**SoulSwipe**

Ishaaq Shaik,

JayaChandra Narala,

Jyothsna Chaparala,

Vasavi Devineni,

Venkata Rayudu Adapa, Venkata Sravan Telu



A documentation for the application of Soul Swipe

Computer Science and Information Systems

Northwest Missouri State University

Maryville, MO

# Abstract

In today’s hectic environment, finding a life partner can be difficult. Discovering the ideal match might be difficult due to busy schedules, small social networks, and different cultural backgrounds. For this, we have come up with an iOS mobile application developed to help the user find a desirable life partner. The application asks the gender of the user and the gender the user is trying to search. Following this, the user answers the compatibility quiz which is used to search for the desired partners. By the searching algorithm, the application calculates the compatibility score between the user and the rest of the profiles and shows the top matching profiles first in the user feed. A user can manually search for a particular partner and communicate with them. This application has been developed by Swift. We also developed an Admin page which has a shared database with the mobile application used to monitor any inappropriate behavior by the users.

# Introduction

## Background

For emotional support, companionship, and shared duties, having a life partner is essential. Long-term stability, social support, and teamwork are all made possible by having a partner. However, there are several reasons why one might not find a life partner, such as personal preferences, divergent interests in life, difficulties approaching possible mates, limited social circle, or a concentration on one’s objectives. For this, there is a need for a platform where one can find a suitable partner. In a linked and dynamic culture, the digital sphere has emerged as a conduit for individuals in search of connection.

## Motivation

The difficulty in finding a compatible life partner has contributed to the mobile application. The fast-paced modern world might make it challenging to meet a suitable partner through traditional ways due to things like varied personal preferences, small social circles, and busy schedules. By providing a simple venue for people to meet based on compatibility and common interests, Soul Swipe helps people overcome some of the challenges that come with traditional dating.

## Objectives

The main objectives of Soul Swipe are:

* Show suitable partners for the users.
* To offer a practical and easily available means for individuals to gather and communicate.
* Provide secure communication among users.
* Creating a positive and user-friendly experience

## Scope

The scope of the project includes the design, development, and implementation of the mobile application for the user. And a web application to overview the health of the mobile application. The mobile application features include:

* User creation and authentication.
* Searching for a partner.
* In-app messaging.
* Notifications and alerts for users.

The web application features include:

* Blocking users who behave inappropriately.
* Notifying the users of any updates.
* Responding to any request by the user.

The scope also includes the integration of third-party software for chat and the integration of the mobile application with the database.

# Project Description

A dating app called ”Soul Swipe” aims to transform how people interact and discover lasting partnerships. This project includes a responsive online application made with HTML, CSS, and JavaScript, as well as a dynamic iOS mobile application created in Swift. The solid database foundation provided by Google Firebase guarantees safe and effective data management for user profiles and interactions.

A compatibility algorithm that determines and assigns a compatibility score between users and possible partners is at the heart of Soul Swipe. The matchmaking system generates a customized and curated list of profiles with the greatest compatibility scores by analyzing user preferences, interests, and other pertinent data. As a result, users are more equipped to make judgments, which raises the possibility of making a sincere and long-lasting relationship.

The iOS-only mobile application provides people on the go with a smooth and intuitive UI. Soul Swipe makes communication simple and safe by integrating with CometChat to enable real-time messaging. This promotes interpersonal interactions and engagement in a secure and stimulating atmosphere.

From an administrative perspective, the mobile application’s health and data may be tracked using an easy-to-use dashboard that can be accessed through the online application. Administrators can make wise decisions and improve the user experience by learning more about user engagement, app performance, and other important indicators.

Soul Swipe is more than just a dating app; it’s a platform that uses innovative technology and well-thought-out design to help people connect meaningfully. With its cutting-edge features and all-inclusive matchmaking methodology, Soul Swipe seeks to transform the dating scene by providing users with a more satisfying and customized dating experience as they pursue a genuine relationship.

# Methodology

## Requirements Analysis:

* Analyze the project requirements in detail, considering both functional and non-functional factors.
* Describe the user profiles and essential features of a modern dating app.

## Design Phase:

* To visualize the user interface and user experience, create paper wireframes for the web admin interface and the iOS application.
* Create an intuitive user interface that complements Soul Swipe’s branding.

