

Freelancer Project

Group 8

May 23, 2018

Abstract

This file is a milestone report about Group 8's Project for CmpE 352 class. Information in this file is up to end of the design part of the project.

1 Executive Summary

1.1 Summary of the Project Status

Project Introduction

The project is a freelancing platform that will be implemented on both Web and Android with necessary API implementation. And the main objective of this project is providing an environment for clients and freelancers to collaborate in a single platform. The clients may create their projects and jobs about programming, writing, design, development etc. and post the details on this platform to find an expert on such topics.

So the things we have done in this semester are:

Learned Git and Github

The platform we use to share our code and reports is Github, also most of our communication takes place on Github by giving feedbacks using issue system. For that reason learning Git and Github was crucial.

Created Communication Plan

The team has 9 members. In order to get the best performance, we came up with a planned meetings that were feasible to all of us.

Created Project Requirements

One of the most important things was Project Requirements. We created functional and non-functional requirements to meet both our customer's needs and also standards of W3C.

Created Project Plan

We have created a comprehensive Project Plan on the upcoming tasks and shared responsibilities among us fairly. The plan may be changed along the way.

Created User Personas and Scenarios

We have created user personas and scenarios to ground our mockups on. We created those personas and scenarios as realistic and detailed as possible.

To come up with a solid presentation of the project requirements to our customer we have created mockups. The mockups are based on the user personas and scenarios.

Created Web Mockups

Web mockups presented how the Web application will look like according to the requirements to our customer.

Created Mobile Mockups

Mobile mockups presented how the Mobile application will look like according to the requirements to our customer.

Created Class Diagrams

The class diagrams give the underlying design of the application. All of the classes and their corresponding functions, structures, etc. were depicted on the diagrams.

Created Sequence Diagrams

The sequence diagrams were created to represent the interaction between the user and the application.

Created Use Cases

To represent the requirements in a step-by-step manner and then create test cases on those we have created Use Cases.

Created Test Cases

To test the normal behaviour of the application and also to test the application's response to abnormal behaviour we have created comprehensive test cases.

Created Twitter API Application

To get familiar with API usage and server deployment we have created a Twitter API application.

1.2 Changes Planned

- The requirements will always be updated.
- All the requirements will be reviewed according to W3C.
- The plan will be reviewed if the team will have new members.
- Communication plan will be updated according to the free times of all members in the new semester.
- Along the way, some of the class diagrams may be changed in the coding part.

2 List and Status of Deliverables

List Of Deliverables	Status
• Requirements	Delivered
• Project Plan	Delivered
• Scenario	Delivered
• View Design	Delivered
• Class Diagram	Delivered
• Sequence Diagram	Delivered
• Use Cases	Delivered
• Test Cases	Delivered

3 Evaluation of the Status of Deliverables

In brief, we believe that the current status of deliverables is quite well satisfactory. We have studied to present all the deliverable we have promised to our customers. We have evaluated our status on each deliverable that was promised on this milestone and the ones that we have started to work on.

3.1 Requirements

Like all project, we have to specify what we need in our project detailed. Initially we search the examples of similar project's requirements, after that we have focused on our projects and we indicated our project's requirements in different stages. We have classified our requirements under the two main topics, Functional and Non-Functional requirements and that we explained our requirements apparently with subtitles.

3.2 Project Plan

In this project we have to implement our project as a group, not individual. In other words, all participants of our group must be in the game. Due to the fact that we have to work together so we conceive a communication plan according to that communication plan we were in contact each other and always keep in touch, according to that plan we work together and we finished what we must do.

3.3 Scenarios

Process of creating scenarios the most important thing is different scenarios should contains different client types. We pay attention to make different types of people when we created customers of freelancer project. In this area actually we succeeded. We have three people one of them is very informed about freelancer system one of them is medium level one of them is minimum level.

3.4 View Design

The most important part in this task is actually the previous task which is scenarios. If we create a good scenario this leads us to well functioning mockups. With a good scenario we could think every possible situation when a person use our application. In mockups we try to think every possible scenario and add them up in view design.

3.5 Class Diagram

Like the other parts in this project we try to make all decisions as a group so in meeting 6 we decide almost every main classes of this project. In this part most important thing is making this class diagram sensible and making every classes and variables consistent with each other. We can say that we succeeded this requirements.

3.6 Sequence Diagrams

We have showed all sequences of the system with different perspectives, like registration, creating a job, bidding, interaction and so on. Thus by this diagrams we showed the systematical explanations of all processes which have done by the users and admins.

3.7 Use Cases

In use cases, we have used some cases which explains some situations for users while they using our application. With some examples we tried to make implication about our application while working and our cases like presentations for users and also a documentation of how they can use this application.

3.8 Test Cases

In this part of project we try to create possible functions of our application like sign up or creation of job. Most essential part of this task is take almost every main function into account and think about their outcome for different situation. I can say that for two essential part we make sufficient test cases suitable for every main request from our application.

4 Coding Work Done

Group Member	Work Done
Orkan Akısü	
Semih Arı	
Abdullah Coşkun	<ul style="list-style-type: none">· Write the breakingNews function which shows breaking news in twitter in 4 different languages· Write the specific Locations function which shows tweets from a given specific location.
İsmet Dağlı	
Kübra Eryılmaz	<ul style="list-style-type: none">· Create virtual environment for the project.· Write the boilerplate code for Django framework.· Implemented a function that uses Twitter API for getting users follower numbers given that their twitter user name by using Django.· Deployed this little project using Digital Ocean.
Abdurrahim Eskin	

Selim Karaduman	
Yaman Kındap	<ul style="list-style-type: none"> · Write the chain follow function using the Twitter API. The function sends follow requests to all users that are being followed by the developer twitter account. · Write the follow back function using the Twitter API. The function sends follow request to all account that follow the developer twitter account. · Write the post tweet function using the Twitter API which posts a new tweet from the developer twitter account. · Write the search query function using the Twitter API which searches a given query in the twitter database and returns the results.

5 Requirements

These are the current requirements of our project.

Glossary

System : The entire application with every feature, for both web and Android.

Guest : A person who is not signed in or signed up yet.

User : A person who is registered to the system.

Username : A unique string that identifies a user.

Password : A secret string that is used to prove users' right to enter the system.

Profile page : A page that contains information about the corresponding user.

Project : An assignment that is created by a user on the system.

Tag : A distinctive string that describes a feature about projects.

Admin : A person who is able to control the system.

Client : A user who creates new projects on the system.

Freelancer : A user who develops the projects that are given by the client.

Bidding : A mechanism with which freelancer states his desired salary for a project.

Rating : A mechanism which helps users to evaluate another user.

Messaging : A mechanism with which freelancer and client will be able to communicate privately.

Dashboard : A collection of data displayed in a graphical layout.

1. Functional Requirements

1.1 System Requirements

1.1.1 Search Engine

1.1.1.1 Semantic Search

Semantic tagging shall be used for description of posts. Wikidata API may be utilized.

