

1. SYNOPSIS

The “**Job Portal**” provides a comprehensive guide to understanding and utilizing a sophisticated online job portal built using Laravel/PHP with MySQL database. This documentation serves as a roadmap for users, companies, and administrators, offering insights into the website's features, architecture, and functionalities.

1.1 Introduction:

The introduction outlines the purpose and objectives of the project, targeting a broad audience interested in exploring or utilizing the job portal website. It provides a brief overview of the website's functionalities and highlights its significance in connecting job seekers with potential employers.

1.2 Features:

This section delves into the various features offered by the job portal website, categorizing them based on user roles (User, Company, Admin). It describes how users can browse job listings, apply for jobs, and interact with posts and other users. Additionally, it outlines the capabilities of companies in posting jobs, managing their profile, and engaging with users. Admin functionalities, such as user and job management, are also detailed.

1.3 Architecture and Technologies Used:

The architecture section provides insights into the underlying technology stack, including Laravel/PHP framework and MySQL database. It explains the project's structure, emphasizing Laravel's MVC architecture and the database schema.

1.4 User Guide:

The user guide offers step-by-step instructions for users, companies, and administrators on how to navigate and utilize the job portal website effectively. It covers account registration, login procedures, job application processes, and interaction with posts and other users.

1.5 Conclusion:

In conclusion, the documentation summarizes the key aspects of the job portal website, highlighting its impact and significance in facilitating job search and recruitment processes. It encourages users to explore the website further and offers contact information for additional support or inquiries.

The “**Job Portal**” serves as a comprehensive resource for understanding, utilizing, and further developing the job portal website, catering to the diverse needs of users, companies, and administrators alike.

2. PREAMBLE

2.1 General Introduction

The Job Portal is a platform that connects job seekers with potential employers. The website offers a range of features for users, companies, and administrators, and is built using Laravel/PHP with a MySQL database. This documentation provides an overview of the system, its functionality, and how it was developed.

2.2 Statement of Problem

Finding a job or hiring the right candidate can be a challenging and time-consuming process. Job seekers often struggle to find relevant job opportunities, while employers struggle to find qualified candidates. The Job Portal Website aims to address this problem by providing a platform that connects job seekers with potential employers.

2.3 Objective and Scope of the Study

The primary objective of the Job Portal Website is to create a dynamic and interactive platform that streamlines the job search and recruitment process. Key objectives include:

- Providing users with a user-friendly interface for browsing job listings, applying for jobs, and interacting with other users.
- Empowering companies to post job openings, manage their profiles, and engage with potential candidates.
- Equipping administrators with tools for managing users, verifying companies, and overseeing job postings.

The scope of the study encompasses the development and implementation of the Job Portal Website, focusing on its core functionalities and features to meet the needs of users, companies, and administrators.

2.4 Module Description with functionality

The Job Portal Website comprises several modules, each offering distinct functionalities tailored to specific user roles:

- **User Module:** Enables users to browse job listings, apply for jobs, save job listings, follow other users and companies, post content, and engage with posts through comments and likes.
- **Company Module:** Empowers companies to post job openings, manage their company profiles, view user interactions, and engage with potential candidates.
- **Admin Module:** Provides administrators with tools for managing users, verifying companies, moderating content, and overseeing the overall functionality of the website.

2.5 Methodology

The development of the Job Portal Website follows an iterative and collaborative approach, leveraging the Agile methodology. Key phases of the development process include requirements gathering, design, implementation, testing, and deployment. Continuous feedback and iteration ensure that the website meets the evolving needs of its users.

2.6 Feasibility Study

A feasibility study was conducted to assess the viability of this project. The study considered technical, economic, and operational factors. The chosen technologies (Laravel/PHP, MySQL) are widely used and well-supported, making the project technically feasible. The economic feasibility was determined by the use of open-source technologies and the potential return on investment for businesses utilizing the platform. Finally, the operational feasibility considered the ease of use for both job seekers and employers, ensuring smooth website operation.

3. REVIEW OF LITERATURE

The Review of Literature summarizes key findings and trends in the field of online job portals, providing valuable insights for the development of the Job Portal Website:

- **Evolution of Online Job Portals:**

From basic job listings to advanced platforms offering personalized recommendations and social networking integration.

- **User Experience in Job Portals:**

Emphasizes the importance of intuitive interfaces and personalized features to enhance user satisfaction and engagement.

- **Role of Social Media in Recruitment:**

Highlights the impact of social media platforms in sourcing talent and building employer branding.

- **Technology Trends:**

Explores advancements in AI, machine learning, and NLP for automating recruitment processes and improving efficiency.

- **Security and Privacy Concerns:**

Addresses the need for robust security measures and compliance with data protection regulations to protect user data.

- **Emerging Trends:**

Discusses the integration of VR, blockchain, and gig economy platforms as emerging trends shaping the future of online job portals.

The Review of Literature informs the design and implementation of the Job Portal Website, ensuring it aligns with industry best practices and addresses current challenges and opportunities in the field of online recruitment.

4. TECHNICAL DESCRIPTION

This section details the hardware and software requirements necessary to run the Job Portal Website.

4.1 Hardware Requirement

The hardware requirements for this website are relatively modest and can vary depending on the expected user traffic. Here's a general guideline:

- **Processor:** Intel Pentium IV or equivalent (multi-core processor recommended for higher traffic)
- **RAM:** 4 GB minimum (8 GB or more recommended for optimal performance)
- **Hard Disk Space:** 50 GB minimum (additional space needed for data storage as the website grows)
- **Internet Connection:** Reliable internet connection with good bandwidth

4.2 Software Requirement

The website's functionality relies on the following software components:

- **Web Server:** Apache, Nginx, or any other compatible web server software.
- **PHP:** Version 7.4 or higher (check your web hosting provider's supported versions).
- **MySQL Database:** Version 5.6 or higher to store website data.
- **Laravel Framework:** Version 8 or higher for efficient web application development.
- **Composer:** Dependency management tool for PHP (used to install Laravel and other required packages).

5. SYSTEM DESIGN AND DEVELOPMENT

5.1 Architectural Design / System Flow

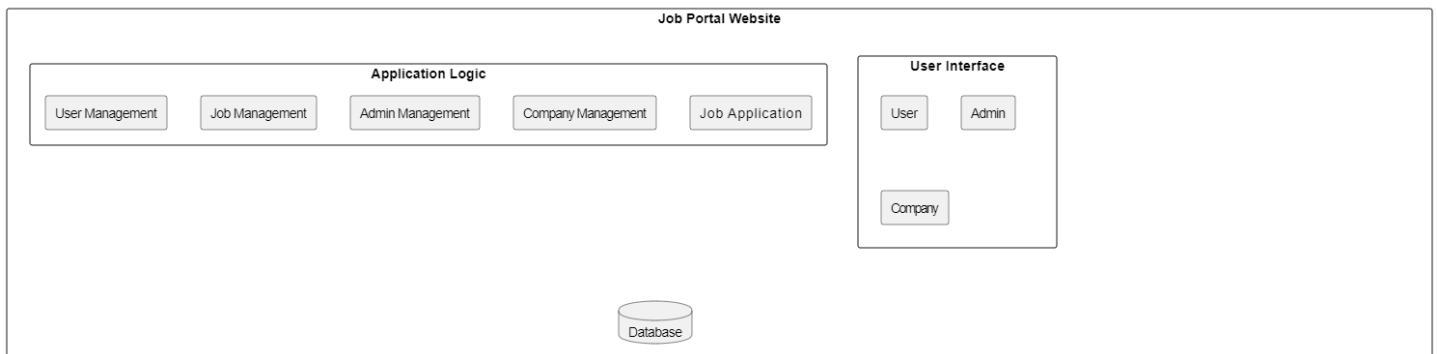
The Job Portal Website follows a layered architectural design, leveraging the Model-View-Controller (MVC) architecture provided by the Laravel PHP framework.

