

# SW Engineering CSC648/848

## WorkWaves

### Job Platform

#### Team 03

**Brenda** – *Team Lead*

**Sidney Thomas** – *Github Master*

**Banting Lin**– *Front-End*

**Jaime** – *Support*

**Brandon** – *Scrum master*

**Abdarrahman** – *Back-End \*

#### “Milestone 2”

October 11<sup>th</sup>, 2023

#### Revision History Table

Revision ID	Revision Date	Revised By

## I. Data Definitions V2

Revision from Milestone 1

<b><u>Primary Data Name</u></b>	<b><u>Definition (&amp; Examples)</u></b>	<b><u>Usage</u></b>
User Profile	User Profiles represent individuals who have registered on your application. They can be either Workers or Employers, each with their own set of characteristics and preferences. (e.g John Smith)	User Profiles store personal information, preferences, and data related to their interactions on the platform. This data is used to personalize their experience, facilitate connections with other users, and display relevant content.

Job Post	Job Posts are listings created by Employers to advertise job opportunities (e.g Baby-sitter)	Job Posts allow Employers to post job opportunities, specifying details such as job titles, descriptions, locations, and payment information. Workers can browse and request (apply) for these job listings.
Worker Post	Worker Posts are listings created by Workers to showcase their availability and skills to potential employers or clients. (e.g Experienced Babysitter Available)	Worker Posts provide information about the services Workers offer, including availability, skills, and payment details. Employers can browse and contact Workers for freelance services or jobs.
Rating	Ratings are feedback and rating scores provided by users to evaluate the performance and reliability of Workers or Employers. (e.g Rating: 4.7 stars)	Ratings help build trust and reputation on the platform. They are used by both Workers and Employers to make informed decisions when selecting or hiring someone for a job or service.

### **User Profiles (which can be Workers or Employers):**

- User ID (unique identifier)
- Username
- Email
- Password (hashed)
- First Name
- Last Name

- Profile Picture
- Contact Information (phone number, address, etc.)
- Type (Worker, Employer)
- Registration Date
- Ratings (if applicable)
- Number of Jobs Completed
- Number of Times Requested
- Location
- Notifications Preferences
- **Job Posts:**
- Job Post ID (unique identifier)
- Title
- Description (Notes)
- Date Posted
- Creator (references User Profile ID of the employer)
- Location
- Payment
- Picture (Image)

#### **Worker Posts:**

- Post ID (unique identifier)
- Title
- Description (Notes)
- Picture
- Age
- Rating
- Skill
- Date Posted
- Creator (references User Profile ID of the Worker)
- Location

- Availability
- Payment

**Ratings (if applicable, for Workers and Employers):**

- Rating ID (unique identifier)
- Rated User (references User Profile ID)
- Rating (1-5 stars)
- Date Rated

Data Definition Continued:

<u>Primary Data Name</u>	<u>Sub-Data</u>
User Profile	user_id username email password first_name last_name profile_picture contact_information user_type registration_date

	ratings jobs_completed times_requested location notifications_preferences
Job Post	job_post_id title description date_posted location payment picture creator #Reference to UserProfile (Employer)
Worker Post	post_id title description picture age rating skill date_posted location availability payment creator # Reference to UserProfile (Worker)

Rating	rating_id rating date_rated rated_user # Reference to UserProfile (Worker or Employer)
--------	--

## II. Functional Requirements V2

	Must-have		Desired		Opportunistic
--	-----------	--	---------	--	---------------

	ID	Functional Requirement Description	Details (As Needed)
	1	Users can register for their own workwave accounts	1.1) user can sign up for a new account  1.2) user can sign in with their existing account

	2	Users can post job listings or advertise themselves	<p>2.1) Users can post a job or freelance themselves/there is a tab to switch, each job posting will have a form that they can fill which makes it easy for the search bar to work</p> <p>2.2) Users can also edit their posting</p> <p>2.3) guest users can see what jobs are out there</p>
	3	Have a search bar, also have a tab to filter out certain jobs	<p>3.1) Users can use the search bar for filtering</p> <p>3.2) a side tab for job filtering</p>
	4	Users can apply for jobs listed	4.1) Users should be able to apply for the job with the details of the job posting
	5	Users who apply for jobs will be notified about the job	<p>5.1) the user will be notified when they received a job request</p> <p>5.2) the user will be notified when someone accepted their job request</p>
	6	Users should be able to rate their employee after the job is complete	<p>6.1) employer should be able to give a start rating</p> <p>6.2) an employee star rating is the average of all ratings</p>
	7	Users who register should be able to change their account information	7.1) employers and employee should be able to adjust account info i.e password and username