1.1.1.2 Filtered Search

The search engine shall support filtering by category and price.

1.1.2 Project Creation Mechanism

The system shall allow clients to create projects.

1.1.3 Bidding Mechanism

There shall be a bidding mechanism such that the biddings of the freelancer shall only be seen by the corresponding client.

1.1.4 Freelancer Client Interaction

There shall be an embedded messaging mechanism between the freelancer and the client throughout the project development. This mechanism shall be disabled once the project is completed.

1.1.5 Recommendation System

There shall be a recommendation mechanism which provides functionalities both for freelancers and clients.

1.1.5.1 The freelancers shall be recommended new projects based on previously completed projects, interest areas and fee.

1.1.5.2 The clients shall be recommended freelancers based on shared project participation, project content and specifications.

1.1.6 Notification System

1.1.6.1 Any change in the project description shall be notified to freelancers who made a bid on the project.

1.1.6.2 All user types shall be notified when their account balance changes.

1.1.7 Browsing

1.1.7.1 A user shall be able to view the public projects on the home page.

1.1.7.2 A client shall be able to view the recommended freelancer profiles.

1.1.7.3 A freelancer shall be able to view the recommended projects.

1.2 User Requirements

1.2.1 Registered User Requirements

1.2.1.1 Project Creation

A user shall be able to create a project.

1.2.1.1.1 Project creation shall require a project description, and categorization of the project.

1.2.1.2 Profile Creation

A user shall have a profile page and he/she shall be able to enter his/her details, experiences and so on to his/her profile.

1.2.1.3 Feedback

1.2.1.3.1 Rating

A client shall be able to rate the freelancer that works on the project after the client approves the project is ended.

1.2.1.3.2 Comment

A user shall be able to leave a comment about his/her freelancer/client, after the project has been completed.

1.2.1.4 Reporting

1.2.1.4.1 A user shall be able to report other users after he/she encounters an inconvenience.

1.2.1.4.2 A user shall be able to report projects after he/she encounters an inconvenience.

1.2.1.5 Searching

1.2.1.5.1 The user shall be able to search a project with a given set of keywords.

1.2.1.5.2 The system shall be able to provide a search mechanism that sorts the results according to the similarities with the provided search keywords.

1.2.1.6 Messaging

Freelancer and Client shall be able to communicate privately after agreeing on the job.

1.2.1.7 Changing Mode

A user shall be able to switch modes between Client Mode and Freelancer Mode.

1.2.1.8 Bidding

A freelancer shall be able to bid on an open project.

1.2.1.9 Financial Transaction

1.2.1.9.1 A user shall be able to load money to his/her account.

1.2.1.9.2 A client shall be able to send money to the arranged freelancer.

1.2.1.10 Project Submission

A freelancer shall be able to submit his/her project to a client's storage.

1.2.1.10.1 The client storage should be able to accept submissions up to 5 MBs of data.

1.2.1.11 Freelancer Selection

1.2.1.11.1 A client shall be able to select a freelancer among the users that made a bid to his/her project.

1.2.1.11.2 A client shall be able to select a freelancers bid if there is sufficient funds in his/her account.

1.2.2 Non Registered User Requirements

1.2.2.1 Signing Up

A non-registered user shall be able to sign up to the service.

1.2.2.2 Login

A guest user shall be able to enter the system with his/her username and password.

1.2.2.3 Visibility

A non-registered user shall be able to see limited number of projects then requested to sign up.

1.2.3 Admin

1.2.3.1 Examining Reports

An admin shall be able to examine reports that have been send by registered users.

1.2.3.2 Control Dashboard

An admin shall be able to control dashboard.

1.2.3.3 Blocking

An admin shall be able to block some registered users' some or all activities or their profile who violate some certain rules.

2.Non-Functional Requirements

2.1 Privacy

2.1.1 Anonymity

- A user shall be able to protect his/her anonymity by not sharing his/her real identity.

2.1.2 Project Privacy

- A client shall be able to choose whether he/she wants his/her project to be public or private.

2.2 Security

2.2.1 Financial Transactions

- Financial transactions should be secure and reliable.

2.2.2 Password Requirements

- To be able to sign up users shall be required to have a secure password.

2.3 System Properties

- The implementation should follow the standards introduced by the World Wide Web Consortium(W3C).

2.4 Response Time

- A request made to the server should be responded in no more than 5 seconds.

2.5 Storage Requirements

- A user shall be able to store at least reasonable(to be decided later) amount of data.

2.6 Usability

- The platform shall be easy to use and easily learnable.

6 Mockups

In this part of project we create mockups to illustrate our customer what would his product look like when the project is done. To make a good example first we should have stories about a group of people who might want to use our website. By this way we could know what are a person needs from our website. So we create 3 people and 3 scenario for them.

Melisa Erdem

Melisa was born in Bursa she graduated from an high school in İstanbul. Currently she is an undergraduate Management student. She made jewelry as an hobby but after her friends really liked them she decided to make some money from it. At first she tried to sell over Instagram but she could not create a loyal customer base. Then she wanted to own a website for selling it but she has no qualification about web designing. After that a friend of her recommended to hire a freelancer and this brings her this website. She is using mobile application to hire a freelancer.

User Scenario

- First she has to sign up then she have to login.
- Second she has to create project.
- Third she should see freelancers who are bidding for her project and their qualifications.
- Then she should be able to visit profile of freelancers who bid her project give the project for him/her.
- At last she has to take the project and pay to the website and she should be able to rate and comment about freelancer.

Acceptance Criteria

- As a client she wants to create a project and freelancers to bid for her project.

- As a client she wants to see profile of freelancers and comment about them.

Mehmet Yiğit

Mehmet was born in Mersin and he spent her life in Mersin till he get into Computer Engineering in Boğaziçi University. He was very interested in programming since high school but but he is not very interested in lectures so it is his sixth year in computer engineering. Another things he is interested in are his games. He spent almost exclusively his money for them. However it became an issue so he wanted to solve it, he is very good at a lot of programming language like C++ Python Java etc. He made many web and mobile application but never turned this into money. Then he decided to became a freelancer but he wants a platform that provide him customers who is aware of value of his knowledge and also projects in his area. He is using freelancer system over his computer.

User Scenario

- First he has to sign up then he have to login.
- Second he has to create a profile.
- Third he should see projects which is related to his qualification which is given in his profile.
- Then when he is interested in some projects he has to bid for them.
- Then when client accept his bid, he should be transferred a page where he can upload the project.
- At last after he delivered project he get paid and he should be able to comment and rate.

Acceptance Criteria

- As a freelancer he wants to create a profile where he can inform about his qualification.
- As a freelancer he wants to see projects which is related to his experiences.
- As a freelancer when he is interested in projects he wants to bid for them.