- **Client-Side Interaction:**
 - Users, companies, and administrators interact with the website through their respective user interfaces, accessing various features and functionalities.
 - Client-side technologies such as HTML, CSS, and JavaScript handle the presentation and user interaction aspects of the website.
- **Server-Side Processing:**
 - Requests from clients are routed to the appropriate controllers within the Laravel application.
 - Controllers process the requests, interact with the database, and return responses to the clients.
 - Business logic and application rules are implemented within the controllers, ensuring the integrity and security of the system.
- **Data Storage and Retrieval:**
 - Data is stored and managed in the MySQL database, following the defined database schema.
 - Models in Laravel represent database tables and encapsulate data manipulation operations, providing an abstraction layer for interacting with the database.
 - Eloquent ORM (Object-Relational Mapping) simplifies database operations by mapping database records to PHP objects and vice versa.
- **Presentation Layer:**
 - Views in Laravel are responsible for rendering HTML content to be displayed to users.
 - Blade templating engine facilitates the creation of dynamic and reusable view templates, allowing for efficient content presentation and layout customization.
 - Views are populated with data retrieved from controllers and models, providing users with dynamic and interactive user interfaces.
- **Routing and Middleware:**
 - Laravel's routing mechanism maps incoming requests to corresponding controller actions based on defined routes.
 - Middleware intercepts and processes requests before they reach the controllers, enabling cross-cutting concerns such as authentication, authorization, and request validation.

