

Software Requirements Specification

For The

Enhanced Job and Candidate Application

Prepared by:

Simul Kadakia
Wesley Trescott
Gagandeep Singh

Table of Contents

SOFTWARE REQUIREMENTS SPECIFICATION.....	1
1. INTRODUCTION	1
1.1 PURPOSE	1
1.2 SCOPE.....	1
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	2
1.4 REFERENCES.....	2
2. GENERAL DESCRIPTION.....	3
2.1 PRODUCT PERSPECTIVE	3
2.2 PRODUCT FUNCTIONS	3
2.3 USER CHARACTERISTICS.....	3
2.4 GENERAL CONSTRAINTS	3
2.5 ASSUMPTIONS AND DEPENDENCIES	3
3. SPECIFIC REQUIREMENTS	4
3.1 EXTERNAL INTERFACE REQUIREMENTS.....	4
3.1.1 <i>User Interfaces</i>	4
3.1.2 <i>Hardware Interfaces</i>	6
3.1.3 <i>Software Interfaces</i>	7
3.1.4 <i>Communications Interfaces</i>	7
3.2 FUNCTIONAL REQUIREMENTS	8
3.2.1: <i>User class - Users</i>	8
3.2.2: <i>User class - Admin</i>	9
3.3 NON-FUNCTIONAL REQUIREMENTS	10
3.3.1 <i>Performance</i>	10
3.3.2 <i>Reliability</i>	10
3.3.3 <i>Availability</i>	10
3.3.4 <i>Security</i>	10
3.3.5 <i>Maintainability</i>	11
3.3.6 <i>Portability</i>	11
3.4 DESIGN CONSTRAINTS	11
3.5 LOGICAL DATABASE REQUIREMENTS	12
3.6 OTHER REQUIREMENTS	12
4. ANALYSIS MODELS	12
4.1 DATA FLOW DIAGRAMS (DFD).....	12
4.2 TRACEABILITY MATRIX:	17

1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a detailed description of the functionalities of the 'Enhanced Job and Candidate Application' system. Additionally, it will serve as a tool to communicate the design plans of Enhanced Job and Candidate Application with Computech Corporation so that both parties may fully understand the specifications, requirements, and design of the project. This document is primarily intended to be proposed to the client for its approval and as a reference for developing the first version of the system for the development team.

1.2 Scope

The 'Enhanced Job and Candidate Application' is a web application targeting initially Wayne State University students, and later a general audience, who can search for jobs based on their skill sets and apply to those they are interested in. The web application is not specific to a particular device and needs to be compatible with smartphones, tablets, and desktop environments. A full listing of guaranteed compatible mobile devices is given in the Mobile Platform Compatibility section below, and, in desktop environments, Internet Explorer, Google Chrome, and Safari are guaranteed compatible web browsers.

Users of the application can log into a secure portal where they can access their saved data such as email address, contact information, and experience. They can filter job searches based on specific criteria, such as location and salary, which will narrow the search results of the available jobs. Users can apply to the jobs that they are interested in, with the option of uploading a resume. To apply, users can either manually enter their data or populate the fields from saved data previously entered in their user profile. Based on users' saved information, the application will provide the user with an automatic job matching functionality that displays to the user a list of jobs potentially fitting his or her credentials.

Additionally, the application will feature an admin mode that will interface with Computech's Pinpoint system to delete or ban users, lock accounts, and search for users based on name, email, or location.

1.3 Definitions, Acronyms, and Abbreviations

Term	Definition
EJCA	Enhanced Job and Candidate Application
User	Job seekers who use the application
Admin/Administrator	System administrator who is given specific permission for managing and controlling the system
Admin portal	Part of the web application that provides special facilities to Admin/Administrator
Stakeholder	Any person who interacts with the application except developers
API (Application Programming Interface)	A generalized interface used to allow EJCA to interact with external software/applications (Pinpoint)
Front end	The part of the application the user interacts with
Back end	The part of application that handles data and is maintained by developers
UI (User Interface)	Front end of EJCA
DESC	Description
DEP	Dependency
MUST	The minimum level required to avoid failure
PLAN	The level at which an endeavor is successful
WISH	A desirable level of achievement that may not be attainable through available means
DEFINED	The official definition of a term

1.4 References

“Pinpoint.” *Computech Corporation Products*. Computech Corporation, n.d. Web. 9 Feb 2015.

