System and Software Architecture Description (SSAD)

REFERsy.com

Team No.10

Sreenarayan Ashokkumar - Project Manager, Feasibility Analyst

Guanhui Chen - Software Architect

Dongyoung Jung - Life Cycle Planner

Chunming Lu - Prototyper

Donglin Pu – Prototyper

Sharanya Radhamohan - Requirements Engineer

Fangjie Zhu - Operational Concept Engineer

Suchit Mathur - IIV&V, Quality Focal Point

Date: 12/05/2014

Version History

Date	Author	Version	Changes made	Rationale
08/25/05	PA	0.0	Original template for use with Instructional ICM-Sw v1.0	Initial draft for use with Instructional ICM-Sw v1.0
05/22/09	SK	0.1	Embedded description in each Table	To be consistent with ICM EPG template set standard V2.1
10/13/14	SR	1.0	First Version of the SSAD draft.	To present the SSAD report as a part of the Foundation Commitment Package.
10/20/14	SR	2.0	Revised section 1 and 2 after FED	Ready for Foundation Commitment Package with feedback from ARB
12/01/14	SR, DP, CL, GC, SA	3.0	Added section 3, 4, 5.	Ready for draft Development Commitment Package
12/05/14	CL, GH, DJ	3.1	Changes Hardware Component Diagram and Deployment Diagram Change the classes design Change use case diagram	Feedback from ARB

Table of Contents

Table of Tables

Table of Figures

1.   Introduction

1.1 Purpose of the SSAD

The purpose of the System and Software Architecture Description is to construct further detail than is contained in the OCD, WinWin Agreements, and prototype regarding what the system will do and at what level it will perform. It is used by the developers as a reference to the system architecture.

1.2 Status of the SSAD

This is version 3.0 if the System and Software Architecture Description. It has completed all 5 sections of this document.

2.   System Analysis

2.1 System Analysis Overview

The primary purpose of REFERsy.com is to create a better social job referencing system than LinkedIn. It is a job search site which allows a user to sign in to Facebook, LinkedIn, and G+ and find available jobs within their network of friends. Every job that the user sees listed will also tell them who, in their network, can refer them to that job.

The user creates an account on REFERsy.com and signs in to at least one social media account (Facebook / Google+ / LinkedIn). The user also updates their profile information, based on which job search results are refined. REFERsy.com fetches the available jobs from Indeed.com based on the job title or location keywords and checks to see if there are people on the user's networks who work in organizations listed in the search results.

The results are then displayed, allowing the user mark a job as favorite, save the current search, email a friend who can provide a referral or subscribe for email notifications.

2.1.1   System Context
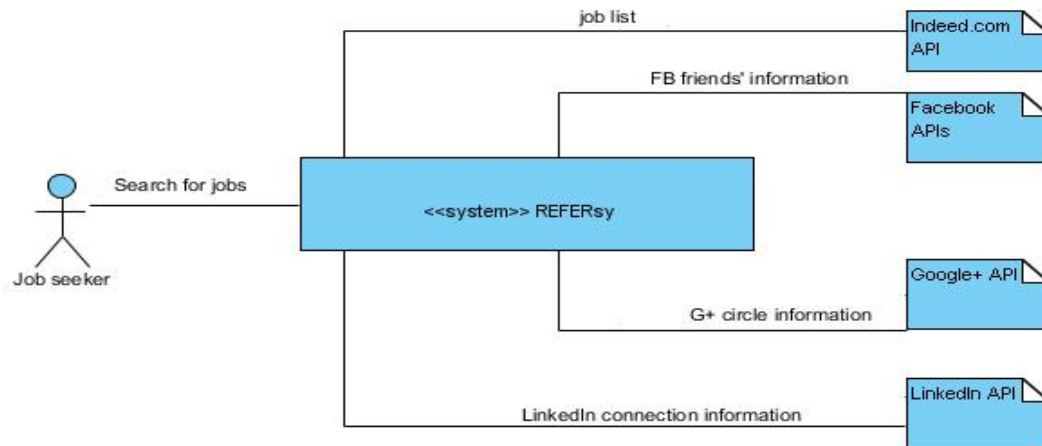
Figure 1: System Context Diagram

Table 1: Actors Summary

Actor	Description	Responsibilities
Job Seeker	Any user who is looking for jobs on REFERsy.com	Can login to/ logout from REFERsy.com, sign in to social media accounts, search for jobs, update their profile, subscribe or unsubscribe from email notifications, save search results and mark jobs as favorites,