## Database Design:

• Using Google Firebase, plan and create the database structure to effectively store and retrieve user data, such as compatibility scores, profiles, and preferences.

## Mobile Application Development:

• Use Swift to implement the iOS mobile application, integrating CometChat real-time messaging, swiping functionality, and user profiles, among other features, into the designed user interface.

## Web Application Development:

• Using HTML, CSS, and JavaScript, create the web application for the admin interface, which will include a dashboard for monitoring the health and statistics of the mobile application.

## Compatibility Algorithm:

• Create and put into use a compatibility algorithm that uses the Jaccard index to determine compatibility scores based on user preferences and other pertinent information.

## Integration of Third-Party Services:

• Improve user interaction and communication by integrating CometChat for real-time messaging within the mobile application.

## Testing:

* To find and fix any errors or problems, thoroughly test the web and mobile applications.
* To guarantee a seamless experience across various devices and browsers, conduct compatibility testing.

## Documentation

• Create detailed documentation for mobile application and web application for admin for general usage and future development.

## Deployment:

* Release the mobile app for iOS users through the Apple App Store.
* Place the web application on a safe server that the administrative team can access.

## Monitoring and Maintenance:

* Use monitoring tools to keep tabs on the mobile application’s functionality and user engagement.
* Maintain and support the product continuously to fix any problems that arise after launch and roll out updates and new features in response to user input.

# Implementation

## Mobile Application Development (iOS using Swift):

The iOS mobile application’s decision to use Swift as its primary programming language was crucial. An app that is feature-rich, responsive, and easy to use was made possible by Swift’s effectiveness and performance. Storyboard has been used with minimum iOS development version 16. To handle user interactions, data processing, and fluid navigation, the solution made use of Swift’s capabilities. Account registration is done by Firebase authentication service.

**5.1.1 Required Packages:**

The development of our iOS mobile application using Swift relies on certain essential packages to ensure optimal performance, seamless user interactions, and effective data processing. Below is the list of required packages and dependencies, along with instructions on how to install them.

1. **Firebase Authentication:**

This application utilizes Firebase Authentication for secure and efficient user registration. This package is crucial for handling user authentication seamlessly. To install Firebase Authentication, follow the below steps:

1. In XCode, open the project which requires the Firebase Authentication.
2. Navigate to **File > Add Packages**
3. When prompted, add the Firebase Apple platforms SDK repository: <https://github.com/firebase/firebase-ios-sdk>.
4. **Lottie:**

Lottie is used to add beautiful and interactive animations to the application. It simplifies the process of incorporating animations to the user interface.

1. In XCode, open the project which requires the Lottie.
2. Navigate to **File > Add Packages**
3. When prompted, add the Lottie SDK repository: <https://github.com/airbnb/lottie-ios.git>.
4. **Comet Chat:**

Comet Chat is an SDK used to add the messaging feature to the iOS Application. It adds all the features like sending text messages, audio messages. Interacting through calls, block or unblock a profiler in chats etc.,

1. To integrate the Comet Chat to the project, follow the steps in the link: <https://www.cometchat.com/docs/ios-uikit/integration>
2. Along with this, we need a software named “Rosetta”. To install this, open the terminal in Mac book and execute the below command: *“softwareupdate --install-rosetta”*

**5.1.2 App Navigation:**

The application is designed with a user-friendly interface that seamlessly guides users through various screens, each serving a specific purpose. Below is an overview of the screens within the application:

1. **Splash Screen:**

The Splash screen is the initial visual users encounter when launching the application. It serves as a brief introduction, displaying the app’s logo. The primary purpose of the Splash screen is to provide a smooth transition into the main content of the application while creating a positive first impression. The Splash screen is typically displayed for a short duration, ensuring a swift and engaging entry into the application.