Pliasas, Vygandas. “Creating a custom user login form with .NET C# MVC 4 Razor.” *Code Project: For those who code*. Code Project, 23 Oct 2012. Web. 9 Feb 2015.

Additional helpful sources not directly referenced in this document:

- Microsoft ASP.NET MVC - <http://www.asp.net/mvc>
- Microsoft SQL Server - <https://msdn.microsoft.com/en-us/sqlserver/aa336270.aspx>
- Razor - [http://www.asp.net/web-pages/overview/getting-started/introducing-razor-syntax-\(c\)](http://www.asp.net/web-pages/overview/getting-started/introducing-razor-syntax-(c))
- JQuery - <http://jquery.com/>

2. General Description

2.1 Product Perspective

The project is built from scratch, and thus there is no initial foundation to build our system upon. The system will consist of a backend and a frontend. The backend will handle the storage of job listings and user information data, as well as the querying and delivery of this data back to the frontend to be displayed. The frontend will allow a user to interact directly with a webpage which will allow a user to specify a particular filter and then send a query to the backend to retrieve data.

2.2 Product Functions

- User can log in
- User can browse open job listings
- User can search/filter based on certain criteria
- User can provide profile information and upload resumes
- User can apply to jobs
- User can check what jobs he or she has applied to
- Admin can login
- Admin can search for users by his or her first and last name
- Admin can lock/ban/delete users

2.3 User Characteristics

We expect two different types of users for this project: Users and Admins.

Users are considered the most important interactors because they are the ones who search for jobs and apply to the ones that they are interested in. They are also going to be the majority of site views because the web app is mainly focused on delivering content to them. Most potential users will simply browse job listings until they find job listing(s) they are interested in, at which point they can apply.

Admins are the second most important users of our site. They can search for users based on their first and last names. They can also lock, ban and remove users who abuse the application and apply to multiple jobs in a short period of time without any relevant experience/skills.

2.4 General Constraints

This is a web application and the details about the operating environment are not important, except in the context of rendering the user interface based on the type of device used. End users will use the application from a modern internet browser such as Safari, Chrome, or Internet Explorer.

2.5 Assumptions and Dependencies

Assumptions to properly use the application include:

- Internet connection with enough bandwidth (about 1 to 2 mbps) to fully render all web application pages

- A modern internet browser with an up to date JavaScript engine and support for HTTP cookies to remember returning user logins

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

In total, the web application will have eleven screens: a home screen, a user dashboard screen, a job info page, a job application page, a contact page, a login page, a register page, a help/general info page, a job search page, an update profile page, and an admin dashboard page. Preliminary renderings for the first four pages are shown below.

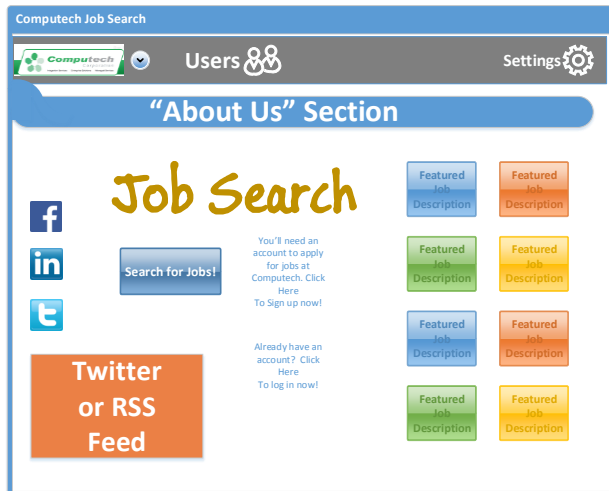
Each page of the site will contain a top gray menu bar with the Computech logo, a chevron dropdown menu, a Users dropdown menu, and a Settings dropdown menu. Clicking on the Computech logo takes the user to the home screen. The chevron dropdown will have hyperlinks to the Contact page, as well as to the Computech homepage, the Computech Products page, and the Computech Services page. Based on whether a user is logged in, either the user's name or the "Users" label will be displayed on the next dropdown. This menu will have hyperlinks to the Register and Login pages if the user is not logged in, and a hyperlink to log out if the user is logged in. Finally, the Settings dropdown will display a hyperlink to the Help page, and, if the user is logged in, a hyperlink to the Update Profile page.