5.2 Data Flow Diagram (DFD)

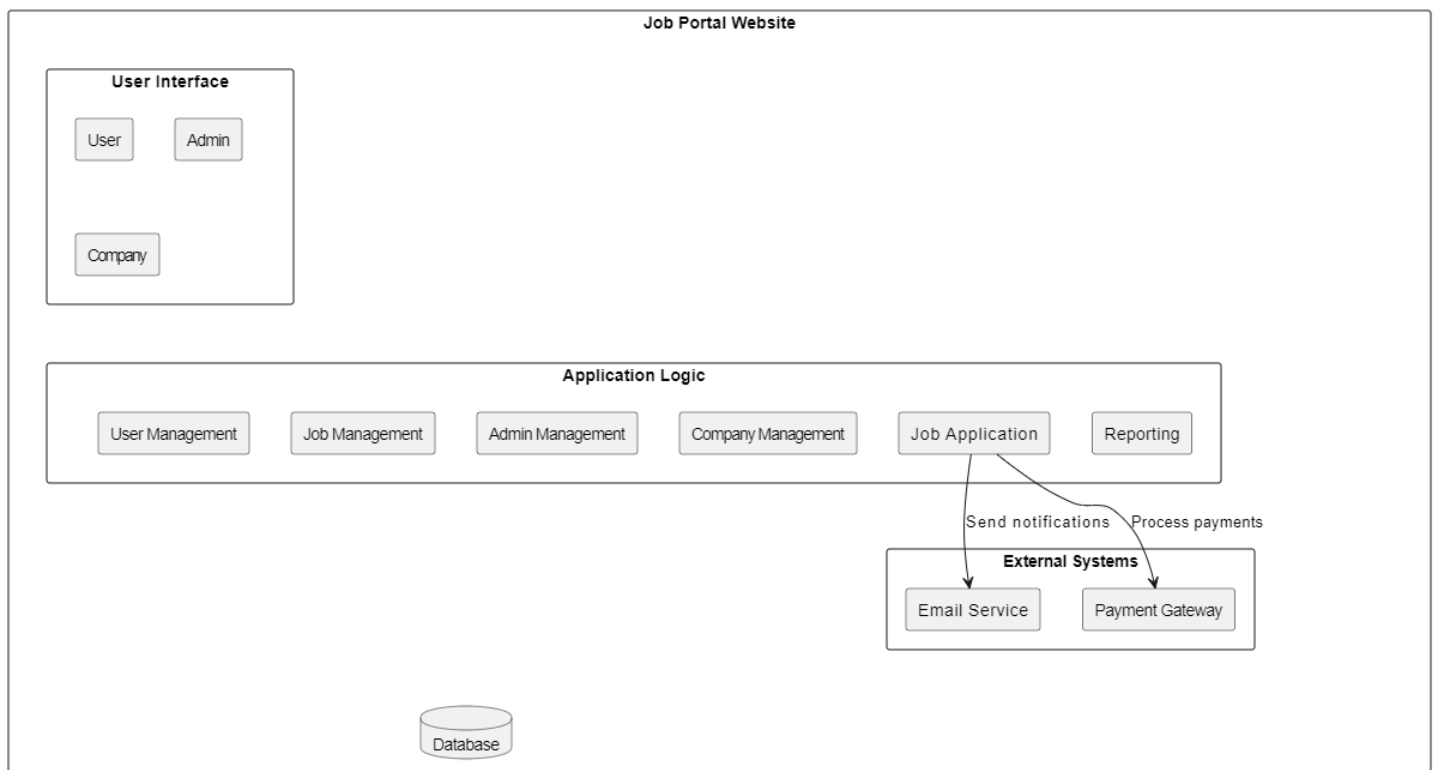
5.2.1 0 Level DFD

0 Level DFD - Job Portal Website



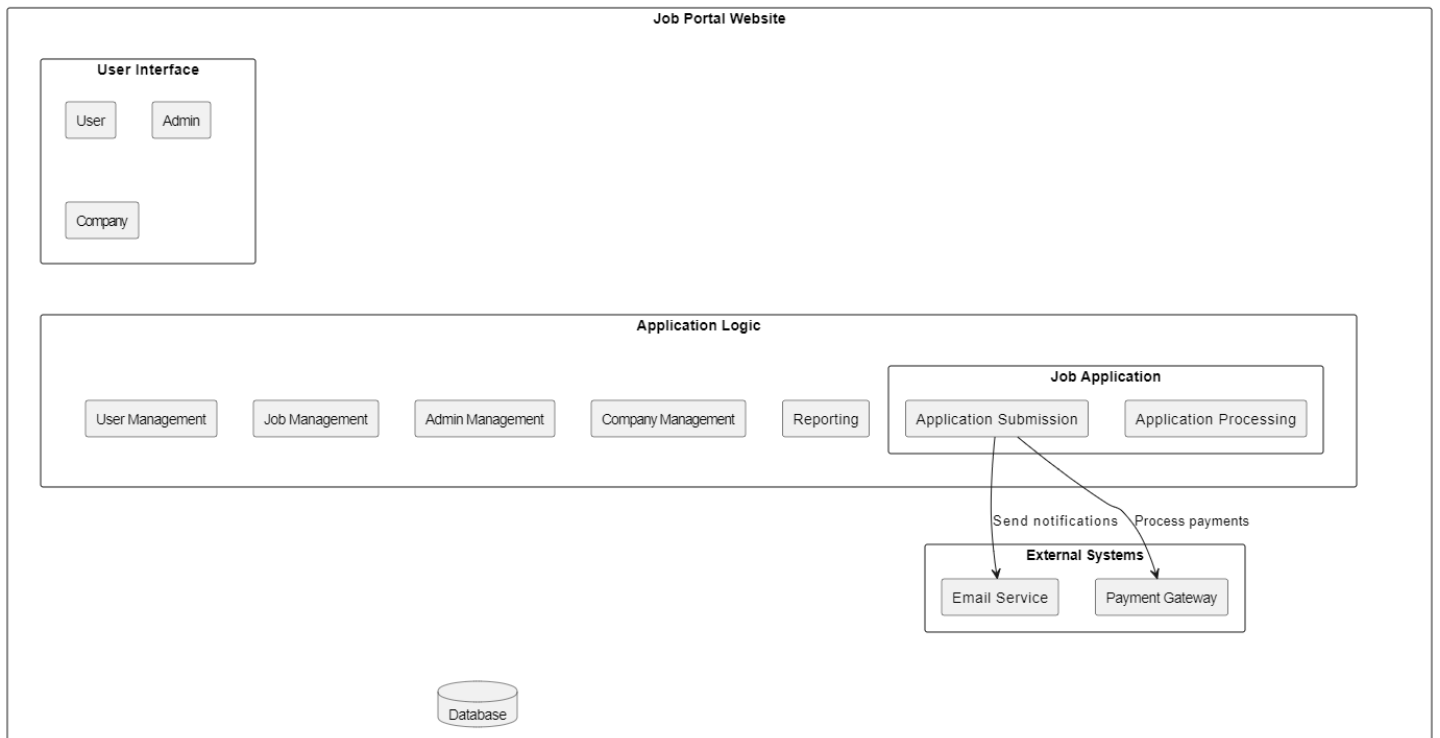
5.2.2 1 Level DFD

1 Level DFD - Job Portal Website



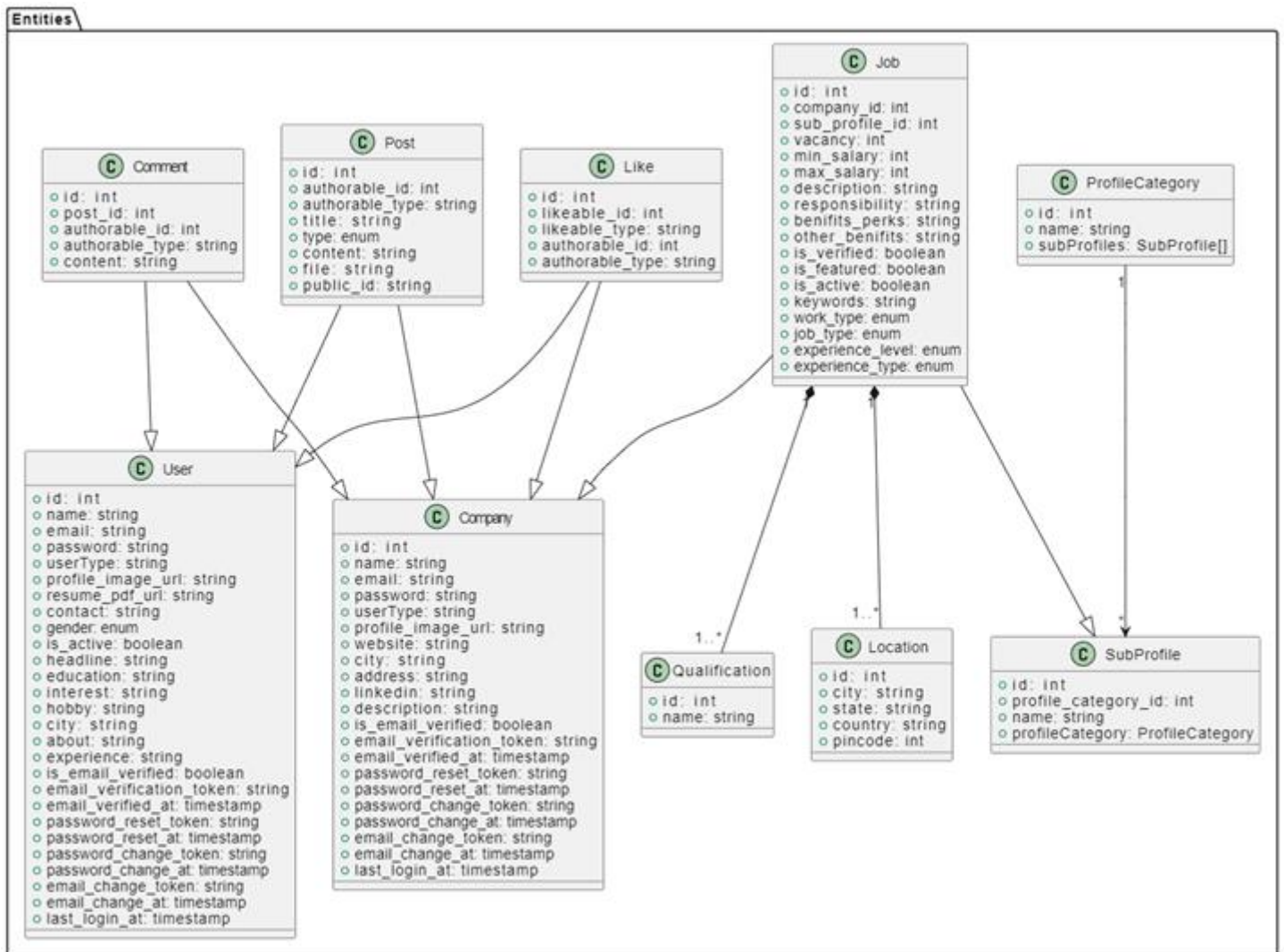
5.2.3 2 Level DFD

2 Level DFD - Job Portal Website



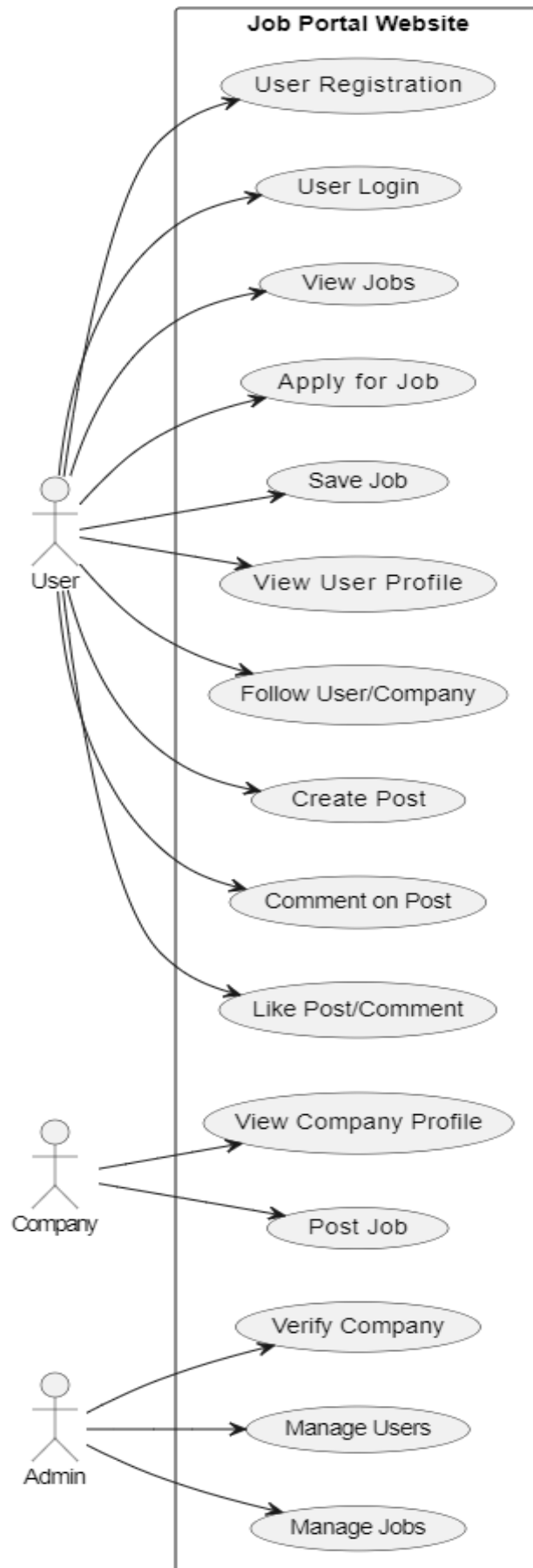
5.3 Structural Modeling

5.3.1 Class Diagram

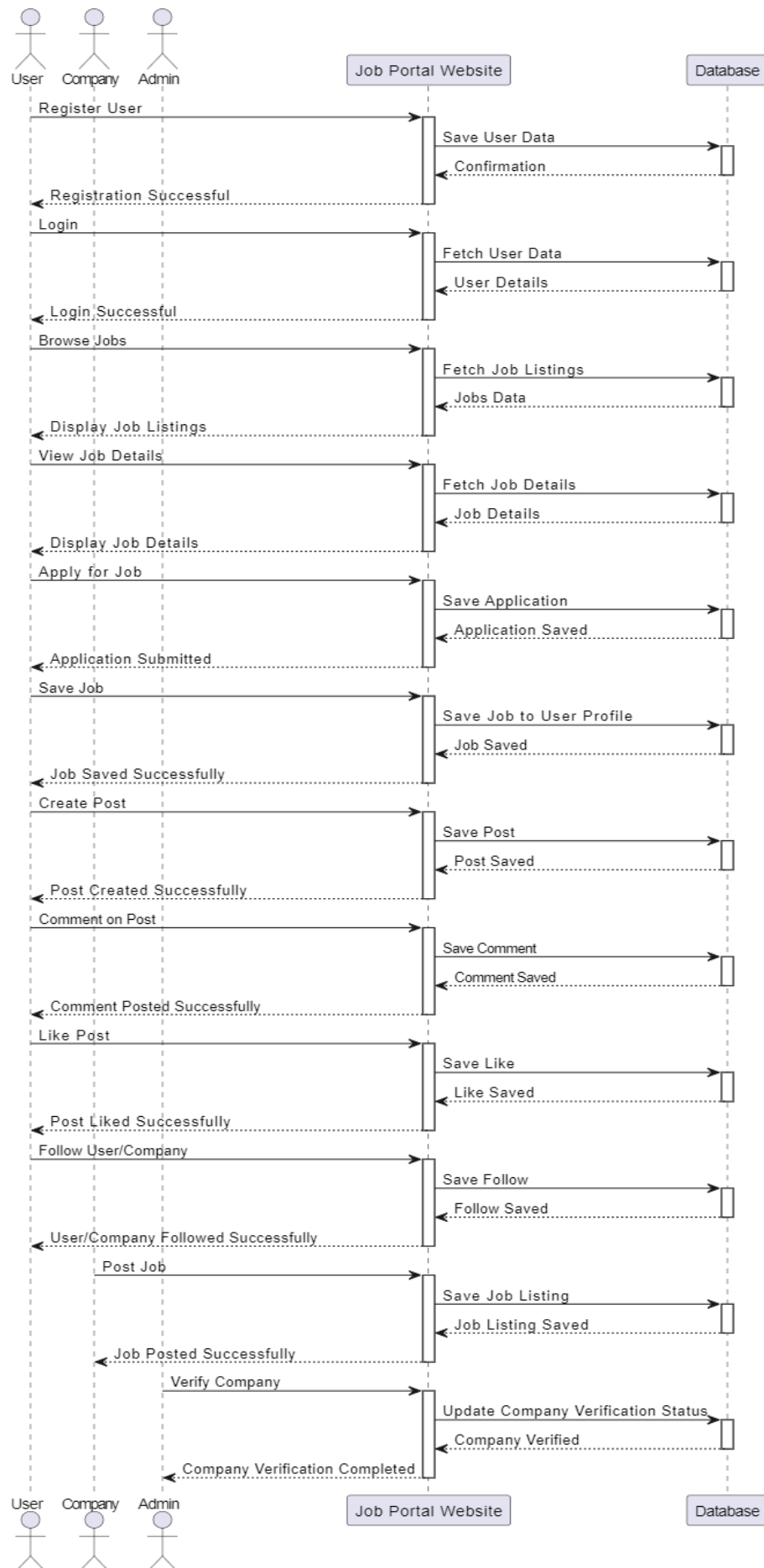


5.4 Behavioral Modeling

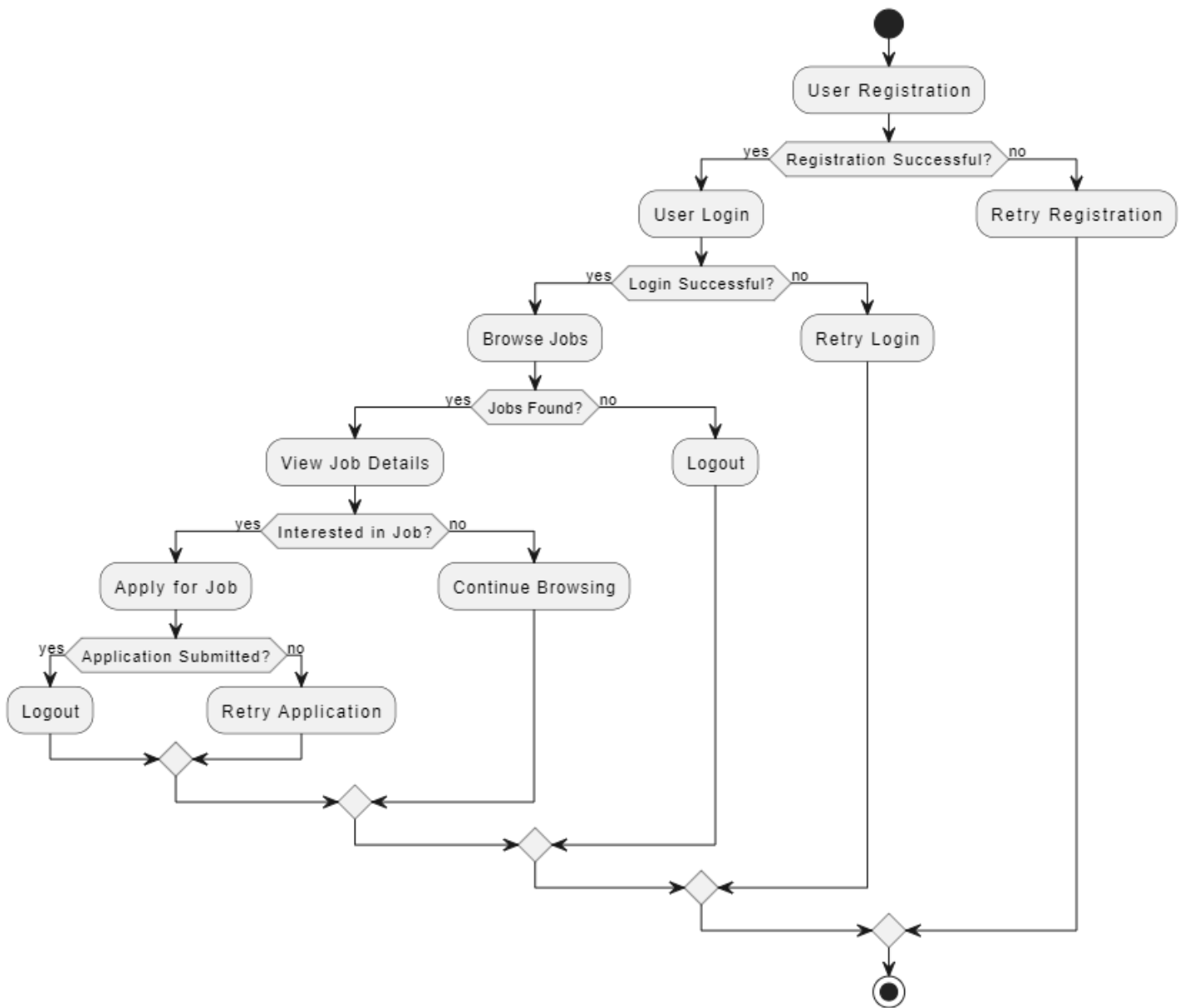
5.4.1 Use Case Diagram



5.4.2 Sequence Diagram



5.4.3 Activity Diagram



5.5 Database Design

5.5.1 Table Structure

E users
◦ id: bigint unsigned
◦ profile_image_url: text
...
● name: varchar(25)
● email: varchar(50)
● password: varchar(100)
● userType: varchar(10)

E companies
◦ id: bigint unsigned
◦ is_verified: tinyint
...
● userType: varchar(10)
● name: varchar(60)
● email: varchar(100)
● password: varchar(100)

E admins
◦ id: bigint unsigned
...
● userType: varchar(10)
● name: varchar(20)
● email: varchar(100)
● password: varchar(100)

E profile_categories
◦ id: bigint unsigned
...
● name: varchar(100)

E sub_profiles
◦ id: bigint unsigned
◦ profile_category_id: bigint unsigned
...
● name: varchar(100)

E qualifications
◦ id: bigint unsigned
...
● name: varchar(100)

E locations
◦ id: bigint unsigned
◦ pincode: int
...
● city: varchar(20)
● state: varchar(25)
● country: varchar(25)

E jobs
◦ id: bigint unsigned
◦ company_id: bigint unsigned
◦ sub_profile_id: bigint unsigned
◦ vacancy: tinyint
◦ min_salary: mediumint
◦ max_salary: mediumint
◦ description: text
...

