**JobPlus: A Web-based Job Placement System**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

A Capstone Project Presented to the Faculty of

the College of Information, Computer and Communications Technology

University of San Jose- Recoletos

Cebu City, Philippines

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

In Partial Fulfillment

of the Requirements for the Degree

Bachelor of Science in Information Technology

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

by

**Baguion, Kent Michael M.**

March 2017

Table of Contents

**CHAPTER PAGE**

[CHAPTER 1 1](#_Toc477862911)

[INTRODUCTION 1](#_Toc477862912)

[Rationale of the Study 1](#_Toc477862913)

[Review of Related Studies 3](#_Toc477862914)

[Project Objective 5](#_Toc477862915)

[Scope and Limitation of the Study 5](#_Toc477862916)

[CHAPTER 2 6](#_Toc477862917)

[SOFTWARE REQUIREMENTS AND DESIGN SPECIFICATION 6](#_Toc477862918)

[Use Case Diagram 6](#_Toc477862919)

[Use Case Narrative: UC 02 View Posted Jobs 8](#_Toc477862920)

[Use Case Narrative: UC 03 View Job Applications 8](#_Toc477862921)

[Use Case Narrative: UC 04 Job Seeker Application Response 9](#_Toc477862922)

[Use Case Narrative: UC 05 Recommend Job Seekers 10](#_Toc477862923)

[Use Case Narrative: UC 06 Request Job Offer 10](#_Toc477862924)

[Use Case Narrative: UC 07 Browse Job Post 11](#_Toc477862925)

[Use Case Narrative: UC 08 Generates Filtered Jobs 12](#_Toc477862926)

[Use Case Narrative: UC 09 Job Seeker Job Application 13](#_Toc477862927)

[Use Case Narrative: UC 10 User Manages Profile 13](#_Toc477862928)

[Use Case Narrative: UC 11 User Rating 14](#_Toc477862929)

[ENTITY RELATIONSHIP DIAGRAM 15](#_Toc477862930)

[ACTIVITY DIAGRAM 16](#_Toc477862931)

[USER EXPERIENCE & INTERFACE DESIGN 20](#_Toc477862932)

[CHAPTER 3 29](#_Toc477862933)

[SOFTWARE DEVELOPMENT AND TESTING 29](#_Toc477862934)

[DEVELOPMENT SOFTWARE PLATFORMS, DEVELOPMENT ENVIRONMENT AND TOOLS 29](#_Toc477862935)

[DEVELOPMENT AND TESTING PROCESS 30](#_Toc477862936)

[DEVELOPMENT PROCESS 30](#_Toc477862937)

[Job Recommendation 31](#_Toc477862938)

[Relevancy Ranking 32](#_Toc477862939)

[Location Scoring 32](#_Toc477862940)

[Work History Scoring 33](#_Toc477862941)

[Skills Scoring 34](#_Toc477862942)

[Job Feeds 35](#_Toc477862943)

[Recommended Jobs 37](#_Toc477862944)

[Nearby Job 38](#_Toc477862945)

[Conflict Checking 39](#_Toc477862946)

[TESTING PROCESS AND CASES 41](#_Toc477862947)

[TEST CASE 1: EMPLOYER 41](#_Toc477862948)

[TEST CASE 2: JOB SEEKER 42](#_Toc477862949)

[CHAPTER 4 44](#_Toc477862950)

[SUMMARY, CONCLUSION AND RECOMMENDATIONS 44](#_Toc477862951)

[SUMMARY OF FINDINGS 44](#_Toc477862952)

[CONCLUSION 44](#_Toc477862953)

[RECOMMENDATIONS 45](#_Toc477862954)

[FUTURE WORKS 45](#_Toc477862955)

[BIBLIOGRAPHY 46](#_Toc477862956)

[CURRICULUM VITAE 47](#_Toc477862957)

**TABLE OF FIGURES**

**USE CASE DIAGRAM**

**Figure 2.1**: Use Case Diagram for JobPlus 6

**Figure 2.2:** Use Case Narrative – Create job Posts 7

**Figure 2.3:** Use Case Narrative – View Posted Jobs 8

**Figure 2.4:** Use Case Narrative – View Job Applications 9

**Figure 2.5:** Use Case Narrative – Job Seeker Application Response 9

**Figure 2.6:** Use Case Narrative – Recommend Job Seekers 10

**Figure 2.7:** Use Case Narrative – Request Job Offer 11

**Figure 2.8:** Use Case Narrative – Browse Job Post 12

**Figure 2.9:** Use Case Narrative – Generates Filtered Jobs 12

**Figure 2.10:** Use Case Narrative – Job Seeker Job Application 13

**Figure 2.11:** Use Case Narrative – User Manages Profile 14

**Figure 2.12:** Use Case Narrative – User Rating 15

**Figure 2.13:** Entity Relational Diagram 15

**Figure 2.14:** User Login 16

**Figure 2.15:** Profile Setup 16

**Figure 2.16:** Employer Job Posting 17

**Figure 2.17:** Employer Application Response 18

**Figure 2.18:** Job Seeker job searching and application 18

**Figure 2.19:** Job start to end 19

**Figure 2.20:** Profile Setup 20

**Figure 2.21**: Profile Setup Step 2 20

**Figure 2.22:** Employer Job Post Step 1 21

**Figure 2.23:** Profile Setup Step 3 21

**Figure 2.24:** Employer Job Post Step 2 22

**Figure 2.25 :** Employer Job Post Step 3 22

**Figure 2.26** : Employer Posted Jobs 23

**Figure 2.27:** Employer's Application Response 23

**Figure 2.28:** Applicant Currently Working 24

**Figure 2.29:** Today's Worker 24

**Figure 2.30**:Job Page 25

**Figure 2.31:** End Job 25

**Figure 2.32:** View Job 26

**Figure 2.33:** Terms and Agreement 26

**Figure 2.34:** Upcoming Job 27

**Figure 2.35:** Active Job 27

**Figure 2.36:** Rating of Employer 28

**Figure 2.37:** Payment Received 28

**Figure 3.1:** Job Address Coordinates 30

**Figure 3.2:** Getting the profile address and Retrieval of Address Coordinates 31

**Figure 3.3:** Retrieval of Distance and Time 31

**Figure 3.4**: Location Points 32

**Figure 3.5:** Fetching Work History 33

**Figure 3.6** History Points 34

**Figure 3.7:** Skills Scoring 35

**Figure 3.8:** Job Feed Criteria 36

Figure 3.9 Recommended Criteria 37

**Figure 3.10** Nearby Job Criteria 38

**Figure 3.11:** Retrieval of Distance and time 39

**Figure 3.12: Conflict checking** 39

# 

**INTRODUCTION**

Rationale of the Study

With the ever-rising demand for the cost of living in our modern civilization, work-able individuals are always seeking opportunities for not just making a living to meet basic needs but for growth and progress that exceeds beyond daily survival [1]. In the current economic climate, where there are more candidates than a corporate job opening, it is important to stand out from the others and to do so, increasing need for profound skills, academic achievements and attainments are required on a candidate’s application. As a result, this leaves unskilled and lower academic laborers at a loss of working opportunities.

According to the Philippine Statistics Authority (PSA, 2016), the sector of the country which comprises 56.3% of all businesses [2] employs an outstanding 2,118,391 laborers and unskilled workers as of 2016 [3] which is increasing in number every single year but sadly these aspirants are not given the platform for employment and to search for jobs that match their skills and abilities.

(Saluja, et al., 2013) Typical employment scheme is thru manual systems in the form of registers and job listings which are available thru physical and remove agency or recruitment offices. Furthermore, mainstream solutions on the worldwide web are available but are highly prioritizing skilled and degree holder laborers. (Ranjan, 2013) Job Portal Systems are the avenue for recruiters to prune less or unskilled applicants and find the most suitable worker.

The need for automation and ubiquitous information of job opportunities must be available for job seekers and employers beyond the confines of an agency. The opening of menial or service-oriented jobs (e.g., house-care, carpentry, etc.) must be made available to anybody not limited to agencies and other entities. Moreover, developing a job portal mainly focused on unskilled laborer capabilities that enable recommendation of qualified working needs.

This paper aims to develop **JobPlus: A web-based job-placement system**. The platform will create a general process of a job posting, job application, salary disbursement, and advanced working experience portfolio. Also, to enhance productivity, the application will provide a task scheduler with date and time conflict alert. Lastly, a personalized job recommendation will be provided based on the job working history, skills and nearby location. To which all these are focused to unskilled laborers searching for working opportunities.

Review of Related Studies

There have been many Disruptive Technology applications which are emerging and multiple surveys conducted in the Philippines with regards to job portal system. One of which is for the employers to contact an agency that provides menial tasks and services. Looking for someone who we can trust and feel secure working beside us is a very challenging because it’s hard to determine the attribute or values of that person by just their appearance. The fact that our society forces alleged “menial” jobs on people, under the threat of starvation if they don’t perform the jobs, says something about the way we live. Although it’s widely believed that some jobs are so menial that only illegal aliens will take them, 75% of unemployed workers said they would apply for jobs paying the minimum wage. [4]

One good example of job recruitment application is JobStreet, JobStreet is an online job portal founded in Malaysia that provides an online recruitment service. Where applicants may build and post their resume online. Employers may also post their job postings. JobStreet mostly caters professional individuals who are looking for a full-time or part-time job. What made this application different from Jobstreet is that JobPlus only cater menial jobs for non-professionals. Applicants in JobPlus doesn’t need to post or upload their resume because their respective profiles already serve as a resume.

Freelancer is an online job portal solely for freelance professionals. Freelancer highly encourages their users to post their projects and portfolios to boost further their chance of being hired. Furthermore, What made JobPlus different from Freelancer is that JobPlus users doesn’t need to post their portfolios and projects because the job postings are mostly menial tasks. A freelancer may still work even at home, but JobPlus users certainly need to be at the workplace physically to start their respective jobs.

Mynimo is an online job bulletin that connects employers and job seekers.  
Job seekers may post their respective resumes and wait for an employer to hire them. Employers post their job postings and wait for respective applicants to apply. What made JobPlus different from Mynimo is that JobPlus users may take the role of an employer or a job seeker while in Mynimo, user roles are different. JobPlus users need to post their resume because their skills are already used to highlight their profile. JobPlus user profiles serve as their resumes.

### Project Objective

This study aims to build a web-based job portal system that mainly focuses on employing unskilled workers and being a platform for employers to communicate with hired employees.

Specifically, the paper aims to:

1. Integrate SMS notification.
2. Recommends a job based on an applicant’s skills, work history, and current location.
3. Develop a task scheduler that enhances the working productivity and attendance.
4. Create a general system of a job posting, recruitment, and employment of unskilled workers.

### Scope and Limitation of the Study

The paper aims to create a job portal mainly grounding on working opportunities for unskilled workers that are highly relevant in the Philippine and other service-based countries.

