< به نام آزادی>





*Software Requirements Specification (SRS) for "Karisho" Job Shift Finding Application*

*Table of Contents* 1.Introduction.................................................................................................................................3

2.Background..................................................................................................................................3

3.Scope...........................................................................................................................................4

3.1.Insurance…………………………………………………………………………………………………………………………….4

3.2.Data Management……………………………………………………………………………………………………………….4

3.3.Regulatory Compliance………………………………………………………………………………………………….……5

3.4.Unique Feature……………………………………………………………………….……………………………………………5

4.User Requirements…………………………………………………………………………………….…………………………..5

4.1.User Types……………………………………………………………………………………………………………….………….5

4.1.1.Job Seekers (Employees)………………………………………………………………………………………………..…5

4.1.2.Companies (Employers)…………………………………………………………………………………………………….6

4.2.User Needs ………………………………………………………………………………………………………………………….6

4.3.Company Needs…………………………………………………………………………………………………………………..7

5.Functional Requirements………………………………………………………………………………………………………..7

5.1.User Registration & Profiles…………………………………………………………………………………………………7

5.1.1.Registration process for jobseekers ………………………………………………………………………………….7

5.1.2.Registration process for companies…………………………………………………………………………………..8

5.2.Job Search & Filters………………………………………………………………………………………………………………9

5.2.1.Location Filter……………………………………………………………………………………………………………………9

5.2.2.Minimum hourly rate filter………………………………………………………………………………………………..9

5.2.3.Industry Filters………………………………………………………………………………………………………………….9

5.3.Job posting & management……………………………………………………………………………………………….10

5.3.1.Create a shift…………………………………………………………………………………………………………………..10

5.3.2.Accept the best candidate……………………………………………………………………………………………….10

5.3.3.Rate the professional………………………………………………………………………………………………………10

5.3.4.Flexpool management…………………………………………………………………………………………………….11

5.4.Application and selection process………………………………………………………………………………………11

5.4.1.Job seeker’s job feed……………………………………………………………………………………………………….11

5.4.2.Job application………………………………………………………………………………………………………………..11

5.4.3.Candidate selection…………………………………………………………………………………………………………12

5.4.4.Digital agreement……………………………………………………………………………………………………………12

5.5.Shift confirmation and reminders………………………………………………………………………………………12

5.5.1.candidate notification……………………………………………………………………………………………………..12

5.5.2.Email notification…………………………………………………………………………………………………………….12

5.5.3.Reminder notification……………………………………………………………………………………………………..13

5.6.Payment & Feedback………………………………………………………………………………..……………………….13

5.6.1.Payment options……………………………………………………………………………………………………………..13

5.6.2.Rating system………………………………………………………………………………………………………………….14

6.usecase………………………………………………………………………………………………………………………………...14

7.nonfunctional requirements…………………………………………………………………………………………………16

8.System architecture………………………………………………………………………………………………………………17

9.Assumptions and dependencies……………………………………………………………………………………………18

9.1.Assumptions………………………………………………………………………………………………………………………18

9.2.Dependencies…………………………………………………………………………………………………………………….19

10.Contraints…………………………………………………………………………………………………………………………..19

10.1.Time constraints……………………………………………………………………………………………………………...19

10.2.Budget constraint…………………………………………………………………………………………………………….20

10.3.Technology constraints………………………………………………………………………………………………….…20

10.4.Regulatory and compliance constraints…………………………………………………………………………...20

10.5.Scalibility constraints……………………………………………………………………………………………………….20

10.6.Security constraints…………………………………………………………………………………………………………20

10.7.Usability constraints………………………………………………………………………………………………………..21

10.8.Integration constraints…………………………………………………………………………………………………….21

10.9.Resource constraints………………………………………………………………………………………………………..21

11.Testing Requirements ………………………………………………………………………………………………………..21

11.1.Functional testing…………………………………………………………………………………………………………….21

11.2.Performance testing………………………………………………………………………………………………………..22

11.3.Security testing………………………………………………………………………………………………………………..22

11.4.usability testing………………………………………………………………………………………………………………..22

11.5.compability testing…………………………………………………………………………………………………………..22

11.6.Integration testing……………………………………………………………………………………………………………23

11.7.Regression testing…………………………………………………………………………………………………………...23

11.8.User Acceptance Testing(UAT)…… …………………………………………………………………………………..23

11.9.Localization and internationalization testing…………………………………………………………………...23

12.Deployment and installation……………………………………………………………………………………………….24

12.1.Deployment…………………………………………………………………………………………………………………..…24

12.2.Installation………………………………………………………………………………………………………………….……25

13.Maintenance & Support……………………………………………………………………………………………..……...26

13.1Maintenance………………………………………………………………………….………………………………………...26

13.2.Support………………………………………………………………………………………………………………………….…27

14.UML Diagrams…………………………………………………………………………………………………………………….29

**1. Introduction**

"Karisho" is a revolutionary job finding application that sets itself apart from other job apps by focusing exclusively on shift-based work opportunities. As an employee, you have the freedom to choose where and when you want to work, earning money for individual shifts. This unique approach provides a remarkable opportunity for those in need of additional income, individuals seeking a flexible work arrangement, and anyone looking for a side job.