E job_qualifications
◦ id: bigint unsigned
◦ jobs_id: bigint unsigned
◦ qualifications_id: bigint unsigned
...

E job_locations
◦ id: bigint unsigned
◦ jobs_id: bigint unsigned
◦ locations_id: bigint unsigned
...

E applied_jobs
◦ id: bigint unsigned
◦ user_id: bigint unsigned
◦ job_id: bigint unsigned
...

E saved_jobs
◦ id: bigint unsigned
◦ user_id: bigint unsigned
◦ job_id: bigint unsigned
...

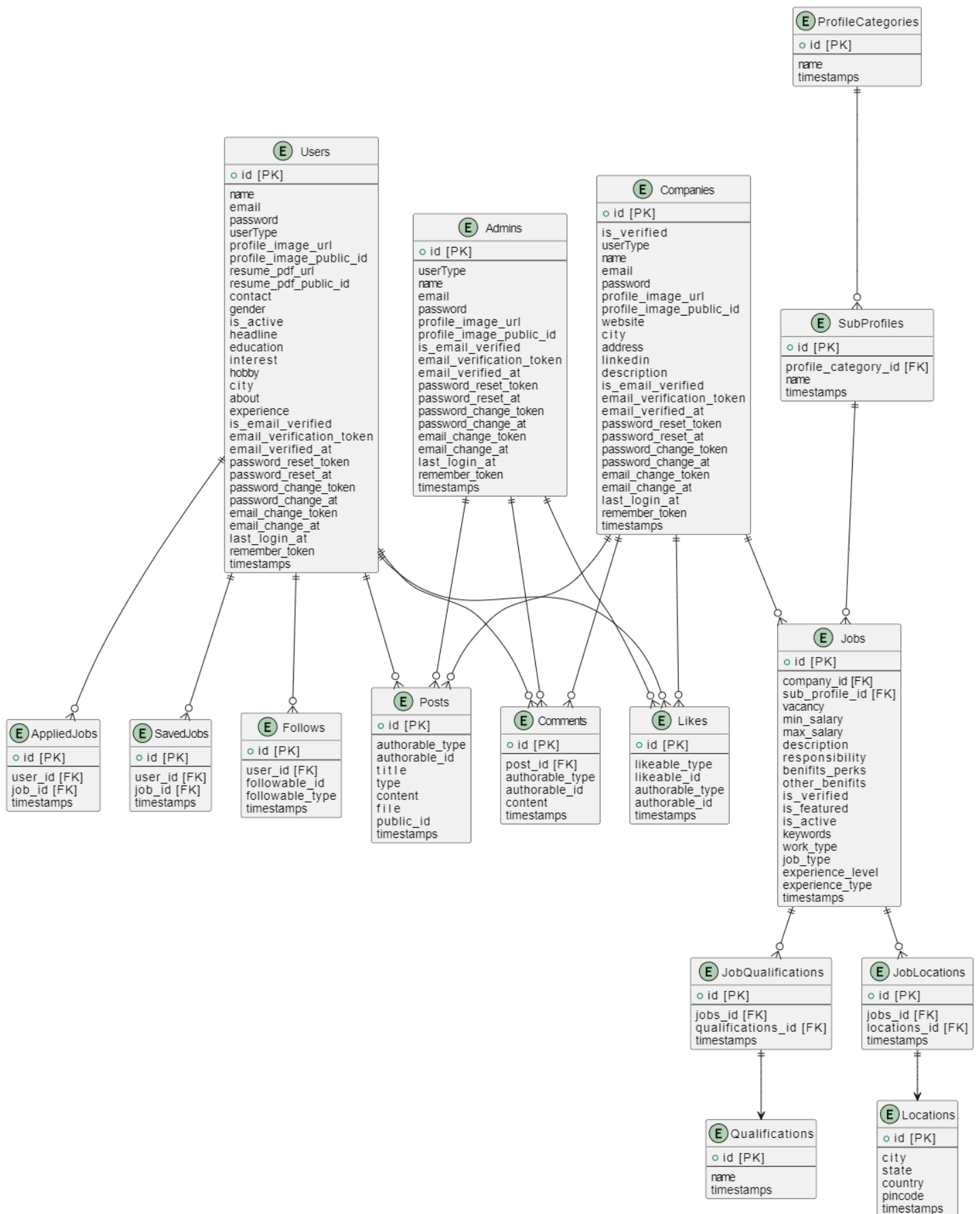
E follows
◦ id: bigint unsigned
◦ user_id: bigint unsigned
◦ followable_id: bigint unsigned
...
● followable_type: varchar(255)

E posts
◦ id: bigint unsigned
◦ authorable_id: bigint unsigned
...
● authorable_type: varchar(255)
● title: varchar(255)
● type: enum('TEXT', 'IMAGE', 'VIDEO')

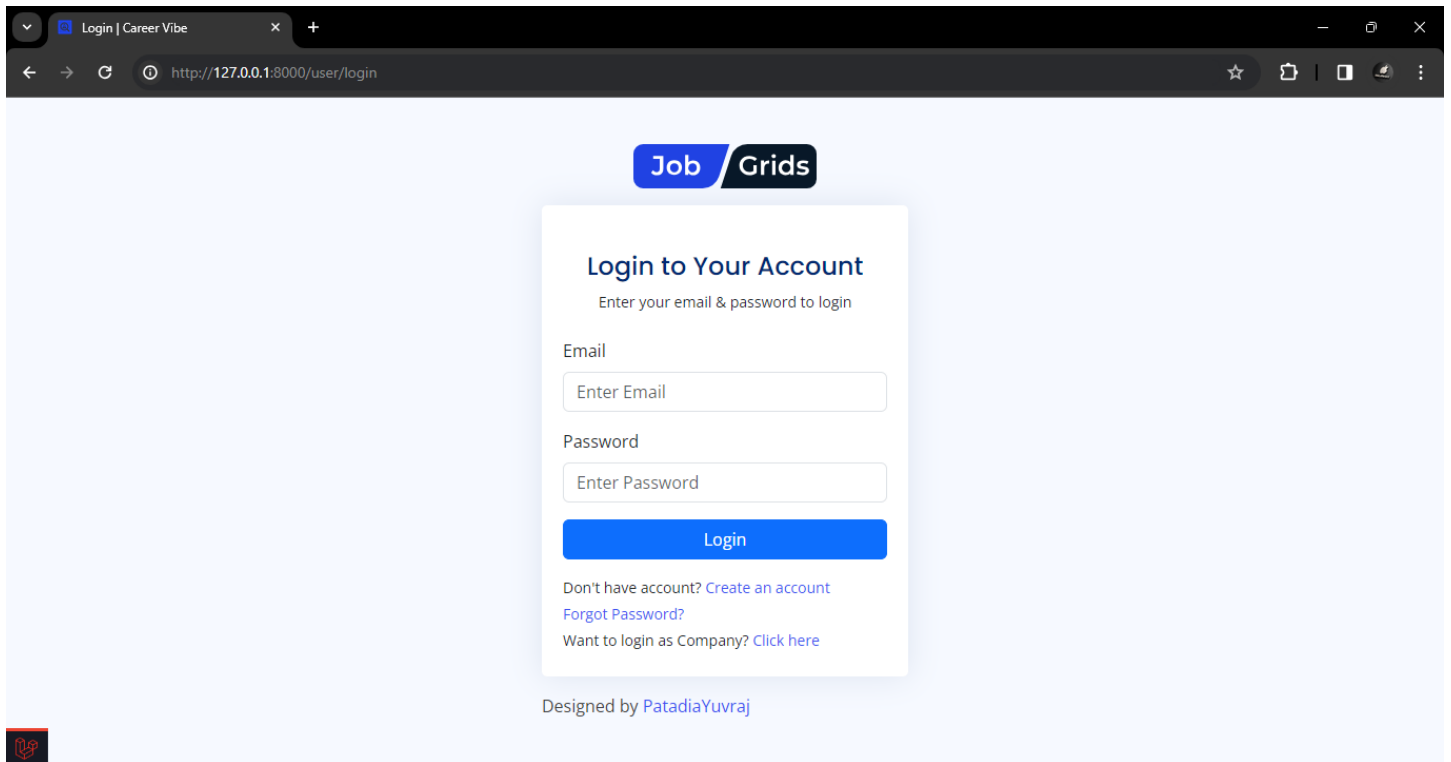
E comments
◦ id: bigint unsigned
◦ post_id: bigint unsigned
◦ authorable_id: bigint unsigned
◦ content: text
...
● authorable_type: varchar(255)