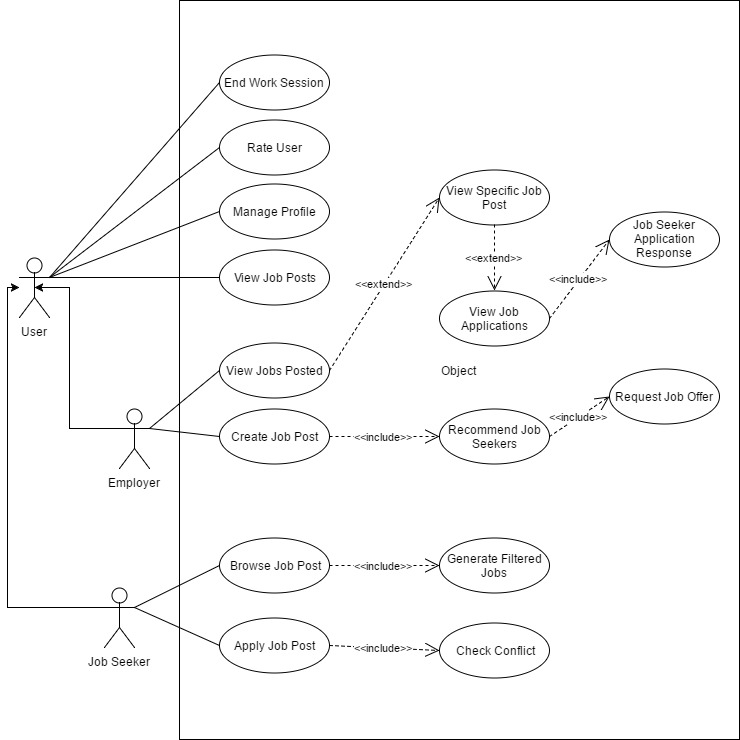
Ownership of a mobile device that has SMS capability is required for the job application so as to enable JobPlus to notify and validate the existing user.

# 

SOFTWARE REQUIREMENTS AND DESIGN SPECIFICATION

This chapter shows what the system requires and functions are needed for the Job Portal to work. It includes Use Case Diagram, Use Case Narrative, Activity Diagram, Class Diagram and the system’s User Interface Design.

**Use Case Diagram**



**Figure 2.1**: Use Case Diagram for JobPlus

Figure

Use Case Diagram for JobPlus

Figure 2.1 represents the use case diagram of the study. The employer creates a job post; the system will then generate a list of recommended applicants that suits the skills of the job posted. The employer can decide to whether request a job offer based on the generated list or wait for a job seeker to apply for that precise job post. On the other hand, the job seeker can browse for a job or choose on the generated list of jobs based on the job seeker’s skills, location and work history.

Use Case Narrative

Use Case Narrative: UC 01 Create Job Posts

|  |  |  |
| --- | --- | --- |
| **Use Case** | Create Job Posts | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to create a job post. | |
| **Overview:** | Job Posts are created by the employer which the jobs seekers will apply. | |
| **Type:** | Essential | |
| **Precondition:** | The Employer has to be logged in. | |
| **Post condition:** | The Job post information will be stored in the database. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks the Post Job button | | 1. Shows a modal containing the job information |
| 1. User selects the job type | |  |
| 1. User selects the Job category | | 1. Shows the skills matching the selected job category |
| 1. User selects the skills required for the job post. | |  |
| 1. User fills in the job title | |  |
| 1. User fills in the job description | |  |
| 1. User pinpoints the job location | |  |
| 1. User picks the start date and time | |  |
| 1. User picks the end date and time | |  |
| 1. User click the Next Step button | | 1. Shows the step two of the form which is the payment information |
| 1. User selects the number of applicants needed for the job | |  |
| 1. User fills in the job salary | |  |
| 1. User selects the pay type of the job | |  |
| 1. User clicks the Next Step button | | 1. Shows the step three of the form which is the job summary |
| 1. User clicks the Finish button | | 1. Stores the job information in the database. |

**Figure 2.2:** Use Case Narrative – Create job Posts

### Use Case Narrative: UC 02 View Posted Jobs

|  |  |  |
| --- | --- | --- |
| **Use Case** | View Posted Jobs | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to view all jobs posted. | |
| **Overview:** | Employer views all the jobs posted. | |
| **Type:** | Essential | |
| **Precondition:** | The employer has to be logged in. | |
| **Post condition:** | All jobs posted by the employer will be displayed on a page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Job Postings tab in the upper right navigation. | | 1. The system retrieves all the job posts created by the current employer. |
|  | | 1. The system displays the retrieved data. |
| 1. User clicks the View Job button | | 1. The system displays a modal containing the job information. |

**Figure 2.3:** Use Case Narrative – View Posted Jobs

### Use Case Narrative: UC 03 View Job Applications

|  |  |  |
| --- | --- | --- |
| **Use Case** | View Job Applications | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to browse the applications of the jobs posted. | |
| **Overview:** | The employer views all the applications. | |
| **Type:** | Essential | |
| **Precondition:** | The employer has to be logged in. | |
| **Post condition:** | All job applications will be displayed on a page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Applications tab in the upper right navigation. | | 2. Retrieves all the applications of the jobs the employer posted. |
|  | | 3. Displays the retrieved data. |

**Figure 2.4:** Use Case Narrative – View Job Applications

### Use Case Narrative: UC 04 Job Seeker Application Response

|  |  |  |
| --- | --- | --- |
| **Use Case** | Job Seeker Application Response | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to respond to the job seeker’s application. | |
| **Overview:** | Employer responds to the job seeker’s application. | |
| **Type:** | Essential | |
| **Precondition:** | The employer must have posted a job. | |
| **Post condition:** | Job application status changes. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Accept button | | 2. Changes the application status to 1. |
| **Alternative Scenario/s** | | |
| 3. User clicks Decline button | | 4.Deletes the applicant’s application. |

**Figure 2.5:** Use Case Narrative – Job Seeker Application Response

### Use Case Narrative: UC 05 Recommend Job Seekers

|  |  |  |
| --- | --- | --- |
| **Use Case** | Suggest Job Seekers | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to view all suitable job seekers for the job posting. | |
| **Overview:** | Employer views all the recommended job seekers for the job post. | |
| **Type:** | Essential | |
| **Precondition:** | The employer must have posted a job. | |
| **Post condition:** | A list of recommended job seekers will be displayed on a modal. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
|  | | 1. Retrieves all the registered applicants. |
|  | | 2. Sends a request to the Geocoding API to retrieve the coordinates of the user. |
|  | | 3. Retrieves the result of the request made to the Geocoding API |
|  | | 4. Sends a request to the Google Distance Matrix API, to retrieve the duration and distance between the job and job seeker. |
|  | | 5.Get the location points |
|  | | 6.Get the skill points |
|  | | 7.Get the work history points |
|  | | 8.Get the point average |
|  | | 9. Retrieve and order the results by the total average. |

**Figure 2.6:** Use Case Narrative – Recommend Job Seekers

### Use Case Narrative: UC 06 Request Job Offer

|  |  |  |
| --- | --- | --- |
| **Use Case** | Request Job Offer | |
| **Actors:** | Employer | |
| **Purpose:** | Allows the employer to request a job offer to the job seeker. | |
| **Overview:** | Employer offers a job to the job seeker. | |
| **Type:** | Essential | |
| **Precondition:** | The employer must have posted a job. | |
| **Post condition:** | A job offer request is sent to the job seeker. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Hire Applicant button | | 2. Stores the job offer in the database. |

**Figure 2.7:** Use Case Narrative – Request Job Offer

### Use Case Narrative: UC 07 Browse Job Post

|  |  |  |
| --- | --- | --- |
| **Use Case** | Browse Job Post | |
| **Actors:** | Job Seeker | |
| **Purpose:** | Allows the job seeker to browse for a job posting. | |
| **Overview:** | Job seeker browses for a job posting. | |
| **Type:** | Essential | |
| **Precondition:** | Job seeker must be logged in. | |
| **Post condition:** | A list of job postings is displayed in a page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User selects the job category | |  |
| 2. User selects the type of jobs | |  |
| 3. User fills in the location of the job | |  |
| 4.User clicks the Search button | | 5. System retrieves all the job postings. |
|  | | 6. System retrieves all the scheduled works of the job seeker. |
|  | | 7. System filters the job result that conflicts with the job seeker’s scheduled jobs. |
|  | | 8.System filters the job result by matching the job skills and job seeker’s skills. |
|  | | 9.Finally, the system filters the results by eliminating the jobs that does not meet the job seeker’s specified location. |
|  | | 10.Displays the job results to the job page |
| **Alternative Scenario/s** | | |
| User selects the job category | |  |
| User selects the type of jobs | |  |
| User clicks the Search button | | System retrieves all the job postings |
|  | | System retrieves all the scheduled works of the job seeker. |
|  | | System filters the job result that conflicts with the job seeker’s scheduled jobs. |
|  | | System filters the job result by matching the job skills and job seeker’s skills. |
| User selects the type of jobs | |  |
| User fills in the location of the job | |  |
| User clicks the Search button | | Displays the job results to the job page |
|  | | System retrieves all the scheduled works of the job seeker. |
|  | | System filters the job result that conflicts with the job seeker’s scheduled jobs. |
|  | | System filters the job result by matching the job skills and job seeker’s skills. |
|  | | Displays the job results to the job page |
| User fills in the location of the job | |  |
| User clicks the Search button | | System retrieves all the job postings |
|  | | System filters the results by eliminating the jobs that does not meet the job seeker’s specified location. |
|  | | Displays the job results to the job page. |

**Figure 2.8:** Use Case Narrative – Browse Job Post

### Use Case Narrative: UC 08 Generate Recommended Jobs

|  |  |  |
| --- | --- | --- |
| **Use Case** | Generate Recommended Jobs | |
| **Actors:** | System | |
| **Purpose:** | Generate job postings for job seekers. | |
| **Overview:** | Generate job postings based from the three major factors which are: location, work history and skills. | |
| **Type:** | Essential | |
| **Precondition:** | The user has added job postings in the database. | |
| **Post condition:** | Filtered job postings are displayed in the page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
|  | | 1. The system retrieves all job postings from the database. |
|  | | 2. The system sends a request to the Google Distance Matrix API to calculate the distance between the job address and user’s address. |
|  | | 3. The system stores the result in the database. |
|  | | 4. The system sets the result to the different job categories. |
|  | | 5. The system retrieves the job posting results. |

**Figure 2.9:** Use Case Narrative – Generates Filtered Jobs

### Use Case Narrative: UC 09 Job Seeker Job Application

|  |  |  |
| --- | --- | --- |
| **Use Case** | Job Seeker Job Application | |
| **Actors:** | Job Seeker | |
| **Purpose:** | Job seeker sends an application request to the employer. | |
| **Overview:** | An application request is sent to the employer. | |
| **Type:** | Essential | |
| **Precondition:** | The user has viewed the job information. | |
| **Post condition:** | The user receives a prompt. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Apply Job | | 2. The system will display a terms and agreements modal. |
| 3. User confirms the terms and agreement. | | 4. The system stores the job application in the database. |