Users accessing the site reach the home screen with a button to directly access a job search page. Here the user chooses the type of search he or she wants to conduct by filtering with combo boxes and checkboxes based on required skills and experience, location, and pay rate. There will also be a section for featured jobs. The user will also have links to Computech's social network pages on Facebook, LinkedIn, and Twitter. Eventually, the site will also contain a Twitter feed section.

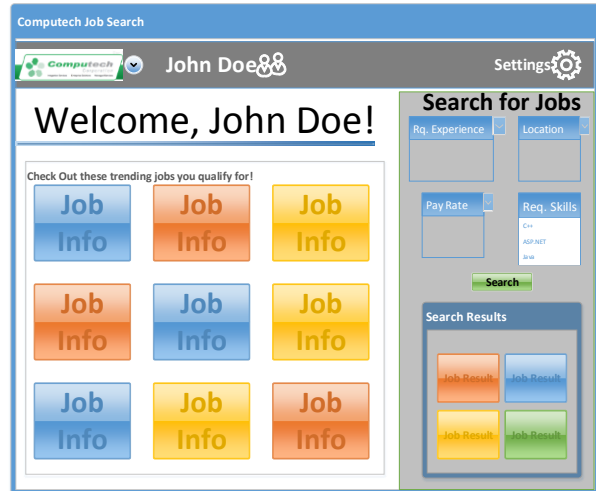
Once logged in, the user will see a dashboard screen displaying jobs matching his or her experience or credentials for which he/she can apply. The user will also have a search functionality to search for additional jobs.

When filling out a job application, the user will receive validation warnings if information entered does not match saved profile information. Also, the user will receive success notifications that the application was successfully submitted as well as warning alert notifications if the user leaves the application without submitting it.

Every page will include the standard header with the aforementioned links. Some preliminary user interface pages are shown below:



A preliminary home screen with the gray header



A preliminary user dashboard with recommended jobs at center and a job search option to the right.

