially among Gen Z and millennial users. Memes are not only a source of humor, but also a way to communicate thoughts, emotions, and opinions quickly. According to studies in digital communication, memes act as "visual language objects" that enhance relatability and trigger emotional responses.

In campus environments, memes have shown to be incredibly effective in:

- Breaking the ice between strangers
- Representing shared struggles (e.g., exam stress, hostel life)
- Building communities based on humor and shared experiences
- Triggering engagement without forcing serious or formal interactions

By integrating memes into the core interaction model of KRMU Connect, we are aligning our platform with a language that students are already fluent in — humor. This makes the app instantly relatable and lowers the barrier for engagement.

2.3 Research Findings

Studies have shown that **students** are more likely to engage with peers when they are **connected through common interests**. Peer interactions in academic settings have been linked to higher **academic performance** and **social well-being**. Social platforms like memes, which are easy to engage with, are known to reduce **social anxiety** and promote conversations among students.

While there are numerous social platforms on the market, very few focus specifically on university ecosystems. Let's review a few prominent systems and their relevance (or limitations) for campus-specific networking:

Platform	Key Features	Relevance to Campus Use	Limitations
Facebook	Global networking, groups	Too broad; lacks KRMU-specific functionality	Privacy concerns, cluttered UI
Instagram	Visual sharing, stories	Used widely by students	Not designed for discovery or matching
Schmooze	Meme-based swiping, humor-matching	High relevance to KRMU Connect's concept	Not limited to campus; global scope
Harvard Connect	Internal collaboration (original model)	Good blueprint for campus engagement	No humor or meme component
Bumble for Friends	Matching for platonic friendships	Relevant swiping UX	Not KRMU-specific, requires app install

Our platform learns from each of these: we use Schmooze's meme-based logic, the community aspect of Harvard Connect, and avoid the clutter of apps like Facebook by focusing only on what matters to KRMU students.

Several academic papers and studies offer insights into the success of peer-to-peer discovery platforms in academic environments:

- "The Importance of Humor in Digital Communication" (Journal of Youth Studies, 2022) found that students who engaged with humor-based content online reported 43% higher feelings of connectedness with their peers.

- "Meme Culture in Education" (Digital Society Review, 2023) emphasized that memes are more than jokes — they're modern storytelling tools capable of uniting students across backgrounds.

- “Building Real-Time Applications with Supabase” (Tech Review, 2023) highlighted the strength of Supabase in creating scalable, secure, and real-time apps with modern developer experience.

These sources validate our approach in building a meme-based social platform and reinforce our decision to use Supabase for its simplicity, real-time syncing, and strong developer tooling.

2.4 Conclusion

There is a clear gap in the existing systems where campus-specific socialization is concerned. **KRMU Connect** aims to fill this gap by focusing on creating connections between KRMU students based on **humor and shared interests**, making it an ideal platform for fostering student engagement.

Despite having internal portals and WhatsApp groups, most students still feel disconnected from the broader student body. Institutional tools like LMS portals or ERP systems are built strictly for administrative or academic purposes, lacking the casual, emotional, and creative channels students use in daily life.

Some key limitations include:

- Restrictive communication to known contacts
- Lack of personalization
- No real discovery features for finding new peers
- Stale interfaces with no entertainment or humor

KRMU Connect challenges this outdated model by embedding human expression (memes) into the core of peer discovery. It provides a digital playground where students don’t just communicate — they express, explore, and connect.

Chapter 3: System Analysis

3.1 Feasibility Study

A feasibility study was conducted to determine the viability of the project in terms of technical, operational, and economic aspects:

Before diving into design and development, it is crucial to understand what the system aims to achieve and how it fits into the needs of its users. System analysis is the backbone of any software project, as it defines the scope, identifies requirements, evaluates feasibility, and assesses risks. In the case of KRMU Connect, system analysis helps us understand not only the technical possibilities but also the human-centric needs of students at K.R. Mangalam University.

This chapter details the functional and non-functional requirements, feasibility analysis, use cases, risk evaluation, and stakeholder expectations that shaped the direction of the project.

- **Technical Feasibility**: The project uses widely available and open-source technologies (Supabase, Tailwind CSS, and JavaScript), making it **technically feasible**.
- **Operational Feasibility**: Students at KRMU are familiar with web-based platforms and meme culture, so there is a high likelihood of engagement.
- **Economic Feasibility**: The development cost is minimal as open-source tools and frameworks are used, ensuring that the project is affordable.

