PROJECT REPORT ON

"HIRELYFT JOB SEARCH APP"

FOR

HIRELYFT INDIA PVT LTD

BY

Kanishk Deshpande

UNIVERSITY OF PUNE

MASTER IN COMPUTER APLLICATION

Maharashtra Education Society's

**INSTITUTE OF MANGEMENT
AND CAREER COURSES (IMCC),
PUNE-411038**

College certificate

MAHARASHTRA EDUCATION SOCIETY'S
(SINCE 1860)

INSTITUTE OF MANAGEMENT & CAREER COURSES (IMCC)

(AUTONOMOUS)

Approved by AICTE and Recognized by Savitribai Phule Pune University, Pune
IMCC Campus, 131, Mayur Colony, Kothrud, Pune 411038, Maharashtra, India | Ph.: 020-2546 6271 / 73 | e-mail: info.imcc@mespune.in | https://imcc.mespune.in

NAAC Accredited with Grade A+

Ref. No. MES IMCC /442/2025-26 **Date:06/05/2026**

CERTIFICATE

This is to certify that the Project Report entitled

"Hirelyft Job Portal Application"

is prepared by

Kanishk Deshpande

M.C.A. Semester IV Course for the Academic Year 2025-26 at M.E. Society's Institute of Management & Career Courses (IMCC), Pune - 411038. M.C.A Course is autonomous and affiliated to Savitribai Phule Pune University.

To the best of our knowledge, this is an original study done by the said student and important sources used by him/her have been duly acknowledged in this report.

The report is submitted in partial fulfillment of M.C.A Course for the Academic Year 2025-26 as per the rules and prescribed guidelines of Savitribai Phule Pune University.

Dr Shilpa Parlikar
Project Coordinator

Ms. Manasi Shirurkar
Programme Coordinator

Dr. Ravikant Zirmite
Head, Dept of MCA MES IMCC

Dr. Santosh Deshpande
Director, MES IMCC

External Examiner

Head Office: 'MES Bhavan', 1214-1215, Sadashiv Peth, Pune - 411030, Maharashtra, India. | Ph.: +91-020-41038100 | www.mespune.in

Company Certificate



www.hirelyft.in hr@hirelyft.in

Hirelyft HIRELYFT INDIA PRIVATE LIMITED

Elevating hiring and opportunities

CIN: U63122PN2025PTC238691

Date: 17/04/2026

Project Certificate

This is to certify that **Kanishk Deshpande**, a student of **MES IMCC**, has been successfully pursuing his project internship at **Hirelyft India Pvt Ltd** since 23/09/2025.
He is currently working on the **project titled: "Hirelyft Job Search App"**

This certificate is issued to confirm that his project work is currently **In-Progress**. During the course of this internship, he has demonstrated a professional approach and has adhered to all organizational protocols, operational standards, and confidentiality requirements.

This certificate is issued for his academic records and for submission to **MES Institute of Management & Career Courses (IMCC) Affiliated to Savitribai Phule Pune University.**

Authorized Signatory,

Ms. Bharti Kabade
Director,
Hirelyft India Pvt. Ltd.

Date: 17/04/2026

DPIIT #startupindia

S.No 16/3/2/6 Shop No.10, Krushna Vihar, Ambegaon BK, Haveli Pune-46, Maharashtra

Guide certificate

Maharashtra Education Society's
Institute of Management and Career Courses (IMCC), Pune
AUTONOMOUS

[CGPA-3.32] in Cycle 1

CERTIFICATE

This is to certify that ***Kanishk Deshpande*** has completed the project work entitled ***"Hirelyft Job Portal Application"*** under my guidance. The report is submitted in partial fulfillment of M.C.A. Course for the Academic Year 2025-2026 as per the rules & prescribed guidelines.

His/her work is found to be satisfactory and complete in all respects.

Date: 09/05/2026

Ms Manasi Shirurkar
(Internal Project Guide)

ACKNOWLEDGEMENT

I take this opportunity to express my sincere gratitude and heartfelt thanks to all those who guided and supported me throughout the completion of this project.

I am deeply grateful to the management of Maharashtra Education Society's IMCC, Pune, for providing the necessary infrastructure and a conducive academic environment for carrying out this work.

I sincerely thank my Project Guide for their invaluable guidance, constant encouragement, and constructive feedback. Their expertise and patient mentorship have been a source of great inspiration throughout this journey.

I extend my gratitude to the Head of Department and all faculty members for their continuous support and academic inputs, which greatly enriched the quality of this project.

I am thankful to the organization and its team for granting me the opportunity to undertake this project and providing the required resources, data, and technical support during the internship.

Name: Kanishk Kumar Deshpande

Roll No.:2401042

Date: 09/05/2026

Application Development Project

Chapter No.	Details	Page No.
1	**Introduction**	
1.1	Company Profile	1
1.2	Problem Statement	1
1.3	Objectives of Proposed System	2
1.4	Scope of the Proposed System	2
2	**Analysis and Design**	
2.1	Entity Relationship Diagram (ERD)	4
2.2	Business Use Case Diagram	5
2.3	Table Structure / Data Dictionary	6
3	**Testing**	
3.1	Test Cases with Results	10
3.2	Defect Report / Test Log	24
4	**Limitations of Proposed System**	25
5	**Proposed Enhancements**	26
6	**References**	27
7	**User Manual (All Input and Output Screens with Consistent Data)**	28

Application Development Project Documentation

1: Introduction

1.1 Company Profile:

Hirelyft India Private Limited is a Pune-based organization dedicated to elevating hiring and career opportunities. The company specializes in developing, maintaining, and enhancing websites and applications across a diverse range of industries. Hirelyft is a DPIIT-recognized startup that provides real-world experience to interns by involving them in complex client projects and internal product development.

1.2 Problem Statement:

In today's competitive job market, both job seekers and employers face significant challenges in the recruitment process. Traditional job portals suffer from several limitations:

- Lack of intelligent matching between candidate skills and job requirements, leading to irrelevant applications and missed opportunities.
- Absence of real-time communication channels between employers and candidates, causing delays in the hiring process.
- Limited candidate presentation options — text-based resumes alone fail to showcase a candidate's personality, communication skills, and practical abilities.

- Fragmented workflow requiring candidates to use multiple platforms for job search, resume creation, application tracking, and interview management.
- Poor notification systems that leave candidates uninformed about application status changes, interview schedules, and new opportunities.
- Absence of profile completeness guidance, resulting in incomplete profiles that reduce candidates' visibility to employers.

There is a clear need for a unified, intelligent job portal application that addresses these pain points by integrating AI-powered matching, real-time messaging, video portfolios, automated resume generation, and smart notifications into a single, intuitive mobile application.

1.3 Objectives of Proposed System

The primary objectives of the Hirelyft Mobile Job Portal Application are:

- To develop a comprehensive mobile application that connects job seekers with employers through an intelligent, user-friendly platform.
- To implement AI-powered skill match analysis using Google Gemini AI, enabling candidates to understand their fit for specific job roles.
- To provide real-time communication between employers and job seekers through Socket.IO-based instant messaging.
- To introduce a video portfolio feature (Shorts) allowing candidates to showcase their skills, personality, and communication abilities beyond traditional resumes.
- To build an automated resume generation system that creates professional PDF resumes directly from user profile data.
- To implement a comprehensive notification system using Firebase Cloud Messaging (FCM) for timely updates on applications, interviews, and messages.
- To create a robust interview management system with scheduling, status tracking, and bidirectional feedback

1.4 Scope of the Proposed System

The Hirelyft Mobile Job Portal Application is a full-stack system comprising a Flutter-based mobile application (frontend), a Node.js/Express.js REST API server (backend), and a PostgreSQL database. The scope encompasses the following functional areas:

A. Job Seeker Module:

- User registration and authentication with OTP verification
- Comprehensive profile creation with education, experience, skills, certifications, projects, and language details
- Job search and discovery with advanced filtering (location, salary, experience, job type, work mode)
- One-click job applications with cover letter and expected salary
- Application status tracking across lifecycle stages (Pending → Viewed → Shortlisted → Interviewing → Hired/Rejected)
- Interview schedule management with details view
- Video shorts creation and sharing for professional showcasing
- Automated PDF resume builder with multiple templates
- AI-powered skill match analysis against job requirements
- Profile completion tracking with progress indicators

B. Employer Module:

- Company profile creation and management with verification workflow
- Job posting with detailed specifications (salary, experience, location, skills required)
- Application review and candidate management
- Interview scheduling with multiple formats (In-person, Phone, Virtual, Technical, Group)

C. Communication Module:

- Real-time messaging using Socket.IO WebSocket technology
- Conversation management with message history
- Push notifications via Firebase Cloud Messaging
- In-app notification center with categorization and read status

