Project Proposal

On

CareerConnect

Guided By:-

Anuj Kumar

Created By:-

Diwakar Sharma

AF05014352

ANP-D2404

WDR-1

Table Of Contents

Index

INTRODUCTION

Career-Connect is a full-stack web-based job portal application designed to bridge the gap between **job seekers** and **recruiters** by offering an interactive and user-friendly platform. The system enables recruiters to post job vacancies, manage applications, review resumes, and update hiring status, while job seekers can search relevant job openings, apply online, upload resumes, and track their application status in real-time.

The application integrates **React.js** for building a responsive and dynamic user interface, **Tailwind CSS and Shadcn UI** for modern styling, **Clerk Authentication** for secure login using email/password and Google OAuth, and **React Hook Form with Zod validation** for accurate and error-proof data entry.

On the backend, **MySQL database** is used to efficiently store and manage user profiles, job postings, applications, saved jobs, and resume records.
The system also uses **Supabase storage** for file uploads such as resumes and company logos.

Career-Connect is designed to support **two primary user roles**: **Recruiter** and **Candidate**. Recruiters can create company profiles, post job openings, update job status (open/closed), and track candidate applications.
Candidates can build their professional profiles, upload resumes, explore job opportunities filtered by skills, experience, and location, and submit applications effortlessly.

The system also includes key features like **job search filters, job bookmarking (save jobs), resume upload, application status tracking, secure authentication, and responsive UI design**, making it suitable for both desktop and mobile environments. By integrating various technologies, Career-Connect provides an efficient and organized recruitment solution with improved accessibility, automation, and real-time communication.
In conclusion, **Career-Connect** not only simplifies the recruitment cycle but also enhances the overall experience of hiring and job searching through smart automation, **real-time tracking**, and effective digital collaboration between recruiters and candidates.

OBJECTIVES OF THE STUDY

The main objective of Career-Connect is to develop a full-stack job portal using modern web technologies that facilitates seamless interaction between recruiters and job seekers through online job posting, resume submission, smart search functionality, real-time application tracking, and efficient data management using MySQL, while ensuring user experience, security, and system scalability.

Primary Objectives

- To develop an interactive web-based job portal that **connects recruiters** and **job seekers** on a single unified platform.
- To enable secure **authentication** with **email/password** and **Google login** using **Clerk authentication** service, ensuring data privacy and role-based access control.
- To provide **recruiters** with a dedicated interface to **post jobs**, **edit** job details, **manage** applications, and **update** job status (open/closed).
- To allow candidates to **search jobs**, **filter** by skills, location, company, and job type, **apply online**, **upload resumes**, **save jobs**, and **track application** status.
- To implement **MySQL** as the backend **database** for storing and managing user profiles, job details, applications, companies, and resume data efficiently.

Functional Objectives

- To provide a **responsive, user-friendly UI** using React.js, Tailwind CSS, and Shadcn UI for recruiters and candidates on both desktop and mobile devices.
- To allow **resume upload functionality**, storing files securely using Supabase or Cloud storage.
- To implement **real-time application status tracking**, allowing recruiters to update and candidates to monitor their job application progress such as **Pending**, **Reviewed**, **Selected**, or **Rejected**.
- To maintain a **Saved Jobs (Bookmark Module)**, so candidates can save job openings for future consideration.

PROJECT CATEGORY

This project belongs to the category of a **Web-Based Full Stack Application**.

It is an **online job portal system** developed using **React.js for the frontend, Supabase/MySQL for the database, and Clerk for authentication**.

It is a **database-driven, role-based application** where both **job seekers and recruiters** can interact through the website.

It is a **client-server model software**, meaning users access it through a browser, and all data is stored securely in a central database.

ANALYSIS

MODULES & DESCRIPTION

A. Authentication Module

- User **signup/login** using **Clerk**.
- Authentication via **Google/email** & **password**.
- Role-based access: **Recruiter**, **Candidate**.

B. Recruiter Module

- Post new job vacancies.
- **Edit/update/delete** job postings.
- Change job status: **Open / Closed**.
- View applications received for each job.

C. Job Seeker Module

- Search jobs by **location**, **title**, **skill**, or **company**.
- Apply for jobs by submitting a **resume** & **details**.
- **Track application** status (**Pending**, **Reviewed**, **Selected**, **Rejected**).
- **Save/Bookmark** jobs for later.

D. Resume Upload & Profile Module

- **Upload** resume (PDF).
- Add **experience, skills**, and **education**.
- **Edit** personal profile.