Sami Şükrü Sayın

Sami Şükrü Sayın was born in Ankara, 1964. He is a retired colonel and he wanted to open a restaurant which especially specialized at meatballs. He planned with his friends at army everything and they opened it in Sultanahmet. They became quite popular in area but still other than the people of the region nobody haven't heard them. A friend of Sami advised him to be a part of yemeksepeti but Sami is so stingy to give them commission and he is a bit paranoiac he doesn't trust them either. Then his friend advised him to hire a freelancer to make his own website he did not trust that too. However after he learnt about recommendation system and reliability of freelancer website he decided to give it a shot. He is using freelancer system over his computer.

User Scenario

- First he has to sign up then she have to login.
- Second he has to create project.
- Third he should see freelancers who are bidding for his project and their qualifications.
- Then he should be able to visit profile of freelancers who bid his project give the project for him/her.
- At last he has to take the project and pay to the website and he should be able to rate and comment about freelancer.

Acceptance Criteria

- As a client he wants to create a project and freelancers to bid for his project.
- As a client he wants to see profile of freelancers and comment about them and their background safety is the most important priority for him.

We created 3 person whose requests differ from each other so our project would be suitable for more people.

Then we create 2 types of mockups web mockups and mobile mockups then we match for these mockups to suitable user (who can use more likely web or mobile).

Web Mockups

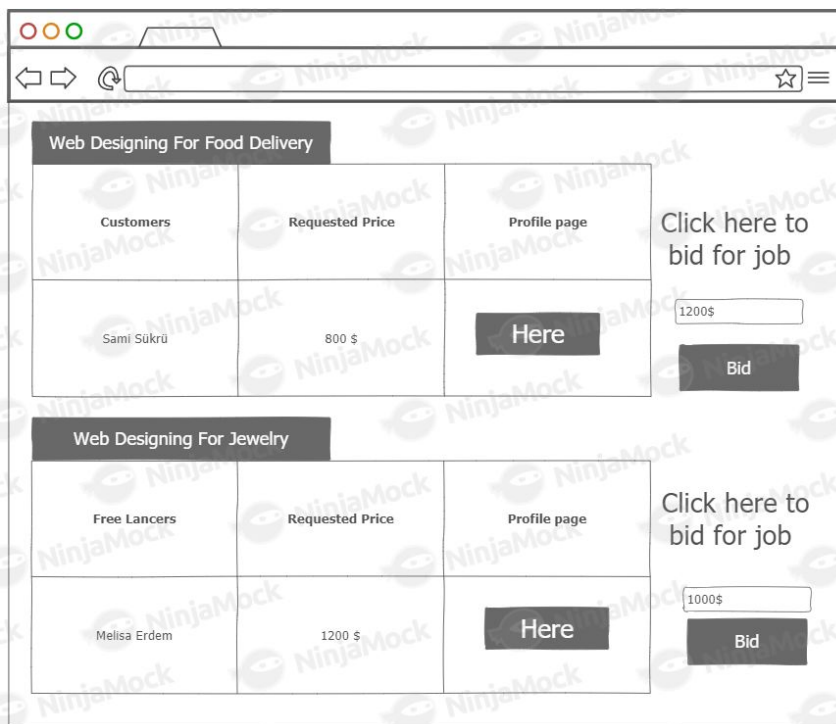
We thought that Mehmet and Sami will be more likely web users.

User Scenario 2(Mehmet Yiğit *Freelancer*)

First he has to sign up then she have to login.



Then when he is interested in some projects he has to bid for them.



Then when client accept his bid, he should be transferred a page where he can upload the project.

Project Upload

Project Name:

ReadMe.md	Read ME	4 days ago
AssetsFolder	Assets Folder	18 hours ago
ImagesFolder	Images	2 days ago

NOTES

- Secure Payment has to be implemented
- Images have no zoom
- Menu obscures the page
- Ads?

At last after he delivered project he get paid and he should be able to comment and rate.

Finishing project

☆☆☆☆☆

User Scenario 3(Sami Şükrü Sayın *Client*)

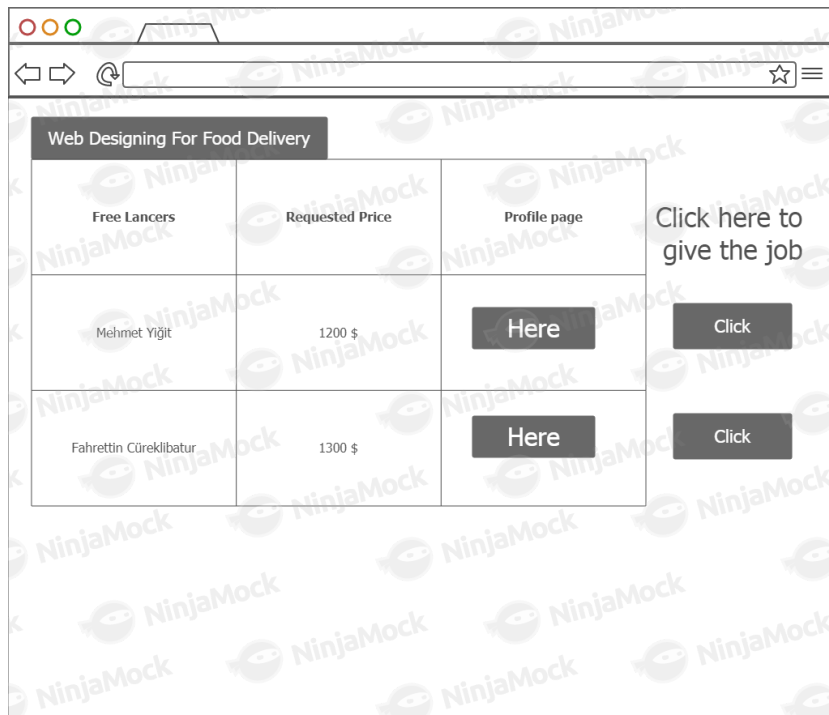
First he has to sign up then he has to login.

The screenshot shows a web browser window with two forms side-by-side. The left form is titled 'Register' and contains the following fields: 'name:' with value 'sami sukru', 'Surname:' with value 'sayin', 'Email:' with value 'sami@gmail.com', 'Password:' with value 'password', and 'Confirm Password:' with value 'password'. There is a 'Register' button at the bottom. The right form is titled 'Log in page' and contains the following fields: 'Email' with value 'sami@gmail.com', 'Password' with value 'Password', a checked 'Remember me' checkbox, a 'login' button, and a 'Forgot password?' link.

Second he has to create project.

The screenshot shows a web browser window with a 'Create A New Project' form. The form has four main sections, each with a header and a text input field: 'Project Name' with the value 'Web Designing For Food Delivery', 'Project Description' with the value 'I want a web designer to create me an online food deliver system. My initial concern is safety so I will accept only 9 point or higher. In this delivery system basicly there should be payment options a menu and prizes. Details will be settled after we get an agreement.', 'Deadline' with the value 'I need it will be done in three months from today', and 'Qualification Requests And Requirements' with the value 'I need a person who has experience at web designing and html language at least for 5 years. Html Web Designing'. There is a 'Done' button at the bottom.