D. Admin Module:

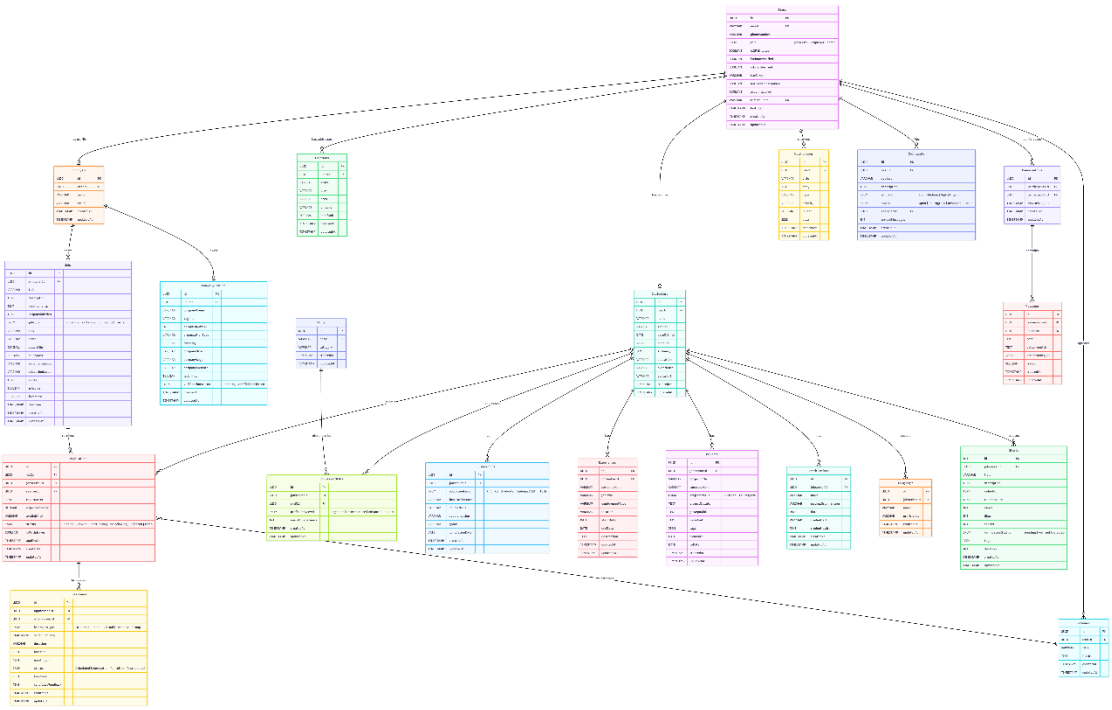
- User management and moderation
- Company verification and status management
- Complaint handling and support ticket resolution
- Account deletion request processing

E. Technology Scope:

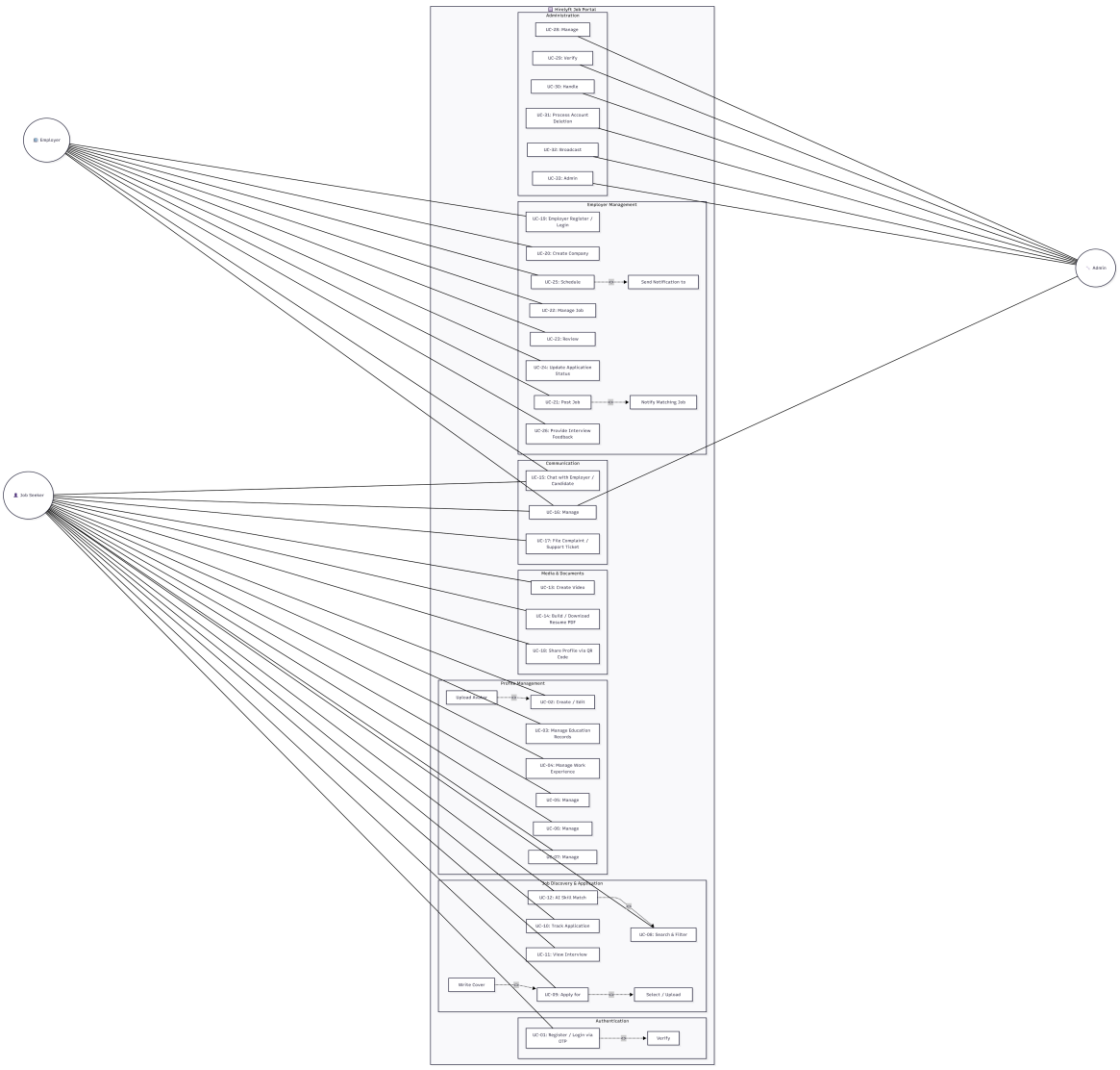
Layer	Technology
Frontend	Flutter (Dart SDK 3.9.2), Material Design 3, Provider
Backend	Node.js, Express.js 5.1.0, RESTful API
Database	PostgreSQL with Sequelize ORM
Real-time	Socket.IO (WebSocket)
Cloud Services	Firebase (FCM), Supabase, Cloudinary
AI/ML	Google Gemini AI, OpenRouter SDK
Security	JWT, bcryptjs, Helmet.js, XSS Protection

2: Analysis and Design

2.1 Entity Relationship Diagram (ERD)



2.2 Business Use Case Diagram



2.3 Table Structure/Data Dictionary

This section outlines the physical database schema designed for the Hirelyft platform. The data dictionary provides a detailed breakdown of the primary tables, their attributes, data types, primary/foreign key constraints, and a brief description of each field's purpose within the system.

1. Table: users (Core Authentication & System Access)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique system identifier for the user account.
phoneNumber	VARCHAR	Unique, Not Null	Mobile number used for primary login and OTP verification.
email	VARCHAR	Nullable	Optional email address for system communications.
password	VARCHAR	Nullable	Bcrypt hashed password for authentication.
role	ENUM	Not Null	Defines access level (jobseeker, employer, admin).
isPhoneVerified	BOOLEAN	Default: False	Flag indicating if the phone number OTP was verified.
fcmToken	VARCHAR	Nullable	Internal token for routing Firebase push notifications.

2. Table: jobseekers (Candidate Profiles)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the job seeker profile.
userId	UUID	**Foreign Key**	Maps 1:1 to the primary users table.
name	VARCHAR	Not Null	Full legal name of the candidate.
headline	VARCHAR	Nullable	Brief professional summary or current job title.
educationLevel	ENUM	Nullable	Highest educational degree obtained.
expectedSalary	DECIMAL	Nullable	Target financial compensation requested by the candidate.
experienceYears	DECIMAL	Default: 0	Total years of accumulated professional experience.

3. Table: employers (Recruiter Profiles)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the employer profile.
userId	UUID	**Foreign Key**	Maps 1:1 to the primary users table.
membersCount	INT	Default: 0	Total number of sub-recruiters acting under this account.
maxJobCount	INT	Default: 1	Maximum allowed job postings restricted by subscription tier.
jobPostedCount	INT	Default: 0	Running total of successfully published jobs.