	8	Employers will be able to request previous users	8.1) employers should be able to dm previous one time employee 8.2) employers should be able to give out contact for long term employment
	9	Users will have the ability to tag precise location for work	9.1) users are able to filter via location for work
	10	Users will have a stats about their performance	10.1) an enhanced star rating system
	11	Work wave will have a help desk and support for issues	11.1) If there are issues we should be able to solve, posting issues  11.2) If the user was not able to do the job we should be able to report them to the help team
	12	Users can	12.1) the user can unsend the message 12.2) the user can reply to a specific message of other person in the chat
	13	Company membership	13.1) we can implement a membership to allow certain jobs to be the front of the page
	14	Algo for targeted users	14.1) users who do the same job over again will get recommended the same jobs that they have done before

### III. UI Mockups and Storyboards (high level only)

Find it in the attachment (of the email)

UX-Mockup/Flow.pdf → interactive pdf where the clickable button has been made.

#### IV. High-level Architecture, Database Organization

##### Users

username
password
email
First name
Last name
Profile picture
location

##### Users if employee

Label as worker
Jobs completed
Rating of completion

##### Users if employer

Label as employer
Previous posting
Rating of the employer*

## Posts

Title of post
location
time posted
Description of post

## Posts if employer looking for employee

Payment type
location
Start time

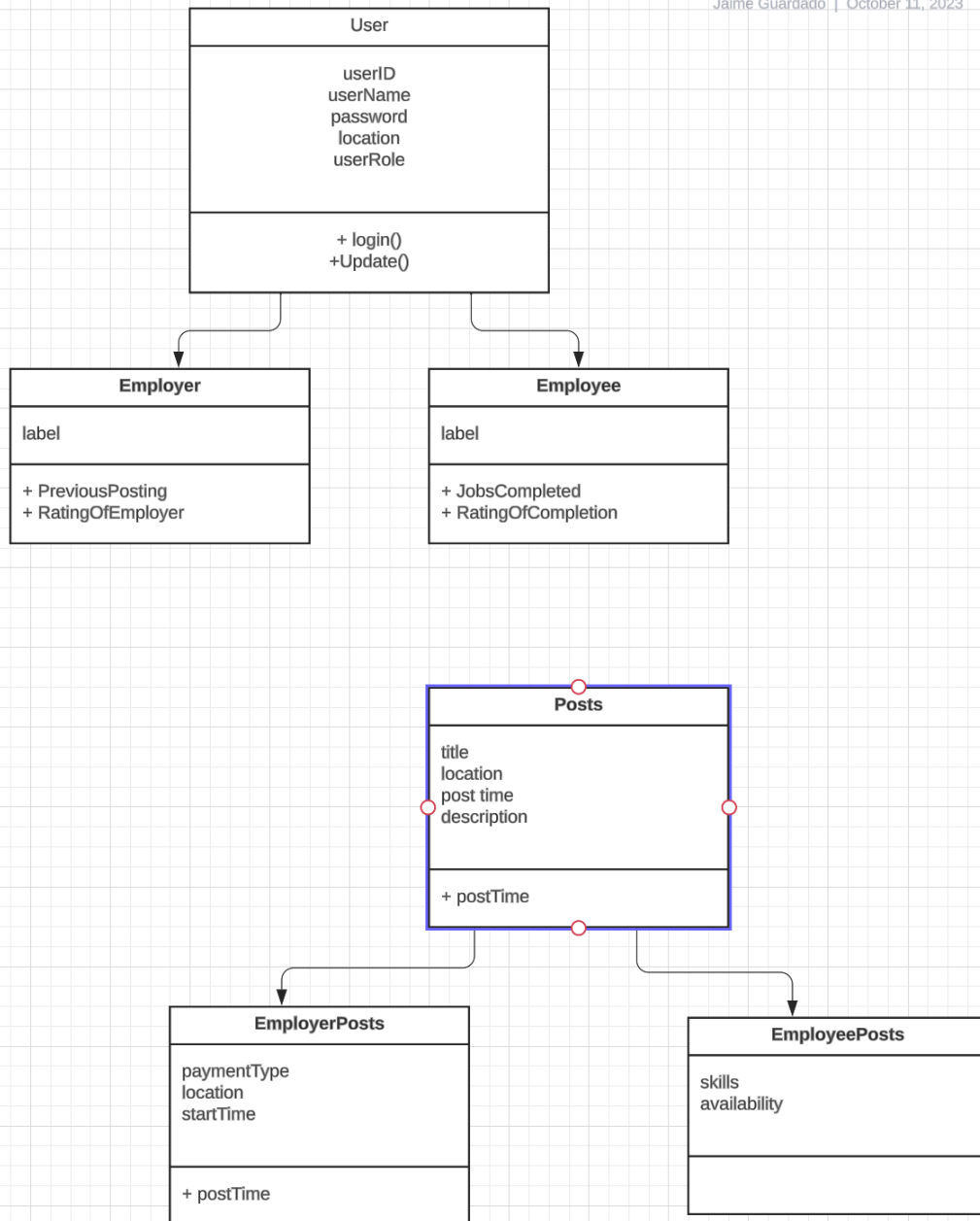
## Posts for self-employed

Skill set
availability

## V. High-level UML Diagrams

## UML class

Jaime Guardado | October 11, 2023



VI. Identify actual key risks for your project at this time

Teamwork Risk:

If team members are not aligned on the project's design and user experience, it can lead to inconsistency and confusion in the final product. For instance, the front-end team may envision a different user interface than what the back-end team is implementing, causing a mismatch in what the final product should be and diminishing the wanted user experience that was initially discussed.

#### Resolution:

- Pair Programming: Cross-team collaboration through pair programming or code reviews. This allows team members from different roles to work closely together and ensure that the front-end and back-end components align effectively.

- Communication Channels: Establish clear communication channels, such as dedicated chat groups (Discord) where team members can easily share design-related updates, questions, and concerns. Encourage active participation and responsiveness within these channels via pinging members on a regular in order to make sure nobody misses the information.