**Figure 2.10:** Use Case Narrative – Job Seeker Job Application

### Use Case Narrative: UC 10 User Manages Profile

|  |  |  |
| --- | --- | --- |
| **Use Case** | User Manages Profile | |
| **Actors:** | User | |
| **Purpose:** | User manages profile. | |
| **Overview:** | The user manages the profile. | |
| **Type:** | Essential | |
| **Precondition:** | The user must be on the profile page. | |
| **Post condition:** | Profile information is updated. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks Profile in the upper right corner of the navigation | | 2. The system retrieves the data of the current user |
| 3. User supplies an info and clicks update button | | 4. The system will override the old data supplied in the database from the newly submitted one. |

**Figure 2.11:** Use Case Narrative – User Manages Profile

### Use Case Narrative: UC 11 User Rating

|  |  |  |
| --- | --- | --- |
| **Use Case** | User Rating | |
| **Actors:** | User | |
| **Purpose:** | User rates another user | |
| **Overview:** | The user rates other users. | |
| **Type:** | Essential | |
| **Precondition:** | The user must have started the work session | |
| **Post condition:** | A rate and feedback is sent. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
| 1. User clicks the end work session | | 2. The system prompts the user to confirm end work session |
| 3. User confirms the end session | | 4. The system will display a work summary and a rating form. |
| 5. User fills in rating form | | 6. The system will store the rating in the database. |

**Figure 2.12:** Use Case Narrative – User Rating

### Use Case Narrative: UC 08 Generate Job Feeds

|  |  |  |
| --- | --- | --- |
| **Use Case** | Generate Job Feeds | |
| **Actors:** | System | |
| **Purpose:** | Generate job postings for job seekers. | |
| **Overview:** | Generate job postings based from the three major factors which are: location, work history and skills. | |
| **Type:** | Essential | |
| **Precondition:** | The user has added job postings in the database. | |
| **Post condition:** | Filtered job postings are displayed in the page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
|  | | 1. The system retrieves all job postings from the database. |
|  | | 2. The system sends a request to the Google Distance Matrix API to calculate the distance between the job address and user’s address. |
|  | | 3. The system stores the result in the database. |
|  | | 4. The system sets the result to the different job categories. |
|  | | 5. The system retrieves the job posting results. |

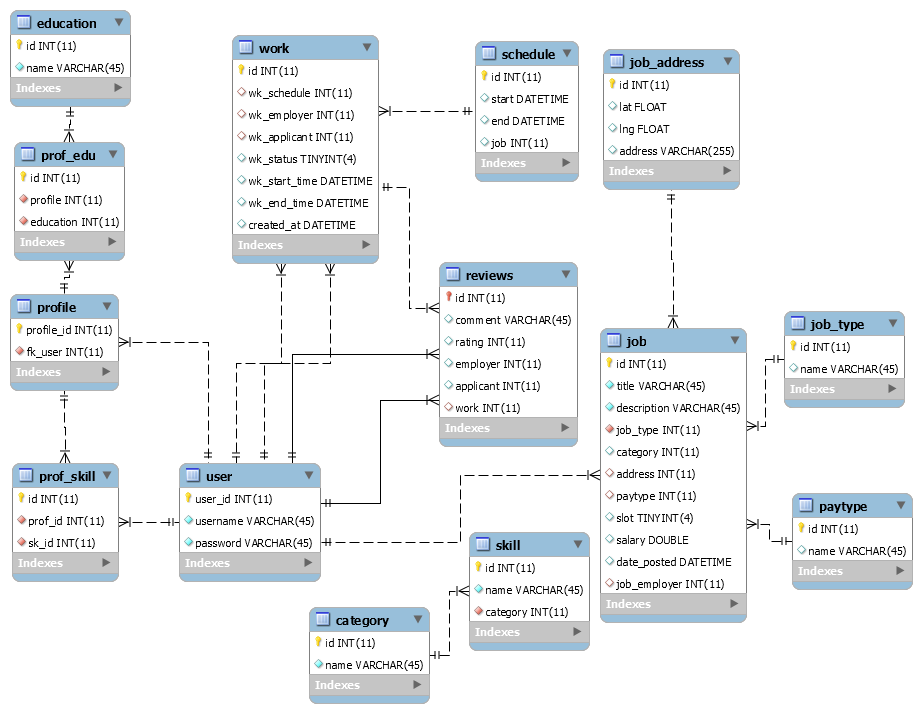
**Figure 2.13:** Use Case Narrative – Generates Filtered Jobs

### Use Case Narrative: UC 08 Generate Nearby Jobs

|  |  |  |
| --- | --- | --- |
| **Use Case** | Generate Nearby Feeds | |
| **Actors:** | System | |
| **Purpose:** | Generate job postings for job seekers. | |
| **Overview:** | Generate job postings based from the three major factors which are: location, work history and skills. | |
| **Type:** | Essential | |
| **Precondition:** | The user has added job postings in the database. | |
| **Post condition:** | Filtered job postings are displayed in the page. | |
| **Flow of Events** | | |
| *Actor Action* | | *System Response* |
|  | | 1. The system retrieves all job postings from the database. |
|  | | 2. The system sends a request to the Google Distance Matrix API to calculate the distance between the job address and user’s address. |
|  | | 3. The system stores the result in the database. |
|  | | 4. The system sets the result to the different job categories. |
|  | | 5. The system retrieves the job posting results. |

**Figure 2.14:** Use Case Narrative – Generates Filtered Jobs

ENTITY RELATIONSHIP DIAGRAM



**Figure 2.15:** Entity Relational Diagram

**CLASS DIAGRAM**



**Figure 2.16:** Class Diagram

ACTIVITY DIAGRAM

C:\Users\iCodes\AppData\Local\Microsoft\Windows\INetCache\Content.Word\AD-login.png

**Figure 2.17:** User Login

Figure 2.14 illustrates the Activity Diagram for User’s authentication. Once the user is already authenticated, the user will then be redirected into the homepage.



**Figure 2.18:** Profile Setup

Figure 2.15 illustrates the Activity Diagram for profile setup. The user fills in basic information and skills, after which the system will send a verification code to the mobile number the user supplied. The user has to enter the verification code received to finish the profile setup.



**Figure 2.19:** Employer Job Posting

Figure 2.16 illustrates the Activity Diagram for job posting. The user fills in basic job information and skills required to finish the task. The system will then give the user a summary of the job he wants to post.



**Figure 2.20:** Employer Application Response

Figure 2.17 illustrates the Activity Diagram for employer’s application response. The employer has choices whether to check the applicants profile or to directly proceed to hiring that applicant.



**Figure 2.21:** Job Seeker job searching and application

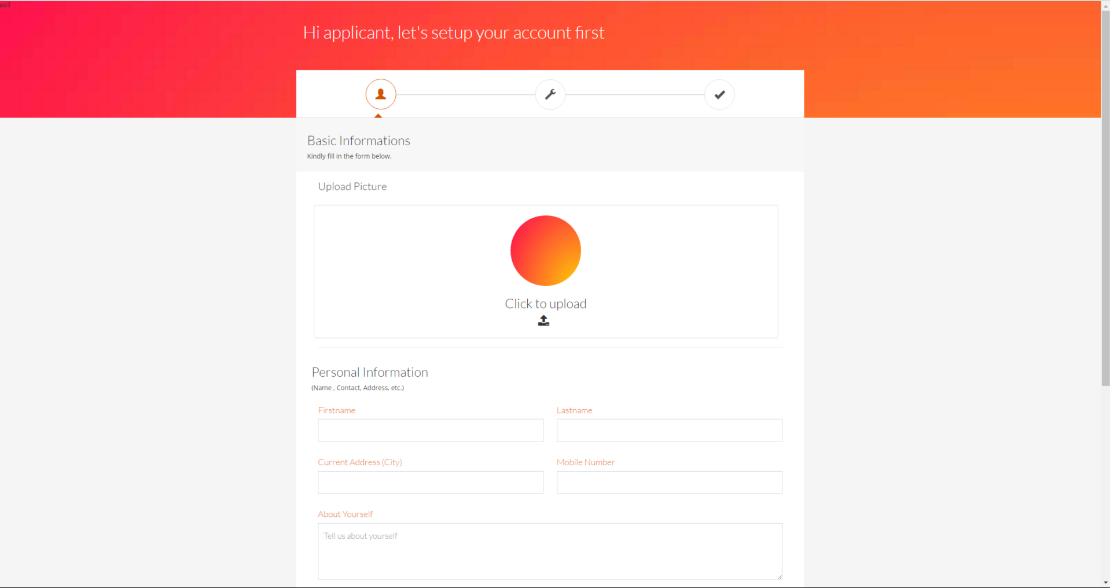
Figure 2.18 illustrates the Activity Diagram for job seeker’s job browsing and application. The job seeker browse for a job then views the job by clicked on a list of job items. Upon application, the job seeker will be shown a terms and agreements that the job seeker has to follow. After which, the user confirms by clicking the confirm button then the application will be sent right away.



**Figure 2.22:** Job start to end

Figure 2.19 illustrates the Activity Diagram for both the job seeker and employer’s transition on start up to the ending of the job. The user has to start the job and after the service have been rendered, by simply clicking the end job button, they user will be given the work summary and rating form to rate.