Home Register Sign In

Register

Username

Password

E-Mail Address

Register New User

© 2015 - Computech Corporation

A preliminary user registration screen

Home Register Sign In

Login

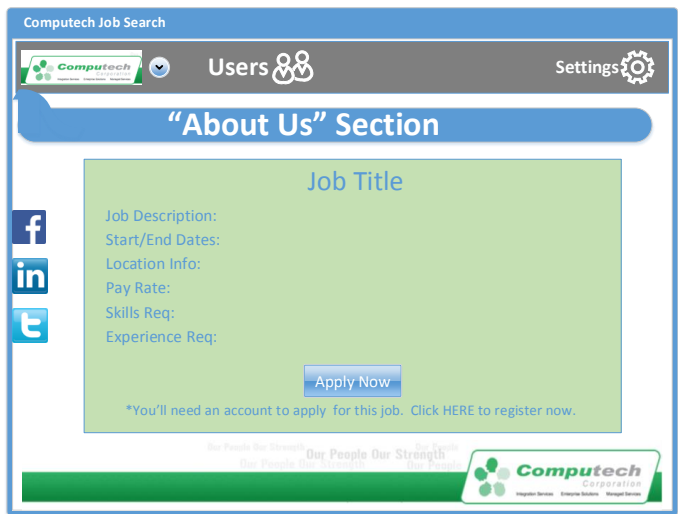
Username

Password

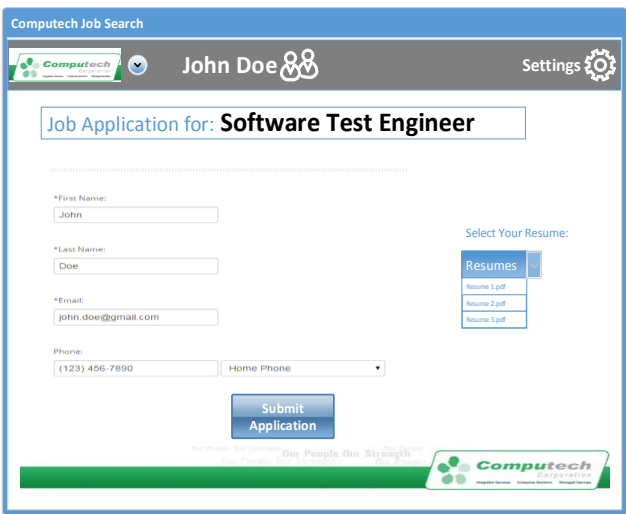
☐ Remember me on this computer

Log In

A preliminary login screen.



A preliminary job details page with an “Apply Now” button and the standard gray toolbar.



A preliminary job application page with populated fields and a resume selection option.

Computech Job Search

Position	Description	City	State	Country	
Summer Internship	Create Web Application using MVC, SQL Server Database	Detroit	MI	United States	Details
Web Developer	Design and develop new applications. Maintain current applications	Silicon Valley	CA	United States	Details
Senior Software Engineer	This position will take a lead role in application design, perform code reviews, assist with delivery estimates, and provide feedback about all aspects of the process.	Troy	MI	United States	Details
Quality Assurance Analyst	This role drives the automated load, performance, and integration testing framework for the development organization. They are responsible for the design, implementation, and continual improvement for this critical aspect of software quality for a product suite that requires an ability to scale to large deployments.	Chicago	IL	United States	Details
Database Engineer	Oversee physical database design and implement performance tuning, Lead code and design reviews, Implement disaster recovery for database servers, Create standards and best practices for database design and development	Dallas	TX	United States	Details

A preliminary page displaying jobs.

3.1.2 Hardware Interfaces

To function properly, the EJCA will require two types of hardware interfaces: a webserver and the user’s machine. The webserver will be the physical machine hosting the site, including the server software it runs as well as the database; the webserver will receive HTTP requests and provide HTTP responses, ensuring the constant availability of the application. This piece of hardware is owned by Computech. In particular, the server to be used is Windows Server 2008 R2 Standard, containing both IIS 7.5 webserver

and the Microsoft SQL Server 2008 R2 database. Additionally, each user of the application will access it using his or her own machine, whether that be in a desktop environment, or through a laptop, smartphone, or tablet.

3.1.3 Software Interfaces

EJCA will run on the MVC architecture, a design pattern offered as part of the Microsoft ASP.NET framework. Since this architecture provides a simple three-tier system for displaying pages and managing data, it is ideal for the needs of EJCA. Being a web application, EJCA will also require a web service, which will run on a web server. Both these software interfaces are vital to the public availability of the application so that it will be accessible to its users.

3.1.4 Communications Interfaces

As discussed above, the web application utilizes the MVC architecture, so communication between the model, view, and controller components of the system is important since they depend on each other. Though communication between each of these components is handled via the .NET framework, accessibility of the application itself to users is handled by the HTTP protocol. The messages passed will be GET and POST requests from the user or admin, as well as calls to insert, delete, or select data from our database. The security of HTTP data transmission is discussed below in the 3.3.4 Security section below.

3.2 Functional Requirements

3.2.1: User class - Users

3.2.1.1 Search:

DESC: User can search for various jobs available. When the user opens the website, the home page will show featured jobs with an option to conduct a job search. Users can also conduct a search for the same jobs once logged into their profiles.

REQ: None.

3.2.1.2 Register:

DESC: Users can create a new account by providing personal information such as email id, password and first and last name. This information will be stored in database and email id and password will be used to log in user.

REQ: None.

3.2.1.3 Login:

DESC: Given that a user has registered, the user can login by entering email id and password entered during registration process.

REQ: FR 3.2.1.2

3.2.1.4 Retrieve password:

DESC: Given that a user has registered, the user should then be able to retrieve password through email. User can request to change password which will send further instruction to his or her email.