A couple of hearts with pink outline

Description automatically generated with medium confidence

**Figure 1:** Splash Screen

1. **Login Screen:**

The Login screen allows users to authenticate and access their accounts. User can enter their credentials, such as username (email) and password, to log in. The app validates the username (email) and password using Firebase authentication. On clicking the login button, the app performs the validations. If the credentials are correct, the screen will be navigated to “Home” screen. The login button has another button “Sign up”, on clicking this button navigates to “Sign up” screen.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 2:** Login and Signup Screen | A screenshot of a login screen  Description automatically generated  **Figure 3:** Login Screen |

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 4:** Validating username (email) | A screenshot of a phone  Description automatically generated  **Figure 5:** Validating password |

A screenshot of a phone

Description automatically generated

**Figure 6:** Validating the credentials.

1. **Signup Screen:**

The Signup screen enables the new users to create accounts and join the application. The features of this screen are:

* Includes fields for user details such as email, first name, last name, and date of birth.
* Application ensures the validation of all the fields before allowing user registration.
* To register for the application, user must be above 18 years.
* On clicking “Log in” button, navigates to Login Screen.
* After a successful registration, a link will be sent to the user’s email for their email verification. User must set a new password for their account.
* After a successful registration, the app will be navigated to “Login” Screen. The user has to login with their email and newly set password.

|  |  |
| --- | --- |
| A screenshot of a login form  Description automatically generated  **Figure 7:** Signup Screen | A screen shot of a message  Description automatically generated  **Figure 8:** Validating Email |
| A screenshot of a phone  Description automatically generated  **Figure 9:** Validating First name of the user | A screenshot of a phone  Description automatically generated  **Figure 10:** Validating Last name of the user |
| A screenshot of a phone  Description automatically generated  **Figure 11:** Verifying the user’s age | A screenshot of a login form  Description automatically generated  **Figure 12:** Successful User Registration |

A screenshot of a computer screen

Description automatically generated

**Figure 13:** Resetting the password.

1. **Gender Selection Screen:**

The Gender Selection screen allows users to specify their gender and the gender they are interested in. The features of this screen are:

* Users can choose their own gender from the menu.
* Users can also specify the gender they are interested in.
* A “continue” button to proceed to the next step after making selections.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 14:** Choosing the preferred genders | A screenshot of a phone  Description automatically generated  **Figure 15:** On clicking on continue |

1. **Questionnaire Screen:**

The Questionnaire screen is a static screen providing information about the next steps related to the compatibility quiz. The features of this screen are:

* Static information explaining the purpose and importance of the compatibility quiz.
* Clear instructions on how users can proceed to the compatibility quiz.
* A “Take Quiz” button to start the compatibility quiz.

A white paper with black text

Description automatically generated

**Figure 16:** Questionnaire Screen

1. **Compatibility Quiz Screen:**

The Compatibility screen presents users with questions about relationships and partners. The features of this screen are:

* Multiple choice questions related to relationships and partner preferences.
* Users can select options that best represent their preferences or experiences.
* Users select number of options based on the question requirement.
* A “Next” button to proceed to the next set of questions or finalize the compatibility quiz.
* Validations are also performed for the questions while answering.
* After the completion of questionnaire, the app will be redirected to “Home” Screen.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 17:** Validating if no options are selected | A screenshot of a phone  Description automatically generated  **Figure 18:** Validating if a smaller number of options are selected than the requirement |
| A screenshot of a phone  Description automatically generated  **Figure 19:** Validating if user tries to select more than required number of options | A screenshot of a phone  Description automatically generated  **Figure 20:** After completing the compatibility quiz |

1. **Home Screen:**

The Home Screen is the central hub of the application, providing access to key features and navigation to other sections. This screen has 4 tabs with respective icons at the bottom of the screen. The tabs are:

* Home: Displays partner profiles with compatibility scores.
* Favorites: Displays the profiles that are liked (or favorited) by the user.
* Chats: Provides messaging features (to chat) with other profiles.
* Profile: Displays the user’s own profile and other settings.

The features of the home screen are:

* Displays partner profiles of the user’s preferred gender with compatibility scores.
* The first row displays the profiles having the top 5 highest compatibility scores. The remaining profiles are divided into multiple rows.
* Each partner profile is presented as a card with an image, compatibility score, partner name, number of likes profile have, and a like button.
* Clicking on a profile navigates to the individual profile of the partner.
* Clicking on the like button adds the profile to the user’s favorites. On tapping second time on the like button, removes the profiles from user’s favorites.