E likes
◦ id: bigint unsigned
◦ likeable_id: bigint unsigned
◦ authorable_id: bigint unsigned
...
● likeable_type: varchar(255)
● authorable_type: varchar(255)

5.5.2 Entity Relationship



5.6 Screen Design or Screen shots



A screenshot of a web browser showing the 'Login | Career Vibe' page. The browser's address bar displays 'http://127.0.0.1:8000/user/login'. The page features a light blue background with a 'Job Grids' logo at the top. A central white card contains the 'Login to Your Account' form. The form includes fields for 'Email' and 'Password', a blue 'Login' button, and links for 'Create an account', 'Forgot Password?', and 'Click here' to login as a company. A small red logo is visible in the bottom left corner.

Job Grids

Login to Your Account

Enter your email & password to login

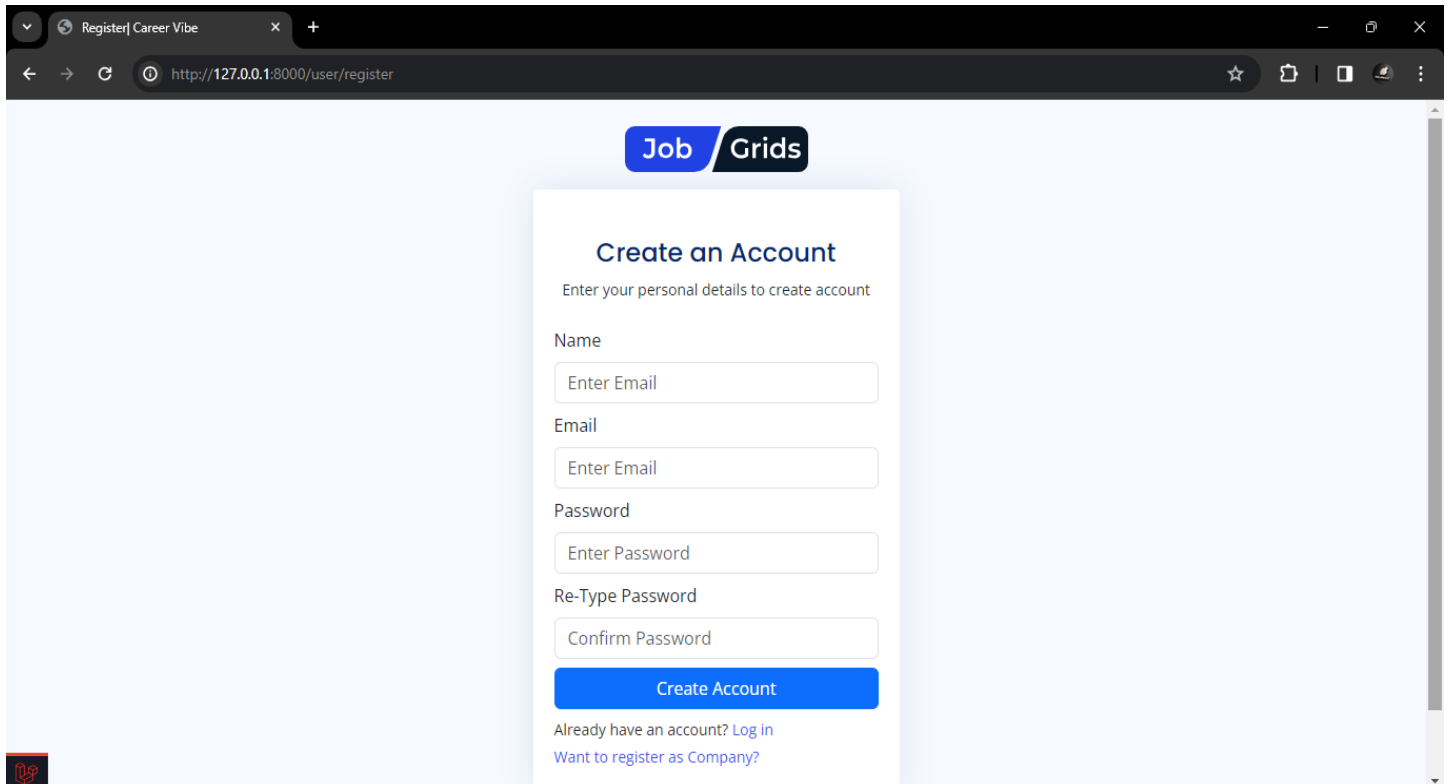
Email

Password

Login

Don't have account? [Create an account](#)
[Forgot Password?](#)
Want to login as Company? [Click here](#)

Designed by [PatadiaYuvraj](#)



A screenshot of a web browser showing the 'Register | Career Vibe' page. The browser's address bar displays 'http://127.0.0.1:8000/user/register'. The page features a light blue background with a 'Job Grids' logo at the top. A central white card contains the 'Create an Account' form. The form includes fields for 'Name', 'Email', 'Password', and 'Re-Type Password', a blue 'Create Account' button, and links for 'Log in' and 'Want to register as Company?'. A small red logo is visible in the bottom left corner.