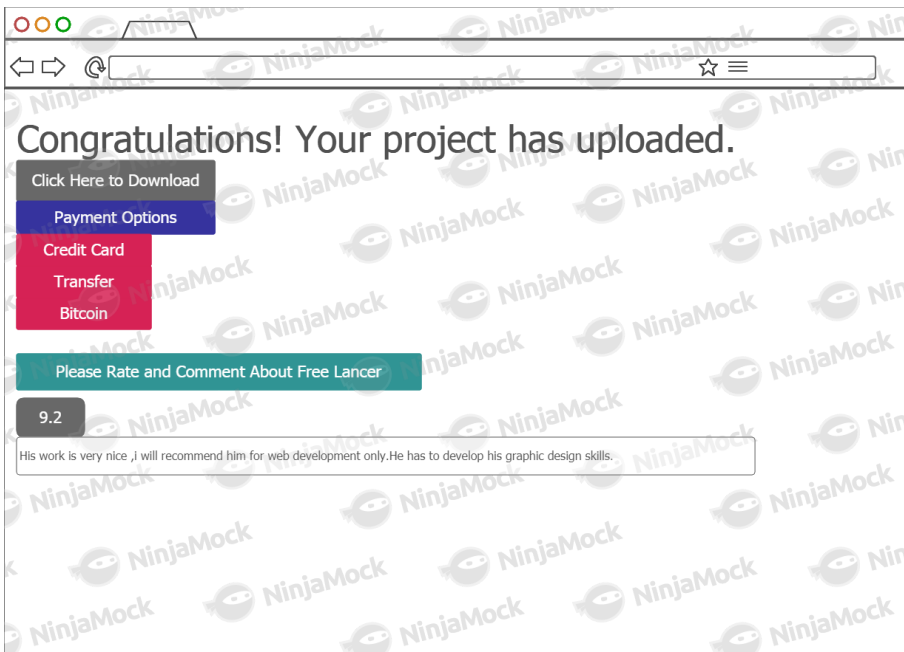
Third he should see freelancers who are bidding for his project and their qualifications.



Then he should be able to visit profile of freelancers who bid his project give the project for him/her.



At last he has to take the project and pay to the website and he should be able to rate and comment about freelancer.



Mobile Mockups:

We thought that Melisa is most likely to be a mobile user.

Melisa unlocks her phone and opens the application. Then,she signs in to her account:



She starts to create her project for her jewelry business:

Create a New Project

Project Name

A Website for Jewelry

Project Description

A need a web developer to design and develop a website for me. The website will have online payment system and it must be secure because i need to create a local customer base. The design of the website shall be compatible with jewelry concept.

Deadline

2019 MARCH 23

Requests and Requirements

I need a web developer who has previous experience in commercial websites.

Skills/Tags

☒ HTML ☒ Web design ☒ PHP

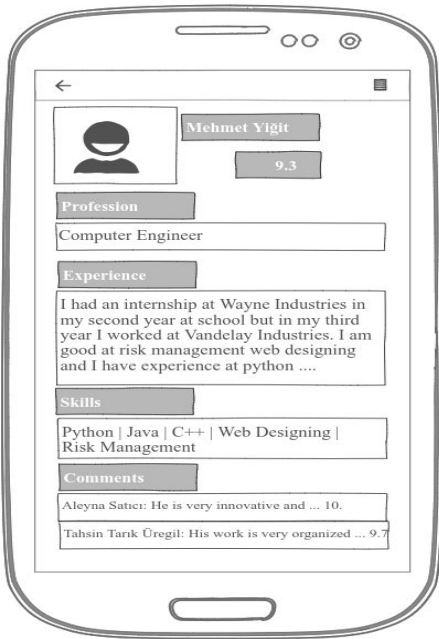
☒ CSS ☒ Javascript

After 2 days she decides to see if anyone has bid on her project:

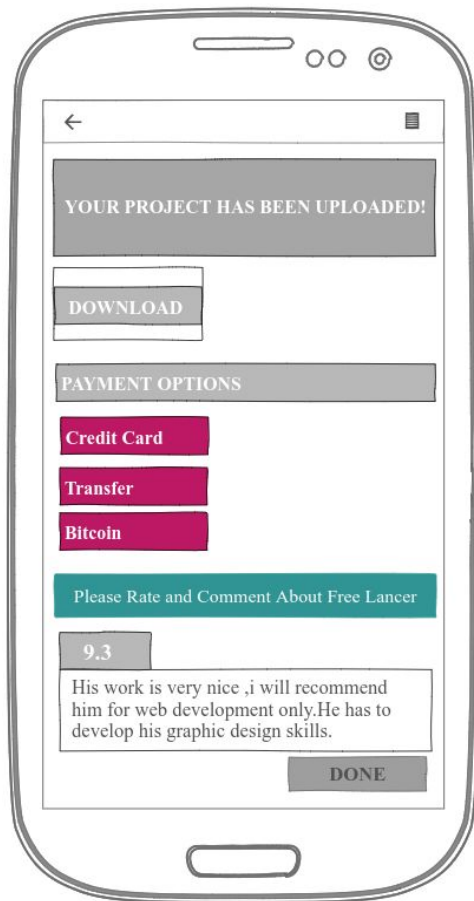
A Website For Jewelry BIDDINGS

Freelancer	Requested Price	Profile Page	
Mehmet Yiğit	1000\$	HERE	✓
Ayşe Demir	1800\$	HERE	✓
Fatma Ak	1200\$	HERE	✓
Ali Yıldırım	800\$	HERE	✓
Ahmet Büyük	3200\$	HERE	✓
Merve Yılmaz	200\$	HERE	✓

She sees Mehmet Yiğit and that name sounds interesting to her,so she looks to his profile:



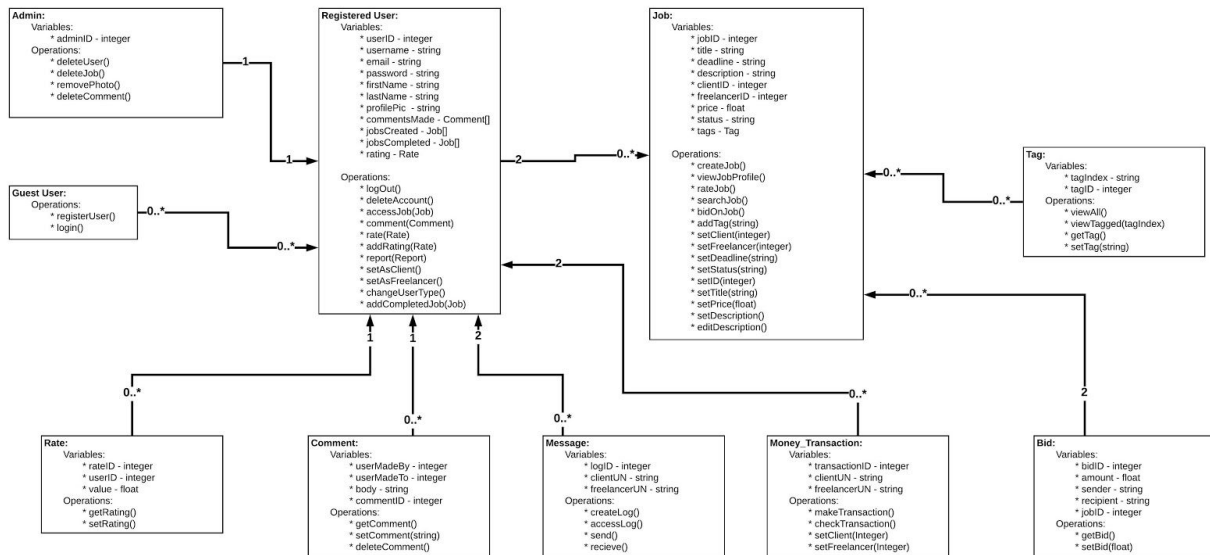
A great web developer Mehmet Yiğit has finished the project on time. Now it's time for Melisa to pay to him and rate him:



7 Software Design

In meeting 6 we decide the design of class and sequence diagram and create a template for that. To make sequence diagrams we should first create the class diagram so we started to think about every functions and classes that we should implement to do this project. Than we take notes about them and at last give task of documenting these notes and uploading that documentation to a group member.

Class Diagram:



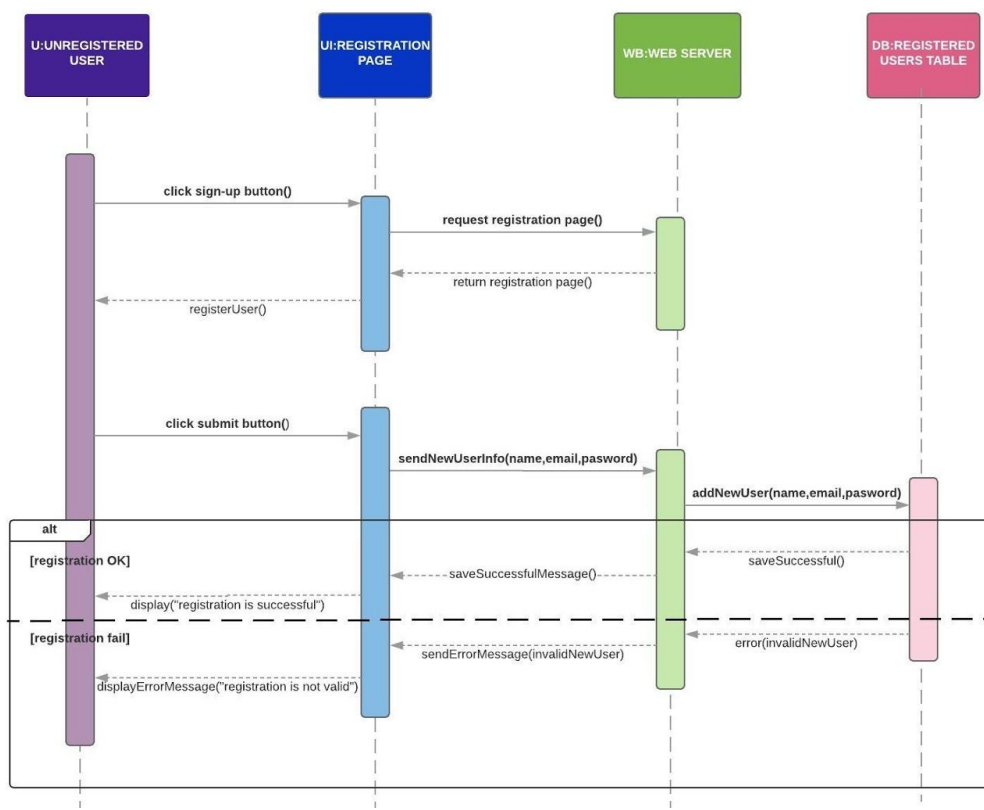
Sequence Diagram:

After creation of class diagram in meeting 7 we decide a scenario which provides us a template of sequence diagrams. At last we decided that for sequence diagram we should have 8 headline:

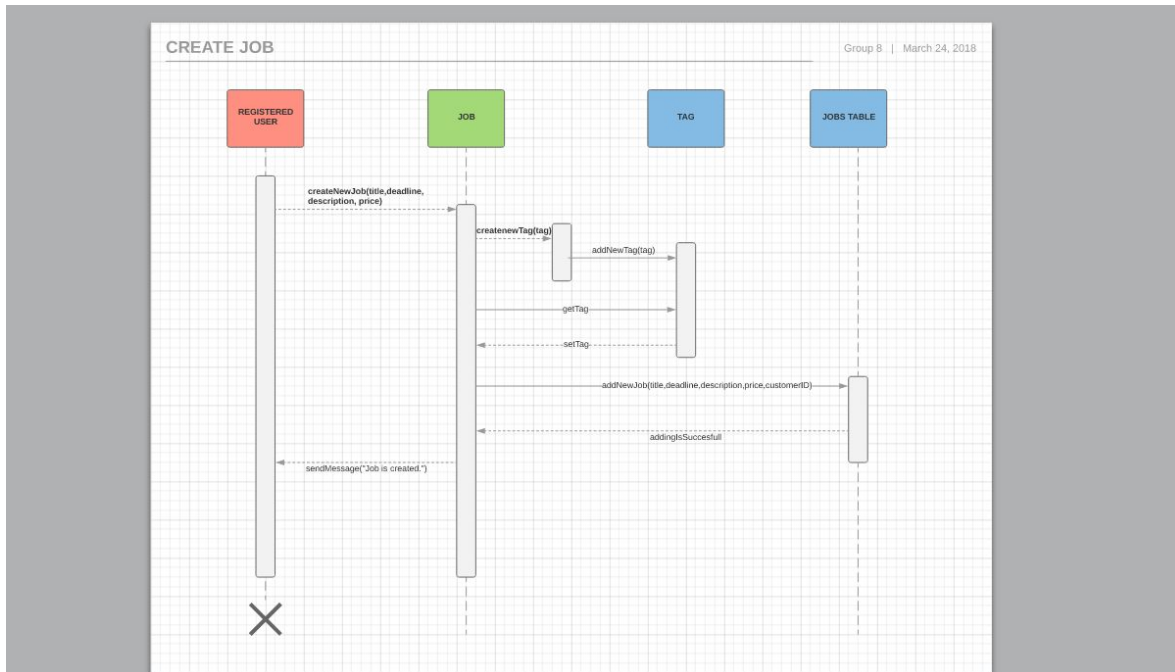
Registration

REGISTRATION

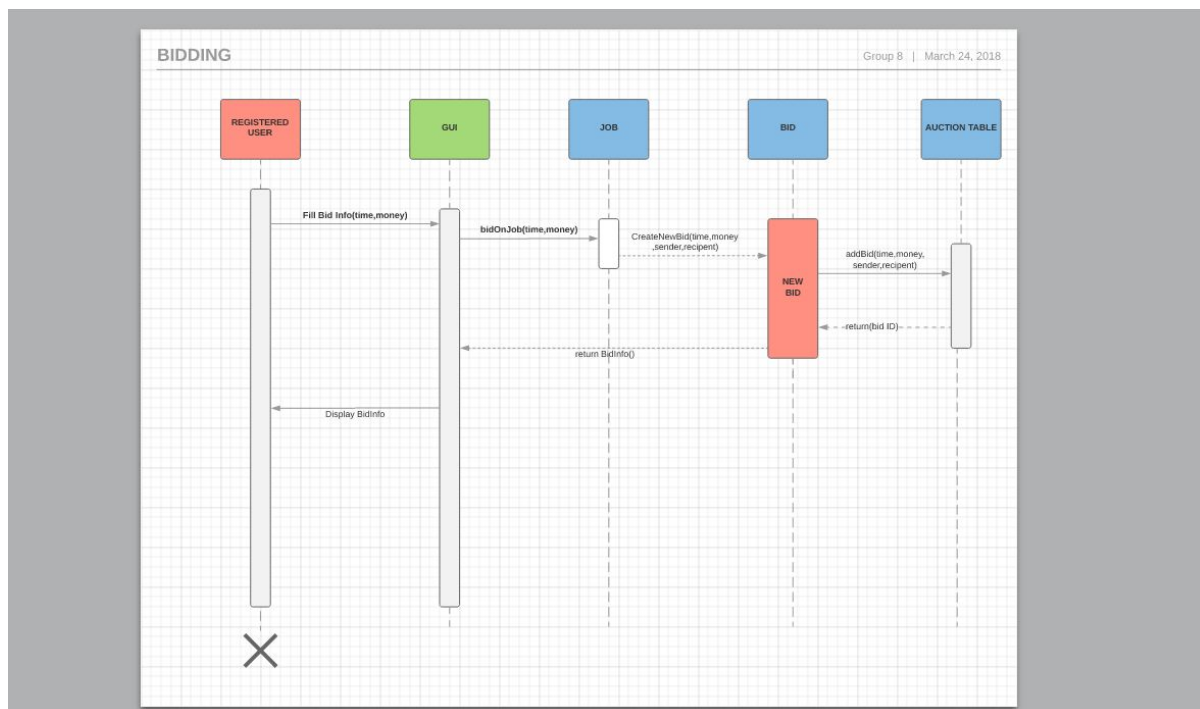
Group 8 | March 24, 2018



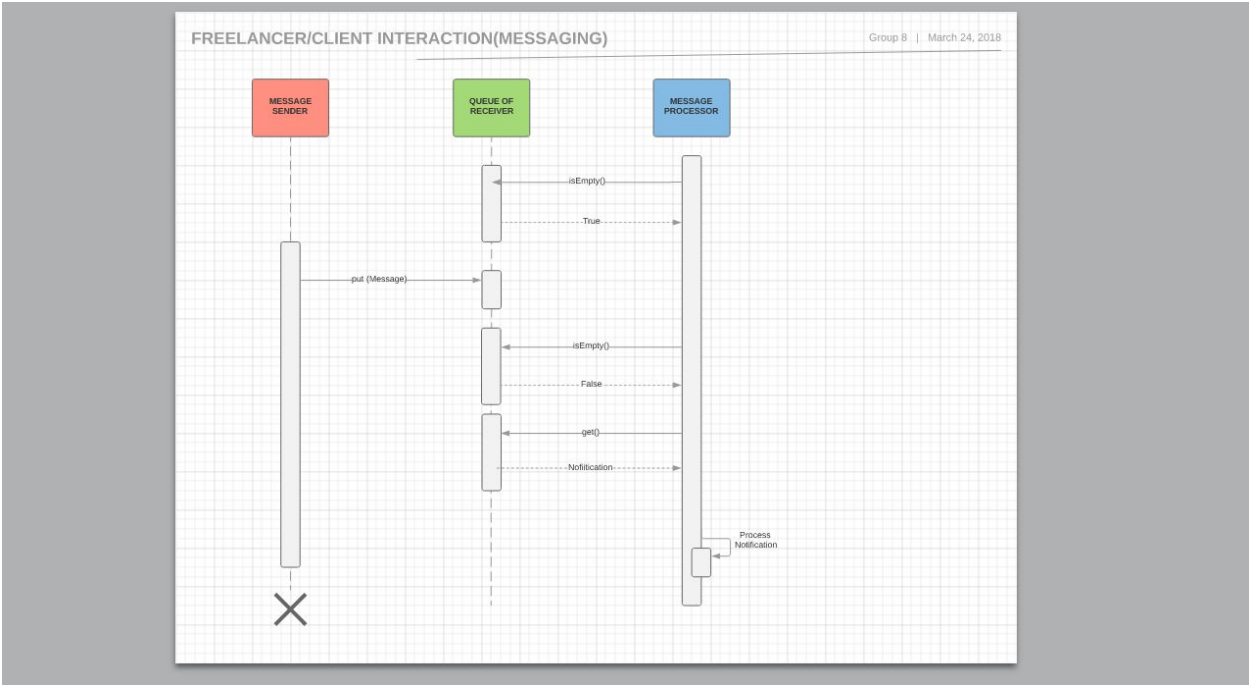
Creating Job



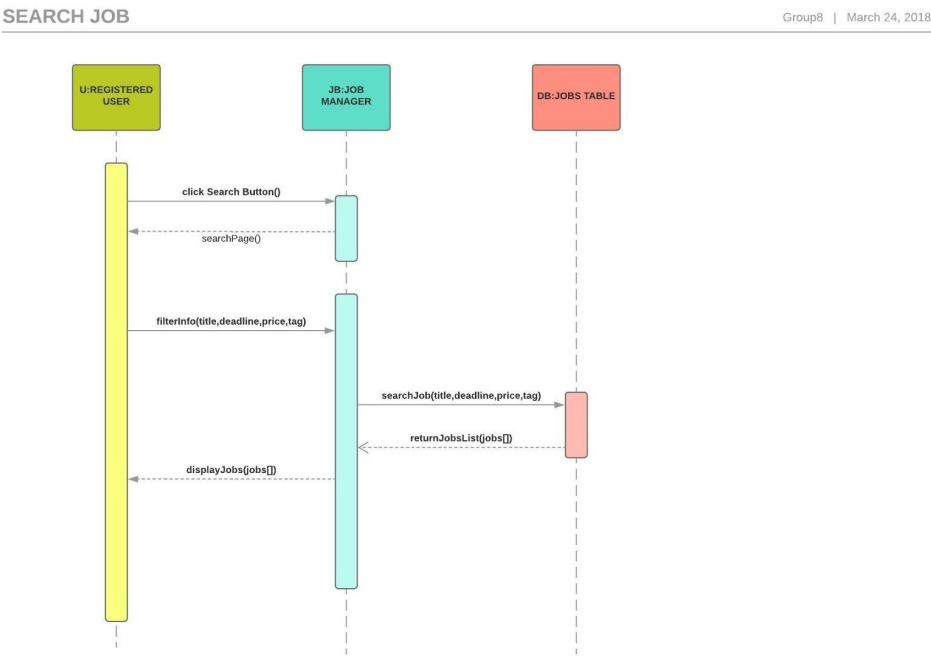
Bidding



Interaction



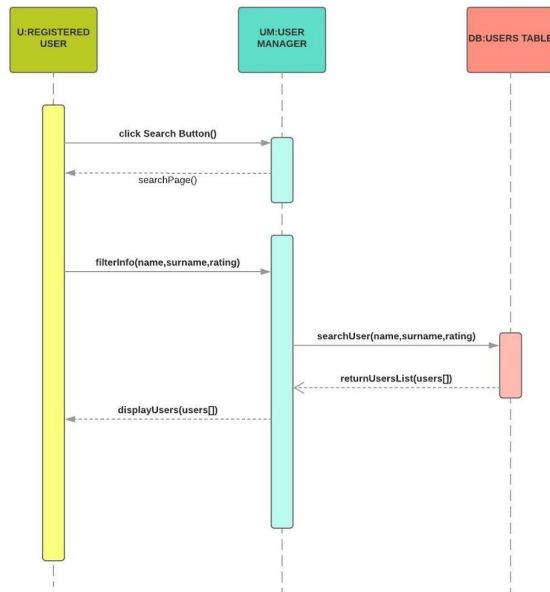
Searching Job



Searching User

SEARCH USER

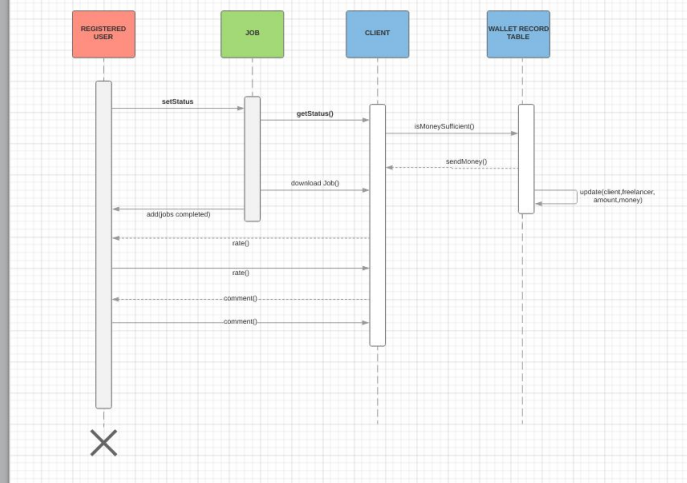
Group 8 | March 24, 2018



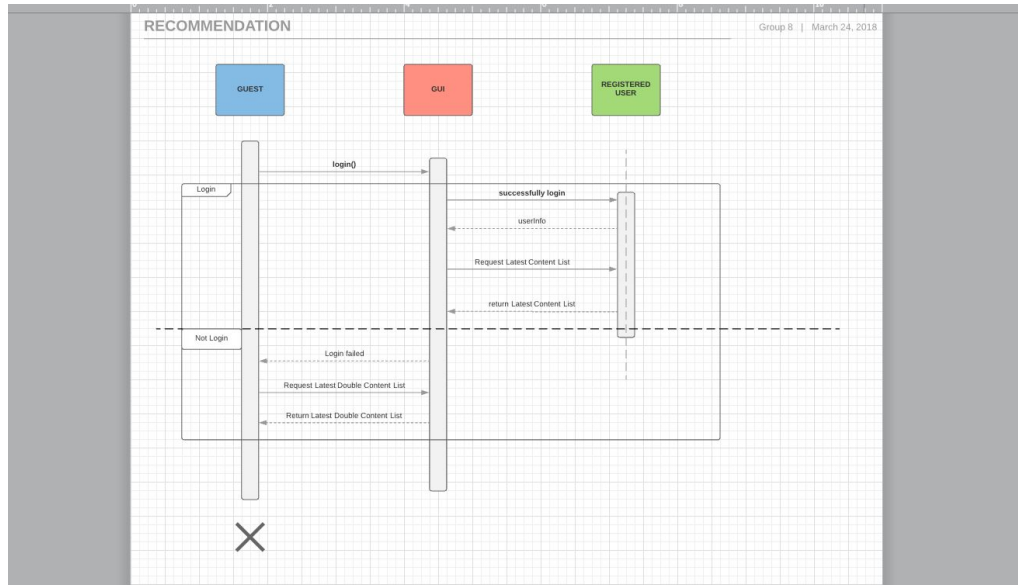
Completion of Job

COMPLICATION OF JOB

Group 8 | March 24, 2018



Recommendation



8 User Tests

In Briefly, according to possible user cases we implemented visual algorithm for the project implementation, processes and we simulate our program to create a idea of how it works. In other words we have showed in general examples, how our program works and which stages our users will encounter and also we show what they must do.

Test Case ID:SIGN UP

Title: Registration

Test Priority: High

Test Scenario: A guest user wants to sign up as a freelancer or a client.

Test Purpose: To test if registration mechanism and remembering registered user mechanism works perfectly.

Designed by: Abdullah Coşkun

Design date: 30/03/18

Executed by: -

Execution date: -

Related Requirements: 1.2.2.1 Signing Up

Preconditions:

1. User is not logged in.
2. Password should be longer than 8 character and contains at least one number one special character.
3. Email should be valid.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Guest User opens Sign Up page		Sign up page will appear	
2	Guest User types email	"mehmet_yigit@gmail.com"	valid	