Karisho not only benefits job seekers but also streamlines the process for employers to find suitable staff.

By offering a vast array of shifts in various industries, we make it easier than ever for companies to fill labor gaps and alleviate concerns about workforce shortages.

**2. Background**

At Karisho, we believe in creating a life that you truly enjoy. Our platform is built on the principles of flexibility, security, and growth opportunities. We empower you to take control of your work schedule, enabling you to decide when and where you want to work. With thousands of available shifts across diverse industries, you can plan your work around your life commitments.

Maximizing your earnings while minimizing uncertainty is a core focus of Karisho. On average, job seekers can earn £12-17 per hour, with the option to negotiate rates. What's more, we ensure prompt payment, with funds transferred to your account within three days. By building an attractive profile, you can become a client magnet, leveraging our platform to develop your personal and professional skills, try different job types, and negotiate higher rates.

Our mission is to empower individuals to gain confidence in their abilities and seize opportunities in life. We make finding work as easy as a tap of a button, equipping you with skills that enable you to negotiate better rates and ensuring that work fits into your life, not the other way around. At Karisho, we are genuinely dedicated to improving the lives of our users. Our platform empowers both individuals and companies, making them more resilient and energized in an ever-changing job market.

We believe in providing everyone with a fair chance, based on their objectively considered qualities rather than superficial factors like appearance or birthplace. Our commitment extends to ensuring full access to the labor market, eliminating hidden shifts, unnecessary hurdles, and burdensome processes. Leveraging the power of technology, Karisho fosters efficiency by connecting companies and workers seamlessly, resulting in better earnings for workers and creating abundant opportunities for all.

**3. Scope**

The primary goal of the "Karisho" application is to create a platform that facilitates easier and faster connections between companies and job seekers, as mentioned in the background of the app. By providing a user-friendly interface and robust features, Karisho aims to streamline the process of finding shift-based employment opportunities.

While striving to achieve this goal, several challenges and considerations need to be addressed:

**3.1 Insurance**

One of the challenges involves providing insurance coverage for both employers and employees. As a job finding application, Karisho may need to navigate legal and regulatory requirements to ensure adequate insurance coverage is in place, protecting the interests of all parties involved. The specific insurance policies and coverage may vary depending on the country or region in which the app operates.

**3.2 Data Management**

To support the functionality of the application, a robust and secure database and server infrastructure must be established. Data management is critical to ensure the smooth operation of the app, including storing user profiles, shift information, and transactional data. The database and server infrastructure should be scalable, reliable, and capable of handling a potentially large volume of users and data.

**3.3 Regulatory Compliance**

Compliance with local regulations is vital for the operation of the app. Different countries may have specific labor laws, employment regulations, and data protection requirements that must be adhered to.

Karisho should ensure that the app and its processes align with the applicable legal frameworks in each target market.

**3.4 Unique Feature:** Flexpool

One of the unique features of Karisho is the flexpool functionality. Employers have the ability to create and manage flexpools, which allow them to assemble a group of preferred employees. By adding favorite employees to the flexpool, employers can automatically accept their shift applications, reducing the time and effort required for manual approval. This feature enhances the efficiency of the shift assignment process and strengthens the relationship between employers and their preferred workforce.

By addressing these challenges and implementing unique features like the flexpool, Karisho aims to provide a comprehensive and efficient platform that benefits both job seekers and companies, enabling easier and faster connections in the shift-based job market.

**4. User Requirements**

**4.1 User Types**

The "Karisho" application caters to two primary user types:

**4.1.1 Job Seekers (Employees)**

Job seekers are individuals looking for shift-based employment opportunities. They have specific needs and requirements when using the app. The key features and functionalities for job seekers include:

* + - *Dashboard*: Job seekers should have access to a personalized dashboard where they can view their previous and upcoming shifts. This feature provides them with an overview of their work schedule and helps them manage their commitments effectively.
    - *Find Job Section*: Job seekers should be able to browse available job shifts posted by companies. This section allows them to search for shifts based on their preferences, such as location, date, industry, or job type. The shifts should be presented in a calendar format, providing a convenient overview of available opportunities.