4. Table: company_profiles (Business Entities)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the business entity.
userId	UUID	**Foreign Key**	Links to the employer account managing the company.
companyName	VARCHAR	Not Null	Official registered name of the business.
industry	VARCHAR	Nullable	Primary sector of business operation (e.g., IT, Retail).
companySize	VARCHAR	Nullable	Bracket denoting employee count (e.g., '100-500').

5. Table: jobs (Job Postings)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the specific job posting.
employerId	UUID	**Foreign Key**	Links to the employers table denoting the author.
title	VARCHAR	Not Null	Designation, role, or position being offered.
jobType	ENUM	Not Null	Nature of employment (Full-time, Part-time, Contract).
openings	INT	Not Null	Number of individual vacancies available.
experienceLevel	ENUM	Default: 'Fresher'	Minimum candidate experience tier required to apply.
salary_min	FLOAT	Nullable	Lower bound of offered financial compensation.
isActive	BOOLEAN	Default: True	System flag determining if the job is actively accepting candidates.

6. Table: applications (Application Lifecycle Tracking)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the application ticket.
jobseekerId	UUID	**Foreign Key**	Maps exactly to the candidate who submitted the application.
jobId	UUID	**Foreign Key**	Maps exactly to the specific job being applied for.
status	ENUM	Default: 'Pending'	Hiring workflow state (Pending, Shortlisted, Rejected, Hired).
resumeId	UUID	**Foreign Key**	Link to the immutable snapshot of the Resume submitted.
appliedAt	DATETIME	Not Null	Automatic timestamp capturing when the submission occurred.

7. Table: interviews (Hiring Communications)

Field Name	Data Type	Key / Constraint	Description
id	UUID	**Primary Key**	Unique identifier for the scheduled interview event.
applicationId	UUID	**Foreign Key**	Link to the target application proceeding to interview.
interviewerId	UUID	**Foreign Key**	Link to the specific user/employer conducting the interview.
scheduledDate	DATETIME	Not Null	The confirmed date and time block for the meeting.
mode	ENUM	Not Null	Format of the interview (Online, Offline).
status	ENUM	Default: 'Scheduled'	State of the interview (Scheduled, Completed, Cancelled).

3 Testing

3.1 Test Cases with Results:

TC-AI-002	AI Skill Match with Empty Profile Skills	FAIL
Objective	Verify that AI skill match returns a meaningful response when user has no skills in their profile.	
Environment	Flutter app, Google Gemini AI API, Node.js backend	
Pre-conditions	User is logged in. Profile has NO skills added. Active job listing with required skills exists.	
Input Specifications	User Skills: None (empty). Job Requirements: Flutter, Dart, Firebase, REST API	
Execution Steps	1. Open a job listing. 2. Tap 'AI Skill Match' button. 3. Gemini API called with empty skills list. 4. Observe response displayed to user.	
Expected Output	Meaningful response: '0% match. Please add your skills to your profile to get an accurate analysis.'	
Actual Output	Gemini API returned a null response. App threw Null check operator used on a null value exception. Screen crashed and navigated back to job listing.	
Pass / Fail Criteria	Fail — No null safety check on Gemini API response. Empty skills payload causes crash instead of a user-friendly fallback message.	

TC-AI-001	AI Skill Match Analysis	PASS
Objective	Verify that the AI skill match feature correctly analyzes candidate skills vs job requirements.	
Environment	Flutter app, Google Gemini AI API, Node.js backend	
Pre-conditions	User has skills added to profile. Job listing has defined required skills.	
Input Specifications	User Skills: Flutter, Dart, Firebase. Job Req: Flutter, Dart, REST API, Firebase	
Execution Steps	1. Open a job listing. 2. Tap 'AI Skill Match' button. 3. Wait for Gemini API response. 4. View match percentage and suggestions.	
Expected Output	Match percentage displayed. Missing skills highlighted. Suggestions shown.	
Actual Output	75% match displayed. 'REST API' flagged as missing. Improvement suggestions rendered.	
Pass / Fail Criteria	Pass if match % is computed and suggestions are returned within 5 seconds.	

TC-ADMIN-001	Admin Company Verification	PASS
Objective	Verify that admin can review and approve a company verification request.	
Environment	Flutter app (Admin role), Node.js backend, PostgreSQL (CompanyProfiles table)	
Pre-conditions	Admin is logged in. At least one company has status 'Pending' verification.	
Input Specifications	Company: TechCorp India, Action: Approve	
Execution Steps	1. Admin navigates to 'Manage Companies'. 2. Selects company with 'Pending' status. 3. Reviews company documents. 4. Taps 'Approve'. 5. Confirms action.	
Expected Output	Company isVerified set to true. Verification status updated to 'Verified'. Employer notified.	
Actual Output	Company status updated to 'Verified'. FCM notification sent to employer account.	
Pass / Fail Criteria	Pass if company isVerified=true and employer receives approval notification.	

TC-SUPP-001	Submit Support Complaint Ticket	PASS
Objective	Verify that a user can file a support complaint ticket from the app.	
Environment	Flutter app, Node.js backend, PostgreSQL (Complaints table)	
Pre-conditions	User is logged in. Support section is accessible.	
Input Specifications	Subject: 'Cannot upload resume', Description: 'File picker crashes on Android 13', Priority: High	
Execution Steps	1. Navigate to 'Support' from drawer menu. 2. Tap 'New Ticket'. 3. Enter subject and description. 4. Set priority to 'High'. 5. Tap 'Submit'.	
Expected Output	Ticket created with status 'Open'. Admin receives notification of new ticket.	
Actual Output	Complaint created with ID and 'Open' status. Ticket visible in support list.	
Pass / Fail Criteria	Pass if ticket is saved and user can view it in the support tickets list.	

TC-SHORT-002	Video Short Upload — File Size Limit Exceeded	FAIL
Objective	Verify that uploading a video larger than the 50MB limit shows a proper validation error.	
Environment	Flutter app, Cloudinary video storage, Node.js backend (50MB limit configured)	
Pre-conditions	User is logged in. A video file of 85MB is available on device gallery.	
Input Specifications	Video: large_demo.mp4 (85 MB), Title: 'My Project Walkthrough'	
Execution Steps	1. Navigate to Shorts section. 2. Tap '+ Upload Short'. 3. Select 85 MB video from gallery. 4. Add title and description. 5. Tap 'Publish' and observe behavior.	
Expected Output	Client-side validation: 'Video must be under 50MB. Please choose a smaller file.' Upload not initiated.	
Actual Output	Upload initiated. Progress bar reached 78% then Cloudinary returned 413 Request Entity Too Large. App showed 'Upload Failed' without file size guidance. User confused.	
Pass / Fail Criteria	Fail — No client-side file size validation before upload. Large file consumes bandwidth and Cloudinary returns 413 with no helpful user message.	

TC-SHORT-001	Video Short Upload by Job Seeker	PASS
Objective	Verify that a job seeker can upload a professional video short to their profile.	
Environment	Flutter app, Cloudinary video storage, Node.js backend	
Pre-conditions	User is logged in. Video file is under 50MB. Device has camera/gallery permissions.	
Input Specifications	Video: portfolio_intro.mp4 (30 seconds), Title: 'My Flutter Skills Demo'	
Execution Steps	1. Navigate to Shorts section. 2. Tap '+ Upload Short'. 3. Select video from gallery. 4. Add title and description. 5. Tap 'Publish'.	
Expected Output	Video uploaded to Cloudinary. Short card appears in profile with pending verification status.	
Actual Output	Video uploaded successfully. Short visible with 'Pending' verification badge.	
Pass / Fail Criteria	Pass if video is uploaded and short appears in the Shorts feed.	

TC-SEC-002	SQL Injection via Job Search Input	FAIL
Objective	Verify that the job search input is sanitized and SQL injection attempts are blocked.	
Environment	Flutter app, Node.js/Express.js backend, PostgreSQL with Sequelize ORM	
Pre-conditions	User is logged in. Job search screen is accessible.	
Input Specifications	Search Input: '; DROP TABLE Jobs; -- (SQL injection payload)	
Execution Steps	1. Navigate to Find Jobs screen. 2. Enter SQL injection string in search box. 3. Tap Search. 4. Monitor backend logs and database. 5. Observe response in app.	
Expected Output	Input sanitized. Query treated as literal text. No DB modification. Search returns empty results safely.	
Actual Output	Backend log showed raw query construction without parameterization in one legacy search route. Sequelize ORM blocked actual DROP TABLE, but error log exposed raw SQL query details in 500 response body visible to client.	
Pass / Fail Criteria	Fail — One legacy search API route uses raw string interpolation. While ORM prevented data loss, internal SQL error details were exposed in API response — a security information leak.	