2.1.2 Artifacts & Information
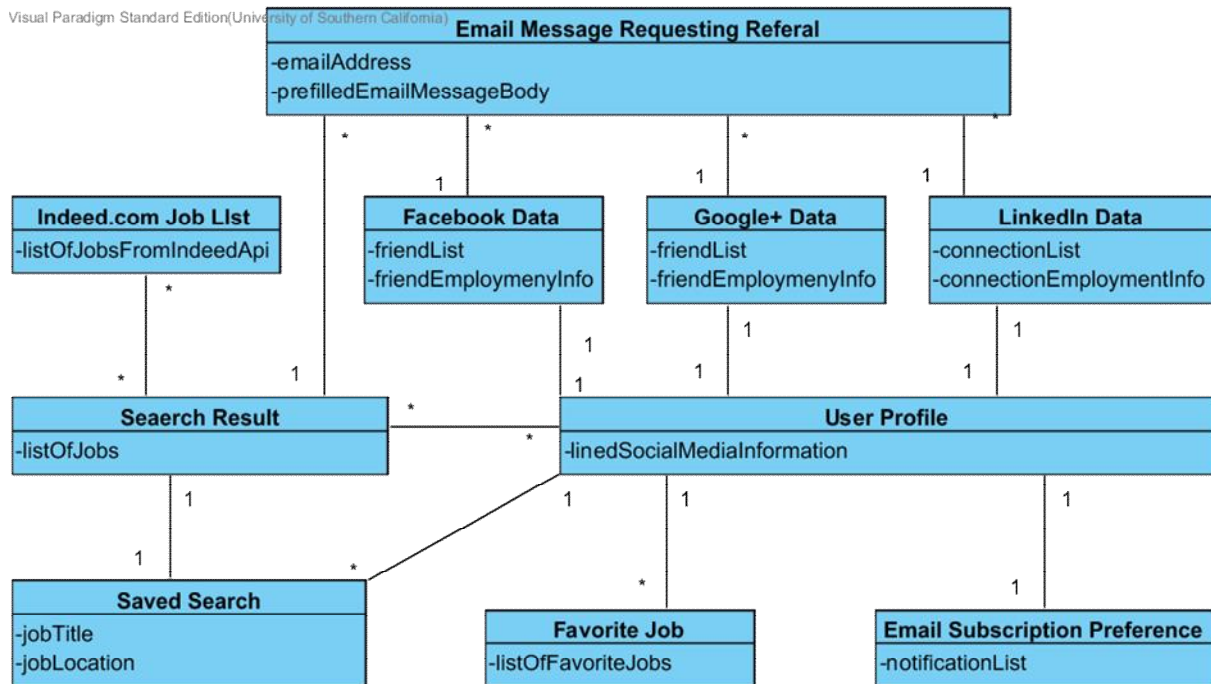
Figure 2: Artifacts and Information Diagram

Table 2: Artifacts and Information Summary

Artifact	Purpose
Indeed.com Job List	Contains a list of available jobs on Indeed.com
Facebook Data	Contains the user's friend's list on Facebook and their employer information
Google+ Data	Contains the list of user's friends on Google+ and their employer information
LinkedIn Data	Contains the list of the user's connections on LinkedIn and their employer information
Search result	Contains a list of available jobs based on job title and/or location keywords
Email message requesting referral	Prefilled email message body requesting a contact for a referral for a particular job
User profile	Contains information about the user such as which social network they have linked to.
Saved search	Contains a list of all the searches that the user has saved
Favorite job	Contains a list of all the jobs that the user has marked as a favorite
Email Subscription preference	Contains the user's preferences regarding receiving notification emails about specific jobs and choose how often they want to receive emails.

2.1.3   Behavior
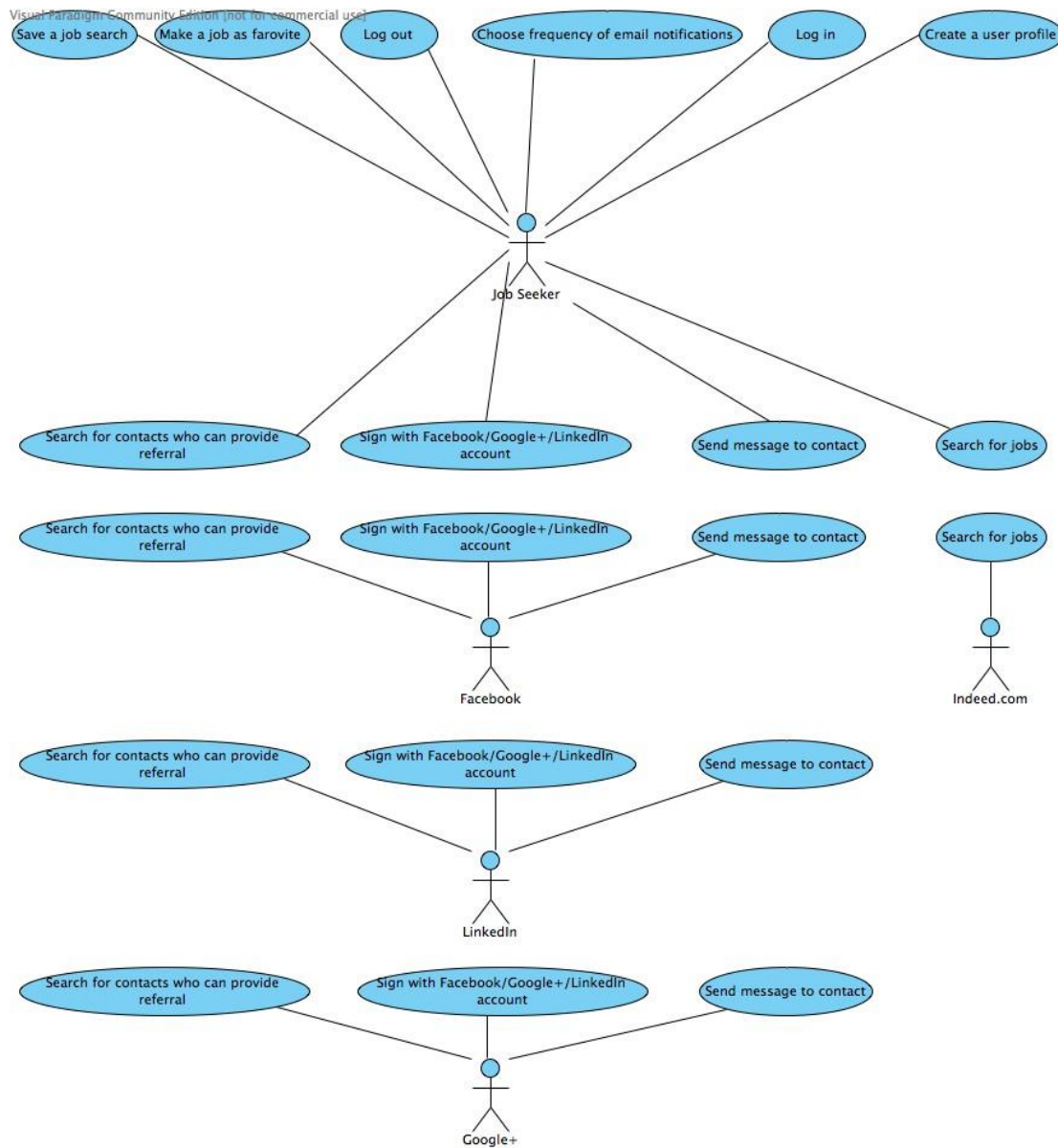
Figure 3: Process Diagram

2.1.3.1 Account Management

2.1.3.1.1 Create a new account

Table 3: Process Description for "Create a new account"

Identifier	UC-1: Create an account
Purpose	Allow a new user to create an account on REFERsy.com
Requirements	The user must use an email address to create the account
Development Risks	None
Pre-conditions	The email address used is unique and valid
Post-conditions	A new user account is created

Table 4: Typical Course of Action for "Create a new account"

Seq#	Actor's Action	System's Response
1	Enter an email address and password	
2	Click on 'Create Account'	
3		System validates the email address and password combination, creates a new user account and displays the user's profile page.