**4.1.2 Companies (Employers)**

Companies are entities looking to hire workers for shift-based positions. They have specific needs and requirements when using the app. The key features and functionalities for companies include:

* + - *Job Posting*: Companies should be able to create and post shifts on the platform. They should provide necessary details such as job requirements, rates, and any specific instructions.
    - *Shift Management*: Companies should have the ability to manage their posted shifts, including reviewing applications, selecting suitable candidates, and assigning workers to shifts.
    - *Flexpool Management*: Companies should be able to create and manage flexpools, where they can add their preferred employees. This feature allows companies to streamline the shift allocation process by automatically accepting shift applications from the flexpool members.

**4.2 User Needs**

In order to effectively use the "Karisho" application, both job seekers and companies need to fulfill certain requirements:

* + - *User Registration*: Both job seekers and companies must sign up for an account to utilize the app's functionalities practically. During the registration process, users will be required to provide necessary information as specified by the app. This may include personal details, contact information, work experience, and any relevant documents (such as resumes or certifications).
* Please note that the specific information and requirements during the registration process will be further detailed in the app's user interface and registration forms.

By catering to the needs of job seekers and companies, "Karisho" aims to provide a user-friendly platform that facilitates the connection between individuals seeking shift-based employment and companies offering such opportunities.

**5. Functional Requirements**

**5.1 User Registration and Profiles**

The user registration process is a crucial step in accessing the features and functionalities of the "Karisho" application. There are separate registration processes for job seekers (employees) and companies (employers), each with specific information requirements. The functional requirements for user registration and profiles area as follows:

**5.1.1 Registration Process for Job Seekers**

During the registration process, job seekers should provide the following information:

* + - *Name and Family Name:* Job seekers are required to enter their first name and last name.
    - *Social Number:* Job seekers should provide their social number or any unique identifier.
    - *Address and Postal Code:* Job seekers need to enter their residential address and postal code.
    - *Card Information:* Job seekers must provide the necessary card information for payment purposes. This may include credit card details or other payment methods supported by the app.
    - *Profile Picture:* Job seekers should upload a profile picture to personalize their account.
    - *Previous Work Experience:* Job seekers need to provide information about their previous work experience, including job titles, companies, and durations.
    - *Resume File*: Job seekers should have the option to upload their resume file, allowing employers to review their qualifications and skills.

**5.1.2 Registration Process for Companies**

During the registration process, companies should provide the following information:

* + - *Company Name:* Companies are required to enter their official company name.
    - *Company Location:* Companies need to provide their physical location or address.
    - *Company Number*: Companies should provide their company number or any unique identifier.
    - *Company Regulations*: Companies must provide information about their specific regulations or policies related to employment and labor.
    - *Owner Information*: Companies need to provide details about the company owner, including name and contact information.
* Note: The specific user interface design and validation rules for each input field should be implemented to ensure accurate and secure data collection during the registration process.
* By capturing the necessary information during user registration, "Karisho" can create personalized profiles for job seekers and companies, enabling smooth user interactions and facilitating effective job matching.

**5.2 Job Search and Filters**

The "Karisho" application provides job seekers with the ability to search for relevant job shifts based on their preferences. The following filters are available for job search:

**5.2.1 Location Filter**

Job seekers can filter job shifts based on the desired location. They can enter a specific location or use the app's location detection feature to find shifts available in their vicinity.

**5.2.2 Minimum Hourly Rate Filter**

Job seekers have the option to set a minimum hourly rate to filter the job shifts. They can specify the minimum rate they are willing to accept for a shift, allowing them to focus on shifts that meet their desired compensation.

**5.2.3 Industry Filters**

Job seekers can choose from various industry filters to narrow down their job search. The following industry filters are available:

* *Hospitality:* Job shifts related to roles such as Assistant Chef, Barista, Bartending, Catering, Cleaning, Cloakroom Assistant, Hosting, Housekeeping, Independent Chef, Kitchen Porter, Room Service, Site Crew - Hospitality, User Research, Waiting.
* *Logistics:* Job shifts related to roles such as Driver, Order Picker, Rider.
* Retail: Job shifts related to roles such as Sales Associate, Stock Associate.

By utilizing these search filters, job seekers can efficiently find job shifts that align with their location, desired hourly rate, and industry preferences.

**5.3 Job Posting and Management**

The "Karisho" application allows companies (employers) to post job shifts and efficiently manage the hiring process. The following functionalities are provided for job posting and management:

**5.3.1 Create a Shift**

Employers can create job shifts by providing detailed requirements and rates for the shift. They can specify the job title, job description, required skills, preferred qualifications, shift duration, and any other relevant details. Additionally, employers can set the hourly rate for the shift, indicating the compensation offered to the job seeker.

**5.3.2 Accept the Best Candidates**

Upon receiving applications from job seekers, employers have the ability to review and evaluate the applicants based on their qualifications, skills, and experience. Employers can compare the applicants and choose the best fit for the job shift. They can consider factors such as availability, suitability, and previous ratings if available.