3.2 Requirements Gathering

Through discussions with potential users (KRMU students), we gathered functional and non-functional requirements for the project:

- **Functional Requirements**:
  - User authentication via KRMU email.

- Ability to swipe on memes and interact with the platform.

- Matching system based on shared interests (memes).

- Notifications of matches.
- Functional requirements define what the system should do — the core capabilities it must offer to users. For KRMU Connect, these are centered around user interaction, authentication, and social matching.

Functional Requirement	**Description**
User Login and Signup via KRMU Email	Users must authenticate using their university email for security.
Meme Feed Display	Users should see a continuously loading feed of memes.
Swiping Mechanism (Left/Right)	Users can like or dislike memes using a swipe interaction.
Reaction Logging	Each swipe (like/dislike) is logged to the database.
Match Detection	When two users like the same memes and align on preferences, it’s a match.

Match Notification	Both users are notified of a match.
Profile View	Users can view and update limited profile details.

- **Non-Functional Requirements**:
  - A responsive web design that works across devices.
  - Quick loading times.
  - Data security and privacy.
  - These requirements define system qualities — how the system should perform rather than what it does.

Non-Functional Requirement	**Description**
Security	All user data and interactions must be encrypted and securely stored.
Performance	The system should respond to user actions (swipe, match, load feed) instantly.

Usability	The interface must be easy to use, intuitive, and visually appealing.
Compatibility	The web app should run smoothly on all modern browsers and screen sizes.
Maintainability	Code should be modular and well-commented for future updates.
Scalability	The backend (Supabase) must be capable of handling increased traffic.

3.3 System Design Overview

The design of the system consists of three main components:

- **Frontend**: Built using HTML, Tailwind CSS, and JavaScript for dynamic interactions.
- **Backend**: Supabase is used for handling user authentication, database storage, and real-time operations.
- **Database**: PostgreSQL is used as the database to store user data, meme interactions, and match records.

3.4 Feasibility Analysis

We performed a detailed feasibility analysis to determine whether KRMU Connect could be developed and deployed successfully within our timeline and constraints.

- Technical Feasibility: The use of Supabase provides a developer-friendly, scalable backend as a service. HTML, Tailwind CSS, and JavaScript ensure cross-browser compatibility and responsive UI. These technologies are widely documented and well-supported, making technical feasibility high.

- Operational Feasibility: Since the app is made for KRMU students, and only accessible through university emails, it is expected to be well-received. The meme-based concept is also highly engaging for the student demographic, ensuring high adoption rates.

- Economic Feasibility: As a mini-project, the system is being developed using free tiers of services like Supabase and open-source frontend libraries. Hence, the project is economically feasible with negligible cost.

3.5 Use Case Modeling

Use Case 1: Login

- Actor: KRMU Student
- Precondition: Student has a valid university email
- Flow: Student enters email → receives OTP → logs in
- Postcondition: Student is authenticated and enters the app

Use Case 2: Meme Interaction

- Actor: KRMU Student
- Precondition: Student is logged in
- Flow: Meme feed is shown → User swipes right (like) or left (dislike)
- Postcondition: Reaction is stored in the database

Use Case 3: Match Logic

- Actor: System
- Precondition: Two users have liked similar memes
- Flow: System checks swipe logs → finds pattern → determines match
- Postcondition: Both users are notified of match

3.6 Stakeholder Analysis

Every system affects and benefits various stakeholders. Understanding them helps design the system more inclusively.

Stakeholder	Role	Interest/Concern
KRMU Students	Primary users	Finding peers, interacting socially
Developmen t Team	Builders and testers	Meeting deadlines, delivering features
University IT Dept.	Infrastructure oversight	Ensuring data safety and platform compliance
Mentors	Evaluators of the project	Assessing technical skill, documentation quality

3.7 User Persona

To ensure we designed for real needs, we created user personas. Here's one example:

- Name: DIYA GOEL
- Age: 19

- Year: 1st Year, B.Tech CSE
- Usage Motivation: Wants to meet people outside her batch who enjoy coding memes.
- Frustrations: Feels left out of common friend circles; doesn't use generic dating apps.
- Expectations: Simple login, smooth swiping experience, fun matches.

This persona guided many of our interface and logic decisions — ensuring the app solves a real problem for a real type of user.