REQ: FR 3.2.1.2

3.2.1.5 Apply:

DESC: Given that a user is logged in, the first page he or she can see is the home page. User can search/filter jobs and apply for jobs.

REQ: FR 3.2.1.3

3.2.1.6 Filter by skills, location, pay rate, and experience:

DESC: User can filter jobs based on skills, location, pay rate, and experience required for the job. Jobs can be filtered based on single criteria or multiple criteria.

REQ: None

3.2.1.7 Profile Information:

DESC: Given that a user is logged in, the user can enter his or her information such as skills and experience in his or her profile page. The user can store his or her information which can be used to autofill fields when applying for a job.

REQ: FR 3.2.1.2

3.2.1.8 Change Information:

DESC: Given that a user is logged in, the user can change information stored in his or her profile page. User should be able to change any fields stored in his or her profile page.

REQ: FR 3.2.1.3

3.2.1.9 Upload Resume:

DESC: Given that a user is logged in, the user can upload up to three resumes and attach them to his or her profile. User can only attach one resume file to each job application.

REQ: FR 3.2.1.3

3.2.2: User class - Admin

3.2.2.1 Login:

DESC: Admin can login by entering email id and password stored in database. Admin information is stored in database and there is no registration page for admin.

REQ: None

3.2.2.2 Search:

DESC: Admin can search for all registered users by their first and/or last names. Admin can search for users which will display basic user information such as name, skills and so on.

REQ: FR 3.2.2.1

3.2.2.3 Deactivate users:

DESC: Admin can deactivate users which will disable them to login into their account. Users who abuse the system can be deactivated by the admin which will lock them to log in to their account.

REQ: FR 3.2.2.1

3.2.2.4 Delete users:

DESC: Admin can delete users which will delete all of their information from the database. Users who abuse the system can be deleted by the admin. All information related to user such as login and profile information will be deleted from the system.

REQ: FR 3.2.2.1

3.3 Non-Functional Requirements

The following requirements specify, for the system level, the mandatory performance, reliability, availability, security, maintainability, and portability of the software product.

3.3.1 Performance

This requirement specifies the mandatory performance metrics of the system during user interaction with the software. This includes response time to user logins, registrations, job searches, and admin logins and activities. Included in this metric are page transition times, as for example from the home screen to the job search screen. Since database queries and requests in the application deal with either basic user information or job listings being retrieved from or entered into the database, load times will be a factor of how much RAM is available on the local machine and the speed of the server hosting the site. The webserver hosting EJCA has 16 GB of RAM, with a target speed, based on this configuration, of 100 KB per second. Additionally, given that the user machine has 1 GB of available RAM, and internet download speeds of at least 2 MBPS, we aim for overall loading quickness to be at a maximum of 3 seconds 100% of the time, with a target of less than two seconds 98% of the time, which will be verified during five hours of testing in our phase 3 system tests, as per our development plan.

3.3.2 Reliability

This requirement specifies the extent to which the software application reliably produces the correct job search results based on valid search criteria, as well as suggests jobs to users based on their skills and experience. Also, this metric encompasses report messages to application users regarding lost internet connections or incorrect registration or search inputs. Each of these requirements will be thoroughly tested and evaluated during the phase 2 testing mentioned in the development plan. In particular, we plan to create at least 25 user accounts with both valid and invalid credentials (invalid credentials being, for example, a duplicate email address or password). These user accounts will contain various experience data, which will be used to assess job recommendation functionality. We will also perform at least 50 job searches using these accounts on a limited jobs database to assess correctness of search algorithms. In these tests, we will require correct error reporting 100% of the time, correct job search results 98% of the time, and correct job recommendations 98% of the time.

3.3.3 Availability

This requirement specifies the readiness of the system for use whenever it is run. This entails the accessibility of the application from an internet browser not considering network failures, meaning that Computech's webserver functions properly and that the application does not require periods of unavailability due to maintenance needs. Thus, this requirement assumes that the user has an internet connection so that the application can connect to the database. The availability requirement will be tested in the phase 3 system tests mentioned in the development plan. In testing, we plan for five hours of active use of the application, during which period our target system availability is 100% of the time, with a 98% of the time availability considered passing.