Table 5: Exceptional Course of Action for "Create a new account"

Seq#	Actor's Action	System's Response
1	Enter an email address and password	
2	Click on 'Create Account'	
3		The system finds that one of the required fields are empty or the email address is already associated with another user account and/or the password does not match specific criteria (like use of a minimum number of characters, special symbols)
4		Displays an error message

Note: There is no Alternate Course of Action for this scenario

2.1.3.1.2 Login to REFERsy.com

Table 6: Process Description for "Login to REFERsy.com"

Identifier	UC-2: Login
Purpose	Allow an existing user to log in to REFERsy.com
Requirements	The user must enter an email address and password
Development Risks	None
Pre-conditions	The user already has a REFERsy.com account
Post-conditions	The user's profile and search page is displayed

Table 7: Typical Course of Action for "Login to REFERsy.com"

Seq#	Actor's Action	System's Response
1	Enter an email address and password	
2	Click on 'Login'	
3		System checks if the user has entered a valid email address and password combination. If valid, display user's profile and search page.

Table 8: Exceptional Course of Action for "Login to REFERsy.com"

Seq#	Actor's Action	System's Response
1	Enter an email address and password	
2	Click on 'Login'	
3		System finds that the user has entered an invalid email address and password combination. Display error message on login page.

Note: There is no Alternate Course of Action for this scenario

2.1.3.1.3 Log out from REFERsy.com

Table 9: Process Description for "Log out from REFERsy.com"

Identifier	UC-3: Logout
Purpose	Allow a logged in user to log out from to REFERsy.com
Requirements	The user must click on the Logout button
Development Risks	None
Pre-conditions	The user is logged in to REFERsy.com
Post-conditions	The user's current session is ended and is logged out of REFERsy.com.

Table 10: Typical Course of Action for "Log out from REFERsy.com"

Seq#	Actor's Action	System's Response
1	Click on 'Logout'	
2		System saves the user's context, ends the current session, logs the user out of REFERsy.com and displays the initial login screen.

Note: There are no Exceptional or Alternate Courses of Action.

2.1.3.1.4 Sign in to social media accounts

Table 11: Process Description for "Sign in to social media accounts"

Identifier	UC-4: Sign in to social media accounts
Purpose	Allow a user to sign into Facebook, Google+ and/or Linkedin to fetch social network information
Requirements	The user must provide login credentials to sign into social media accounts
Development Risks	None
Pre-conditions	The user already has an account in at least one social network out of Facebook, google+ and LinkedIn
Post-conditions	The user's social media login credentials are saved by the system for future use

Table 12: Typical Course of Action for "Sign in to social media accounts"

Seq#	Actor's Action	System's Response
1	Click on 'Sign into Facebook', 'Sign into Google+', 'Sign into LinkedIn'	
2		System displays a popup window asking for user credentials
3	Enter username and password for social media account(s)	
4		System gets permission and access rights from the respective websites if the username and password combination is valid
5		Saves social media account credentials for future use
6		Display success message and user profile/search page

Table 13: Exceptional Courses of action for "Sign in to social media accounts"

Seq#	Actor's Action	System's Response
1	Click on 'Sign into Facebook', ,'Sign into Google+'or 'Sign into LinkedIn'	
2		System displays a popup window asking for user credentials
3	Enter username and password for social media account(s)	
4		Login credentials are invalid
6		Display error message

Note: There are no alternate courses of action for this scenario.

2.1.3.1.5 Update user profile

Table 14: Process Description for "Update user profile"

Identifier	UC-5: Update user's profile
Purpose	Modify profile information such as qualifications, fields of interest and work experience of the user.
Requirements	The user must provide valid information for each required field
Development Risks	None
Pre-conditions	The user is logged into REFERsy.com
Post-conditions	The updated user profile information is saved and displayed

Table 15: Typical Course of Action for "Update user profile"

Seq#	Actor's Action	System's Response
1	Click on 'Edit my profile'	
2	Enter information for required fields	
3	Click on 'Save changes'	
4		Save updated information and display user's profile page.

Table 16: Exceptional Course of action for "Update user profile"

Seq#	Actor's Action	System's Response
1	Click on 'Edit my profile'	
2	Enter information for required fields	
3	Click on 'Save changes'	
4		Finds invalid data or blank fields. Display error message and do no save any changes to the user's profile.

Note: There are no alternate courses of action.

2.1.3.2 Job search

2.1.3.2.1 Search for jobs

Table 17: Process Description for "Search for jobs"

Identifier	UC-6: Search for jobs
Purpose	Fetch available and relevant jobs from Indeed.com
Requirements	The user must provide job title and/or location keyword
Development Risks	Construct valid queries to retrieve good results from Indeed.com API
Pre-conditions	The user is logged into REFERsy.com
Post-conditions	The job search results are displayed

Table 18: Typical Course of Action for "Search for jobs"

Seq#	Actor's Action	System's Response
1	Enter 'job title' and/or 'location' keyword	
2	Click on 'Search'	
3		Use search keywords to filter available jobs fetched from Indeed.com. Display search results.
4	Browse, sort and filter search results	

Table 19: Exceptional Course of action for "Search for jobs"

Seq#	Actor's Action	System's Response
1	Enter 'job title' and/or 'location' keyword	
2	Click on 'Search'	
3		Incorrect search keyword(s) or no results found for keywords. Display error message on search page.

Note: There are no alternate courses of action.