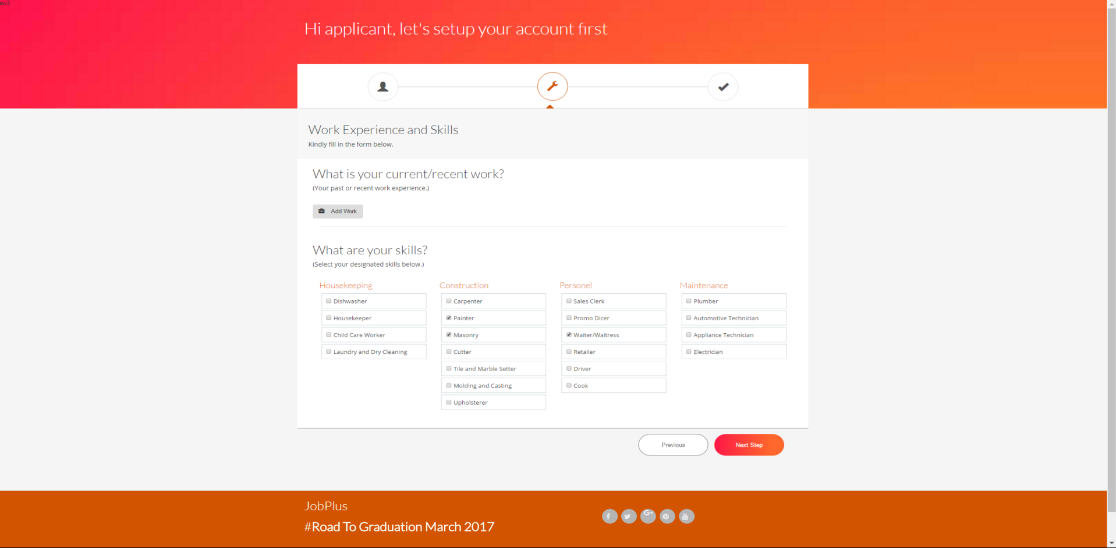
USER EXPERIENCE & INTERFACE DESIGN



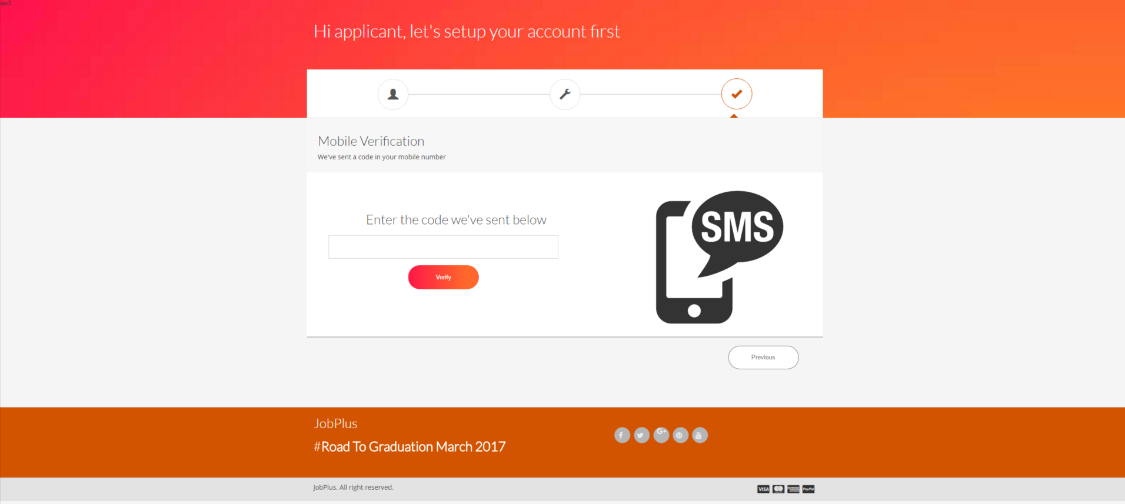
**Figure 2.23:** Profile Setup

This is where the user will be redirected after they have successfully registered. The user has to supply basic information to further enhance the content of the profile.

**Figure 2.24**: Profile Setup Step 2

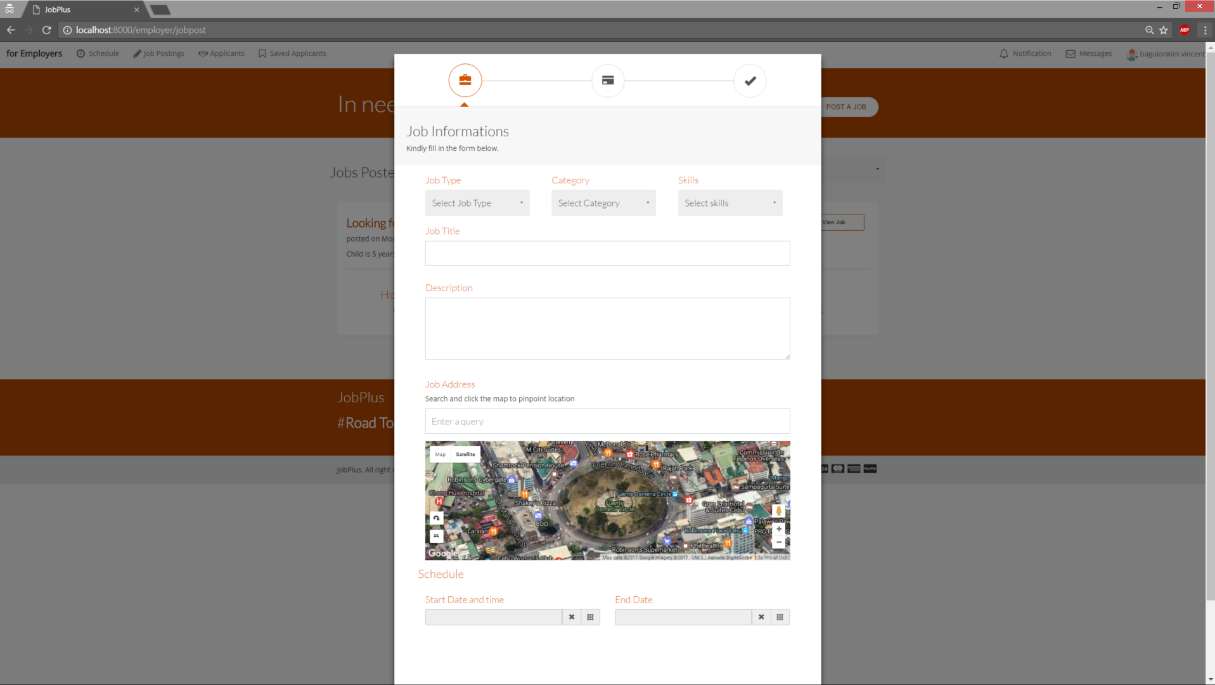


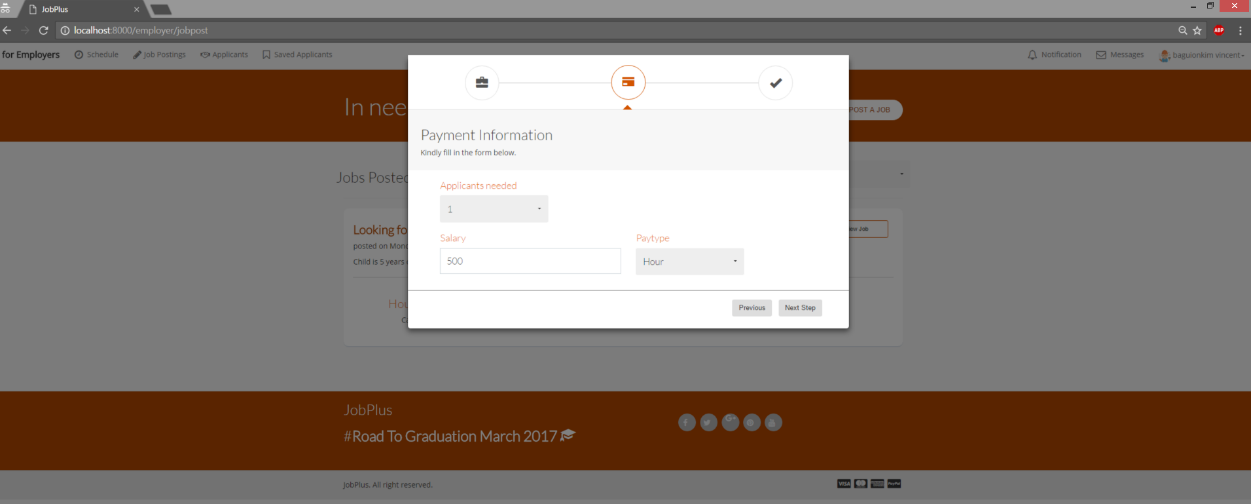
This is the step where various skills are listed and will be selected depending on the user’s skills.

This is the final step before the profile is created, the system will send a unique verification code to the mobile number supplied by the user which will then be used to verify the user.

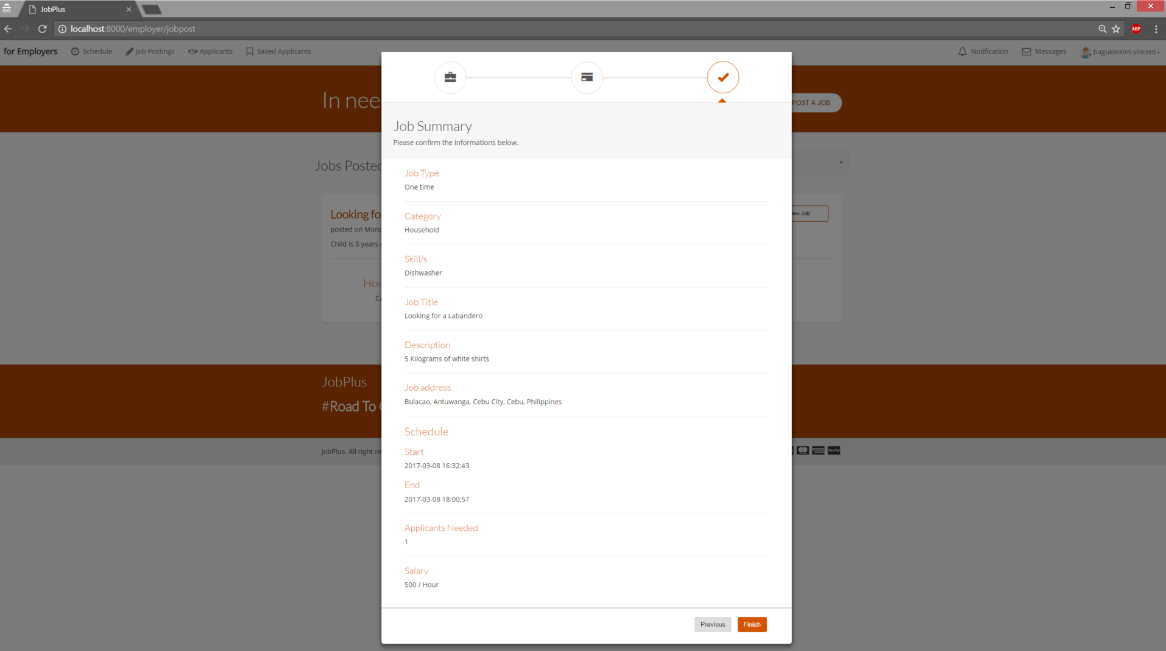
**Figure 2.25:** Employer Job Post Step 1