TC-SEC-001	JWT Token Expiry and Re-authentication	PASS
Objective	Verify that the app handles expired JWT tokens gracefully by redirecting to login.	
Environment	Flutter app, Node.js backend with JWT (7-day expiry), shared_preferences	
Pre-conditions	User's JWT token is expired. User attempts to access a protected route.	
Input Specifications	Expired token used to call GET /api/profile endpoint.	
Execution Steps	1. Simulate expired token in shared_preferences. 2. Launch app and attempt to load dashboard. 3. Observe app behavior on 401 response. 4. Verify redirection to login screen.	
Expected Output	401 Unauthorized response handled. Token cleared. User redirected to Hero/Login screen.	
Actual Output	App received 401, cleared token, showed snackbar 'Session expired', navigated to Hero screen.	
Pass / Fail Criteria	Pass if expired token causes graceful logout and navigation to login without crash.	

TC-RESUME-001	Automated PDF Resume Generation	PASS
Objective	Verify that the system generates a downloadable PDF resume from the user's profile data.	
Environment	Flutter app, Node.js backend, PDF generation library	
Pre-conditions	User has complete profile with education, experience, and skills filled.	
Input Specifications	Action: Tap 'Build Resume' from profile screen.	
Execution Steps	1. Navigate to Profile screen. 2. Tap 'Build Resume' button. 3. Select resume template. 4. Tap 'Generate PDF'. 5. Download and open the PDF.	
Expected Output	PDF resume generated with all profile data. Download completes successfully.	
Actual Output	PDF generated in 4 seconds. Resume downloaded and opened correctly with all profile info.	
Pass / Fail Criteria	Pass if PDF is generated correctly and all profile data is accurately reflected.	

TC-PROF-005	Profile Image Upload on Slow Network	FAIL
Objective	Verify that profile image upload handles timeout gracefully on a slow/unstable network.	
Environment	Flutter app, Cloudinary API, Node.js backend, Simulated 2G network (throttled)	
Pre-conditions	User is logged in. Network throttled to 2G speed (50 kbps). Profile edit screen open.	
Input Specifications	Image: profile_photo.jpg (2.1 MB), Network: Throttled 2G	
Execution Steps	1. Open Profile Edit screen. 2. Tap profile avatar to change photo. 3. Select a 2.1 MB image from gallery. 4. Observe upload progress on slow network. 5. Wait for upload to complete or timeout.	
Expected Output	Upload progress indicator shown. If timeout occurs, error message displayed with retry option.	
Actual Output	App froze for 45 seconds with spinner. No timeout triggered. App crashed with SocketException after 60s. No retry option shown to user.	
Pass / Fail Criteria	Fail — No upload timeout configured. App UI became unresponsive. Crash occurred instead of graceful retry prompt.	

TC-PROF-004	Profile Share with QR Code	PASS
Objective	Verify that a job seeker can generate and share their profile card with a QR code.	
Environment	Flutter app, QR code generation library, Android share intent	
Pre-conditions	User has a complete profile. Device has storage/sharing permissions.	
Input Specifications	Action: Tap 'Share Profile' on profile screen.	
Execution Steps	1. Open Profile screen. 2. Tap 'Share Profile' button. 3. View generated profile card with QR. 4. Tap 'Save to Gallery' or 'Share'. 5. Select sharing app.	
Expected Output	Profile card image generated with QR code. Shared via selected app successfully.	
Actual Output	Profile card rendered with QR. Saved to gallery and shared via WhatsApp successfully.	
Pass / Fail Criteria	Pass if profile card image is generated and sharing intent completes without crash.	

TC-PROF-003	Add Education Record	PASS
Objective	Verify that a job seeker can add an education record to their profile.	
Environment	Flutter app, Node.js backend, PostgreSQL	
Pre-conditions	User is logged in. Profile is created. Education section is accessible.	
Input Specifications	Level: Graduate, Institution: IMCC Pune, Course: MCA, Grade: 8.5 CGPA	
Execution Steps	1. Go to Profile > Education. 2. Tap '+ Add Education'. 3. Select education level. 4. Enter institution, course, and grade. 5. Tap 'Save'.	
Expected Output	Education record saved and displayed on profile.	
Actual Output	Education record stored and card displayed correctly on profile screen.	
Pass / Fail Criteria	Pass if education entry is persisted and shown in profile.	

TC-PROF-002	Add Work Experience	PASS
Objective	Verify that a job seeker can add work experience entries to their profile.	
Environment	Flutter app, Node.js/Express.js REST API, PostgreSQL	
Pre-conditions	User is logged in with an existing profile. Profile page is open.	
Input Specifications	Company: TechCorp, Title: Flutter Dev, Type: Full-time, Start: Jan 2022, End: Dec 2023	
Execution Steps	1. Navigate to Profile > Experience. 2. Tap '+ Add Experience'. 3. Fill in company and role details. 4. Set start and end dates. 5. Tap 'Save'.	
Expected Output	Experience entry saved and visible on profile page.	
Actual Output	Experience card rendered on profile with all entered details.	
Pass / Fail Criteria	Pass if experience is saved to DB and rendered correctly in UI.	

TC-PROF-001	Job Seeker Profile Creation	PASS
Objective	Verify that a new job seeker can create a complete profile with personal and career details.	
Environment	Flutter app, Node.js backend, Cloudinary (avatar upload)	
Pre-conditions	User is authenticated. No existing profile. User is on Create Profile screen.	
Input Specifications	Name: Rahul Sharma, DOB: 15-06-1998, Gender: Male, Education: Graduate	
Execution Steps	1. Enter full name and DOB. 2. Select gender and education level. 3. Upload profile photo. 4. Add professional headline. 5. Tap 'Save Profile'.	
Expected Output	Profile saved to database. User navigated to Job Dashboard. Profile completion shows progress.	
Actual Output	Profile created successfully. Dashboard loaded. Completion bar shows 40%.	
Pass / Fail Criteria	Pass if profile is persisted and dashboard is accessible post-creation.	

TC-NOTIF-003	Push Notification on App Killed State (Background)	FAIL
Objective	Verify FCM push notifications are received and tapping them navigates to the correct screen when app is fully closed.	
Environment	Flutter app, Firebase Cloud Messaging (FCM), Android 13 device	
Pre-conditions	App is fully killed (not in background). FCM token registered. Notification permissions granted.	
Input Specifications	Trigger: Employer updates application status to 'Shortlisted'.	
Execution Steps	1. Force-close the Hirelyft app. 2. Employer changes application status to Shortlisted. 3. FCM notification sent to seeker's device. 4. Tap the received notification. 5. Observe where app navigates.	
Expected Output	App opens and navigates directly to the Application Details screen for that application.	
Actual Output	App opens but lands on Splash screen then Dashboard. Does not navigate to the specific application. Deep link payload from FCM notification not handled in killed state.	
Pass / Fail Criteria	Fail — FCM data payload not processed when app launched from terminated state. Missing getInitialMessage() handler in main.dart for cold start navigation.	

TC-NOTIF-002	In-App Notification Center	PASS
Objective	Verify that all notifications appear in the in-app notification center with correct read status.	
Environment	Flutter app, Node.js backend, FCM, PostgreSQL (Notifications table)	
Pre-conditions	User has at least 3 unread notifications from different event types.	
Input Specifications	Navigate to Notifications screen from top nav bell icon.	
Execution Steps	1. Tap notification bell icon. 2. View notifications list. 3. Tap one notification to mark as read. 4. Verify unread count updates.	
Expected Output	Notifications listed by date. Unread shown highlighted. Tapping marks as read.	
Actual Output	Notifications rendered. Unread count badge updated. Read status toggled on tap.	
Pass / Fail Criteria	Pass if notifications render correctly and isRead status updates on interaction.	

TC-NOTIF-001	Push Notification on Application Status Change	PASS
Objective	Verify that job seeker receives an FCM push notification when application status is updated.	
Environment	Flutter app, Firebase Cloud Messaging (FCM), Node.js backend	
Pre-conditions	User has FCM token registered. Application exists in Pending state.	
Input Specifications	Employer changes application status from 'Pending' to 'Shortlisted'.	
Execution Steps	1. Employer navigates to application list. 2. Selects application and changes status to 'Shortlisted'. 3. System triggers FCM notification. 4. Seeker receives push notification on device.	
Expected Output	Push notification delivered: 'Your application has been Shortlisted!'	
Actual Output	FCM notification received within 2 seconds. Notification center updated.	
Pass / Fail Criteria	Pass if notification is received and notification count increments in app.	