2.1.3.2.2 Save search results

Table 20: Process Description for "Save search results"

Identifier	UC-7: Save a search
Purpose	Save a particular search
Requirements	The user must click on 'Save Search'
Development Risks	Implementing the save feature to save a search with all the filters and conditions that the user has specified
Pre-conditions	A job search has been performed
Post-conditions	Search is saved

Table 21: Typical Course of Action for "Save search results"

Seq#	Actor's Action	System's Response
1	Click on 'Save Search'	
2		Save the current search with all the applied filters and conditions specified by the user
3		Display a success message and a list of all saved searches

Table 22: Exceptional Course of action for "Save search results"

Seq#	Actor's Action	System's Response
1	Click on 'Save Search'	
2		Finds the search has already been saved or no search results to save
3		Display an error message and a list of all previously saved searches

Note: There are no alternate courses of action.

2.1.3.2.3 Mark favorite jobs

Table 23: Process Description for "Mark favorite jobs"

Identifier	UC-8: Mark a job as favorite
Purpose	Keep track of favorite jobs
Requirements	The user must click on 'Favorite Job'
Development Risks	None
Pre-conditions	A job search has been performed
Post-conditions	The job is added to a list of favorite jobs

Table 24: Typical Course of Action for "Mark favorite jobs"

Seq#	Actor's Action	System's Response
1	Click on 'Favorite Job'	
2		Save the current job
3		Display a success message

Table 25: Exceptional Course of action for "Mark favorite jobs"

Seq#	Actor's Action	System's Response
1	Click on 'Favorite Job'	
2		Find that the job has already been added to the list of favorite jobs
3		Display an error message and display a list of all favorite jobs

Note: There are no alternate courses of action.

2.1.3.2.4 Send a referral request email

Table 26: Process Description for "Send a referral request email"

Identifier	UC-9: Send a referral request email
Purpose	To request a referral for a particular job listed in the job search results
Requirements	The user must be able to view the list of people who are working in organizations, who can provide referrals
Development Risks	None
Pre-conditions	The email address of the contact must be available
Post-conditions	A prefilled email message body is displayed which the user can edit and send to the contact.

Table 27: Typical Course of Action for "Send a referral request email"

Seq#	Actor's Action	System's Response
1	Click on 'Request referral'	
2		Display all friends who work in the organization and can provide a referral
3	Select whom to ask for referrals	
4	Edit message body, if necessary	
5		Send the email message and display success message.

Table 28: Exceptional Course of action for "Send a referral request email"

Seq#	Actor's Action	System's Response
1	Click on 'Request referral'	
2		Find no contacts who can provide a referral or email found
3		Display an appropriate error message

Note: There are no alternate courses of action.

2.1.3.2.5 Email subscription

Table 29: Process Description for "Email subscription"

Identifier	UC-10: Email subscription
Purpose	To sign up for email notifications regarding some specific jobs
Requirements	The user must be logged into REFERsy.com
Development Risks	None
Pre-conditions	A valid email address is provided during sign in.
Post-conditions	Successfully subscribe to emails.

Table 30: Typical Course of Action for "Email subscription"

Seq#	Actor's Action	System's Response
1	Click on ' Subscribe for notification emails'	
2		Save user's preferences for receiving emails
3		Display success message and add email address to subscription email list.

Table 31: Exceptional Course of action for "Email subscription"

Seq#	Actor's Action	System's Response
1	Click on ' Subscribe for notification emails'	
2		Invalid email address. Display error message.

Note: There are no alternate courses of action.

2.1.4 Modes of Operation

The system operates in only one mode.

2.2 System Analysis Rationale

The most crucial aspect of REFERsy.com is scraping information from the APIs of the user's social media networks to determine who can provide referrals for jobs pulled from Indeed.com's API.

The system analysis allowed the developers get into more detail about how the user searches for jobs and uses their social media connections to request a referral.

In this analysis of REFERsy.com,

1. Identified the type of user profile information required to refine search results.
2. Identified scenarios when the user may not be able to view their friend's employer information.
3. Identified scenarios when the user may not be able to view their friend's email address due to privacy settings.

This enabled us to discuss possible scenarios in which the system may not work as expected and establish an alternate course of action using feedback from the client.