Job Grids

Create an Account

Enter your personal details to create account

Name

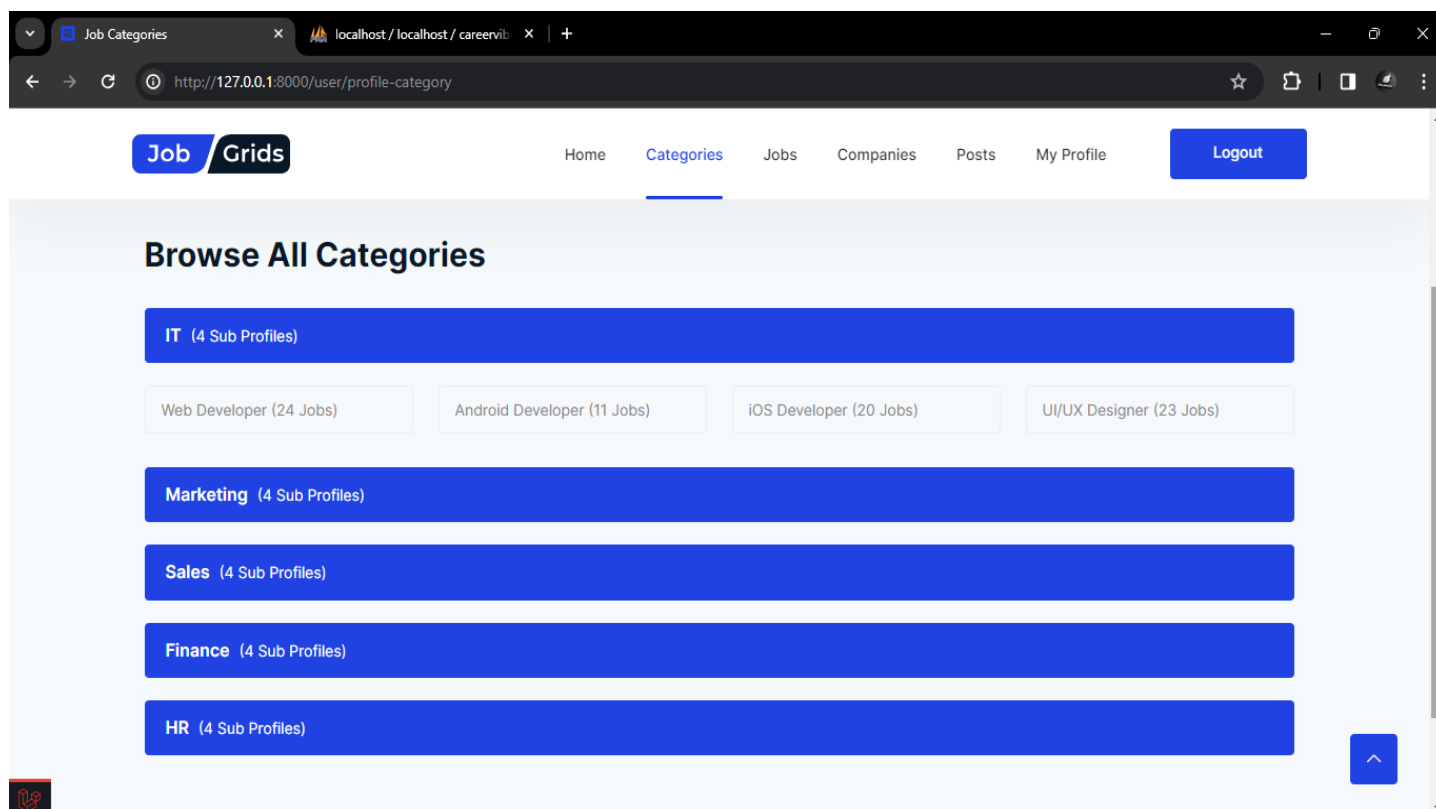
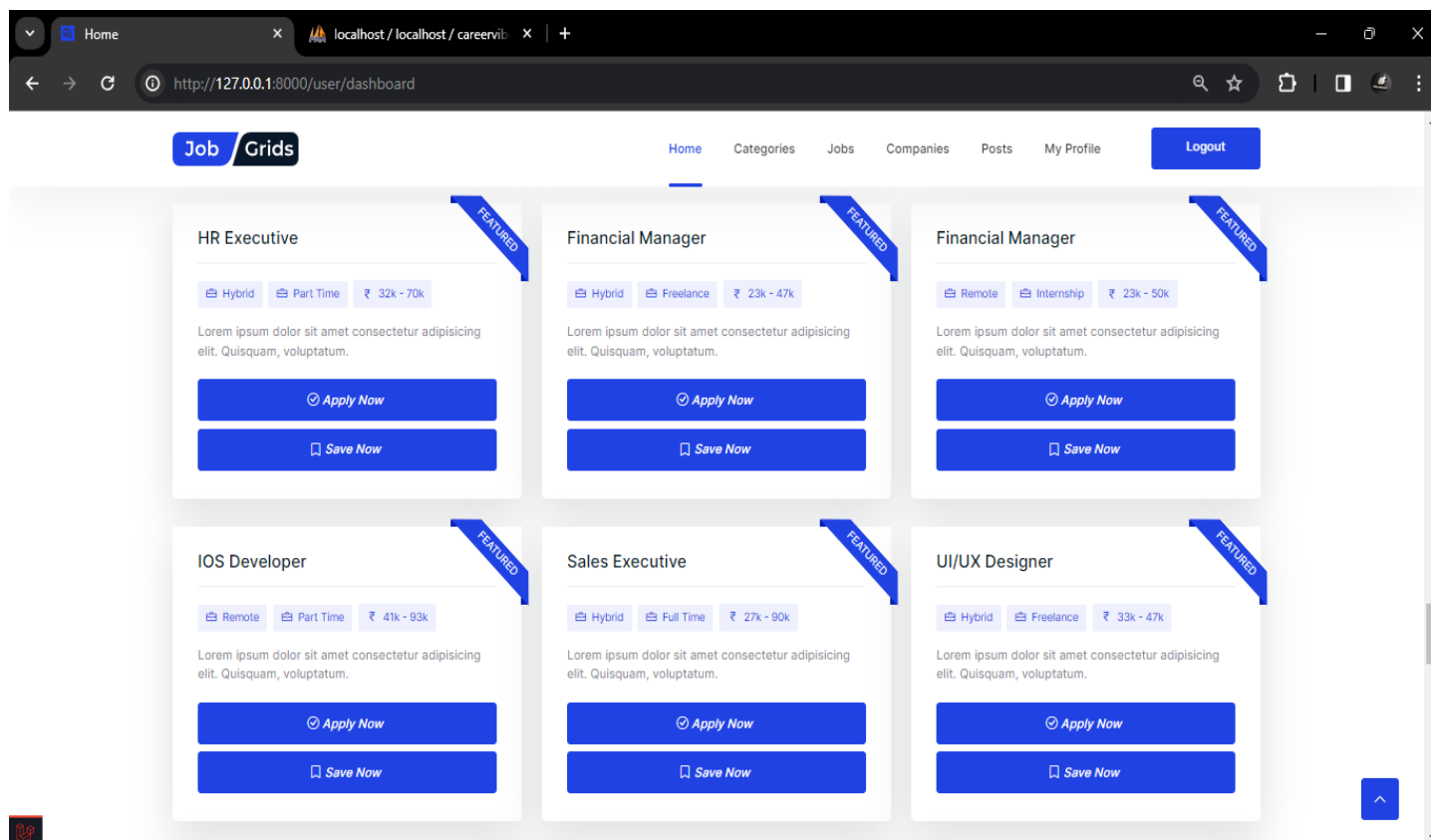
Email

Password

Re-Type Password

Create Account

Already have an account? [Log in](#)
[Want to register as Company?](#)



Job List

localhost / localhost / careervib

http://127.0.0.1:8000/user/job

Job Grids

[Home](#)
[Categories](#)
[Jobs](#)
[Companies](#)
[Posts](#)
[My Profile](#)

Logout

Sales Manager

Save Now

Apply Now

Offered By: Company 1 Sales

Lorem ipsum dolor sit amet consectetur adipisicing elit. Quisquam, voluptatum.

₹ 16K-83K

Full Time

Hybrid

Experienced

Bachelor of Business Administration

Bachelor of Engineering in Mechanical

Bachelor of Engineering in Electronics

Bachelor of Engineering in Mechatronics

Bachelor of Engineering in Production

Surat

Sales Head

Save Now

Apply Now

Offered By: Company 1 Sales

Lorem ipsum dolor sit amet consectetur adipisicing elit. Quisquam, voluptatum.

₹ 12K-95K

Freelance

Onsite

Experienced

Bachelor of Commerce

Master of Engineering

Master of Science

Diploma

Your Profile

localhost / localhost / careervib

http://127.0.0.1:8000/user/me

Job Grids

[Home](#)
[Categories](#)
[Jobs](#)
[Companies](#)
[Posts](#)
[My Profile](#)

Logout

Manage Account

My Profile

Applied Jobs

Saved Jobs

My Followers

My Following

Edit Profile

Add Profile Image

Add Resume

Personal Info

Name

User 1

Email Address

user1@user.com

Profile Image

Not Uploaded

Contact Info

Location

Profile Image

View Option ▾

Phone

Resume PDF

View Option ▾

18 | Page

Dashboard | Admin

http://127.0.0.1:8000/admin/job

Search

Admin 1

Job Grids

Dashboard

Company

User

User LiveWire

Job

Profile Category

Profile Category LiveWire

Job Profile

Job

Job Profile	Job Vacancy	Salary	Work Type	Verified	Featured	Active	Action
Sales Manager	12	16k - 83k	HYBRID	Verified	Featured	Active	Edit Delete
Sales Head	16	12k - 95k	ONSITE	Verified	Featured	Active	Edit Delete
Business Development Executive	22	27k - 46k	HYBRID	Verified	Featured	Active	Edit Delete
Financial Manager	9	28k - 76k	HYBRID	Verified	Featured	Active	Edit Delete
SEO	21	14k - 93k	REMOTE	Verified	Featured	Active	Edit Delete
Sales Executive	9	17k - 99k	REMOTE	Verified	Featured	Active	Edit Delete
Sales Head	17	42k - 56k	REMOTE	Verified	Featured	Active	Edit Delete
Sales Executive	25	27k - 64k	HYBRID	Verified	Featured	Active	Edit Delete
Financial Analyst	7	21k - 81k	ONSITE	Verified	Featured	Active	Edit Delete
Content Marketing	18	17k - 48k	REMOTE	Verified	Featured	Active	Edit Delete

6. SYSTEM TESTING

6.1 Testing and Implementation

System testing is a critical phase in the development lifecycle of the job portal project. It ensures that the entire system, including its components and functionalities, operates as intended before deployment to production. Testing involves identifying and fixing defects, validating system requirements, and ensuring user satisfaction.