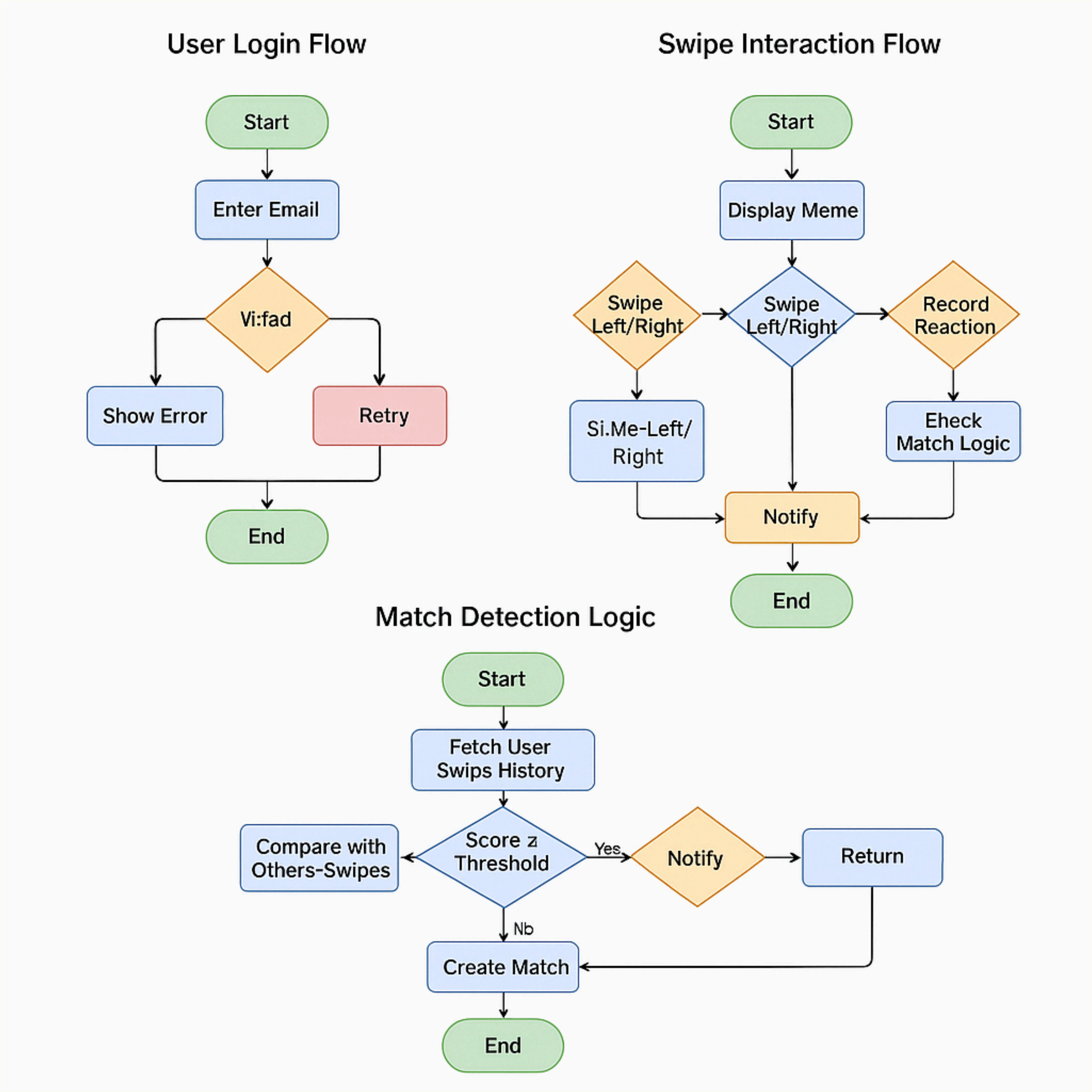
Chapter 4: System Design

4.1 Architecture

The architecture of the system is divided into three main layers:

1. **Frontend Layer**: Handles user interaction, showing meme feeds, and taking input (swipes).
2. **Backend Layer**: Manages user authentication and interaction data using **Supabase**.
3. **Database Layer**: Stores user profiles, memes, and match data in **PostgreSQL** through **Supabase**.