3.   Technology-Independent Model
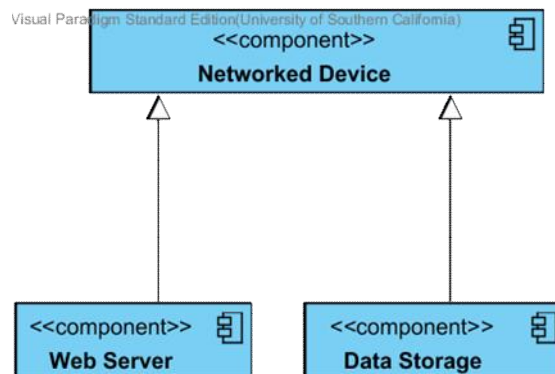
3.1 Design Overview

3.1.1   System Structure

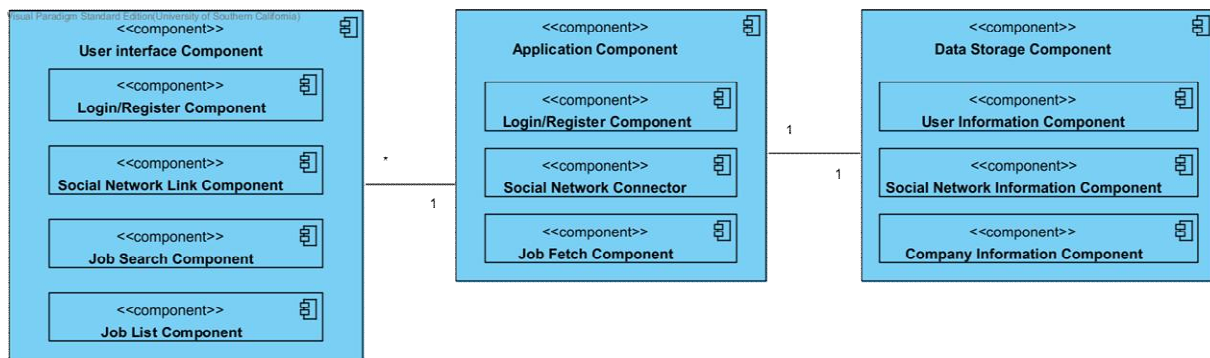
Figure 4: Hardware Component Class Diagram
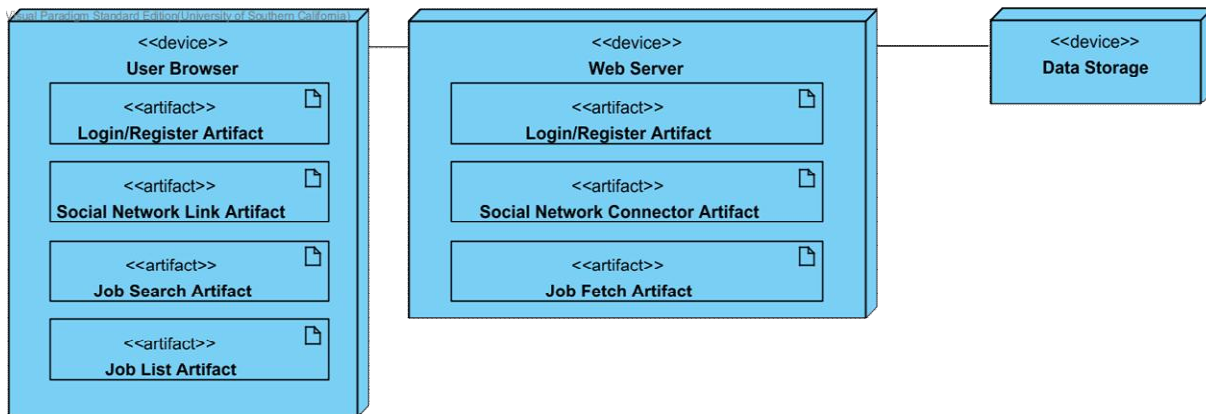

Figure 5: Software Component Class Diagram

Figure 6: Deployment Diagram

Table 32: Hardware Component Description

Hardware Component	Description
Networked Device	The device that connects to the internet to serve the system. It contains server components and data storage units.
Web Server	It receives request, interacts with the data storage and returns response.
Data Storage	It stores data. It could be a database, a flat file or a cloud service.

Table 33: Software Component Description

| Software Component	Description
User Interface Component	This component contains the REFERsy webpages that users can interact with.
Application Component	This component receives request from the User Interface Component, handles login/register functions, retrieves Social Network information and fetches job lists.
Data Storage Component	This component stores User information which stores registered user credentials, Social Network Information which stores registered users' friends and their employment information, and Company Information which stores company name and location.

3.1.2   Design Class
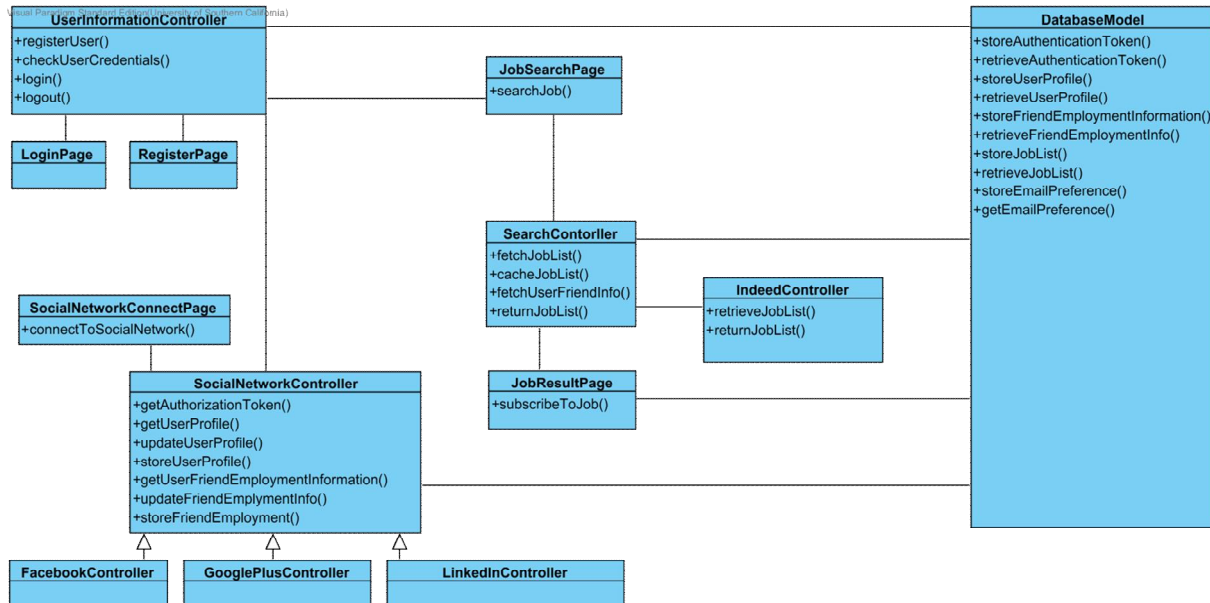
Figure 7: Design Class Diagram

Table 34: Design Class Description

| Class	Type	Description
Search Controller	Control	Handles search request and response.
User Information Controller	Control	Handles user registration, login and logout.
Social Network Controller	Control	Handles social network connection and information retrieval.
Indeed Controller	Control	Handles job search from Indeed.com API
Login Page	Boundary	The user login page.
Register Page	Boundary	The new user register page.
Social Network Connect Page	Boundary	The page that allows users to connect to Facebook, Google+ and LinkedIn.
Job Search Page	Boundary	The job search page.
Job Result Page	Boundary	The page that lists all matching jobs.
Database Model	Entity	This class stores user information, friends employment information and company information.