3	Guest User types password	5801a.38	valid	
4	Guest User clicks sign up		Registration is completed	

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Guest User opens Sign Up page		Sign up page will appear	
2	Guest User types email	"mehmet_yigit"	not valid	

3	Guest User types password	588	not valid	
4	Guest User clicks sign up		"Registration is not completed please check if your email and password is valid"	

Test Case ID:Creation of Job

Title: Job Creation

Test Priority: High

Test Scenario: A Client wants to create a job.

Test Purpose: To test if creation of job mechanism works perfectly.

Designed by: Abdullah Coşkun

Design date: 30/03/18

Executed by: -

Execution date: -

Related Requirements: 1.1.2 , 1.2.1.1

Preconditions:

1. Client is logged in.
2. Job creation page is opened.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Client opens job creation page		Job creation page will appear	

2	Client enters description	"I need a website where I can sell my jewellery ..."	valid	
3	Client enters qualifications	"Web designing, Html ..."	valid	
4	Client clicks create		"Job Creation is completed click here to see freelancers who bids on this project"	

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Client opens job creation page		Job creation page will appear	
2	Client enters description	" "	not valid	
3	Client enters qualifications	"jsva Asembly"	not valid because of mistype	
4	Client clicks create		"Job Creation is not completed please check description and qualification part"	

Test Case ID:BIDDING

Test Priority: High

Test Scenario: Freelancer wants to bid for a project that is opened by a client.