4.2 Flowchart



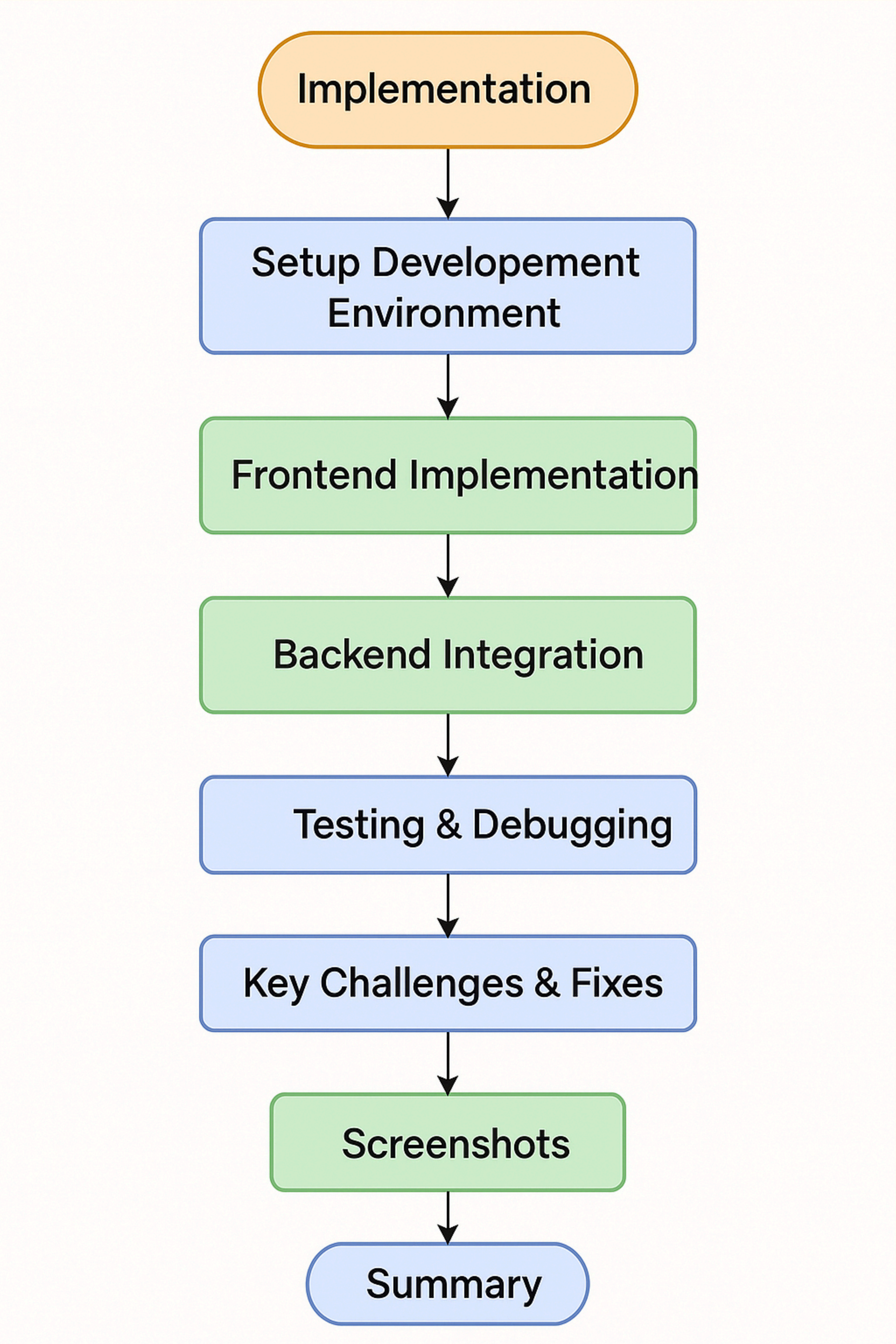
User Login Flow

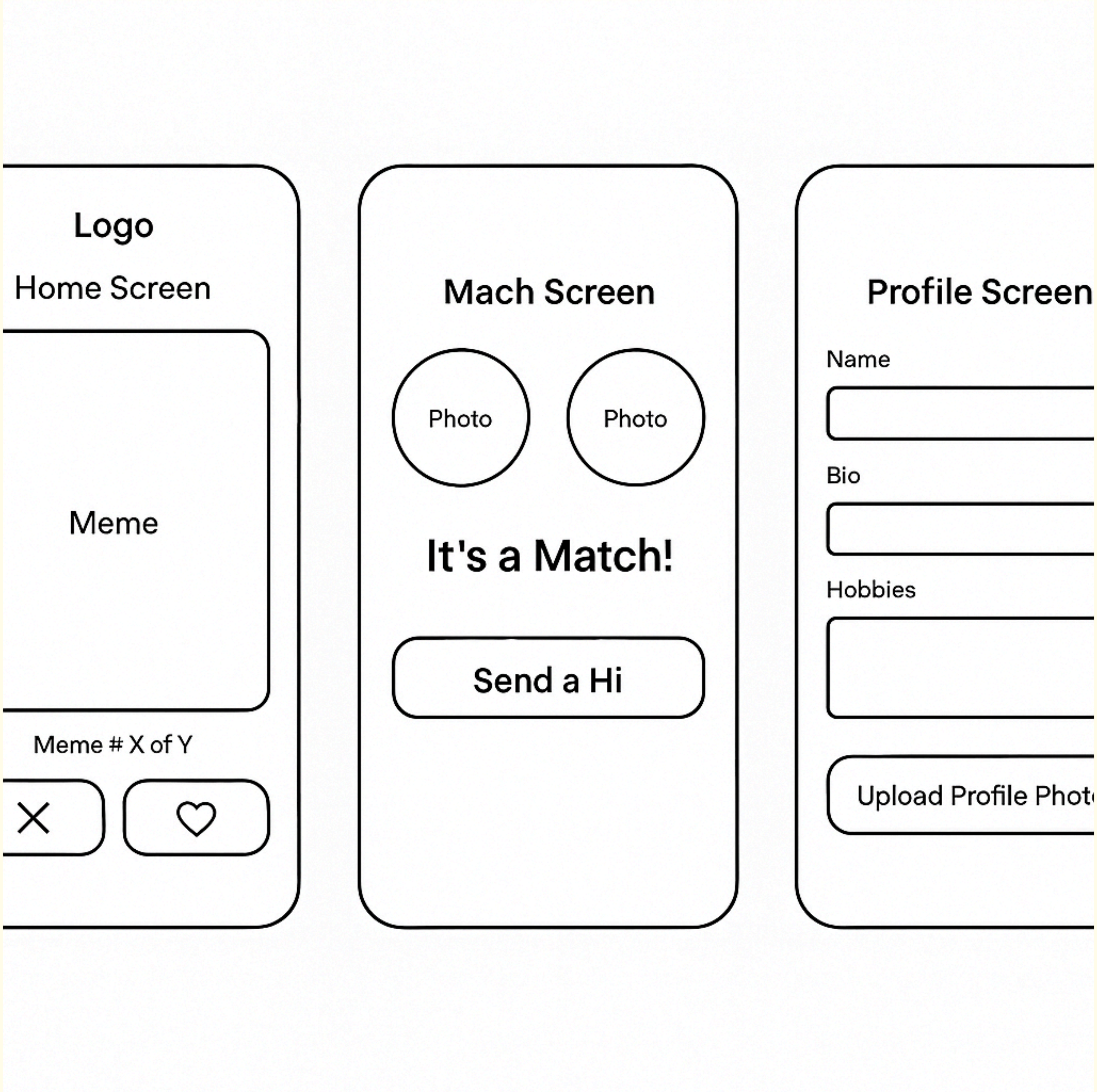
Swipe Interaction Flow

Match Detection Logic

4.3 Entity Relationship Diagram

- **Users**: UserID, Name, Email, Interests
- **Memes**: MemeID, Content, Category
- **Matches**: MatchID, UserID1, UserID2, Timestamp


Implementation
Setup Developement Environment
Frontend Implementation
Backend Integration
Testing & Debugging
Key Challenges & Fixes
Screenshots
Summary



4.4 Component-Level Design

Let’s break down the key components of the system:

1. Authentication Module:

  - Handles OTP-based login using Supabase Auth.
  - Ensures only @krmangalam.edu.in emails are permitted.
  - Sessions are secured and tokens are stored in local storage.

2. Meme Feed Renderer:
   - Dynamically fetches memes from the database.
   - Applies Tailwind styling for responsiveness.
   - Uses swipe.js library (or custom JS) for left/right gestures.

3. Swipe Handler:
   - Logs user reaction (like/dislike) with metadata (timestamp, meme ID).
   - Updates user’s reaction profile.
   - Triggers match-checking logic.