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated  **Figure 21:** Home Screen | A screenshot of a cell phone  Description automatically generated  **Figure 22:** Liking a Profile |

1. **Favorites Screen:**

The Favorites Screen has the list of profiles that are liked by the user. The features of the screen are:

* Each cell of the list contains a partner profile with an image, name, and their profession.
* On tapping on any profile, navigates to the individual profile of the partner.
* On left swiping the cell in the list, can remove the profile from favorites.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated | A screenshot of a phone  Description automatically generated |

**Figure 23:** Favorites Screen

1. **Partner Seeker Profile Screen:**

The Partner Seeker profile screen displays the details of the partner with an image and their bio. A button on the top right corner “Report” is used to report a partner profile. Any partner can be reported only once. On clicking on “Message” screen will pop-up the chat screen of that partner.

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a phone  Description automatically generated |

**Figure 24:** Individual Partner Seeker Profile

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 25:** Reporting a user | A screenshot of a cell phone  Description automatically generated  **Figure 26:** After a user has been reported. |

1. **Chats Screen:**

The Chats screen is the space where users can engage in conversations with other users they have interacted with. The features of the screen are:

* Displays a list of ongoing and past conversations.
* Each conversation entry includes a preview of the recent messages.
* Displays the date of the last message in each conversation.
* Tap on a conversation to open and continue chatting.
* On clicking the “write” icon, users can search profiles and start new conversations.
* Users can interact using text messages, audio messages, emojis, gifs. Users can interact with audio, video calls.
* On clicking on “i" button, user can block or unblock a particular profile.
* User can also be able to view whether the partner is online or offline.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 27:** Chats Screen | A screenshot of a phone  Description automatically generated  **Figure 28:** Individual Conversation with a partner |
|  |  |
| A screenshot of a phone  Description automatically generated  **Figure 29:** After clicking on “+” sign | A screenshot of a phone  Description automatically generated  **Figure 30:** After clicking on “♥️” symbol |
| A screenshot of a phone  Description automatically generated  **Figure 31:** Recording Audio Messages | A screenshot of a phone  Description automatically generated  **Figure 32:** Sending Audio Messages |
| A grey circle with white text  Description automatically generated  **Figure 33:** Making Audio or Video Calls | **A screenshot of a phone  Description automatically generated**  **Figure 34:** Starting new conversation by searching a partner |
| A white screen with red text  Description automatically generated  **Figure 35:** Unblocking a partner | A screen shot of a white screen  Description automatically generated  **Figure 36:** Blocking a partner |

1. **Profile Screen:**

The Profile screen showcases the user’s own profile and provide other options too. The features of the screen are:

* Edit Profile: On clicking on this button, navigates to “Edit Profile” screen, where user can update their profile info.
* Likes: Navigates to “Likes” screen, displays the profile that liked the user’s profile.
* Edit Questionnaire: Navigates to “Edit Quiz” screen, where user can update their answers of compatibility quiz.
* Messages: Navigates to “Messages” screen, displays the messages from the admin.
* Announcements: Navigates to “Announcements” screen, displays the announcements related to the features of the app from the admin.
* Change Password: On clicking on this option, a mail will be sent to the user’s email to reset their password.
* Delete Account: On clicking this option, user can be able to delete their account.
* Logout: Logouts the user from the app and redirects to “Login” screen.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 37:** Profile Screen | A screenshot of a phone  Description automatically generated  **Figure 38:** Profile Screen |

1. **Edit Profile Screen:**

The Edit Profile screen enables the user to update their own profile information. Exception is that user cannot update their email and date of birth.

|  |  |
| --- | --- |
| A screen shot of a phone  Description automatically generated  **Figure 39:** Edit Profile Screen | A screenshot of a phone  Description automatically generated  **Figure 40:** Edit Profile Screen |
| A screenshot of a cellphone  Description automatically generated  **Figure 41:** Updating user profile picture | A screenshot of a phone  Description automatically generated  **Figure42:** After setting user’s profile picture |

A screenshot of a phone

Description automatically generated

**Figure 43:** Updating user’s profile details.

1. **Likes Screen:**

The Likes screens displays list of profiles who liked the logged user’s profile. Each cell of the list contains a partner profile with an image, name, and their profession.

