

# Lovelink - A Dating App

## Project Documentation

### Project Overview

Lovelink is a modern dating application designed to connect individuals based on mutual interests and preferences. The app features a step-by-step registration process, allowing users to input their details, upload profile pictures, and complete their profile before accessing the platform.

### Technology Stack

**Frontend:** Kotlin (Jetpack Compose)

**Backend:** Laravel (REST API)

**Database:** MySQL

**Authentication:** Laravel Breeze

### Core Features

#### 1. User Registration & Authentication

Users register using their username , email, phone number, and gender.

Password authentication is managed via Laravel Breeze. Validation ensures all required fields are correctly filled.

#### 2. Profile Photo Upload

Users must upload at least two photos.

Uses Jetpack Compose with `rememberLauncherForActivityResult` to select images.

Images are displayed using `AsyncImage` (ensure correct dependency is included).

### User Profile Management

Users can update their details via the profile screen.

Profile information is stored in a MySQL database through Laravel APIs.

### REST API (Laravel)

The backend is developed using Laravel, providing endpoints for:

#### User Registration & Login

## **Profile Updates**

## **Photo Uploads**

## **Matching & Messaging**

## **Project Structure**

### **Frontend (Kotlin - Jetpack Compose)**

Registrationscreens (Handles user onboarding, including forms and image upload)

Viewmodel (Handles user state, form validation, and API calls)Navigation.kt (Manages screen transitions)

### **Backend (Laravel)**

#### **Routes (api.php)**

POST /register → Register a new user

POST /login → Authenticate user

PUT /profile/update → Update user details

POST /message/send -> processes send message logic

PUT /setting ->handles profile settings

#### **Controllers**

AuthenticatedSessionController.php → Manages authentication

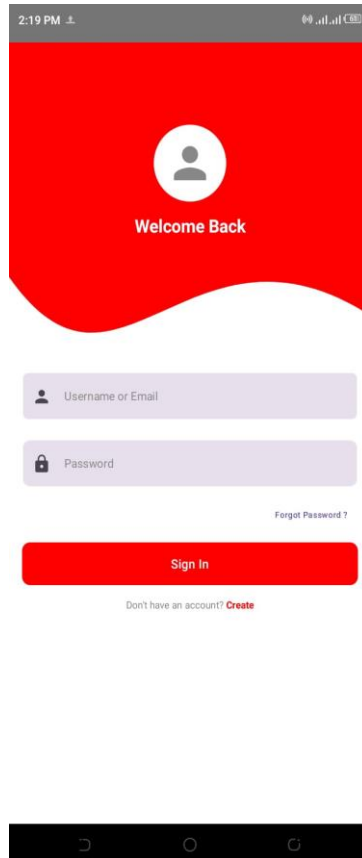
UserController.php → Handles profile updates

PhotoController.php → Manages image uploads

MessageController -> manages sending and receiving of messages

SettingsController ->handles settings options

-




[illegible]


2:18 PM

Signal


60%

# Who Are You? Let's Discover!


 Your info personalizes your experience.



what's your username ?



What's your email ?



What's your phone number ?

## Select your gender

Male

Female

Others

Already have an account? [Login](#)

→ Next