**Figure 2.26:** Profile Setup Step 3



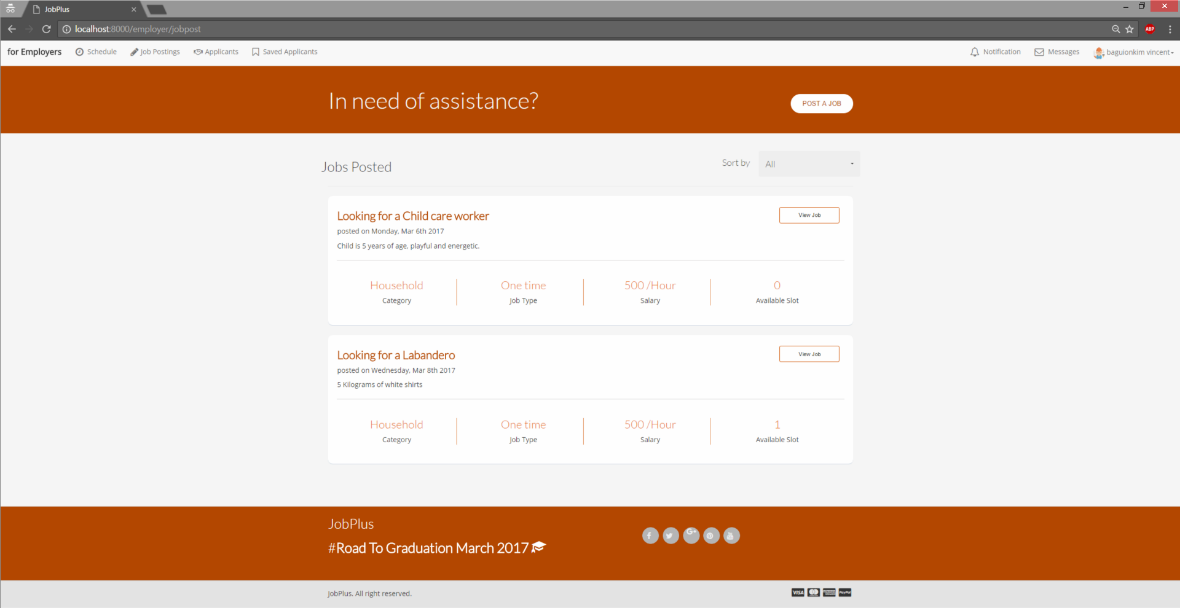
This is the modal page that the user needs to fill in to post a job, this is where the job types, categories and skills are selected. There is also a map to exactly pinpoint the jobs location for better user experience.

**Figure 2.27:** Employer Job Post Step 2

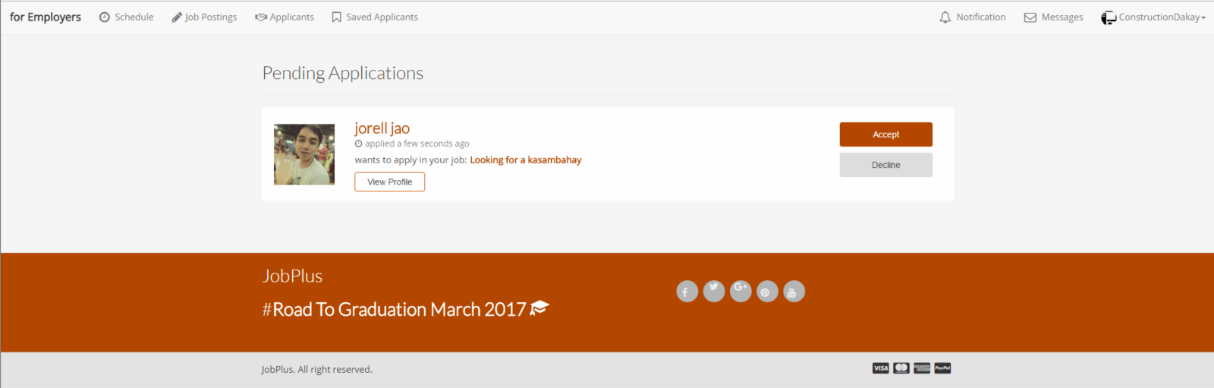
This is where the job salary, pay type and the number of applicants are supplied.

**Figure 2.28 :** Employer Job Post Step 3

This is where the summary of the job information the employer supplied that ensures all the fields are correctly filled.

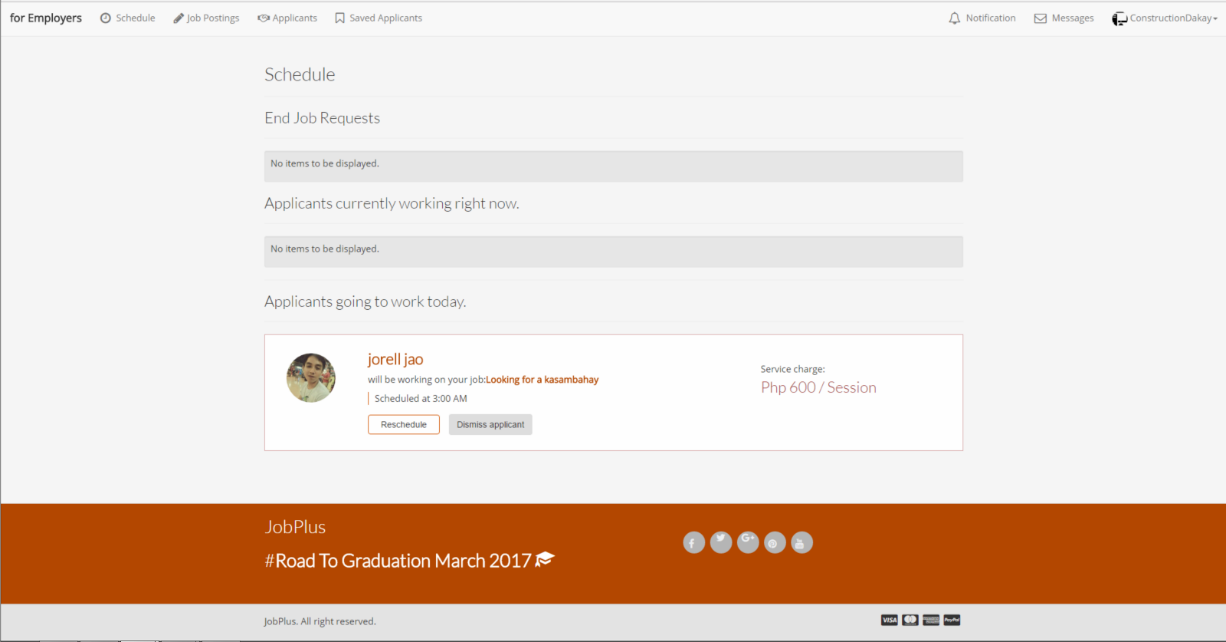
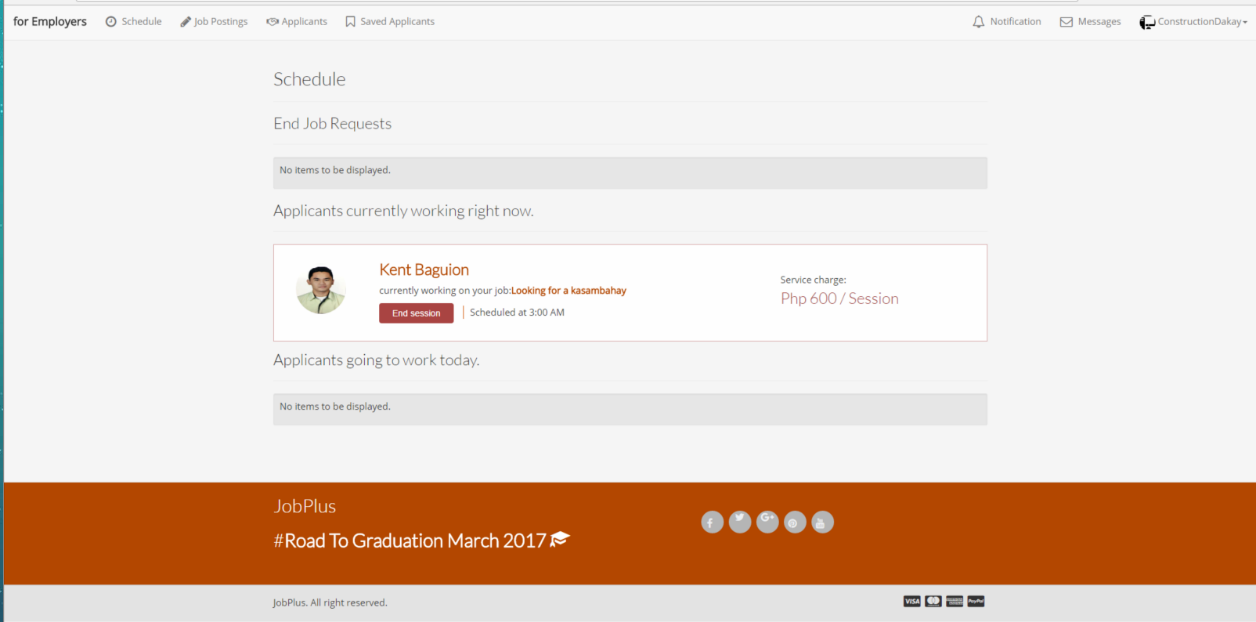


**Figure 2.29** : Employer Posted Jobs

This page shows the list of the job posted by the employer and where the employer views and reschedule a job.

**Figure 2.30:** Employer's Application Response

This page shows the list of the applicants that applied for the jobs posted by the employer. This is where the employer accepts or decline the application requests.