A screenshot of a phone

Description automatically generated

**Figure 44:** Likes Screen

1. **Edit Questionnaire Screen:**

The Edit Questionnaire screen enables the user to update their answers of compatibility quiz. They can choose a particular question and can edit their choice of options. (Scroll the screen to view more questions).

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 45:** Edit Questionnaire Screen | A screenshot of a phone  Description automatically generated  **Figure 46:** After clicking on a particular question |

* On double clicking on the selected option, will unselect the option.
* User can choose new options for the question. On clicking on update, the new answers will be updated to the database.
* Validations are even checked at this screen too.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 47:** Validating the question | A screenshot of a phone  Description automatically generated  **Figure 48:** Updating the question to database |

1. **Announcements Screen:**

The Announcements screen displays a list of announcements from the admin. The announcements can be about the app. The announcements are displayed in form of cells. On tapping any cell, will expand all the cells and can see the announcement message. Scroll the screen to view more announcements.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 49:** Announcements Screen | A screenshot of a phone  Description automatically generated  **Figure 50:** Expanding the cells in Announcements Screen |

1. **Messages Screen:**

The Messages screen displays a list of messages from the admin to the user. The messages are displayed in form of cells. On tapping any cell, will expand all the cells and can see the message. Scroll the screen to view more messages.

|  |  |
| --- | --- |
| A white rectangular object with a white text  Description automatically generated  **Figure 51:** Messages Screen | A screenshot of a phone  Description automatically generated  **Figure 52:** Expanding the cells in Messages Screen |

1. **Change Password:**

On clicking change password, send a link to the user’s email to reset their password.

A screenshot of a phone

Description automatically generated

**Figure 53:** Change Password

1. **Delete Account:**

On clicking delete account, the app displays an alert to confirm the action. If the user chooses “Yes”, then their account will be deleted from the database. But the data related to their database will be stored for until 6 months.

If the deleted user wants to join the app again, then he/she must again register to the app using their previous mail to retrieve the data.

|  |  |
| --- | --- |
| A screenshot of a phone  Description automatically generated  **Figure 54:** Deleted user tries to login | A screenshot of a phone  Description automatically generated  **Figure 55:** Deleted user after successful registration |

1. **Logout:**

On clicking logout, the app will be again navigated to “Login” Screen.

A screenshot of a phone

Description automatically generated

**Figure 56:** Logout

## Web Application Development (HTML, CSS, JavaScript):

HTML, CSS, and JavaScript were used in tandem to create the web application. JavaScript is the technology utilized in the web application’s backend. The goals of the online implementation were to create a streamlined experience that was like the mobile app, preserve brand aesthetics, and create a uniform user interface.

The mobile application’s health and performance could be thoroughly monitored with the help of the web application that was created for administrative purposes. Administrators can monitor user activity, quickly address problems, and guarantee Soul Swipe’s general stability.

**5.2.1 Pages:**

The admin web application is created to manage the iOS application. The web application is user-friendly, and it has the following pages designed in it:

* 1. **Login Page:**

Login Page is the first page of the web application, where the admin credentials will be verified to access the web application.

A screenshot of a login screen

Description automatically generated

**Figure 57:** Login Page of Admin

* 1. **Dashboard:**

Dashboard is the home page of the admin web application. Dashboard has the side-menu for navigation on the left-hand side. On the right-hand side, it has the count of registered users, male users, and female users.

A screenshot of a computer

Description automatically generated

**Figure 58:** Dashboard of Admin

* 1. **Registered Users:**

Registered users page has the list of all registered users of the application. The page also contains a search bar to search the partner seeker based on their name. The page offers multiple filters like based on gender or order based on age, likes or reports. On clicking on “View Profile”, the page will be navigated to individual partner seeker profile.

A screenshot of a computer

Description automatically generated

**Figure 59:** Registered Users Page of Admin

A screenshot of a computer

Description automatically generated

**Figure 60:** Searching a particular partner seeker.

A screenshot of a computer

Description automatically generated

**Figure 61:** Filters in Registered User Page.

* 1. **Individual Partner Profile:**

Admin has the details of each partner seeker, can view in this page. Admin can dynamically update the details of the partner seeker. Admin can also view the profiles who liked the partner seeker and reported the partner seeker.