F. Application Tracking Module

- **Candidate**: track status (Pending → Reviewed → Selected/Rejected).
- **Recruiter**: update application status.

DATABASE DESIGN

1. Companies Table

Field Name	Datatype	Key	Description
id	INT	Primary Key	Unique ID for each company
created_at	TIMESTAMP		Record creation timestamp
name	TEXT		Company name
logo_url	TEXT		Company logo or image URL

2. Jobs Table

Field Name	Data Type	Key	Description
id	INT	Primary Key	Unique ID for each job
created_at	TIMESTAMP		Job posting date/time
recruiter_id	TEXT	Foreign Key -> user.id	Job posted by recruiter (user id)
title	TEXT		Job title/position
company_id	INT	Foreign Key -> companies.id	Company associated with the job
description	TEXT		Job description
location	TEXT		Job location (city/remote)
requirements	TEXT		Skills/qualifications
isOpen	BOOLEAN		Job status (open/closed)

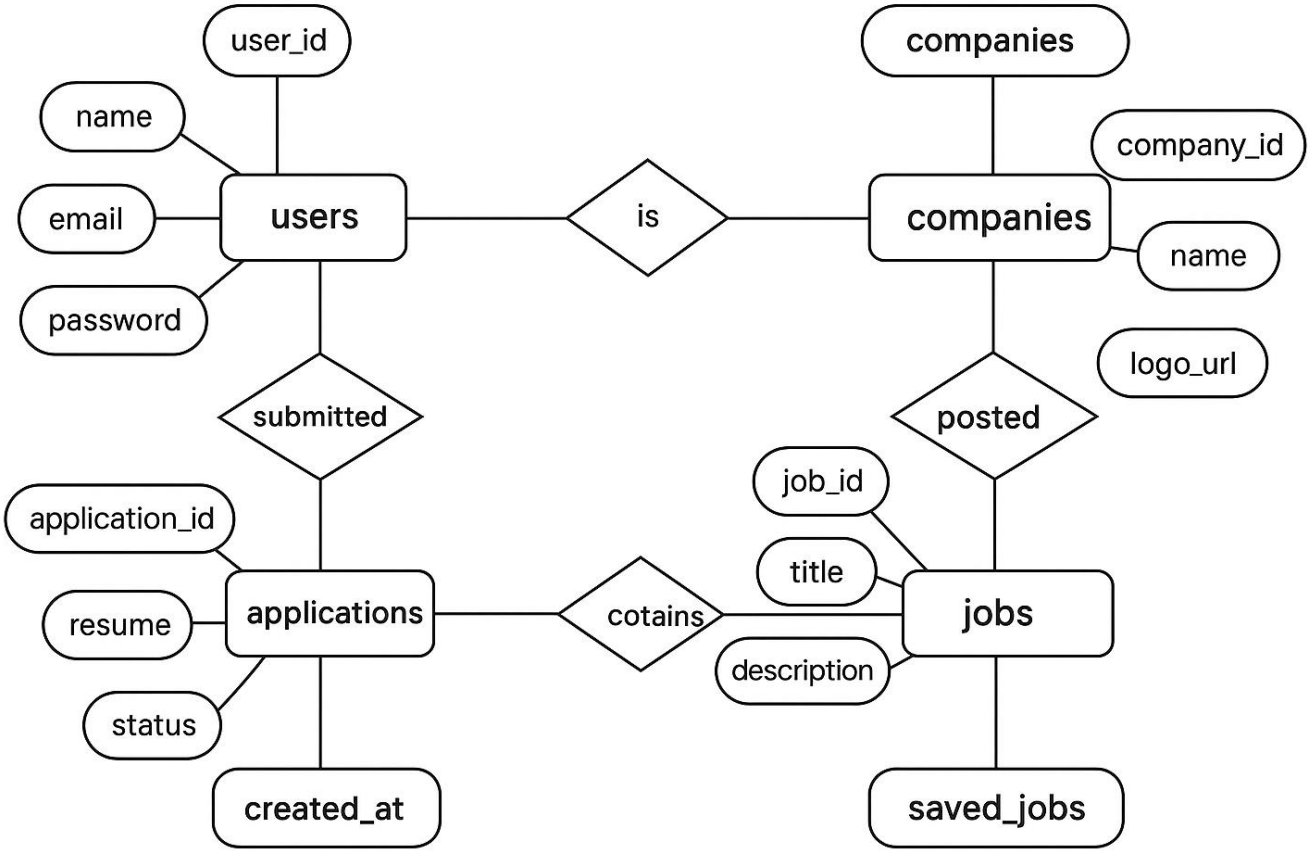
3. Applications Table

Field Name	Data Type	Key	Description
id	INT	Primary Key	Unique ID for each application
created_at	TIMESTAMP		Application date/time
job_id	INT	Foreign Key -> jobs.id	Job for which applied
candidate_id	TEXT	Foreign Key -> users.id	Candidate who applied
status	TEXT		(Pending/Reviewed/Selected/Rejected)
resume	TEXT		URL of uploaded resume
skills	TEXT		Skills of candidate
experience	TEXT		Experience details
education	TEXT		Educational qualifications
name	TEXT		Name of applicant

4. Saved_jobs Table

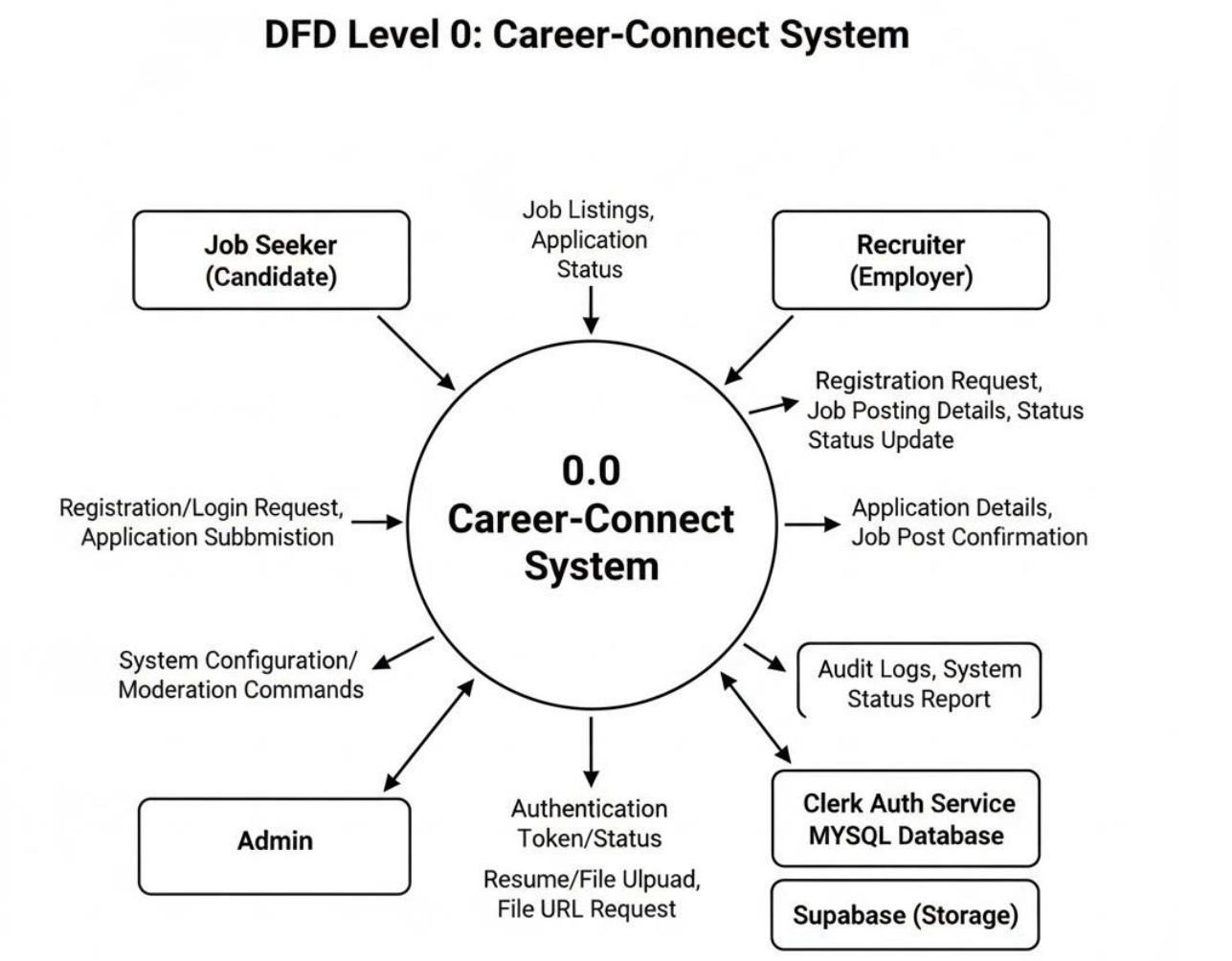
Field Name	Datatype	Key	Description
id	INT	Primary Key	Unique ID for saved job record
created_at	TIMESTAMP		Date/time when saved
user_id	TEXT	Foreign Key -> users.id	Candidate who saved the job
job_id	INT	Foreign Key -> jobs.id	Job that was saved