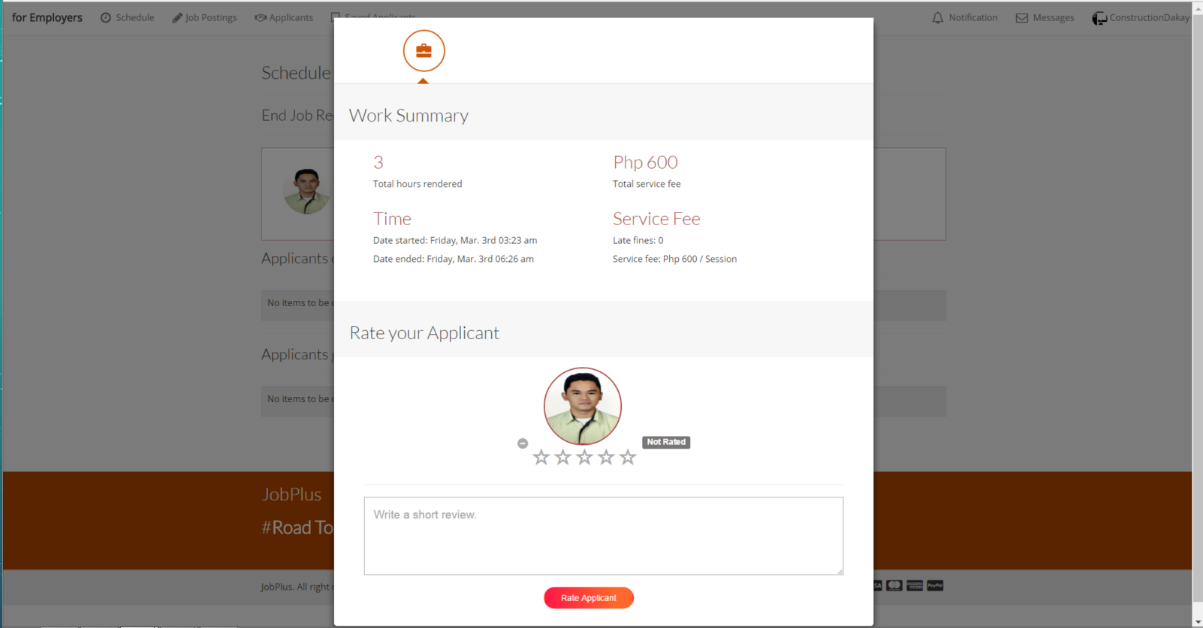
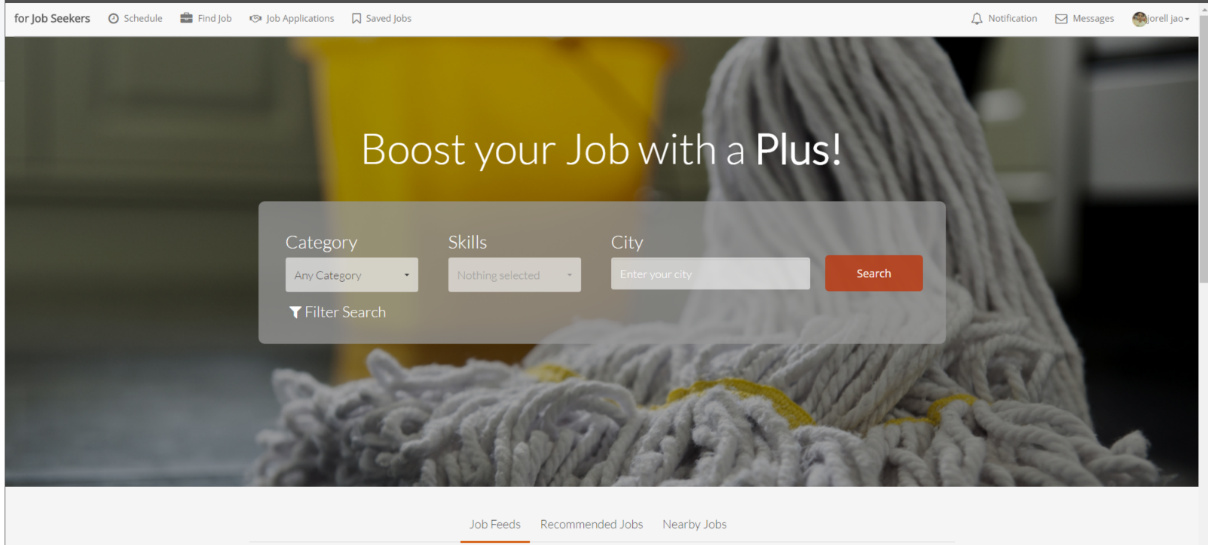


**Figure 2.31:** Applicant Currently Working

**Figure 2.32:** Today's Worker

Figure 2.29 shows the list of the applicants who are scheduled to work on that day.

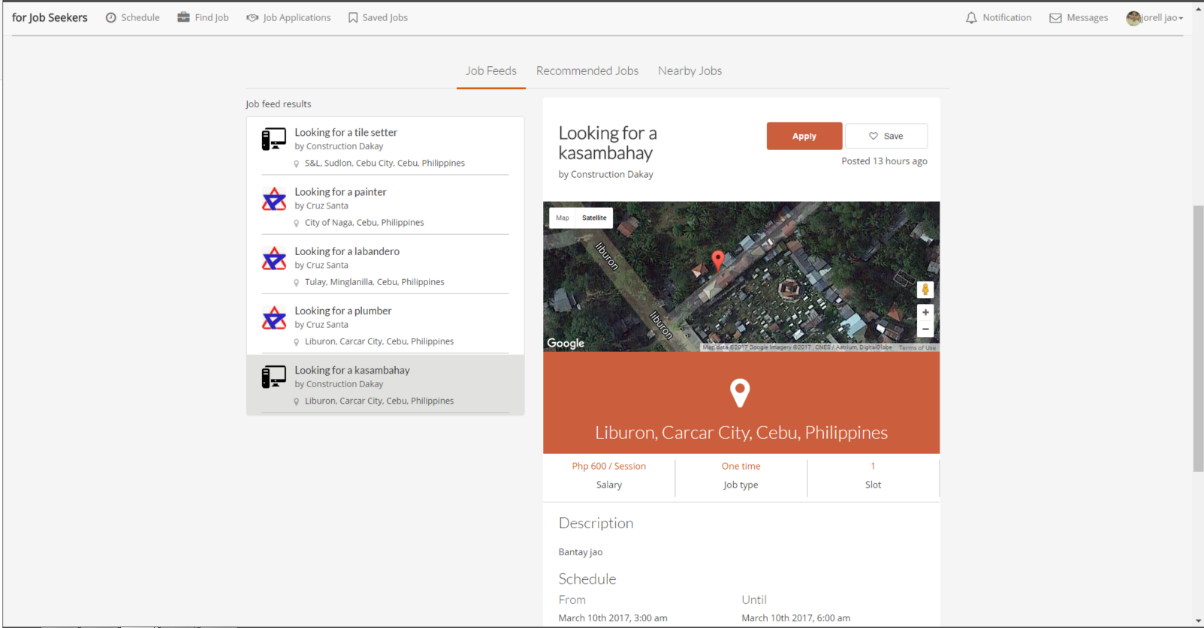
This page shows the list of the applicants that is currently working right now.

This modal generates the summary of the work rendered by the applicant and the venue where the employer rates the applicant.

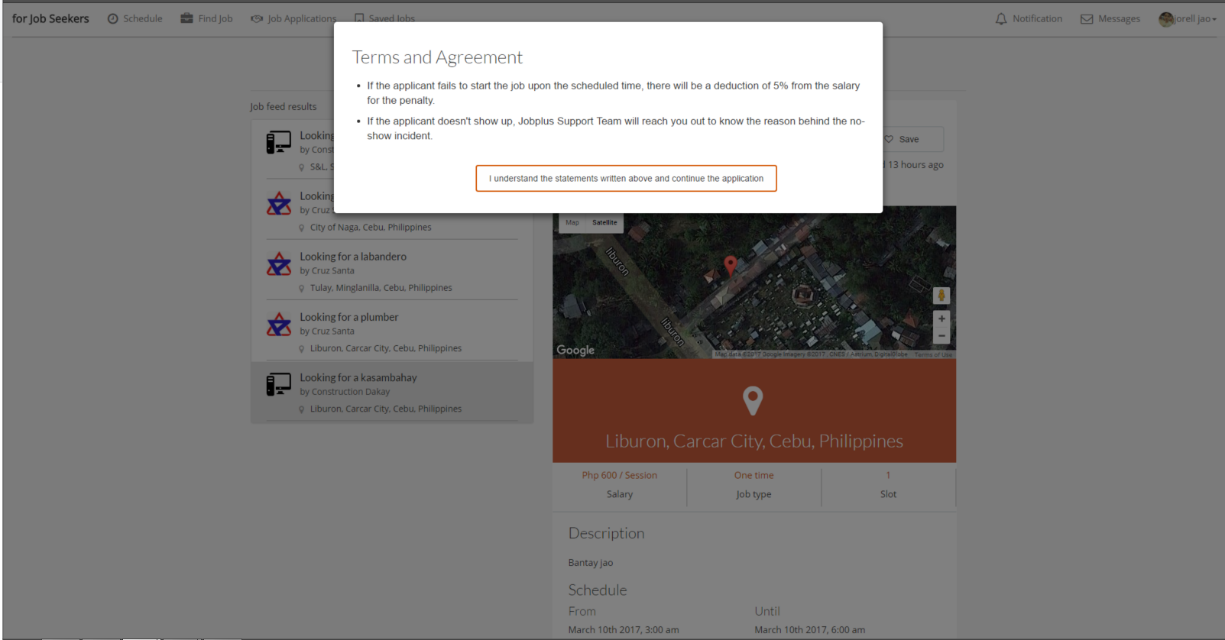
**Figure 2.33**:Job Page

**Figure 2.34:** End Job

This page is the venue for the job seekers to browse and apply for a certain job.