3.1.3   Process Realization
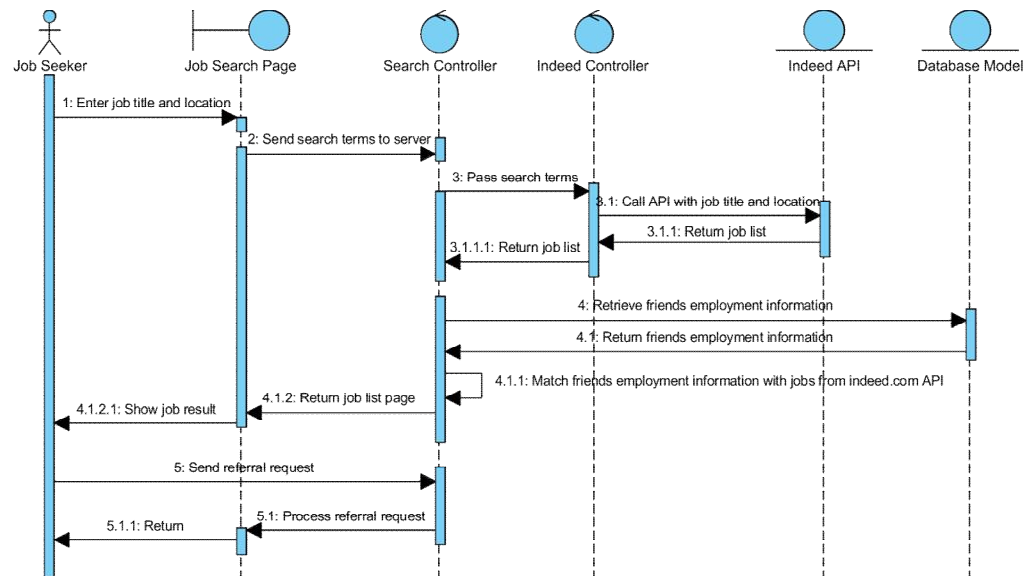
Figure 8: Process Realization Diagram

3.2 Design Rationale

We follow the model-view-controller pattern, where model classes handle the interaction with the database, view classes interact with users and controller classes handle request, response and intermediate operations. By adopting this pattern, user interface, system logic and database are separated and thus modification in one area would not affect the functionalities of the other areas. If in the future a mobile App version is needed, the developer team only needs to add additional view classes for the mobile App platform. If additional information needs to be stored in the database, only the model classes needs to be changed.

The sequence diagram for job search is listed above because job search is one of the success-critical requirements of this project and it involves all three MVC components. Therefore, a sequence diagram for the job search function can clearly demonstrate the flow of this function.