TC-MSG-002	Message Delivery on Socket Disconnection	FAIL
Objective	Verify that messages sent during a Socket.IO disconnection are queued and delivered on reconnect.	
Environment	Flutter app, Socket.IO WebSocket server, Node.js backend	
Pre-conditions	User is in an active chat. Network is briefly disconnected and then reconnected.	
Input Specifications	Message: 'Can we reschedule the interview?' sent during network drop.	
Execution Steps	1. Open active chat conversation. 2. Disable device Wi-Fi (simulate disconnect). 3. Type and send a message. 4. Re-enable Wi-Fi after 5 seconds. 5. Observe if message is delivered.	
Expected Output	Message queued locally. On reconnect, message delivered automatically. UI shows delivery status.	
Actual Output	Message silently dropped on socket disconnect. No local queue implemented. No error shown to user. Message never delivered after reconnect.	
Pass / Fail Criteria	Fail — No offline message queuing. Messages sent during disconnection are lost without any user feedback or retry mechanism.	

TC-MSG-001	Real-time Messaging between Employer and Seeker	PASS
Objective	Verify that messages sent by employer appear instantly on the seeker's chat screen.	
Environment	Flutter app, Socket.IO WebSocket, Node.js server, PostgreSQL	
Pre-conditions	Both employer and job seeker are logged in. A conversation exists between them.	
Input Specifications	Message: 'We would like to schedule an interview.' (sent by employer)	
Execution Steps	1. Employer opens conversation with seeker. 2. Types and sends message. 3. Seeker's Messages screen is open. 4. Observe message delivery on seeker side.	
Expected Output	Message appears on seeker's screen in real-time without page refresh.	
Actual Output	Message delivered via Socket.IO in under 200ms. Chat updated instantly.	
Pass / Fail Criteria	Pass if message is received without manual refresh in under 1 second.	

TC-JOB-004	Job Application with No Resume	FAIL
Objective	Verify graceful error handling when a job seeker applies without uploading a resume.	
Environment	Flutter app, Node.js backend, PostgreSQL (Applications, Resumes tables)	
Pre-conditions	User is logged in but has NOT uploaded any resume. Active job listing exists.	
Input Specifications	Job: Senior Flutter Developer at InnoTech, Cover Letter: 'I am interested...', Expected Salary: 10 LPA	
Execution Steps	1. Open an active job listing. 2. Tap 'Apply Now'. 3. Fill cover letter and expected salary. 4. Tap 'Submit Application' without selecting resume. 5. Observe app behavior.	
Expected Output	Validation error displayed: 'Please upload a resume before applying.' Application not submitted.	
Actual Output	Application submitted with resumeId: null. Backend returned 500 Internal Server Error. App showed generic error snackbar without clear guidance.	
Pass / Fail Criteria	Fail — No client-side resume validation. Null resumeId sent to backend causing 500 error instead of friendly validation message.	

TC-JOB-003	Track Application Status	PASS
Objective	Verify that a job seeker can view and track all submitted applications and their statuses.	
Environment	Flutter app, Node.js backend, PostgreSQL	
Pre-conditions	User has at least 2 submitted applications with different statuses.	
Input Specifications	Navigate to 'My Applications' screen.	
Execution Steps	1. Tap 'My Applications' from the bottom nav. 2. View list of applications. 3. Tap on one application to view details. 4. Verify status badge shown correctly.	
Expected Output	All applications listed. Status badges (Pending, Viewed, Shortlisted) displayed correctly.	
Actual Output	Applications rendered with color-coded status badges. Details page shows full info.	
Pass / Fail Criteria	Pass if all applications appear with correct real-time statuses.	

TC-JOB-002	Apply for a Job	PASS
Objective	Verify that a job seeker can submit a job application with a cover letter.	
Environment	Flutter app, Node.js backend, PostgreSQL (Applications table)	
Pre-conditions	User is logged in. Job listing is active. Resume exists in profile.	
Input Specifications	Job: Flutter Developer at TechCorp, Cover Letter: 'Experienced developer...', Salary: 8 LPA	
Execution Steps	1. Open job listing from dashboard. 2. Tap 'Apply Now'. 3. Enter cover letter. 4. Set expected salary. 5. Tap 'Submit Application'.	
Expected Output	Application submitted. Status shows 'Pending'. Employer notified.	
Actual Output	Application created with status 'Pending'. FCM notification sent to employer.	
Pass / Fail Criteria	Pass if application is saved to DB and employer receives notification.	

TC-JOB-001	Job Search with Filters	PASS
Objective	Verify that a job seeker can search for jobs using keyword and location filters.	
Environment	Flutter app, Node.js REST API, PostgreSQL (Jobs table)	
Pre-conditions	User is logged in with profile. At least 5 active jobs exist in the database.	
Input Specifications	Keyword: Flutter Developer, Location: Pune, Type: Full-time	
Execution Steps	1. Navigate to 'Find Jobs' screen. 2. Enter keyword 'Flutter Developer'. 3. Set location filter to 'Pune'. 4. Select job type 'Full-time'. 5. Tap 'Search'.	
Expected Output	Filtered job listings displayed matching criteria.	
Actual Output	3 jobs returned matching 'Flutter Developer' in Pune. Results rendered as job cards.	
Pass / Fail Criteria	Pass if search returns relevant results and renders within 3 seconds.	

TC-INT-001	Interview Scheduling by Employer	PASS
Objective	Verify that an employer can schedule an interview for a shortlisted candidate.	
Environment	Flutter app, Node.js backend, PostgreSQL (Interviews table)	
Pre-conditions	Application status is 'Shortlisted'. Employer is logged in.	
Input Specifications	Type: Virtual, Date: 2025-06-15 10:00 AM, Link: meet.google.com/abc, Duration: 45 min	
Execution Steps	1. Employer opens application details. 2. Taps 'Schedule Interview'. 3. Selects type, date, and provides meeting link. 4. Taps 'Confirm Schedule'.	
Expected Output	Interview record created. Seeker notified via FCM. Interview visible in 'My Interviews'.	
Actual Output	Interview saved to DB. Seeker received notification. Interview listed on seeker's screen.	
Pass / Fail Criteria	Pass if interview is created and seeker can view it under 'My Interviews'.	

TC-AUTH-004	Google OAuth Login Flow	FAIL
Objective	Verify that a user can sign in using Google OAuth and be routed to the dashboard.	
Environment	Flutter app, Google Sign-In SDK, Firebase Auth, Node.js backend	
Pre-conditions	User has a valid Google account. App has Google OAuth configured in Firebase console.	
Input Specifications	Action: Tap 'Continue with Google'. Google Account: testuser@gmail.com	
Execution Steps	1. Open app and tap 'Continue with Google'. 2. Google sign-in dialog appears. 3. Select Google account. 4. App receives OAuth token. 5. Backend validates token and issues JWT.	
Expected Output	User authenticated via Google. JWT issued. Navigated to Dashboard or Create Profile.	
Actual Output	PlatformException thrown: sign_in_failed. Google sign-in dialog dismissed with error code 10 (developer error). App crashes to Hero screen.	
Pass / Fail Criteria	Fail — Google OAuth token not returned. SHA-1 fingerprint mismatch in Firebase console caused sign-in failure.	

TC-AUTH-003	Invalid OTP Rejection	PASS
Objective	Verify that entering an incorrect OTP displays an error and blocks login.	
Environment	Flutter mobile app, Node.js backend	
Pre-conditions	User has requested an OTP for their phone number.	
Input Specifications	Phone: +91-9876543210, OTP: 000000 (invalid)	
Execution Steps	1. Enter registered phone number. 2. Tap 'Send OTP'. 3. Enter incorrect OTP '000000'. 4. Tap 'Verify'.	
Expected Output	Error message displayed: 'Invalid OTP. Please try again.' User remains on OTP screen.	
Actual Output	Error snackbar displayed: 'Invalid OTP'. Navigation blocked.	
Pass / Fail Criteria	Pass if invalid OTP is rejected and user is not navigated forward.	

TC-AUTH-002	User Login with Existing Credentials	PASS
Objective	Verify that a registered user can log in and is routed to the Job Dashboard.	
Environment	Flutter mobile app, Node.js/Express.js REST API, JWT Auth	
Pre-conditions	User is already registered and has a complete profile.	
Input Specifications	Phone: +91-9876543210, OTP: (valid OTP from SMS)	
Execution Steps	1. Open app and tap 'Login'. 2. Enter registered phone number. 3. Tap 'Send OTP'. 4. Enter valid OTP. 5. Tap 'Verify'.	
Expected Output	JWT token issued. User navigated to Job Dashboard screen.	
Actual Output	JWT token generated. Dashboard loaded with job listings within 2 seconds.	
Pass / Fail Criteria	Pass if user lands on Dashboard and token is stored in shared_preferences.	