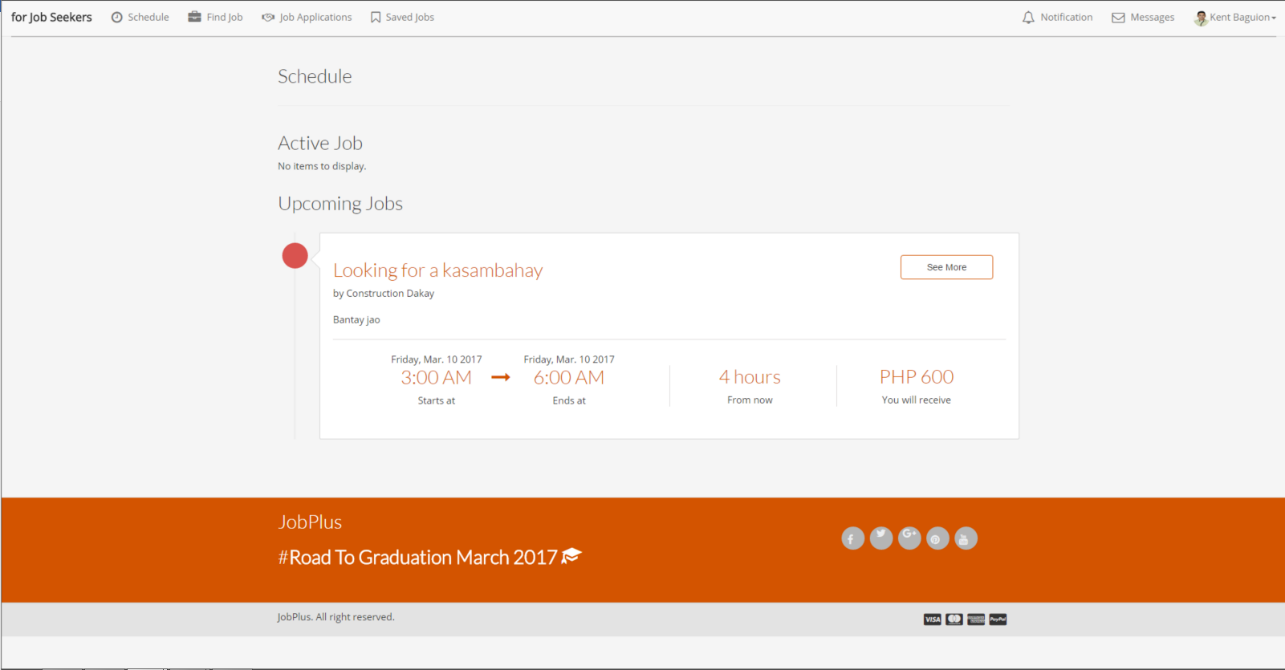
****

**Figure 2.35:** View Job

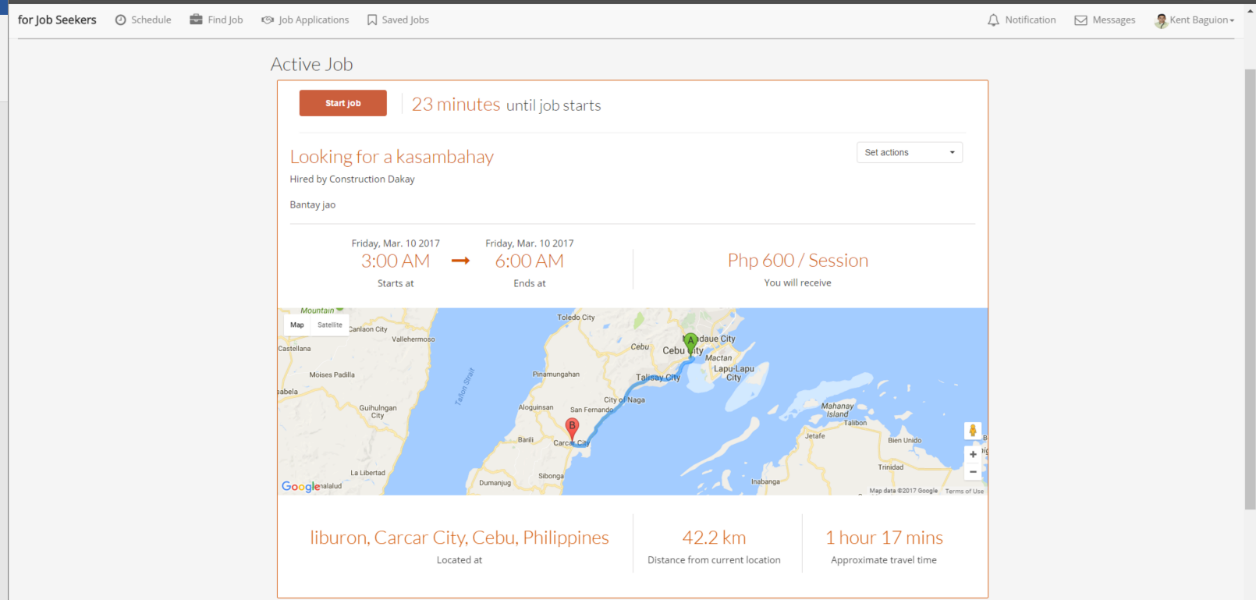
This is a part of the page where the job seeker views the information of the job and to where they can apply and bookmark that job posting.

**Figure 2.36:** Terms and Agreement

Figure 2.33 indicates where the job seeker views the information of the job and to where they can apply and bookmark that job posting.

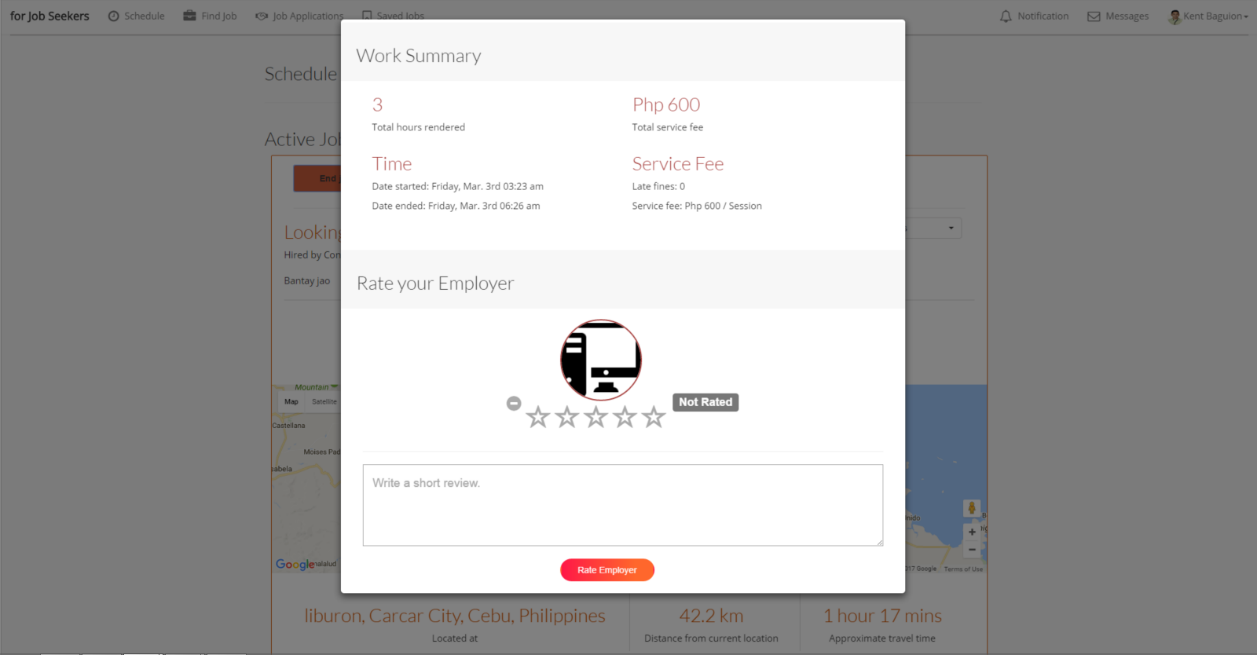
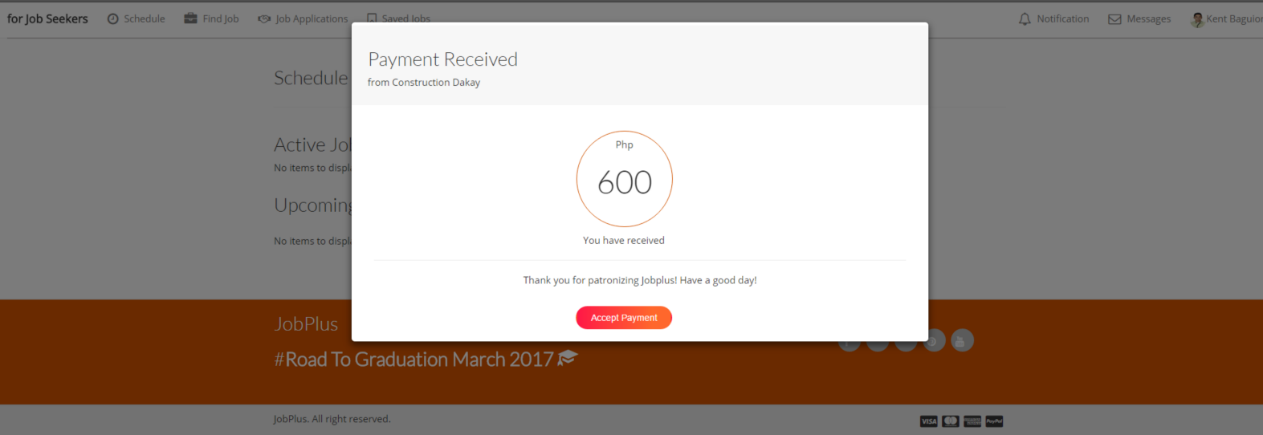
****

**Figure 2.37:** Upcoming Job

This is the page where the job seeker determines the jobs ahead of time.

**Figure 2.38:** Active Job

This is where the jobs that are currently scheduled an hour before or on the current time is found. The job seeker can determine the distance and time of the previous job from the next job by looking at the active job tab.

****This modal generates the summary of the work rendered by the applicant and the venue where the applicant rates their respective employers.

**Figure 2.39:** Rating of Employer

**Figure 2.40:** Payment Received

This is a notification the job seeker gets when the employer already confirmed the end job request and already paid for the compensation of the service rendered. The compensation will then be transferred to the job seeker’s account.

# 

SOFTWARE DEVELOPMENT AND TESTING

This chapter summarizes the platforms and tools used in the project development and testing process.

### DEVELOPMENT SOFTWARE PLATFORMS, DEVELOPMENT ENVIRONMENT AND TOOLS

The web application is developed using Sublime text. [5] Sublime Text is a [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) [cross-platform](https://en.wikipedia.org/wiki/Cross-platform)source code edito[r](https://en.wikipedia.org/wiki/Source_code_editor) with a [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [application](https://en.wikipedia.org/wiki/Application_programming_interface) programming interfac[e](https://en.wikipedia.org/wiki/Application_programming_interface) (API). It natively supports many [programming languages](https://en.wikipedia.org/wiki/Programming_languages) and [markup languages,](https://en.wikipedia.org/wiki/Markup_languages) and its functionality can be extended by users with [plugins.](https://en.wikipedia.org/wiki/Plugins) Our application mainly utilizes Sublime text as the programming medium. Laravel 5 was the framework used to develop our web application, laravel is a PHP framework similar to CodeIgniter. Laravel is accessible, yet powerful, providing powerful tools needed for large, robust applications. [6] It offers the standard Model-View-Controller (MVC) capabilities and provides features that targets specifically for designer and developers.

MySQL Workbench is a visual database design tool that integrates SQL development, Administration, database design, creation and maintenance into a single integrated development environment for MySQL database system. MySQL is the one responsible for securing data the application needs [7].

Google Web APIs service is a beta web program that enables developers to easily find and manipulate information on the web. It’s also for developers and researchers interested in using Google as a resource in their applications. The Google Web APIs service allows software developers to query more than 3 billion web documents directly from their own computer programs.

DEVELOPMENT AND TESTING PROCESS

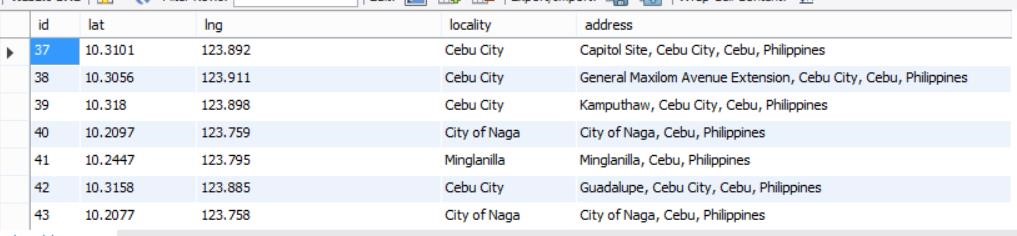
DEVELOPMENT PROCESS

In this section, the development process of retrieving data from Google’s Web Service APIs, processing, and utilizing the results of the study is presented.

Data Retrieval

The system has two approaches in retrieving data. The first approach is for Google’s Web Service APIs and the second one is from the application’s database which uses MySQL.

The system uses Google’s Geocoding API to retrieve the coordinates of the jobs posted by the employer and Google’s Distance Matrix API to retrieve the distance and time between the user and the job. For the system to completely configure the Google’s Web Service API, it needs an API key which is obtained by activating the API in the Google API Console.