ER DIAGRAM

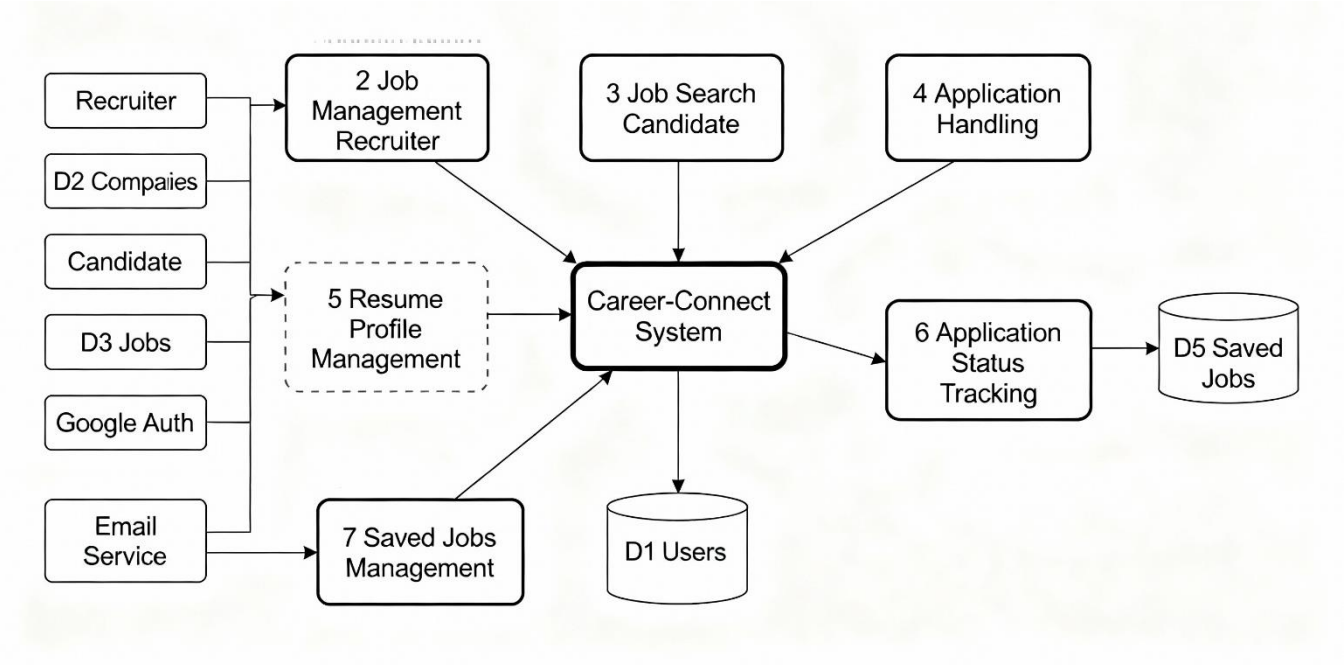

user_id
name
email
password
users
is
companies
company_id
name
logo_url
companies
submitted
posted
application_id
resume
status
applications
cotains
job_id
title
description
jobs
created_at
saved_jobs