3.3.4 Security

This requirement specifies that the admin mode and all user passwords will be encrypted using the SHA1 encryption algorithm before being stored in the database. SHA1 was developed by the NSA and is

an example of a cryptographic hash function, considered nearly impossible to decode, thus ensuring the security of user password information from the system administrator. This requirement also encompasses the validation of user logins, so that incorrect user credentials will not result in access to the application. Furthermore, the functionality of the admin mode ensures that users abusing the application will be removed. Each of these security requirements will be expected to function correctly 100% of the time in our tests. Additionally, the Computech webserver, IIS 7.5, uses an SSL certificate to authenticate and encrypt data transmitted between users and the site using HTTP, so that user data flows securely between the client and server.

3.3.5 Maintainability

This requirement specifies the degree of extensibility and testability of the EJCA. Firstly, extensibility implies both the easy addition of new features to the application and also the ease of portability to new, previously unsupported devices. This requirement is satisfied by our use of popular technology (ASP.NET), which is a framework widely used in web application development across multiple platforms, therefore attesting to its adaptability to new environments. Also, our program will be written using the MVC design pattern, which allows for flexible addition of new features through new controllers and models. Furthermore, our application supports easy testability by taking advantage of the test suites generated by ASP.NET for all of the functionality of the EJCA.

3.3.6 Portability

This requirement specifies that the EJCA be portable to multiple devices. Given the up-front project requirement of compatibility with both mobile and desktop environments, the application will be designed and coded so that all its functionality is compatible with web browsers in PC, Mac, Android, iPhone, and iPad. The effect of this multi-platform design intent is that the application will be easily portable to each supported environment. Complete system testing of all functionality of the application will be carried out on each supported device during phase 3 testing.

3.4 Design Constraints

The largest design restraints in our team's view are the amount of RAM the web application consumes during normal use as well as the application's rendering in the default internet browsers of multiple devices. Consuming too much memory will slow down the user's device and provide a less than optimal user experience. Since database queries and requests in the application use ASP.NET's Entity Framework to store entries in cache memory, RAM is used on the user machine. Also, additional RAM will be used when we implement the user interface using the Bootstrap UI framework, which uses JavaScript objects in its configuration of pages. Given the limited main memory resources of most mobile devices, we will mirror the well documented Gmail web application example, which consumes around 100 MB of RAM during normal operation, and set this amount as the upper bound of acceptable main memory usage for our web application. As the EJCA is a much simpler application than the Gmail email client, the EJCA should not use more RAM than this amount.

Also, due to differences in browser software across devices, and differences in screen size and resolution, our user interface is constrained in size and complexity to support portability to touchscreen devices and devices with small screens. To resolve this constraint, we will use the Bootstrap UI framework, which was written using the mobile first approach. This means that Bootstrap style sheets

are built to render on smaller screens but are also scalable to larger screen sizes, based on the browser displaying the page. Using Bootstrap thus enables us to simultaneously create the desktop and mobile version of the site without having to implement two significantly different versions.

3.5 Logical Database Requirements

A Microsoft SQL Server database will be used in the backend to store information entered by the user, display jobs to users, and display registered users to admin. As mentioned above, the specific database management system to be used will be Microsoft SQL Server 2008 R2 Standard hosted on the Windows Server 2008. The front-end will be on the user/admin machine. Database will be hosted on the same server as the application and there is no limit on the amount of data that can be stored in the database as, when necessary, the database can be hosted on multiple servers to accommodate data.

3.6 Other Requirements

3.6.1 Mobile Platform Compatibility

The web application is guaranteed compatible with following mobile phones and tablets:

- iPhone 6 (iOS 8.1.3)
- iPad Air (iOS 8.1.3)
- Samsung Galaxy S5 (Android v5.0 Lollipop)

The user view is configured to the device's screen resolution and can be accessed from any supported web browser installed on the device. The user has the ability to use all functionality in the app on these mobile devices. However, the user is unable to upload resumes from iOS based devices. To support this compatibility, a mobile version of the web pages using Bootstrap will be rendered specific to these devices, which will be shown to the mobile user.