 The API key is used to send a request to Google’s Web Service API. The system specifically requests coordinates of the job’s address, which are retrieved after the job posting of the employer. Google Maps Distance Matrix API has a standard limit of 2,500 API request per day.

**Figure 3.1:** Job Address Coordinates

The system also uses MySQL in storing the application’s data. It enables application-to-application interaction over the web by using HTTP request methods such as “GET” and “POST” for posting data in the database.

### Job Recommendation

To start with the job recommendation, the system will first get the address of the user and sends a request to Google’s Geocoding API for the retrieval of the user’s coordinates which will be used later on.



**Figure 3.2:** Getting the profile address and Retrieval of Address Coordinates

The jobs posted by the employer needs to be filtered out to suit the needs of the job seekers. The system will fetch all the jobs from the database, specifically the job’s address which will be used to send a request to the Distance Matrix API for the retrieval of distance and time between the user and the location of the job. After the request, the result will then be stored in the database.

**Figure 3.3:** Retrieval of Distance and Time

The results which are stored in the database will be used to initialize the filtering of the jobs for the relevancy ranking.

### Relevancy Ranking

The system utilizes the recommendation of the jobs by the use of relevancy ranking. In the application, there are three major factors that greatly affects the recommendation. First is the skills provided in the user’s profile. Second is the location given in the user’s profile and lastly, the work history of the user in the application. These three factors will be given different individual weights by categories, these categories are: job feeds, recommended jobs and nearby jobs.

### Location Scoring

After the results in figure 2.31 have been stored in the database, the system will then retrieve and calculates the data. The researcher created a function that accepts distance in meters which is the result in figure 2.31 as the parameter.

**Figure 3.4**: Location Points

Initially there’s a point worth a hundred and a counter that worth zero. The counter will then loop with the condition of counter is less than or equal to the distance. Every cycle will have a one point deduction and additional five hundred for the counter. Which means that in every five-hundred-meter interval, there’s a one point deduction.

### Work History Scoring

The work history will be based on the user’s work sessions ended officially. The system retrieves all the work history and proceeds to the calculations of the points. Below is the function used to retrieve the scoring of history.



**Figure 3.5:** Fetching Work History

To get the history points, the researcher created a function that retrieves all the user’ work history together with the skills associated and compares them to all available jobs. If the past employers matches with the compared job’s employer and the past work skills matches with the job’s skills.

A point is given and will be sent back to figure 2.33, below is the code snippet of the process.

**Figure 3.6** History Points

### Skills Scoring

The system retrieves all the skills specified in the user’s profile and matches them with the job’s skills. If the user has matches all the skills needed for that job, a perfect score is given. A point deduction is given if there’s a lack of user’s skills with the needed job skills. Below is the code snippet of the process.

**Figure 3.7:** Skills Scoring

### Job Feeds

Job feeds is the job category which prioritizes on the user’s work history in the application, followed by the user’s skill and lastly the user’s location.

**Figure 3.8:** Job Feed Criteria

The criteria in figure 3.8 will be used in the job feed category. Fifty percent is given into the user’s work history, thirty percent on the user’s skills and twenty percent on the user’s location. The results from figures 3.4, 3.6 and 3.7 will be retrieved and multiplied by the respective criteria weights. After which, results will be averaged and stored in the database.

### Recommended Jobs

Recommended job is the job category which prioritizes on the user’s skill, followed by the user’s location and lastly the user’s work history.



Figure . Recommended Criteria

The criteria in figure 3.9 will be used in the recommended job category. Fifty percent is given into the user’s skills, thirty percent on the user’s location and twenty percent on the user’s work history. The results from figures 2.32, 2.33 and 2.35 will be retrieved and multiplied by the respective criteria weights. After which, results will be averaged and stored in the database.

### Nearby Job

This section is the job category which prioritizes on the user’s location, followed by the user’s skills and lastly the user’s work history.



**Figure 3.10** Nearby Job Criteria

The criteria in figure 3.10 will be used in the nearby job category. Fifty percent is given into the user’s location, thirty percent on the user’s skills and twenty percent on the user’s work history. The results from figures 2.32, 2.33 and 2.35 will be retrieved and multiplied by the respective criteria weights. After which, results will be averaged and stored in the database.

### Conflict Checking

The system ensures that job seekers will not be checking their schedules for conflicts manually. First, the system retrieves all the job seeker’s schedule and sends a request to the Google’s Distance Matrix API. This method will retrieve the duration time between the previous work to the next work session.



**Figure 3.11:** Retrieval of Distance and time

After the request, the system will add the duration time to the job seeker’s scheduled end time. After which, the system will then compare the added previous work schedule to the job schedule applied. Figure 3.12: Conflict checking

TESTING PROCESS AND CASES

### TEST CASE 1: EMPLOYER

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Module** | **Test Scenario** | **Expected Result** | **Actual Result** |
| Create Job Post | Incomplete form | Must show the incomplete field message | Displayed the incomplete field message |
| Required fields completely filled | Must show a list of job seekers | Displayed the list of job seekers |
| View Posted Jobs | Employer clicks view button | Must show the job information | Displayed the job information |
| Employer Application Response | Employer clicks Accept button | The application’s status in the database must be changed to one. | Application’s status changed to one. |
| Employer clicks Decline button | The job application must be deleted | Job application was deleted. |
| Work | Start Work | The state of the button must be changed. | Button state was changed. |
| End Work | Must show a popup for confirmation | Displayed the popup confirmation |
| Confirm End Work | Must show a modal containing job summary and employee rating | Displayed the modal containing job summary and employee rating. |
| Rating | Incomplete form | Must show an error message | Displayed the error message |
| Required fields completely filled | Must show a confirmation message | Displayed confirmation message |

TEST CASE 2: JOB SEEKER

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Module** | **Test Scenario** | **Expected Result** | **Actual Result** |
| Browse Job | Job seeker filled in filters | Must show a list of filtered job posts. | Displayed the list of filtered job posts. |
| Job seeker clicks recommended job tab | Must show a list of recommended job posts. | Displayed the list of recommended job posts. |
| Job seeker clicks nearby job tab | Must show a list of nearby job posts. | Displayed the list of nearby job posts. |
| Job seeker clicks a job | Must show the job information. | Displayed the job information. |
| Schedule conflict | Must show a conflict message. | Displayed the conflict message. |
| Job Application | Job seeker clicks apply button | Must show a terms and agreements information. | Displayed the terms and agreements information. |
| Job seeker clicks the confirm button | Must show a success message. | Success message displayed. |
| Work | Start Work | The state of the button must be changed. | Button state was changed. |
| Late | Must show a late message | Late message displayed. |
| End Work | Must show a popup for confirmation | Displayed the popup confirmation |
| Confirm End Work | Must show a modal containing job summary and job seeker rating | Displayed the modal containing job summary and job seeker rating. |
| Rating | Incomplete form | Must show an error message | Displayed the error message |
| Required fields completely filled | Must show a confirmation message | Displayed confirmation message |

# CHAPTER 4

SUMMARY, CONCLUSION AND RECOMMENDATIONS

### SUMMARY OF FINDINGS

Job portals are becoming more and more popular in this millennial generation. Employers are already given ample of options to choose or select job seekers effortlessly. Same goes for job seekers, browsing for available jobs nearby and being able to land on a job have increased dramatically.

The results that were based on the testing we have gathered has passed at least 90% of the test case. The development team conducted different tests like getting the probability on when the bug will occur. This application needs high connection to the internet, this also works better if using the Google Chrome and with a medium-end screen resolution.

### CONCLUSION

The study aims to provide digital innovation of employment of individuals. Assuring that this study will somehow make an impact to the people of the community that is in need of jobs or looking for someone’s assistance. The application will help those busy individuals to further boost their productivity by solely focusing on their work. The application may contribute to those people who have a free time and wants to have an extra income.

### RECOMMENDATIONS

Integrating job placement system solely on a web platform may be one step too late because of the fast-paced development of job portals that caters cross- platform environment. The researcher would recommend to future developers and researchers to implement both mobile and web platforms upon the development of the application. The researcher would also recommend to further tighten the security of the application so that the users can really feel secured when using the application.

### FUTURE WORKS

One of the limitations of this study is it only caters web platform. The portability of the application can exponentially affect the user’s experience. For future works, a cross platform application may be implemented. Thus, user’s access is not only limited on desktop but also on mobile phones.

# BIBLIOGRAPHY

|  |  |
| --- | --- |
| [1] | P. Hudson, "Why Do We Work?," Elite Daily, 17 October 2013. [Online]. Available: http://elitedaily.com/money/entrepreneurship/why-do-we-work/. [Accessed 14 March 2017]. |
| [2] | E. T. De Guzman, "Employment Rate in January 2016," Philippine Statistics Authority, Manila, 2016. |
| [3] | E. T. De Guzman, "Employed Persons by Sector, Occupation and Class of Worker, Philippines January 2016," Philippine Statistics Authority, Cebu, 2016. |
| [4] | A. Sheahen, Basic Income Guarantee, 2012. |
| [5] | "Sublime Text," Sublime, [Online]. Available: https://www.sublimetext.com/. [Accessed 16 March 2017]. |
| [6] | "Laravel," [Online]. Available: https://laravel.com/docs/5.4. [Accessed 16 March 2017]. |
| [7] | "MySQL," Oracle, [Online]. Available: https://www.mysql.com/products/workbench/. [Accessed 16 March 2017]. |
| [8] | https://docs.google.com/forms/d/16-MuY9qwwLnrGdb0CGADwA\_G7oRFsxfjzuq09fLcoLs/edit?usp=drive\_web. [Online]. |
| [9] | D. W. Livingstone, The Education-jobs Gap: Underemployment or Economic Democracy, 1999. |

CURRICULUM VITAE

**PERSONAL INFORMATION**

|  |  |
| --- | --- |
| **Name:** | Kent Michael M. Baguion |
| **Nickname:** | Bagzki, Bravo, Kentoy, Kentot |
| **Date of Birth:** | March 13, 1997 |
| **Address:** | Blk. 5 Lt. 13, Brgy. Tambulilid Ormoc City, Leyte |
| **Status:** | Single |
| **Citizenship:** | Filipino |
| **Religion:** | Roman Catholic |
| **Email Address:** | kmbaguion@gmail.com |
| **Contact Number:** | 09335532300 |

**ACADEMIC INFORMATION**

|  |  |
| --- | --- |
| **Tertiary:** | *University of San Jose-Recoletos (2013-2017)* |
|  | Bachelor of Science in Information Technology |
| **Secondary:** | *Saint Peter’s College of Ormoc (2009-2013)* |
| **Primary:** | *Ormoc City Central School (2003-2009)* |

**PROFESSIONAL SKILLS**

|  |  |
| --- | --- |
| **Language:** | Java, C, C++ , PHP , Javascript |
| **Software:** | Netbeans, Eclipse, Visual Studio , Sublime Text |
| **Operating System:** | Windows Variants |