DATA FLOW DIAGRAM

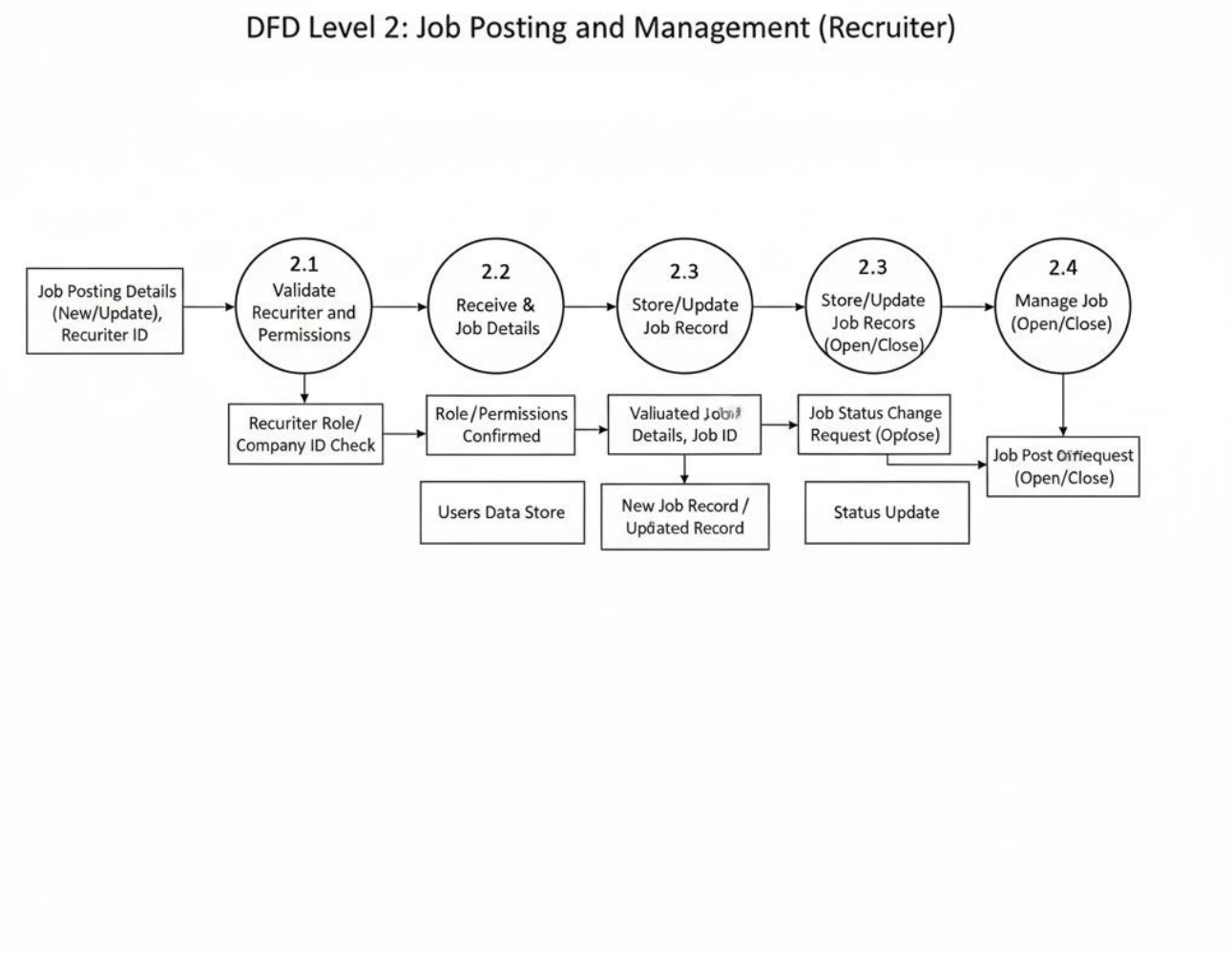
1. 0 – Level DFD



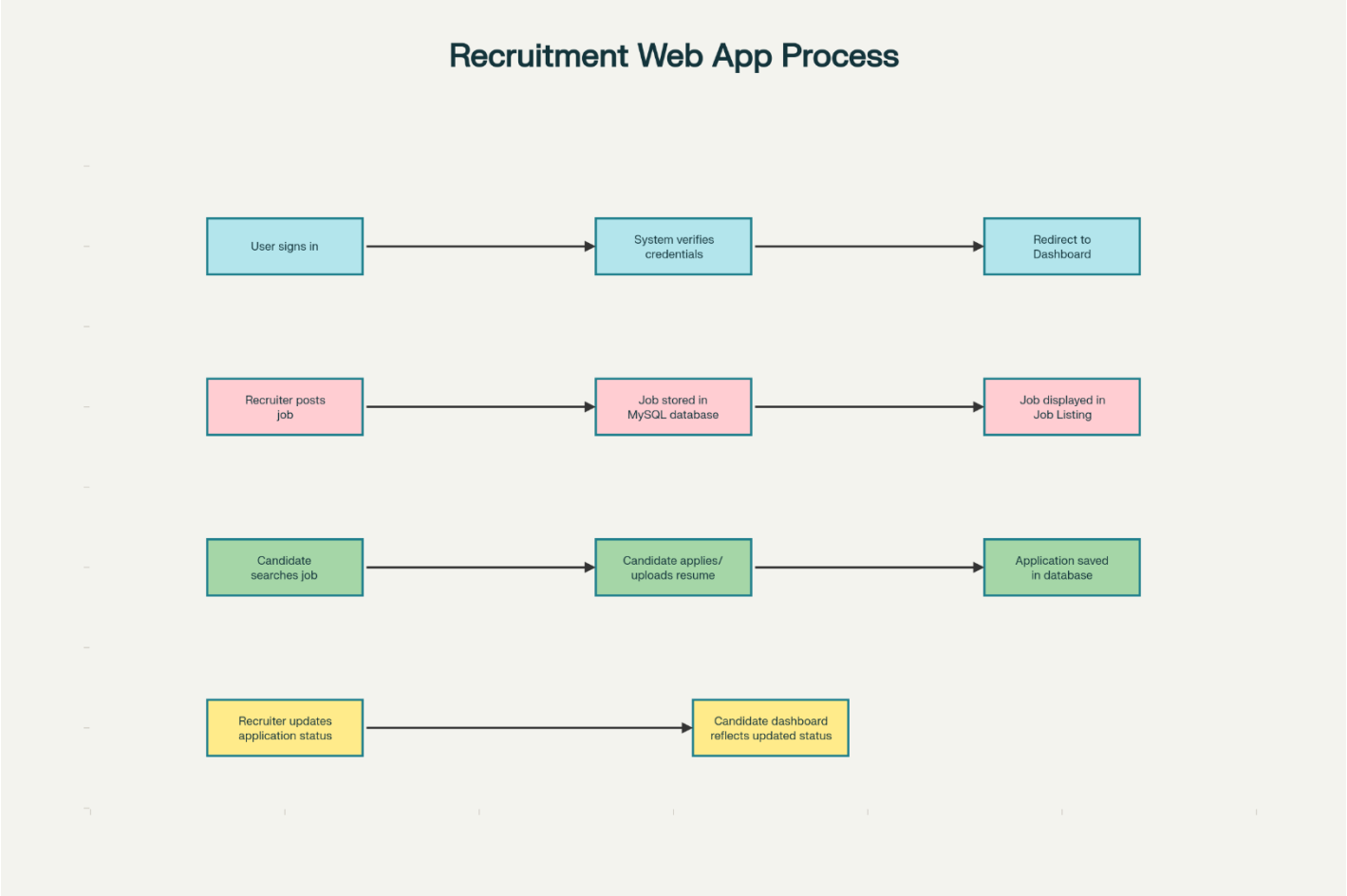
2. 1-Level DFD


Recruiter
D2 Compaiies
Candidate
D3 Jobs
Google Auth
Email Service
2 Job Management Recruiter
3 Job Search Candidate
4 Application Handling
5 Resume Profile Management
Career-Connect System
6 Application Status Tracking
D5 Saved Jobs
7 Saved Jobs Management
D1 Users

3. 2-Level DFD


DFD Level 2: Job Posting and Management (Recruiter)
Job Posting Details (New/Update), Recuriter ID
2.1 Validate Recuriter and Permissions
2.2 Receive & Job Details
2.3 Store/Update Job Record
2.3 Store/Update Job Recors (Open/Close)
2.4 Manage Job (Open/Close)
Recuriter Role/ Company ID Check
Role/Permissions Confirmed
Valiuated Job Details, Job ID
Job Status Change Request (Oplose)
Job Post Offrequest (Open/Close)
Users Data Store
New Job Record / Updated Record
Status Update

COMPLETE STRUCTURE



PLATFORM USED

Software:

- React 19 – Component-based UI
- Tailwind CSS – UI styling
- Shadcn UI – Component library
- Supabase
- MYSQL
- Clerk – Authentication
- GitHub – Version control
- VS Code – Development environment

Hardware Requirements:

Personal Laptop with:

- 8GB RAM
- Intel i3
- Stable internet connection

Development Environment:

- Windows 11
- VSCode
- Node.js V22.17.0
- Browser (Chrome)

Deployment:

- Vercel

FUTURE SCOPE

- AI-based resume screening
- Chatbot for interview preparation
- Video interview integration
- Real-time chat between Job seekers and recruiters
- Automated offer letter generation
- Mobile app (React Native)

BIBLIOGRAPHY

1. React.js official documentation
2. Tailwind CSS documentation
3. Supabase & MySQL documentation
4. Clerk Authentication documentation
5. MDN Web documentation
6. StackOverflow