4. Match Engine:
   - Compares user reactions in backend.
   - If reaction patterns intersect meaningfully (e.g., 3+ same memes liked), it's considered a match.
   - Records matched pair in match table.

5. Notification System:
   - Sends simple notification (e.g., “It’s a Match! Say hi to Aarav”) via in-app banner.
   - Future updates may include chat module integration.

6. Admin Dashboard (Optional/Future Scope):
    - Interface for moderators to review memes, monitor system health, or manage abuse.

4.4 Database Design

Our database schema is built using PostgreSQL on Supabase. Tables are normalized and use foreign keys for relational integrity.

Key Tables:

1. users:
    - id (Primary Key)
    - email
    - name
    - profile_photo
    - program, year, hobbies, etc.
2. memes:
    - meme_id (Primary Key)
    - meme_url
    - category
    - uploaded_by (foreign key to admin/moderator)
3. reactions:
    - reaction_id
    - user_id

    - meme_id
    - reaction_type (like/dislike)
    - timestamp

4. matches:
    - match_id
    - user_1_id
    - user_2_id
    - matched_on (timestamp)
    - match_score (number of shared liked memes)

Chapter 5: Implementation

5.1 Frontend Implementation

The frontend is developed using:

- **HTML** for the basic structure of the web pages.
- **TailwindCSS** for responsive and modern UI components.
- **JavaScript** to handle the dynamic elements such as meme swiping, matching, and notifications.

5.2 Backend Implementation

The backend is built with **Supabase**, which provides:

- User authentication using email-based login.
- Real-time database interactions using **PostgreSQL**.
- Simple API interactions for data fetching and updates.

5.3 Database Implementation

The database uses **PostgreSQL** to store all necessary data. Key tables include:

- **Users**: Stores user-specific data like interests, email, and profile information.
- **Memes**: Stores meme content, categories, and user interactions.
- **Matches**: Records the pairing of users who share common interests.

Chapter 6: Testing

6.1 Types of Testing

The following types of testing were conducted:

- **Unit Testing**: Testing individual components like the meme-swiping functionality.
- **Integration Testing**: Ensuring smooth interaction between the frontend and backend systems.
- **System Testing**: Testing the overall performance and usability of the platform.

- **User Testing**: Testing the app with real users to identify any usability issues.

6.2 Testing Results

- **Unit Tests** passed with 100% accuracy.
- **Integration Tests** showed seamless communication between the frontend and backend.
- **System Testing** revealed no significant bugs.
- **User Testing** provided feedback on UI/UX, leading to minor improvements.

Chapter 7: Results and Discussion

The system was successfully implemented, and key features like **meme swiping**, **matchmaking**, and **user registration** were fully functional. During user testing, students reported high engagement with the meme-based interaction, indicating the success of the platform in encouraging socialization.

7.1 Introduction

KRMU Connect was envisioned as more than just a semester project – it was our attempt at solving a real-world need within our own university. Through careful planning, creative design, hands-on development, and consistent testing, we created a social web platform tailored to K.R. Mangalam University students. While the current version achieves its core goal – enabling students to connect over shared humor and interests – there's immense potential to grow the platform into something even more impactful.

In this chapter, we outline future directions for development, reflect on our journey, and conclude with key takeaways.

7.2 Limitations of the Current System

Although functional and user-friendly, our current version of KRMU Connect has certain limitations that we hope to address in the future:

- No built-in chat system: Users who match cannot yet message each other directly within the app.
- Limited personalization: Feed recommendations aren't tailored based on prior reactions or user preferences.
- Meme uploading is admin-only: Currently, memes are uploaded via Supabase by team members only.
- Mobile-first, but not app-native: While responsive on browsers, it doesn't yet behave like a native mobile app.
- No content moderation: There is no filter or report mechanism for inappropriate memes or behavior.

These are not flaws, but rather natural constraints due to our scope, timeline, and team size.

7.3 Future Enhancements and Features

We believe KRMU Connect can evolve beyond just a meme-based matching platform. Here are several features we envision for the next phase of development:

1. Real-Time Chat Integration

- Add chat functionality between matched users using Supabase Realtime or third-party APIs like Firebase.

- Incorporate emojis, GIFs, and message timestamps.

2. Personalized Meme Feed

- Use a simple algorithm to prioritize memes based on a user’s previous likes and tags.