4.   Technology-Specific System Design

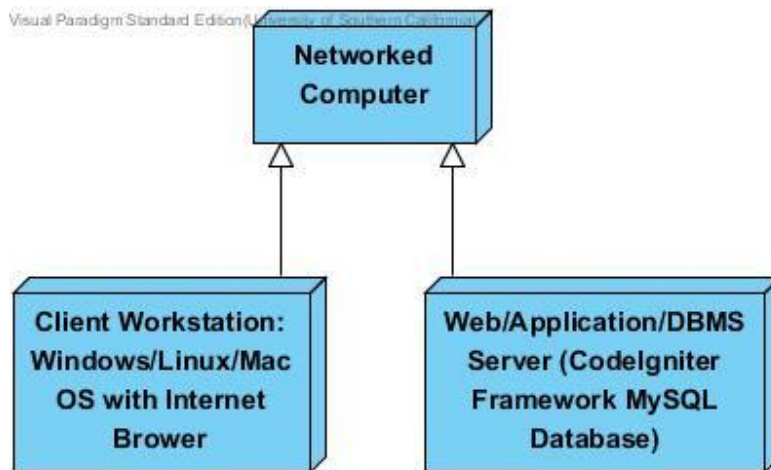## 4.1 Design Overview

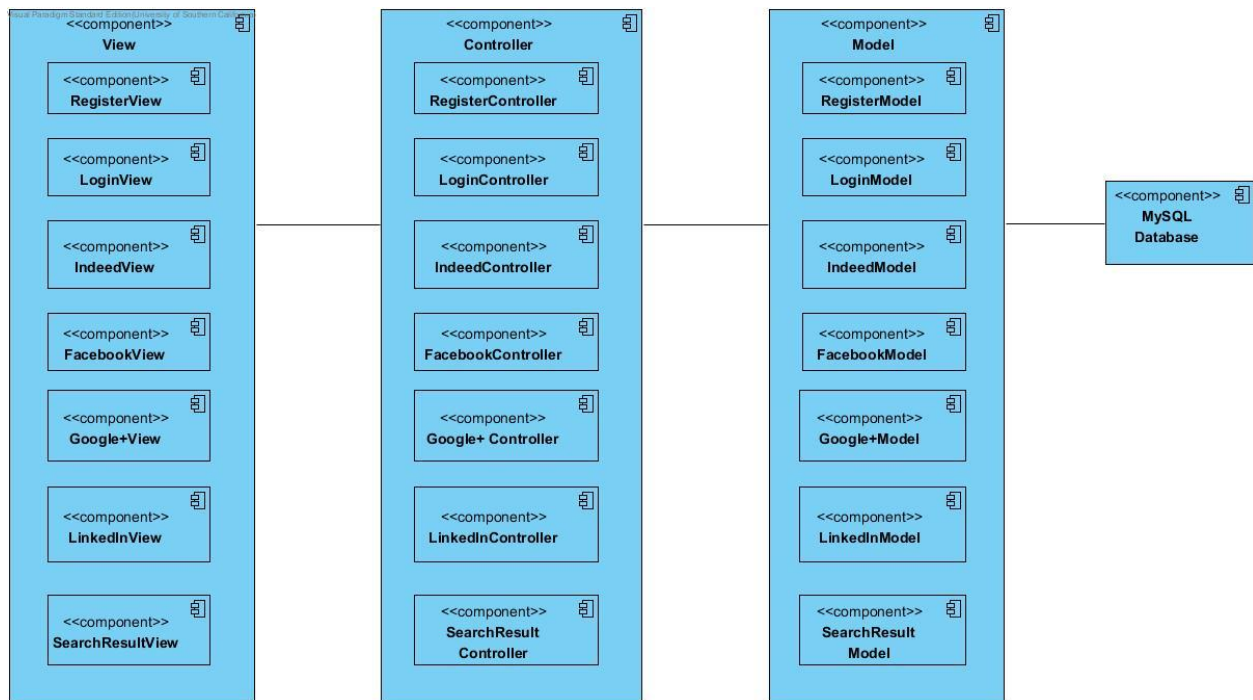### 4.1.1   System Structure

Figure 9: Hardware Component Class Diagram
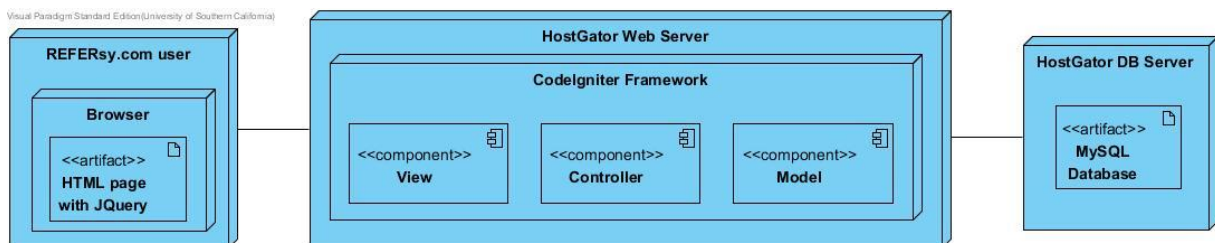
Figure 10: Software Component Class Diagram

Figure 11: Deployment Diagram

Table 35: Hardware Component Description

Hardware Component	Description
Web/Application/DBMS Server	This component accepts connections from all users. Web and Application server are provided by CodeIgniter framework. Our client may offer a server service from a hosting company (HostGator) which provides database service as well, so the Database Server is integrated with Web/Application Server.
Client workstation	This is a user computer. It must have a web browser (Mozilla Firefox) and it must have Internet network.

Table 36: Software Component Description

Software Component	Description
CodeIgniter View	This component is one block of CodeIgniter framework, which contains all pages that should shown to the users.
CodeIgniter Controller	This component is one block of CodeIgniter framework, which controls what data to fetch and what views to send to user.
CodeIgniter Model	This component is one block of CodeIgniter framework, which implements all the functionality at backend.
MySQL Database	This the database that stores REFERsy users' data.

4.1.2   Design Class

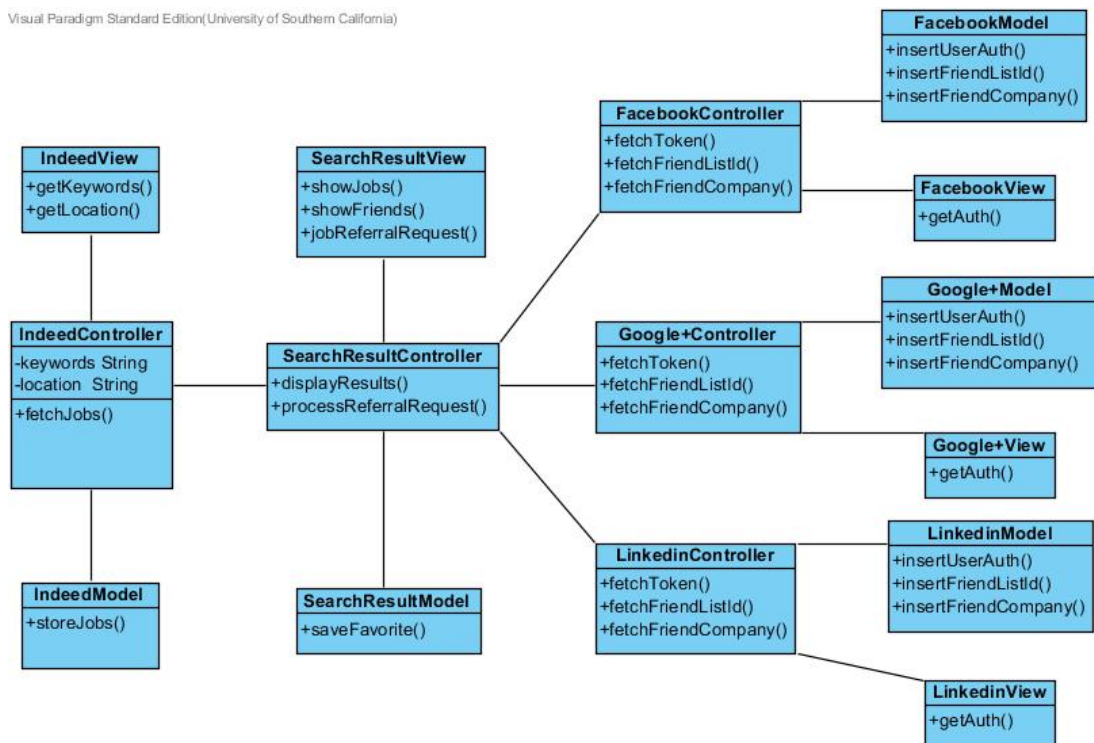Visual Paradigm Standard Edition(University of Southern California)

FacebookModel
+insertUserAuth()
+insertFriendListId()
+insertFriendCompany()

FacebookController
+fetchToken()
+fetchFriendListId()
+fetchFriendCompany()

FacebookView
+getAuth()

IndeedView
+getKeywords()
+getLocation()

SearchResultView
+showJobs()
+showFriends()
+jobReferralRequest()

Google+Model
+insertUserAuth()
+insertFriendListId()
+insertFriendCompany()

IndeedController
-keywords String
-location String
+fetchJobs()

SearchResultController
+displayResults()
+processReferralRequest()

Google+Controller
+fetchToken()
+fetchFriendListId()
+fetchFriendCompany()

Google+View
+getAuth()

LinkedinModel
+insertUserAuth()
+insertFriendListId()
+insertFriendCompany()

LinkedinController
+fetchToken()
+fetchFriendListId()
+fetchFriendCompany()