A screenshot of a computer

Description automatically generated

**Figure 62:** View Profile Page

A screenshot of a computer

Description automatically generated

**Figure 63:** Liked By

A screenshot of a computer

Description automatically generated

**Figure 64:** Reported By

* 1. **Announcements Page:**

Admin can send announcements to all the registered users at the same time. The announcement can be anything related to the mobile application. It can be demonstrating any new feature or update of the application.

A screenshot of a computer

Description automatically generated

**Figure 65:** Announcements Page

A screenshot of a facebook profile

Description automatically generated

**Figure 66:** Posting a new announcement

A screenshot of a computer

Description automatically generated

**Figure 67:** After successful posting of an announcement

* 1. **Messages Page:**

Admin can send messages to individual partner seekers through the web application. Admin can choose the partner seekers to whom he wants to send the messages. Admin can send a same message to multiple users at the same time.

A screenshot of a computer

Description automatically generated

**Figure 68:** Messages Page

A screenshot of a computer

Description automatically generated

**Figure 69:** Posting a new message

A screenshot of a chat

Description automatically generated

**Figure 70:** Adding partner seekers to the message

A screenshot of a computer

Description automatically generated

**Figure 71:** Successful posting a message

## Database Management (Google Firebase):

To securely storing and managing user data, Google Firebase as the back-end database solution was essential. Assuring that consumers receive the most recent information and promoting a dynamic user experience, Firebase provides real-time data synchronization. Compatibility scores, user profiles, and preferences were all considered when the database was reorganized for implementation.

* To add Firebase to the application, follow the steps in the link: <https://medium.com/swlh/add-firebase-to-your-ios-app-quickly-updated-2020-d673ea96552e>
* To integrate the web application, update the constant “firebaseConfig” in **assets > js > app.js** with the project details.
* The Firebase must contain the below Collections:
  + Announcements
  + CompatibilityQuiz
  + CompatibilityScores
  + FavouriteUsers
  + Notifications
  + Reports
  + UniqueID
  + Users

## Compatibility Algorithm:

One unique aspect of Soul Swipe was the use of the Jaccard Index for compatibility scoring. Utilizing similar interests as a basis, compatibility scores between users were determined by integrating this algorithm into the matchmaking process. The implementation aimed to improve the matchmaking process overall for relevance and accuracy.

* If User A, chooses options A, C among the 3 options A, B, and C.
* If User B, chooses options A, B among the 3 options A, B, and C.
* Then the compatibility score for both the users is calculated as below:

J(A, B) = 1/3 = 0.33

## Integration of Third-Party Services (CometChat):

Any dating app must allow for real-time communication, and Cometchat’s integration allowed in-app chatting easy. Cometchat’s features were integrated into the Soul Swipe platform to facilitate instantaneous connections between users and meaningful interactions.

## Testing Strategy:

A thorough testing plan is put in place to guarantee Soul Swipe’s dependability and functionality. This includes end-to-end testing to mimic actual user scenarios, integration testing to evaluate how various modules interact, and unit testing for specific components. Frequent testing cycles are carried out to find and fix any possible defects or performance problems, resulting in a reliable and stable application.

## Security Measures:

Security is of the utmost importance while implementing Soul Swipe. Firebase uses encryption technologies to protect user data both in transit and in storage. To stop unwanted access, authentication procedures are strengthened, and frequent security audits are carried out to find and fix any possible weaknesses. The group complies with industry norms to offer consumers a haven for their private data.

## Challenges and Solutions:

After overcoming development roadblocks in Soul Swipe, the team skillfully handled tasks like integrating GitHub and XCode, integrating Cometchat while adhering to the app’s design, creating interesting compatibility quiz questions, putting the Jaccard Index into practice for accurate compatibility scores, managing Firestore completion handlers, and choosing a backend language for the admin web app. The team overcame these challenges by careful testing, smart planning, and cooperative efforts. Their flexible methodology guarantees a strong, customer-focused dating site where obstacles turn into chances for creativity, culminating in a flawlessly merged and feature-rich Soul Swipe app. The group solves problems quickly by using creative problem-solving techniques. They are quick to adapt and show flexibility. Iterative development enables adaptation to shifting project dynamics, and regular communication channels promote prompt decision-making.