3.6.2 Social Media Integration

The web application also will allow users to login with their Facebook and Google accounts from the login screen. After the user enters valid account login credentials, the web app will pull the user information from his or her Google/Facebook account and auto fill the user profile information with, allowing the user to bypass entering his data manually. The user has the ability to change his or her profile information at any time. This functionality will be achieved using Google and Facebook APIs, specifically Google's OpenID Connect and Facebook's OAuth 2.0.

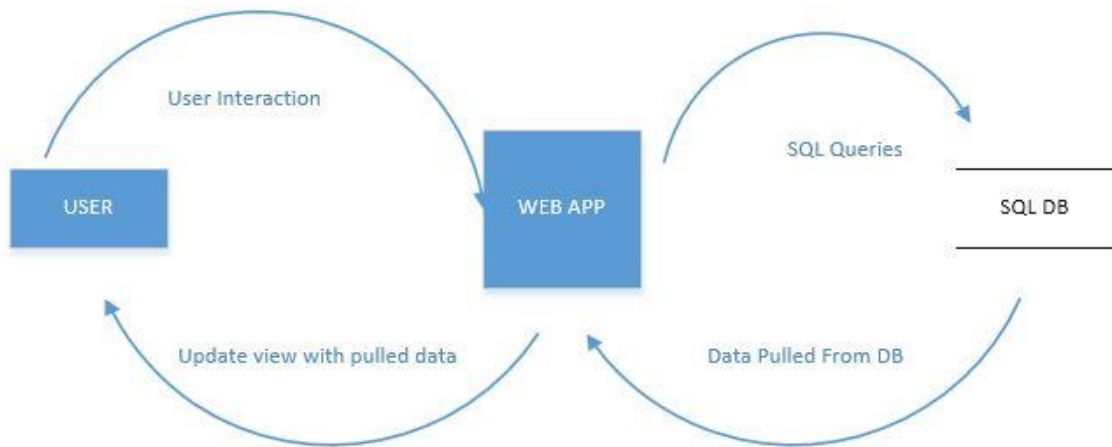
4. Analysis Models

4.1 Data Flow Diagrams (DFD)

The following diagrams model the processes of our application, depicting both how data flows between the user and the application, and how data flows within the application.

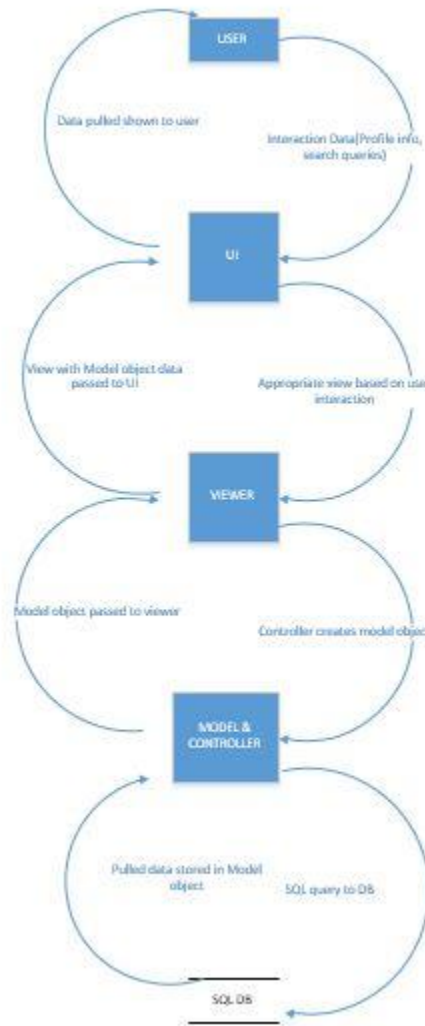
4.1.1 Level1 DFD

The Level1 DFD shows the flow of data among the user, web application, and the SQL database.