**5.3.3 Rate the Professional**

After the completion of a job shift, employers can rate the performance of the job seeker. They can provide a rating or feedback based on factors such as punctuality, quality of work, professionalism, and other relevantattributes. This rating system helps to maintain transparency and accountability in the platform.

**5.3.4 Flexpool Management**

Employers have the option to create a flexpool, which is a group of their preferred employees. They can add their favorite employees to the flexpool, allowing for automatic acceptance of their applications for job shifts. This feature helps employers maintain a pool of reliable and trusted professionals for their staffing needs.

By providing these jobs posting and management functionalities, "Karisho" streamlines the process for employers to create job shifts, select the best candidates, and rate the professionals' performance. The flexpool feature further enhances the efficiency of managing preferred employees.

**5.4 Application and Selection Process**

The "Karisho" application provides a streamlined process for job seekers to browse and apply for job ads, and for companies to select candidates. The following functionalities are included in the application and selection process:

**5.4.1 Job Seeker's Job Feed**

Job seekers have access to a personalized job feed within the application. The feed displays available job ads based on the job seeker's location, preferences, and skills. Job seekers can browse through the job feed to find relevant job shifts.

**5.4.2 Job Application**

Job seekers can apply for job shifts directly from the job ad. They can submit their application by providing necessary details, such as their availability, a brief cover letter if required, and any additional information requested by the employer. The application is sent to the respective company for review.

**5.4.3 Candidate Selection**

Companies have the authority to review the applications received for their job ads. They can evaluate the candidates based on their qualifications, experience, and any other specified criteria. The company can select the most suitable candidate from the pool of applicants for the job shift.

**5.4.4 Digital Agreement**

Once the company has selected a candidate, both parties (company and job seeker) will enter into a digital agreement. The agreement will outline the terms and conditions of the job shift, including the shift details, compensation, working hours, and any other relevant contractual terms. Both the company and the job seeker will digitally sign the agreement to confirm their commitment.

* By incorporating these functionalities, "Karisho" simplifies the application and selection process for job seekers and companies, allowing for efficient job ad browsing, seamless application submission, candidate evaluation, and the establishment of digital agreements.

**5.5 Shift Confirmation and Reminders**

To ensure smooth coordination between the selected candidate and the company, the "Karisho" application provides shift confirmation and reminder functionalities. The following features are included:

**5.5.1 Candidate Notification**

Once a candidate is selected for a job shift, they will receive a notification within the application. The notification will inform them about their acceptance for the shift and provide essential details, such as the date, time, location, and any specific instructions related to the job.

**5.5.2 Email Notification**

In addition to the in-app notification, the selected candidate will also receive an email confirmation for the job shift. The email will serve as a reminder and include the same details mentioned in the candidate notification, ensuring that the candidate has multiple channels of communication to stay updated.

**5.5.3 Reminder Notification**

To ensure punctuality, candidates will receive a reminder notification prior to the scheduled shift time. The reminder will be sent within a reasonable timeframe, such as half an hour before the shift, reminding the candidate to arrive at the designated location on time.

* These shift confirmation and reminder functionalities aim to improve communication and minimize the chances of any miscommunication or missed shifts. By providing timely notifications and reminders, "Karisho" ensures that both the candidate and the company are well-informed and prepared for the job shift.

**5.6 Payment and Feedback**

The "Karisho" application offers a convenient payment system and a two-way rating system for both job seekers and companies. The following functionalities are included:

**5.6.1 Payment Options**

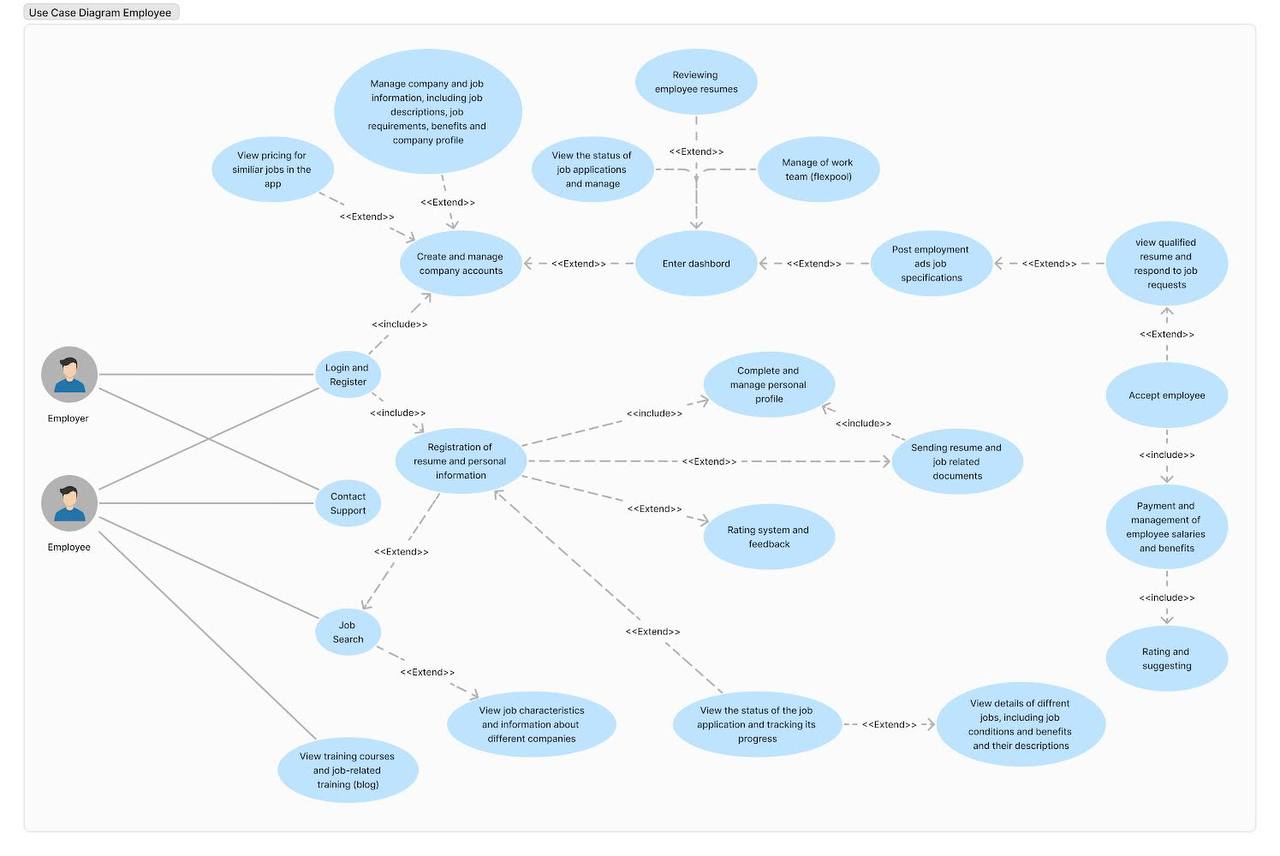
The application provides two payment options for job seekers:

* *Immediate Payment*: Job seekers have the option to receive immediate payment for their completed shifts. In this option, the job seeker will be paid the promised sum minus a percentage deducted by a third-party investment company. This enables the employing company to fulfill their financial obligations at the end of the month.
* *Monthly Payment*: Job seekers can choose to receive their full promised sum at the end of the month directly from the employing company through the "Karisho" app.
* The availability of these payment options provides flexibility to job seekers based on their financial needs and preferences.

**5.6.2 Rating System**

The application incorporates a two-way rating system to gather feedback from both job seekers and companies. This rating system allows the parties to evaluate each other's performance and maintain a reliable and trustworthy platform. The rating system includes the following attributes for job seekers:

* + - *Time of Attendance*: Companies can rate the job seeker's punctuality and adherence to the scheduled shifts.
    - *Quality of Work*: Companies can rate the job seeker's performance and the quality of their work.
    - *Absences*: Companies can provide feedback on the job seeker's attendance record and any instances of absence.
* By implementing this rating system, "Karisho" encourages transparency and accountability among job seekers and companies, ensuring a fair and productive working environment.

**6. Use Cases**

**6.1 Use Case 1:** Job Seeker Registration

Description: This use case describes the process of job seekers registering on the Karisho application.

Actors: Job Seeker

Preconditions: The job seeker has downloaded the Karisho app and has access to a compatible device.

Flow:

1. The job seeker opens the Karisho app.

2. The app displays the registration screen.

3. The job seeker enters their personal information (name, address, etc.) and provides the required documents.

4. The app validates the information and documents.

5. Upon successful validation, the job seeker's account is created.

Postconditions: The job seeker is registered on the Karisho app and can now access the available job listings.

**6.2 Use Case 2:** Job Search

Description: This use case describes the process of job seekers searching for available job shifts.

Actors: Job Seeker

Preconditions: The job seeker is registered on the Karisho app and logged in.

Flow:

1. The job seeker opens the Karisho app.

2. The app displays the job search interface with various search filters.

3. The job seeker applies the desired filters (location, minimum hourly rate, etc.).

4. The app retrieves and displays the relevant job shifts based on the applied filters.

5. The job seeker can view the job details and choose to apply for a specific shift.

Postconditions: The job seeker has applied for a job shift and awaits confirmation from the employer.