6.2 Testing Methodology

The testing methodology employed for the job portal project includes:

- **Functional Testing:** Verifying the functionality of each feature, such as user registration, job posting, job search, etc., to ensure they meet the specified requirements.
- **Integration Testing:** Testing the integration of individual components to ensure they function together as a cohesive system.
- **Performance Testing:** Evaluating system performance under varying loads to ensure it can handle expected user traffic.
- **Security Testing:** Identifying and addressing security vulnerabilities to protect user data and system integrity.
- **Usability Testing:** Assessing the user interface and experience to ensure it is intuitive and user-friendly.
- **Regression Testing:** Re-testing previously validated features after changes to ensure new developments do not adversely affect existing functionality.

6.3 Test Case Design

Test cases are designed to cover various scenarios and validate system functionality. Some examples of test cases for the job portal project include:

- **User Registration:** Verify that users can register successfully with valid information.
- **Job Posting:** Test that companies can post jobs with accurate details and required information fields.
- **Job Search:** Ensure users can search for jobs using different criteria and obtain relevant results.
- **Application Submission:** Validate that users can apply for jobs and receive confirmation of successful submission.

- **Security Measures:** Check for vulnerabilities such as SQL injection, cross-site scripting (XSS), and ensure proper authentication and authorization mechanisms are in place.
- **Performance:** Test system response times under varying loads and ensure it meets defined performance criteria.

6.4 System Implementation Strategy

The implementation of system testing will follow these steps:

- **Test Environment Setup:** Establish a dedicated testing environment mirroring the production environment to conduct testing without affecting live operations.
- **Test Plan Creation:** Develop a comprehensive test plan outlining testing objectives, scope, resources, and timelines.
- **Test Case Development:** Design test cases covering functional, integration, performance, security, and usability aspects.
- **Test Execution:** Execute test cases systematically, documenting test results and identifying any defects or deviations from expected behavior.
- **Defect Resolution:** Address identified defects promptly, prioritizing critical issues and ensuring resolution before proceeding to deployment.
- **User Acceptance Testing (UAT):** Engage stakeholders to validate system functionality and ensure it meets their requirements and expectations.
- **Deployment:** Once testing and UAT are successful, proceed with deployment to the production environment following a predefined rollout plan.

7. CONCLUSION

This document has presented a comprehensive overview of the Job Portal Website developed using Laravel/PHP and a MySQL database. The website caters to three user types: Users (job seekers), Companies (employers), and Admins.

The functionalities have been designed to streamline the job search and recruitment process. Users can search for jobs based on their preferences, apply for jobs online, and manage their applications. Companies can post jobs, attract qualified candidates, and manage the application process. Admins have overall management privileges, ensuring smooth website operation.

The documentation covered the system architecture, user roles and permissions, database design, functionalities for each user type, testing strategies, deployment process, and potential future enhancements. Throughout the development process, valuable learning experiences were gained in areas like Laravel development, project management, and soft skills.

This Job Portal Website serves as a functional platform that can be further enhanced with additional features based on user feedback and future trends. By continuing development and staying updated with technological advancements, this website has the potential to become a valuable resource for both job seekers and employers in the ever-evolving job market.

8. LEARNING DURING PROJECT WORK

Developing this Job Portal Website has been a valuable learning experience. Here are some key takeaways:

1. Technical Skills:

- Gained proficiency in Laravel development framework, particularly in areas like routing, controllers, models, and database interactions using Eloquent.
- Improved understanding of object-oriented programming principles for code reusability and maintainability.
- Strengthened knowledge of working with a relational MySQL database for data storage and retrieval.

2. Project Management:

- Enhanced skills in project planning, breaking down the development process into manageable tasks.
- Learned the importance of code documentation for better understanding and future maintenance.
- Gained experience in testing different functionalities to ensure a smooth user experience.

3. Soft Skills:

- Developed problem-solving abilities by encountering and resolving technical challenges throughout the development process.
- Improved time management skills by effectively juggling development tasks and meeting deadlines.
- Strengthened independent learning skills by seeking solutions and referring to online resources effectively.

9. FUTURE ENHANCEMENTS

9.1 Advanced Job Matching Algorithm

Implement an advanced job matching algorithm leveraging machine learning techniques to enhance job recommendations for users. Incorporate user preferences, past interactions, and job characteristics to provide more personalized and relevant job suggestions.

9.2 Enhanced User Profiles

Enhance user profiles by adding additional fields such as skills, certifications, and preferences. Enable users to showcase their expertise and qualifications more comprehensively, facilitating better matches with job opportunities.

9.3 Social Media Integration

Integrate social media platforms to enable users to import and showcase their professional profiles from platforms such as LinkedIn. Allow users to share job postings, network with connections, and leverage social media for enhanced job discovery and engagement.

9.4 Mobile Application Development

Develop a dedicated mobile application for the job portal to provide users with seamless access to job listings, application submissions, and other features on their smartphones. Ensure cross-platform compatibility to cater to a wider audience.

9.5 Advanced Analytics and Reporting

Implement advanced analytics capabilities to track user engagement, job application trends, and other key metrics. Provide comprehensive reporting features for companies to analyze the effectiveness of their job postings and recruitment strategies.

9.6 Virtual Job Fairs

Organize virtual job fairs within the platform to facilitate direct interaction between job seekers and recruiters. Incorporate features such as live video interviews, virtual booths, and networking sessions to simulate the experience of physical job fairs.

9.7 Gamification Elements

Introduce gamification elements such as badges, achievements, and leaderboards to incentivize user engagement and participation. Reward users for completing profile sections, applying for jobs, and

engaging with the platform, fostering a sense of accomplishment and community.

9.8 Enhanced Candidate Screening

Implement advanced candidate screening tools to streamline the recruitment process for companies. Utilize AI-driven resume parsing and screening algorithms to identify top candidates based on predefined criteria, reducing manual effort and improving efficiency.

9.9 Multilingual Support

Introduce multilingual support to cater to a diverse user base across different regions and languages. Enable users to access the platform in their preferred language, enhancing accessibility and inclusivity.

9.10 Continuous Feedback Mechanism

Implement a continuous feedback mechanism to gather user feedback and suggestions for ongoing improvement. Encourage users to provide feedback on their experience with the platform, features they would like to see, and any issues encountered, facilitating iterative enhancements and refinement.

10. BIBLIOGRAPHY

10.1 Online References

1. Laravel Documentation. Available online at: <https://laravel.com/docs>
2. MySQL Documentation. Available online at: <https://dev.mysql.com/doc/>
3. W3Schools - Web Development Tutorials. Available online at: <https://www.w3schools.com/>
4. Stack Overflow - Community-based Q&A platform for programming. Available online at: <https://stackoverflow.com/>
5. GitHub - Version control platform. Available online at: <https://github.com/>

10.2 Offline References

1. Martin, Robert C. Clean Code: A Handbook of Agile Software Craftsmanship. Prentice Hall, 2008.
2. Gamma, Erich et al. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, 1994.
3. Fowler, Martin. Refactoring: Improving the Design of Existing Code. Addison-Wesley, 1999.
4. McConnell, Steve. Code Complete: A Practical Handbook of Software Construction. Microsoft Press, 2004.
5. Hunt, Andrew and Thomas, David. The Pragmatic Programmer: Your Journey to Mastery. Addison-Wesley, 1999.

These offline references provide foundational knowledge and best practices in software engineering and web development. They cover topics such as clean code, design patterns, refactoring, software construction, and programming mastery, offering valuable insights for the development of the Job Portal Website.