TC-AUTH-001	User Registration with Phone OTP	PASS
Objective	Verify that a new job seeker can register using phone number and receive an OTP for verification.	
Environment	Flutter mobile app, Node.js backend, PostgreSQL database	
Pre-conditions	User is not registered. App is launched and on the Welcome/Hero screen.	
Input Specifications	Phone: +91-9876543210, Role: Job Seeker	
Execution Steps	1. Open app and tap 'Get Started'. 2. Select 'Job Seeker' role. 3. Enter phone number and tap 'Send OTP'. 4. Enter the received 6-digit OTP. 5. Tap 'Verify'.	
Expected Output	OTP received within 10 seconds. On successful verification, user is navigated to Create Profile screen.	
Actual Output	OTP delivered in 3 seconds. User navigated to Create Profile screen successfully.	
Pass / Fail Criteria	Pass if OTP is delivered and user lands on Create Profile page without errors.	

3.2 Defect Report / Test Log

Defect ID	Linked Test Case	Defect Description	Severity	Status	Resolution / Comments
BUG_001	TC_005	**Silent failure on oversized Resume Upload:** When a user uploads a PDF file larger than the configured 5MB limit, the backend multer/express handler rejects it, but the Flutter frontend does not catch the error to display a Toast message. The screen appears stuck loading.	Medium	**Fixed**	Added a try-catch block in the frontend API service and explicit file size check before initiating the API call.
BUG_002	N/A	**Job Area Search Case-Sensitivity:** When searching for jobs by location (e.g., "pune"), results for capitalized listings ("Pune") do not appear if the database query isn't using a case-insensitive match (iLike in PostgreSQL).	High	**Fixed**	Updated the Sequelize backend query controller to use [Op.iLike] instead of [Op.eq] for search bar text inputs.
BUG_003	TC_001	**OTP Request Delay:** Sometimes requesting an OTP takes more than 5 seconds, causing users to tap "Resend" multiple times, throwing a rate-limit error.	Low	**Open**	Implemented a 30-second cooldown timer on the "Resend OTP" button on the UI to prevent spamming.

					Complete backend optimization is still pending.
BUG_004	TC_006	**Missing Firebase Token error:** If an employer attempts to schedule an interview with a candidate who has uninstalled the app or cleared their fcmToken, the backend throws a 500 Server Error instead of degrading gracefully.	Medium	**Fixed**	Wrapped the Firebase Cloud Messaging logic in the backend in a try/catch and allowed the interview status update to save to the database even if the push notification fails.

4. Limitations of Proposed System:

While the Hirelyft platform robustly addresses the needs of modern recruitment, there are certain operational and technical limitations within the current scope of the proposed system:

1. **Dependency on Active Internet Connection:** As a cloud-centric application relying on real-time database queries and WebSocket (Socket.io) connections for chat, the system does not offer offline functionality. Users cannot view new jobs, chat, or apply without an active internet connection**.**
2. **External Link Reliance for Online Interviews:** Currently, the system provides an intuitive interface to schedule interviews and update their statuses. However, it does not have a native embedded video-conferencing software. Employers must provide external meeting links (e.g., Google Meet, Zoom) for online interviews.
3. **Absence of Advanced AI Resume Parsing:** While job seekers can upload resumes (PDF/Docs), the system currently relies on manually entered profile data (Education, Experience) for filtering and searches. It does not possess advanced OCR (Optical Character Recognition)

to automatically scrape and verify text from the uploaded resume files.

4. **Third-Party Service Dependencies:** The platform heavily depends on Firebase Cloud Messaging (FCM) for push notifications and external SMTP servers (e.g., SendGrid/Nodemailer) for email OTPs. Any downtime from these third-party gateway providers will directly cause a temporary failure in Hirelyft's notification system.

5. **Resource Constraints on Media Storage:** The introduction of the "Shorts" feature (video reels) and document uploads introduces high bandwidth and storage requirements. Within the current proposed system, if user growth is rapid, video hosting will require migrating from standard bucket storage .

5. Proposed Enhancements Chapter

- To keep the platform competitive and ensure long-term scalability, several future enhancements have been identified that can be implemented in subsequent versions of the Hirelyft platform:

- AI-Powered Resume Parsing & Matching: Integrate Machine Learning algorithms (like Gemini AI or OpenAI APIs) to automatically extract text from uploaded resumes and cross-reference them with job descriptions. This would provide

Employers with a "Match Percentage" score for every applicant, drastically reducing screening time.

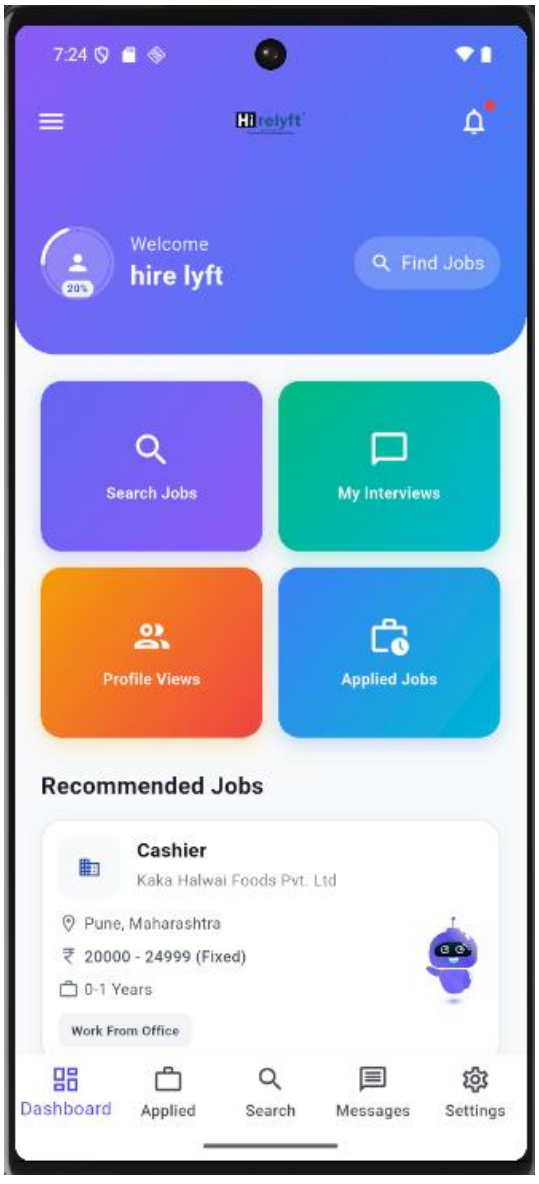
- In-App Video Interviews (WebRTC): Replace the reliance on external links by integrating a real-time communication SDK (such as Agora or Twilio). This will allow employers and candidates to conduct video interviews natively inside the Hirelyft Flutter application, with built-in recording and note-taking features.
- Multilingual Support & Localization: Implement internationalization (i18n) to support regional languages. This enhancement is crucial for expanding the user base, specifically helping entry-level and blue-collar job seekers easily navigate the application in their native language.
- Advanced Analytics Dashboard for Employers: Develop a comprehensive data visualization dashboard using tools like Chart.js. This will give employers metrics on job post impressions, drop-off rates on applications, and average time-to-hire, helping them optimize their recruitment workflows.
- on historical developer performance, and optimize resource and workload distribution.