IndeedModel
+storeJobs()

SearchResultModel
+saveFavorite()

LinkedinView
+getAuth()

Figure 12: Design Class Diagram

Table 37: Design Class Description

| Class	Type	Description
FacebookController	Control	This class call Facebook API to fetch user's access token and retrieve user's friends' job information
FacebookModel	Entity	This class store user's access token and friends' job information
FacebookView	Boundary	This webpage fetch user's facebook authorization
Google+Controller	Control	This class call Google + API to fetch user's access token and retrieve user's friends' job information
Google+Model	Entity	This class store user's access token and friends' job information
Google+View	Boundary	This webpage fetch user's Google+ authorization
LinkedinController	Control	This class call Llinkedin API to fetch user's access token and retrieve user's friends' job information
LinkedinModel	Entity	This class store user's access token and friends' job information
LinkedinView	Boundary	This webpage fetch user's Linkedin authorization
IndeedController	Control	This class call Indeed API to fetch jobs given keywords and/or location
IndeedModel	Entity	This class stores the jobs results in database
IndeedView	Boundary	This webpage allow user input keywords or location for search jobs
SearchResultController	Control	This class display the search results and process referral requests
SearchResultModel	Entity	This class stores the user's favorite search keywords and/or location
SearchResultView	Boundary	This webpage shows the job search results and allow user to send the job referral request

4.1.3 Process Realization
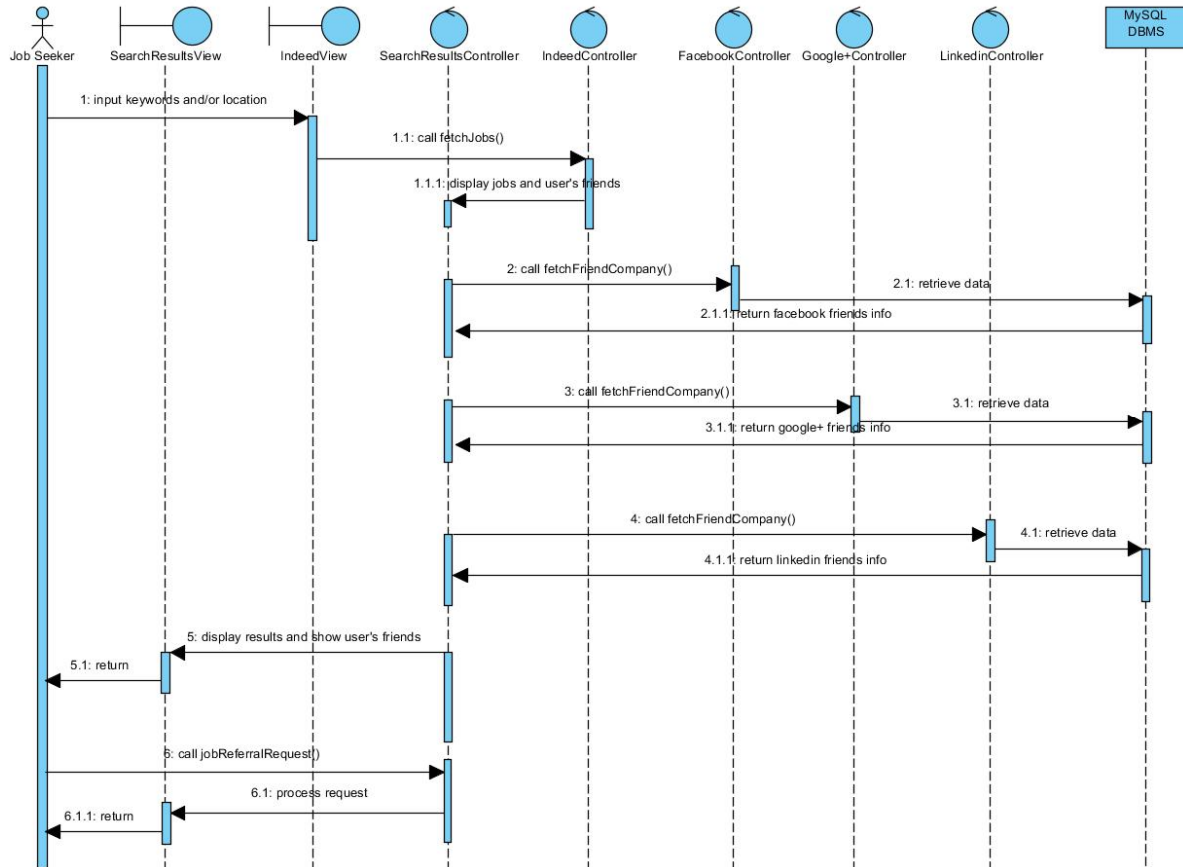
Figure 13: Process Realization Diagram

4.2 Design Rationale

In order to make job seekers search jobs and find their social network friends who can give them job referral, it must be accessible via internet for all users. Therefore, we design the REFERsy as a web application.

We choose codeIgniter as our PHP framework because of its simplicity. It's a MVC design pattern which helps us manage our code efficiently.

We choose MySQL as our database since it's the one of the most wide-used open source DBMS. In additional, it's well compatible with our serve HostGator and PHP framework.

5.   Architectural Styles, Patterns and Frameworks

Table 38: Architectural Styles, Patterns, and Frameworks

Name	Description	Benefits and Limitations
PHP	Server Side scripting language designed to build interactive web applications.	Benefits: - Easy to setup and deploy - Fast Implementation - Flexibility to scale - Free Limitations: - Team members just learnt the language
CodeIgniter	Rapid Development MVC framework using with PHP.	Benefits: - Easy to setup and use - Model, View and Controller can be programmed - Very flexible to changes - Free Limitations: - Team members new to MVC and CodeIgniter
MySQL	Open-source RDMS system.	Benefits: - Easy to setup and use - Very easy to use - Free Limitations: - Slow when handling huge chunks of data
JQuery	Client side scripting language. Makes Javascript codes very easy to code.	Benefits: - Easy to use Limitations: - Team members are new to this technology