**7. Non-functional Requirements**

* *Performance*: The application should have the capability to handle a minimum of 1000 concurrent users without any noticeable degradation in response time. Job listings should load within 2 seconds on average to ensure a seamless user experience. The system should be able to handle a large number of companies, professionals, and ads, supporting scalability to accommodate future growth and workload demands.
* *Security*: High security measures must be implemented to ensure the privacy and confidentiality of user data. Robust encryption protocols should be employed to secure money transfers and financial transactions within the app.
* *Usability*: The app should prioritize usability, providing a user-friendly interface and intuitive navigation to enhance the user experience. Fast and efficient workflows should be designed to enable users to easily search for jobs, apply, and manage their shifts. The application should be compatible with mobile devices, allowing users to access and use the app seamlessly on both web and mobile platforms.
* Please note that these are general statements based on the information provided. For a comprehensive and detailed performance, security, and usability analysis, it is recommended to conduct a thorough evaluation and analysis specific to the "Karisho" application.

**8. System Architecture**

* *Client-Side*:
  + - * *Web Application*: Develop a web-based user interface for job seekers and companies to access the application using modern web technologies such as HTML, CSS, and JavaScript.
      * *Mobile Application*: Create native mobile applications for popular platforms (iOS and Android) to extend the reach and usability of the application.
* *Server-Side*:
  + - * *Web Server*: Utilize a web server to handle incoming client requests and manage the application's logic.
* *Application Logic*: Implement the core business logic of the application, including user registration, job search and posting, shift management, payment processing, and rating systems.
* *Database*: Use a relational or NoSQL database to store and manage user profiles, job listings, shift data, payment information, and other relevant data.
* *APIs and Services:*
  + - *Authentication and Authorization*: Implement secure authentication and authorization mechanisms to protect user data and ensure access control.
    - *Payment Gateway Integration*: Integrate with a reliable and secure payment gateway to facilitate payment transactions between companies and job seekers.
    - *SMS/Email Notifications*: Integrate with a service provider to send notifications and reminders to users via SMS or email.
* *Security*: Implement encryption protocols (e.g., HTTPS) to secure data transmission between clients and the server. Apply best practices for secure coding and protect against common vulnerabilities such as SQL injection and cross-site scripting (XSS).Regularly update and patch software components to address security vulnerabilities.
* *Scalability and Performance*: Utilize cloud-based infrastructure (such as AWS, Azure, or Google Cloud) to provide scalability, load balancing, and automatic scaling based on demand. Employ caching mechanisms to improve performance and reduce database load. Monitor and optimize system performance using appropriate tools and techniques.
* *Maintenance and Monitoring*: Set up monitoring tools and logs to track system performance, detect errors, and troubleshoot issues promptly. Regularly update and maintain the application to ensure compatibility with new technologies, security patches, and bug fixes.
* It is crucial to conduct a thorough analysis of your specific requirements, expected user load, budget, and other factors to design a system architecture that meets your needs effectively. Consider consulting with experienced software architects and engineers to further refine and tailor the architecture to your specific use case.

**9. Assumptions and Dependencies**

**9.1 Assumptions:**

* *Third-Party Payment Provider*: It is assumed that a partnership or integration with a reliable third-party payment provider will be established to facilitate immediate money transfers between employers and employees.
* *Insurance Company Partnership*: It is assumed that a collaboration with an insurance company will be established to provide the necessary insurance coverage for clients, including both employers and employees.
* *Stockholders and Funding*: It is assumed that the app will have contributing stockholders or investors who provide the necessary funding and support for the development, maintenance, and growth of the application.
* *Sponsorship and Partnerships*: It is assumed that partnerships or sponsorship agreements, such as with "Shahrvand," will be established to incorporate the app into their platforms, expanding its reach and user base.

**9.2 Dependencies:**

* *Integration and API*: The successful integration of the app with the third-party payment provider and insurance company will depend on their availability, technical compatibility, and the establishment of appropriate APIs or communication channels.
* *Legal and Regulatory Compliance*: The app's operation will depend on meeting the legal and regulatory requirements of the country or region where it will be deployed, including compliance with labor laws, privacy regulations, and financial regulations.
* *Funding and Investor Support*: The app's development and ongoing operations will depend on securing funding from stockholders or investors, as well as their continued support and involvement in the growth and success of the application.
* *Sponsorship and Partnership Agreements*: The successful incorporation of the app into platforms like "Shahrvand" will depend on reaching mutually beneficial agreements and maintaining positive relationships with sponsors and partners.
* It is important to actively manage and establish these dependencies and assumptions by conducting thorough research, negotiating partnerships, and ensuring legal and regulatory compliance. Continuous communication and collaboration with stakeholders, investors, and partners will be crucial for the success of the "Karisho" application.

**10. Constraints**

**10.1 Time Constraint:**

The development and deployment of the application must be completed within a specific timeframe to meet market demands and stay competitive. Timely delivery of each phase and milestone is crucial for the success of the project.