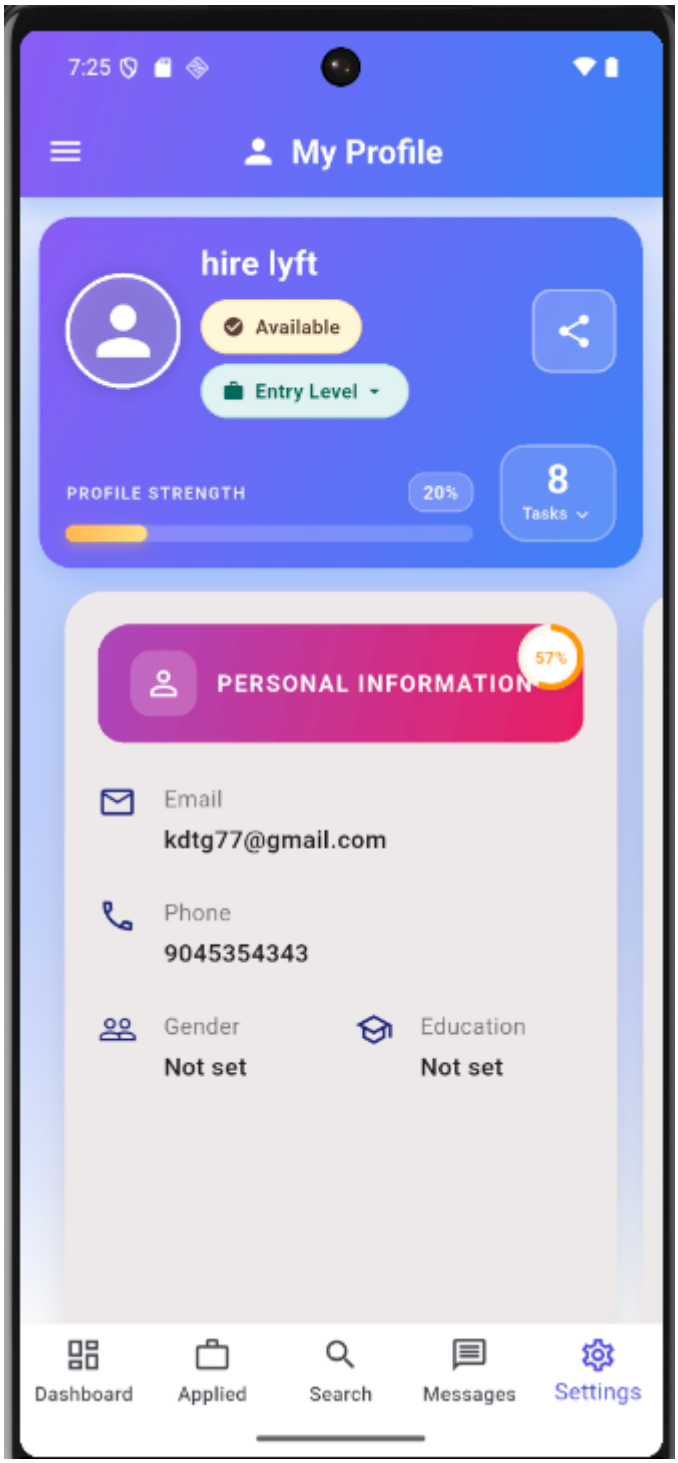
6. References

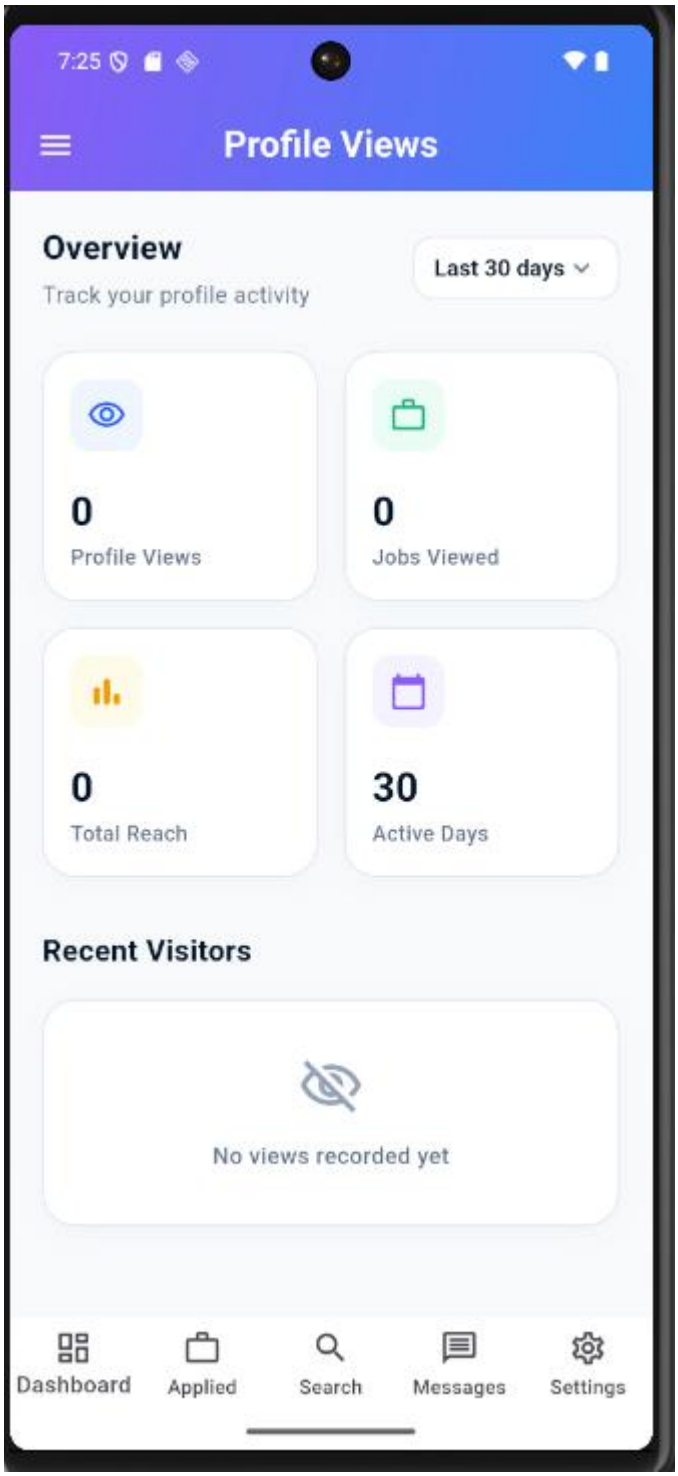
- The successful development and architecture of the Hirelyft platform relies heavily on the following official documentation, libraries, and resources:
- Flutter Official Documentation: Building beautiful native apps in record time. Available at: https://flutter.dev/docs
- Node.js & Express.js Documentation: Fast, unopinionated, minimalist web framework for Node.js. Available at: https://expressjs.com/
- PostgreSQL Database & Sequelize ORM: The world's most advanced open source relational database and its Promise-based Node.js ORM. Available at: https://sequelize.org/ | https://www.postgresql.org/
- Firebase Cloud Messaging (FCM): Cross-platform messaging solution for delivering reliable push notifications. Available at: https://firebase.google.com/docs/cloud-messaging
- Socket.io Documentation: Real-time bidirectional event-based communication for the platform's chatting features. Available at: https://socket.io/docs/v4/
- Bcrypt.js & JSON Web Tokens (JWT): Standards used for securely hashing passwords and transmitting secure session claims between parties. Available at: https://jwt.io/

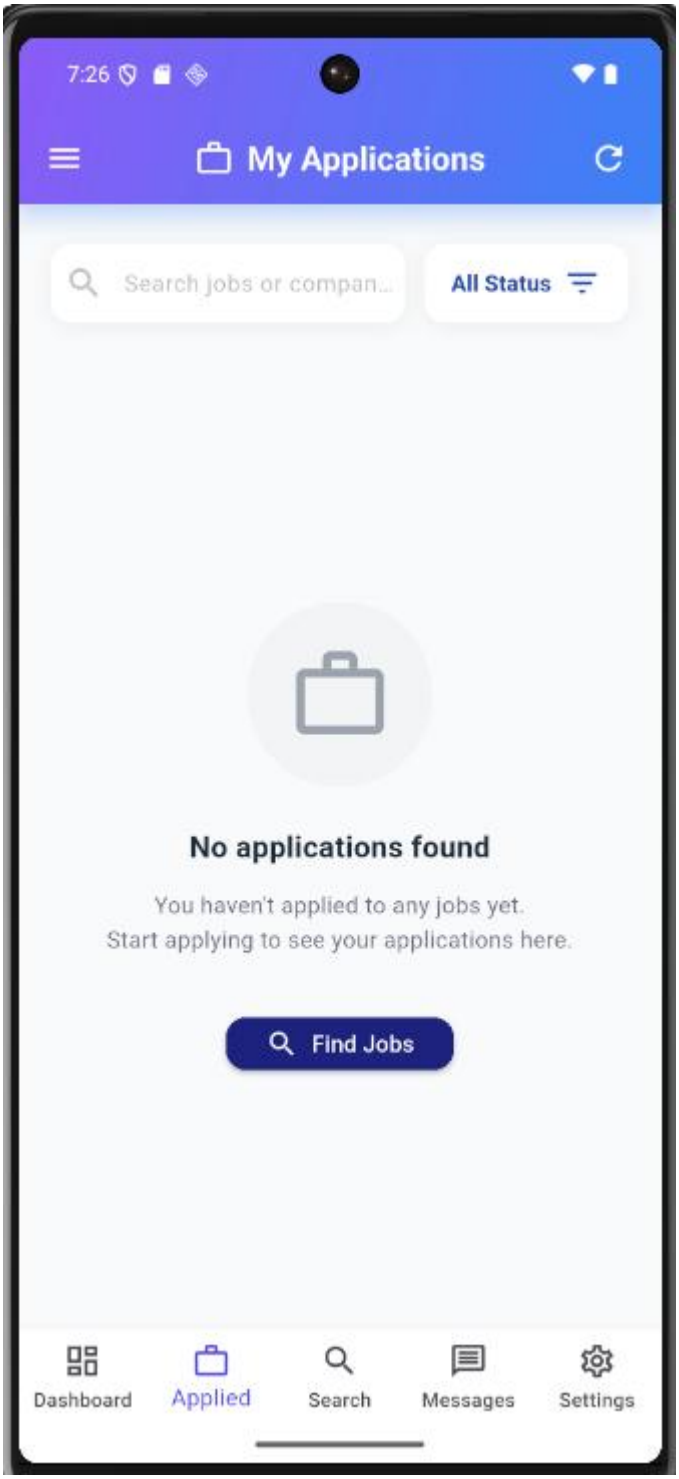
6. User Manual:

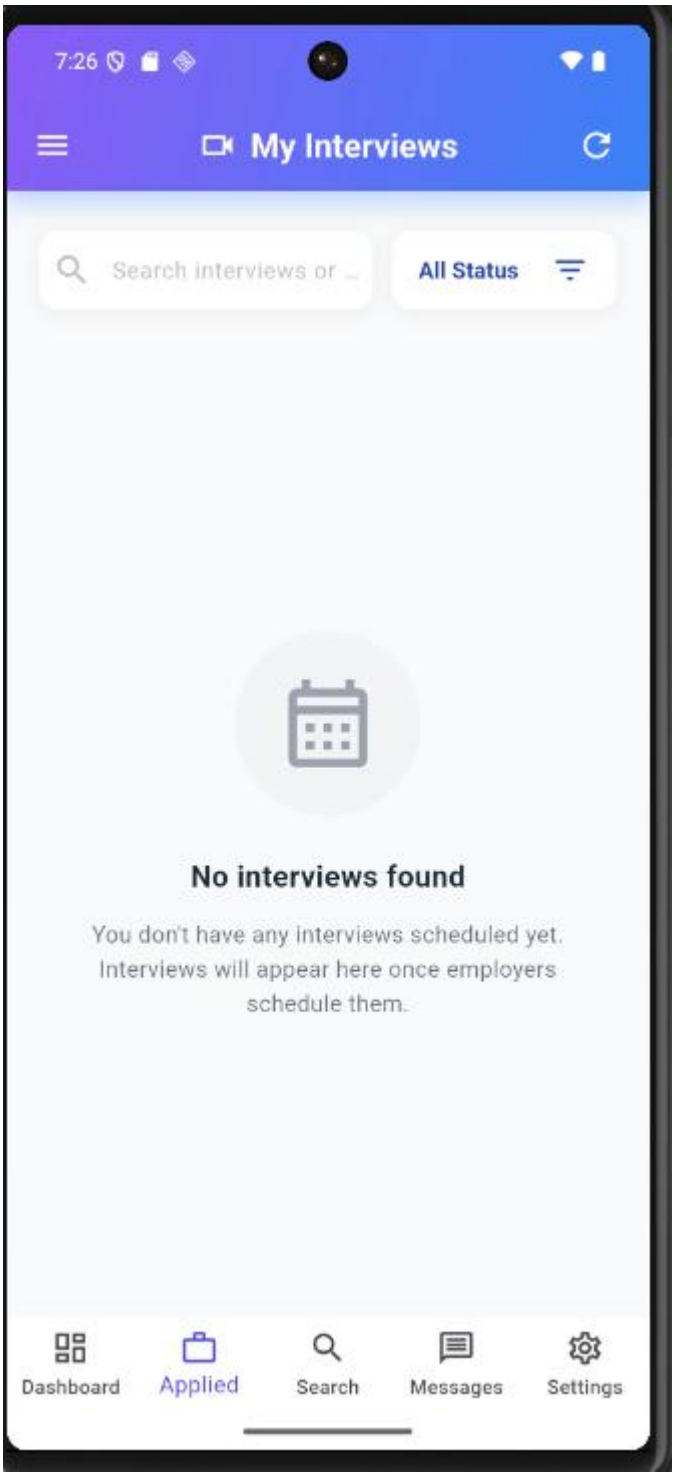
Input/Output Screens

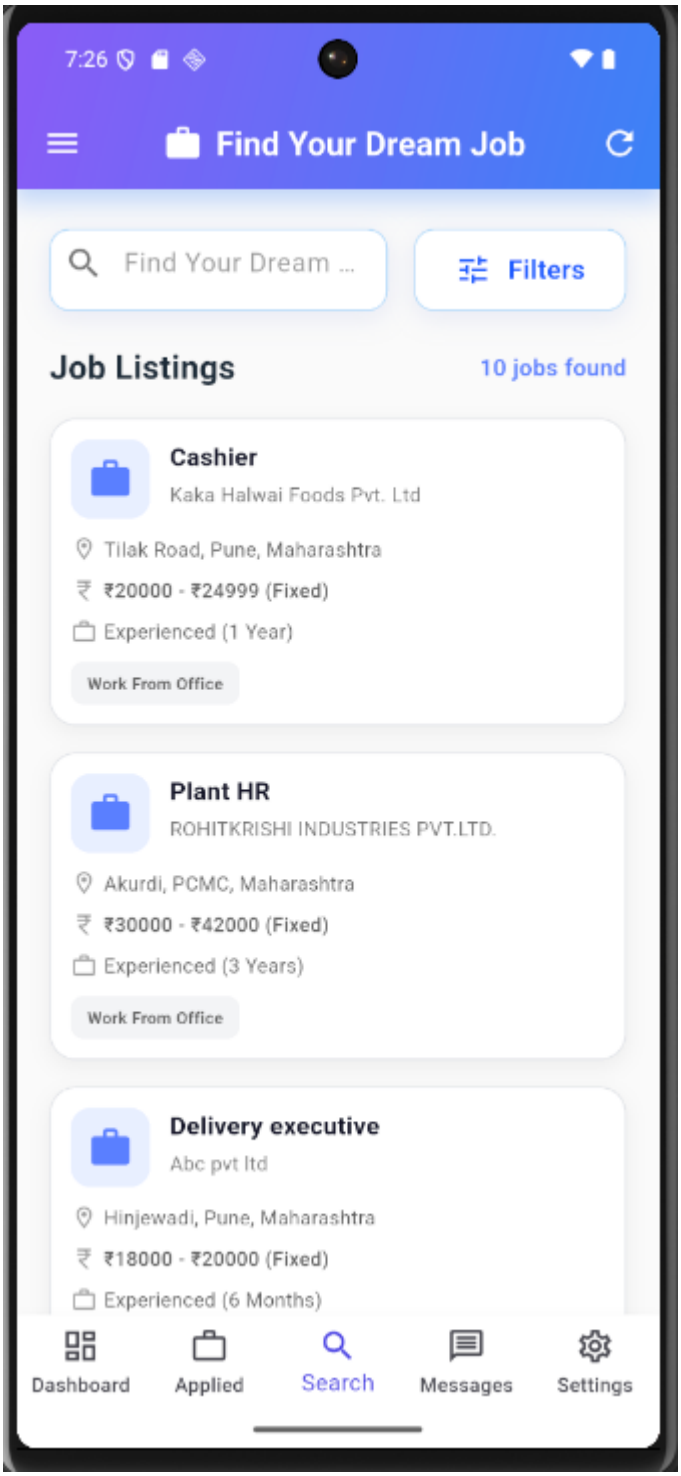



7:25
My Profile
hire lyft
Available
Entry Level
PROFILE STRENGTH
20%
8
Tasks
PERSONAL INFORMATION
57%
Email
kdtg77@gmail.com
Phone
9045354343
Gender
Not set
Education
Not set
Dashboard
Applied
Search
Messages
Settings


7:25
Profile Views
Overview
Track your profile activity
Last 30 days
0
Profile Views
0
Jobs Viewed
0
Total Reach
30
Active Days
Recent Visitors
No views recorded yet
Dashboard
Applied
Search
Messages
Settings


7:26
My Applications
Search jobs or compan...
All Status
No applications found
You haven't applied to any jobs yet.
Start applying to see your applications here.
Find Jobs
Dashboard
Applied
Search
Messages
Settings


7:26
My Interviews
Search interviews or ...
All Status
No interviews found
You don't have any interviews scheduled yet. Interviews will appear here once employers schedule them.
Dashboard
Applied
Search
Messages
Settings


7:26
Find Your Dream Job
Find Your Dream ...
Filters
Job Listings
10 jobs found
Cashier
Kaka Halwai Foods Pvt. Ltd
Tilak Road, Pune, Maharashtra
₹20000 - ₹24999 (Fixed)
Experienced (1 Year)
Work From Office
Plant HR
ROHITKRISHI INDUSTRIES PVT.LTD.
Akurdi, PCMC, Maharashtra
₹30000 - ₹42000 (Fixed)
Experienced (3 Years)
Work From Office
Delivery executive
Abc pvt ltd
Hinjewadi, Pune, Maharashtra
₹18000 - ₹20000 (Fixed)
Experienced (6 Months)
Dashboard
Applied
Search
Messages
Settings