## Code Repositories:

## Documentation:

## Collaboration and Communication:

With JIRA software, the team manages projects more effectively and keeps track of Soul Swipe’s development. JIRA makes it easier to assign tasks, track progress, and resolve issues. In addition, a Microsoft Teams group facilitates better collaboration by acting as a single point of contact for team members to debate, share updates, and make decisions together in real time. This integrated method guarantees smooth collaboration, enabling the team to stay on task, exchange ideas, and quickly resolve any issues that may come up while the app is being developed. For the Soul Swipe project to succeed, a collaborative and open environment is fostered by the integration of Microsoft Teams and JIRA.

# Results

# Discussion

A user-centric approach guided the design choices for both the web and mobile applications, with the goal of producing an entertaining and easy-to-use interface. Visually appealing design of the user interface ensures easy navigation and maximum user engagement. The choice to use Swift to create a specialized iOS mobile application demonstrates the dedication to provide Apple users a smooth and optimized experience. Swift’s performance and efficiency help create a responsive interface that makes it easy for users to browse the app. The web application’s use of HTML, CSS, and JavaScript highlights the team’s commitment to provide a uniform user experience across platforms.

The Jaccard Index compatibility scoring is a noteworthy addition that demonstrates the project’s dedication to provide consumers more relevant matches. The matchmaking process is made more sophisticated by this algorithm, which increases the likelihood that users will find companions who have similar interests and higher compatibility scores. A reliable backend option is Google Firebase, which provides scalability for future growth in addition to safe data storage. Cometchat’s in-app communication integration improves user experience by encouraging in-the-moment contacts and conversations.

CometChat’s real-time messaging integration greatly improves user experience by facilitating smooth communication. Incorporating this third-party service shows that it was a purposeful decision to use specialized tools for important tasks, cutting down on development time and guaranteeing a strong message infrastructure.

An administrative web application further highlights the project’s integrated methodology. Administrators may keep an eye on the functionality and performance of the mobile application, guaranteeing a flawless user experience and quickly resolving any possible problems.

To conclusion, the Soul Swipe dating app initiative is expected to have a influence on the online dating industry. Soul Swipe has the ability to change the digital matchmaking game with its user-centric design, cutting-edge technology, and dedication to deep connections. It provides a novel and successful method of finding love in the contemporary world.

# Conclusion

As we bring the Soul Swipe project to its conclusion, we are proud of how our concept became a smart dating app that skillfully combines creativity and user-centered design. The team overcame obstacles, added cutting-edge features, and guaranteed strong security through careful execution. A dedication to cutting-edge technologies was demonstrated by the usage of Google Firebase for the database, HTML/CSS/JavaScript for the web app, and Swift for iOS. Soul Swipe’s matchmaking experience is improved by Cometchat integration and the Jaccard Index for compatibility scoring. Microsoft Teams and JIRA are examples of collaborative applications that promote effective communication. Soul Swipe’s entry into the digital sphere is evidence of the team’s commitment to reinventing meaningful interactions in the context of online dating.

# Future Work

## Location based profiles:

Profiles are searched across the database and top compatibility-scored profiles are shown to the user. To reduce the load on the database and improve the accuracy of fetching profiles, location-based filtering needs to be integrated. Based on location, the primary filtering process must be done.

## Improving the search algorithm:

The efficiency of the search algorithm must be improved. A more sophisticated search algorithm with the integration of AI must be implemented.

## Expanded Platform Support:

Expand the mobile application’s compatibility to include additional platforms, like Android, to increase its user base.

## Enhanced User Profiles:

Encourage individuals to add more multimedia to their profiles so that viewers can get a more complete picture of their personalities.

## Social Media Integration:

Integrate social media sharing options and login integration so users can link their Soul Swipe profiles to their social media accounts.

## Community and Events Features:

Create a community platform inside the app so that users can interact with one another by attending local and virtual events and joining interest-based groups.

## Personalized Recommendations:

Create a recommendation engine that makes suggestions about possible matches according to the feedback, preferences, and activities of users.

# GitHub Repository

Link to GitHub Repository: <https://github.com/JyothsnaCh0831/SoulSwipe>