**10.2 Budget Constraint**:

The project must adhere to a predefined budget for development, infrastructure setup, maintenance, and operational costs. Financial constraints will impact the selection of technologies, resources, and implementation strategies.

**10.3 Technology Constraints**:

The application must be developed using specific technologies and frameworks that are compatible with the desired features and platform requirements. The selection of technologies may be influenced by factors such as scalability, security, and integration capabilities.

**10.4 Regulatory and Compliance Constraints**:

The application must comply with all applicable laws, regulations, and industry standards. This includes data protection, privacy, labor laws, financial regulations, and any other legal requirements specific to the country or region of operation.

**10.5 Scalability Constraint**:

The application should be designed and implemented in a way that allows for future scalability and growth. It should be able to handle increasing user traffic, job listings, and interactions without significant performance degradation.

**10.6 Security Constraint:**

Ensuring the security of user data, financial transactions, and the overall platform is a critical constraint. The application must employ robust security measures, such as encryption, access controls, and vulnerability management, to protect against unauthorized access, data breaches, and other security threats.

**10.7 Usability Constraint:**

The application must provide a user-friendly and intuitive interface to ensure ease of use for both job seekers and employers. Usability testing and feedback should be incorporated to continually improve the user experience and address any usability constraints.

**10.8 Integration Constraint:**

The application may need to integrate with external systems, such as third-party payment providers, insurance companies, and other platforms. Integration constraints may include technical compatibility, availability of APIs, and adherence to partner agreements.

**10.9 Resource Constraint**:

Availability of resources, including human resources, infrastructure, and technical expertise, may impose constraints on the development, maintenance, and operational aspects of the application. Effective resource management and allocation are crucial to meeting project requirements within the given constraints.

* Note: These constraints are provided as examples. Please assess and identify the specific constraints applicable to your project, considering factors such as business objectives, market conditions, and organizational limitations.

**11. Testing Requirements**

**11.1 Functional Testing**:

Functional testing should be performed to ensure that all the functional requirements of the Karisho application are met. This includes testing the user registration and profile management, job search and filtering, job posting and management, application and selection process, shift confirmation and reminders, payment and feedback, and other functional modules of the application.

**11.2 Performance Testing:**

Performance testing should be conducted to assess the application's performance under various load conditions. This includes testing the application's ability to handle a high number of concurrent users, response time for job listing retrieval, and overall system performance to ensure a smooth and seamless user experience.

**11.3 Security Testing:**

Security testing is essential to identify and address potential vulnerabilities in the application. It should include testing for data protection, secure authentication and authorization mechanisms, prevention of unauthorized access, secure payment transactions, and protection against common security threats such as cross-site scripting (XSS) and SQL injection.

**11.4 Usability Testing:**

Usability testing should be carried out to evaluate the user-friendliness and intuitiveness of the application. It involves gathering feedback from users to assess the ease of navigation, clarity of instructions, and overall user satisfaction. Usability testing helps identify any usability issues and make necessary improvements to enhance the user experience.

**11.5 Compatibility Testing:**

Compatibility testing is necessary to ensure that the Karisho application functions properly across different devices, operating systems, and web browsers. It should be performed to validate the compatibility of the application with popular browsers and mobile platforms, ensuring consistent performance and functionality.

**11.6 Integration Testing:**

Integration testing should be conducted to verify the seamless integration of the Karisho application with any external systems or APIs. This includes testing the integration with third-party payment gateways, insurance providers, and other external services to ensure smooth data exchange and interoperability.

**11.7 Regression Testing**:

Regression testing should be performed after any changes or updates to the application to ensure that existing functionality is not affected. It involves retesting previously tested features and functionalities to verify their continued proper functioning.

**11.8 User Acceptance Testing (UAT**):

User Acceptance Testing should involve real users or representative stakeholders testing the application in a real-world environment. It helps validate the application's readiness for deployment by ensuring that it meets the users' expectations and fulfills the intended business requirements.

**11.9 Localization and Internationalization Testing**:

If the Karisho application is intended for use in multiple regions or countries, localization and internationalization testing should be conducted. This ensures that the application is properly adapted to different languages, cultures, and regional requirements, including date formats, currency symbols, and language translations.

* Note: The specific testing requirements may vary based on the project's scope, development methodology, and other factors. It is recommended to create a comprehensive test plan that covers all the necessary testing activities to ensure the quality and reliability of the Karisho application.