- Design Guidelines: Design guidelines that outline the visual and interaction design principles to be followed. This document can serve as a reference for all team members and ensure consistency in building the project (Github guidelines, using ES-Lint...)

- Prototyping: Create interactive prototypes or wireframes that represent the intended user interface and functionality. Share these prototypes with both members to ensure everyone has a clear understanding of the design direction.

#### Teamwork Risk:

If team members lack a shared understanding of the project's goals and priorities, it can lead to misalignment, wasted efforts, and delays in project delivery. Without clear direction, team members may work on tasks that are not essential.

#### Resolution:

- Project Meeting: Host a meeting at the beginning of each milestone or phase. During this meeting, outline the project's objectives, and high-level goals. Team members can ask questions for clarity.

- Goal Alignment: Ensure that all team members understand how their individual tasks and contributions align with the project's overarching goals. Set up clear tasks for each member with deadlines.

- Regular Goal Review: Periodically review and reaffirm project goals with the team. During team meetings - make any necessary adjustments based on project progress or changing requirements.
- Prioritization Framework: Establish a prioritization framework or methodology that the team can use to determine which tasks or features should be tackled first. This helps prevent the team from working on less critical tasks before addressing essential ones.
- Feedback Mechanism: Feedback mechanism where team members can provide input on project priorities. Encourage team members to voice their opinions and concerns, and consider their feedback when making decisions.
- Goal Tracking: Use project management software to track progress towards project goals and milestones (Taiga). This provides a visual representation of how the team is progressing and helps identify any deviations from the original plan.

## VII. Project management

Our team shares updates in scrum meetings on where they are at with the tasks given to them. We are clear about who does what and who assists with each task. Scrum Master and Team lead take care of the documentation and email the TA after the assignment is done. Through this milestone, the git master and support helped the back and front end with their tasks. Our git master assisted in the wireframe so the rest of us could focus on our tasks. We manage these tasks in Taiga, which is an open source project management tool to assign tasks.