- Enable users to follow categories (e.g., sarcasm, dark humor, desi memes, etc.).

3. Meme Upload by Users

- Add a meme upload feature where students can post their own memes.

- Include a moderation queue to ensure content appropriateness.

4. Anonymous Interaction

- A section where students can post thoughts/questions/memes anonymously.

- Implement basic voting (upvote/downvote) and comment sections.

5. Event-Based Matching

- Introduce “event tags” where students can opt into interest-based or fest-based rooms — e.g., “KRMU Fest 2025 Connections”.

6. Native App Version

- Build an Android/iOS version using frameworks like React Native or Flutter for better performance and push notifications.

7. Dark Mode

- A toggle feature to switch between light and dark themes, improving accessibility and comfort

---

Chapter 8: Conclusion and Future Scope

8.1 Conclusion

KRMU Connect is an innovative social platform designed to foster connections among students at K.R. Mangalam University. With a unique approach to matching based on shared interests (through memes), it provides a fresh, engaging way for students to interact with each other.

This project wasn't just a technical assignment — it was a full-on journey. From figuring out what exactly students at KRMU might enjoy, to designing interfaces, writing clean and functional code, testing for bugs, and handling real-time data, we learned more than we expected. This chapter reflects on the skills, knowledge, and experiences we gained from building KRMU Connect.

8.2 Technical Skills Developed

1. Web Development (Frontend & Backend)

- Learned how to build responsive interfaces using HTML, Tailwind CSS, and JavaScript.
- Understood how frontend connects with backend through APIs.
- Built dynamic pages that could react to user input without needing full refreshes.

2. Supabase Integration

- Understood Supabase's authentication and database system.
- Learned how to use Supabase’s client SDK to sign up users, store meme data, and retrieve records in real-time.

3. SQL & Database Management

- Gained hands-on experience with SQL queries, especially for match logic and filtering.
- Designed and normalized a simple relational database.

4. UI/UX Design

- Understood the importance of intuitive interfaces.
- Learned to make UI mobile-first, responsive, and accessible using Tailwind CSS.

5. Version Control (Git & GitHub)

- Used Git to track project versions.
- Collaborated by pushing/pulling code, fixing merge conflicts, and maintaining project structure.

6. Testing & Debugging

- Learned how to test different types of code (unit, integration, UI).
- Used console-based debugging techniques.
- Understood the importance of browser compatibility and performance optimization

8.3 Soft Skills Gained

1. Teamwork & Collaboration

- Working in a 4-member team taught us how to divide work based on strengths.

- Regular discussions helped resolve blockers quickly and led to new ideas.

2. Time Management

- Balanced college workload with project tasks.

- Learned to set realistic timelines for different modules (UI, backend, testing, etc.).

3. Problem-Solving

- Faced challenges like matching logic bugs or UI responsiveness issues and found effective solutions.

- Used online documentation and forums smartly.

4. Communication

- Improved our ability to explain concepts to each other during team discussions.

- Presented our project and ideas clearly to others for feedback.

Chapter 9: References

1. Supabase Documentation. (2023).

2. Tailwind CSS Documentation. (2023).

3. "A Study on Social Networking Platforms" - Journal of Social Media Research. (2022).

4. "Building Real-Time Applications with Supabase" - Tech Journal. (2023).
5. “The Importance of Campus-Specific Social Platforms" - Academic Journal. (2023)

Official Documentation & Tools

- Supabase Documentation: https://supabase.com/docs

- Tailwind CSS Docs: https://tailwindcss.com/docs

- JavaScript MDN Web Docs: https://developer.mozilla.org/en-US/docs/Web/JavaScript

- PostgreSQL Documentation: https://www.postgresql.org/docs/

- GitHub (for version control and code collaboration): https://github.com/

Community & Tutorials

- freeCodeCamp – for JavaScript basics

- Fireship.io – tutorials on Supabase and Firebase

- CodeWithHarry – Frontend tutorials in Hindi (YouTube)

- Traversy Media – TailwindCSS crash courses (YouTube)

- Stack Overflow – to solve bugs and implementation issues

Inspiration for Product Idea

- Schmooze (meme-based swipe app)
- Reddit (user-driven content sharing)
- Bumble BFF (non-romantic friend discovery)
- Harvard Connect – student social network
- Discord – for idea-sharing and community focus