**12. Deployment and Installation**

**12.1 Deployment:**

The deployment of the Karisho application involves the process of making the application available for use by the intended users. The deployment strategy should ensure a smooth and efficient deployment process while minimizing any disruption to the application's availability. The following considerations should be taken into account:

* *Hosting Environment:* Determine the appropriate hosting environment for the application, whether it is on-premises or cloud-based. Consider factors such as scalability, performance, security, and cost when selecting the hosting environment.
* *Server Configuration*: Configure the servers and necessary infrastructure components to support the Karisho application. This includes setting up the required hardware, operating systems, web servers, databases, and other dependencies.
* *Network Configuration*: Ensure that the network infrastructure is properly configured to enable communication between the application servers, databases, and other relevant systems. Implement appropriate security measure to protect data transmission.
* *Deployment Tools*: Utilize deployment tools and automation scripts to streamline the deployment process. These tools can help automate tasks such as code deployment, database setup, and environment configuration.
* *Staging and Production Environments*: Set up separate staging and production environments to facilitate testing and quality assurance before deploying new releases or updates to the live production environment.
* *Version Control:* Implement version control systems to manage the source code and track changes. This helps in maintaining a history of code revisions and enables easy rollback to previous versions if needed.

**12.2 Installation:**

The installation process involves the steps required to install and configure the Karisho application on the client's device or within their network infrastructure. The installation should be user-friendly and well-documented to guide users through the setup process. Consider the following:

* *Installation Package:* Package the Karisho application into an installation package or container that can be easily distributed and installed on the user's device or server.
* *Installation Instructions:* Provide clear and concise installation instructions that outline the steps required to install and configure the application. Include information on system requirements, dependencies, and any necessary configurations.
* *Configuration Options*: Allow users to customize certain settings during the installation process, such as database connection details, email configuration, or other application-specific configurations.
* *Post-Installation Tasks:* Provide instructions for any post-installation tasks that need to be performed, such as database initialization, setting up administrative accounts, or configuring integrations with external services.
* *Compatibility and Pre-requisites*: Clearly specify the compatibility requirements, including supported operating systems, web browsers, and other dependencies. Ensure that users meet these requirements before proceeding with the installation.
* *Troubleshooting and Support:* Include troubleshooting guides and provide support channels for users to seek assistance in case they encounter any issues during the installation process.
* It is important to thoroughly test the deployment and installation process to ensure that it is robust, efficient, and user-friendly. This will help ensure a successful deployment and smooth adoption of the Karisho application by the users.

**13. Maintenance and Support**

**13.1 Maintenance**:

Maintenance refers to the activities required to keep the Karisho application running smoothly and efficiently after its deployment. It involves addressing any issues, bugs, or errors that arise, as well as making updates or enhancements to the application as needed. The following considerations should be taken into account for maintenance:

* *Bug Fixing:* Promptly investigate and address any reported bugs or errors in the application. This includes identifying the root cause, developing a fix, and deploying the necessary updates or patches to resolve the issues.
* *Performance Optimization*: Continuously monitor and optimize the performance of the application to ensure it meets the expected performance standards. This may involve identifying and resolving bottlenecks, improving database queries, or optimizing code.
* *Security Updates*: Stay updated with the latest security vulnerabilities and patches. Regularly apply security updates to the application and underlying infrastructure to protect against potential security threats.
* *Compatibility Updates*: Keep track of updates to operating systems, web browsers, and other dependencies. Ensure that the Karisho application remains compatible with the latest versions of these components and make necessary updates or modifications to maintain compatibility.
* *Database Maintenance*: Perform routine database maintenance tasks, such as data backups, data integrity checks, and index optimization, to ensure the stability and reliability of the application's data storage.
* *Documentation Updates*: Keep the application documentation up to date, reflecting any changes or enhancements made to the application. This helps users and support teams understand the functionality and usage of the application.

**13.2 Support**:

Providing adequate support to users is crucial for ensuring their satisfaction and resolving any issues or queries they may have. The support process should be well-defined and include the following aspects:

* *Help Desk or Support Channels*: Establish a help desk or support channels, such as email, chat, or a dedicated support portal, to allow users to seek assistance. Provide clear instructions on how to access and utilize these support channels.
* *Response Time*: Define and communicate the expected response time for support requests. Aim to provide timely responses and ensure that users receive the assistance they need within a reasonable timeframe.
* *Knowledge Base:* Develop a comprehensive knowledge base or FAQ section that addresses common user queries and provides self-help resources. This can help users find answers to their questions without requiring direct support.
* *Training and Onboarding:* Offer training materials or resources to help users effectively use the Karisho application. This can include tutorials, user guides, or video demonstrations to assist users in understanding and navigating the application's features.
* *Continuous Improvement:* Collect feedback from users and incorporate it into future updates and enhancements of the application. Regularly assess the support process and make improvements based on user feedback and evolving needs.
* *Escalation Procedure*: Establish an escalation procedure for handling complex or critical support cases that require higher-level expertise or attention. Ensure that there is a clear process in place to escalate and prioritize such cases.
* By maintaining and providing effective support for the Karisho application, you can enhance user satisfaction, address issues in a timely manner, and continuously improve the application's performance and usability.

**14. UML Diagrams**

* Please note that the above SRS document is a template with placeholders for the sections that require