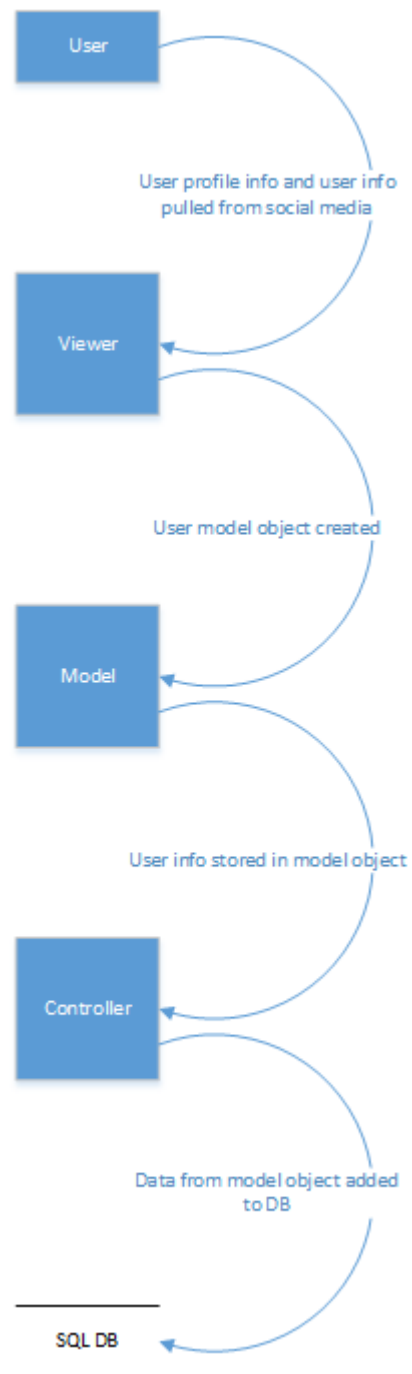
4.1.2 Level2 DFD

Level2 DFD provides in-depth view of the data flow among the various components of the web application such as the UI, Application Core i.e. Model and Controller components, Viewer, and the SQL database.

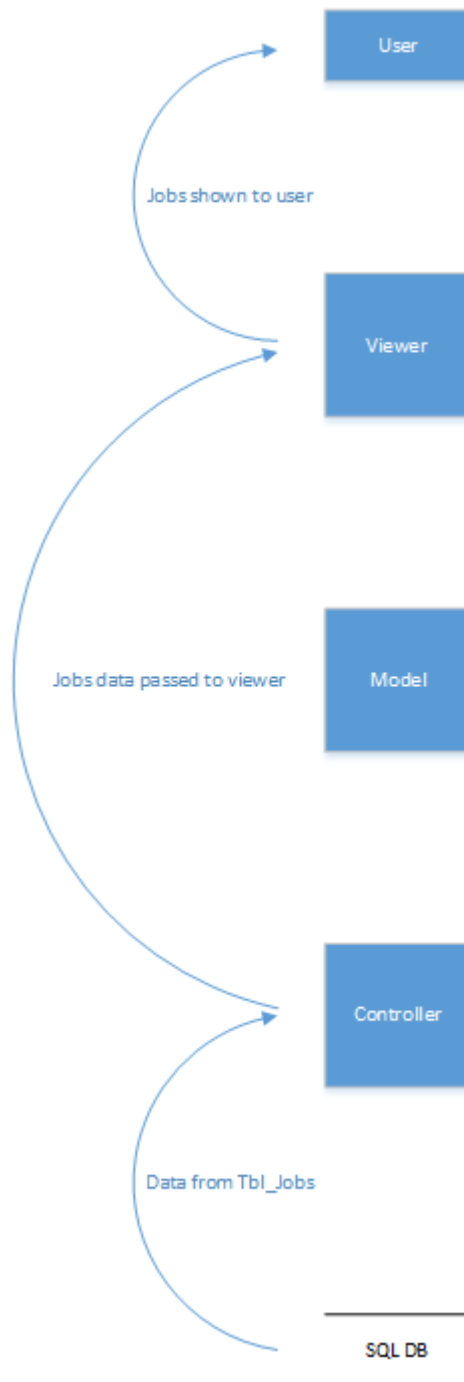


4.1.3 Level3 DFD

User data flow



Jobs data flow



4.2 Traceability Matrix:

[illegible]