Test Purpose: To test if bidding mechanism works perfectly.

Designed by: A.Selim Karaduman

Design date: 30/03/18

Executed by: -

Execution date: -

Related Requirements: 1.1.3

Preconditions:

1. User is logged in.
2. User has switched to the Freelancer Mode.
3. The client has opened the project for bidding

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer clicks "Bid For the Project" button		Bidding page will appear	
2	Freelancer gives his/her plan to complete the work	"Hello I am X and i can finish this project on time because..."		
3	Freelancer clicks "Send" button		The bidding page is sent to the Client	

Test Case ID:INTERACTION

Test Priority: High

Test Scenario: Freelancer and client interacts with each other.

Test Purpose: To test if interaction mechanism works perfectly.

Designed by: A.Selim Karaduman

Design date: 31/03/18

Executed by: -

Execution date: -

Related Requirements: 1.1.4

Preconditions:

1. Freelancer or Client is logged in.
2. Client has at least one job that freelancer is currently working on.

Table

Step s	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer presses the Interact button on the Client's page	(The Interaction has not been started before)	A chat page appears	
2	Freelancer sends a text message	"Hello ! "	The text message is received on the Client's end	
3	Freelancer sends a file	file.fl (<=25 MB)	The file is received on the Client's end	
4	Freelancer sends a file	file.fl (>25 MB)	Error is given to the freelancer due to the big size of the file	

5	Client presses the Interact button on the Freelancer's page	(The Interaction has not been started before)	A chat page appears	
6	Client sends a text message	"Hello ! "	The text message is received on the Freelancer's end	
7	Client sends a file	file.fl (<=25 MB)	The file is received on the Freelancer's end	
8	Client sends a file	file.fl (>25 MB)	Error is given to the Client due to the big size of the file	

Test Case ID:SEARCHJOB

Test Priority: High

Test Scenario: A freelancer wants to search for a job.

Test Purpose: To test if searching engine works correctly for both semantic search and filtered search.

Designed by: Kübra Eryılmaz

Design date: 1/04/18

Executed by: -

Execution date: -

Related Requirements: 1.1.1.1 Semantic Search, 1.1.1.2 Filtered Search, 1.1.7.3, 1.2.1.5.1, 1.2.1.5.2

Preconditions:

1. User is logged in.
2. User has switched to the Freelancer Mode.
3. The freelancer has opened searching page.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer opens Search page		Search page will appear	
2	Freelancer types some tags	"machinelearning", "Android"	Semantic tagging recommends some related tags like "datascience", "deeplearning" , "mobile"	
3	Freelancer clicks recommended tags		Semantic tagging recommends some other related tags "ai", "machine", "learning"	

4	Freelancer clicks search button		Jobs which tagged with "machinelearning", "Android", "datascience", "deeplearning", "mobile" shows up sorted by users past interests	
---	---------------------------------	--	--	--

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer opens Search page		Search page will appear	
2	Freelancer choose some filters	deadline<23 Apr, price between 100\$-1000\$		
3	Freelancer clicks search button		Jobs which holds filtering conditions shows up sorted by users past interests	

Test Case ID:SEARCHFREELANCER

Test Priority: Medium

Test Scenario: A client wants to search for a freelancer.

Test Purpose: To test if searching engine works correctly for filtered search.

Designed by: Kübra Eryılmaz

Design date: 1/04/18

Executed by: -

Execution date: -

Related Requirements: 1.1.1.2 Filtered Search, 1.1.7.2, 1.2.1.5.2

Preconditions:

1. User is logged in.
2. User has switched to the Client Mode.
3. The client has opened searching page.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Client opens Search page		Search page will appear	
2	Client writes a name	"Mr. Robot"	"there is no such user"	
3	Client writes another name	"Elliot Alderson"	Link to all Elliot Aldersons' page	

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Client opens Search page		Search page will appear	
2	Client choose some filters	rate between 6-10		
3	Client clicks search button		Freelancers which holds filtering conditions shows up sorted by users past interaction	

Test Case ID:Completion of Job

Title: Completion of Job

Test Priority: Medium

Test Scenario: Both users want to give a feedback.

Test Purpose: To test if delivery of project and giving feedback mechanism works perfectly.

Designed by: İsmet Dağlı

Design date: 02/04/2018

Executed by: -

Execution date: -

Related Requirements: 1.2.1.3.1 Rating, 1.2.1.3.2 Comment

Preconditions:

1. Two registered user is logged in, one freelancer mode and one client mode.
2. A formerly created project.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer sets the status of project as finished		Job is done	
2	Client gets the status of project when having enough money		Downloading the job	
3	The client rates the freelancer and clicking the rate button		Adding rate the rate field of freelancer	
4	The freelancer rates the client and clicking the rate button		Adding rate the rate field of client	

5	The client writes a comment and clicking the comment button	"It is done just like I order"	Adding comment the class of freelancer	
6	The freelancer writes a comment and clicking the comment button	"paid in time"	Adding comment the class of client	

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Freelancer sets the status of project as finished		Job is done	
2	Client gets the status of project		Job is available	
3	The client doesn't have enough money		"Error: dont have enough money to get job"	

Test Case ID: Recommendation System

Title: Recommendation

Test Priority: High

Test Scenario: A registered or guest user is at homepage and recommendant projects are displayed.

Test Purpose: To test if recommendation mechanism works perfectly.

Designed by: İsmet Dağlı

Design date: 02/04/2018

Executed by: -

Execution date: -

Related Requirements: 1.1.5 Recommendation System

Preconditions:

1. Guest user is not logged in.
2. Latest visited content list is available.

Table

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Guest User opens the login page		Login page will appear	
2	Guest User types email and password	"ID: mehmet_yigit@gmail.comPW: 5801a.38"	valid	
3	Guest User click the login page		successful login	
4	The system requests the latest content list		Recommendant contents are displayed at homepage	

Steps	Steps of Testing	Test Data	Expected Output	Actual Output
1	Guest User opens the login page		Login page will appear	
2	Guest User types email and password	"ID: mehmet_yigit PW: 5801a.38"	invalid	

3	Guest User click the login page		unsuccessful login	
4	The system requests the latest double content list		Recommendant contents are displayed at homepage	

9 Project Plan

The first thing to do is making a good communication plan before starting the project so we design a communication plan in meeting 2 and document it after that. Communication plan should contain at least 4-5 communication method. Also in an urgent situation these methods should provide us multiple different ways to communicate both with each other and customer.

Audience	Purpose/Message	Delivery Method	Place	Frequency
Every Team Member	discussing any issue about project	Whatsapp	Mobile/Place	Every time needed
Every Team Member	discussing any issue about project	Piazza	Mobile/Place	Every time

				needed
Every Team Member	Updating documentation, sharing research and adding code	Github	Mobile/PC	Every time needed and every week after meetings
Every Team Member	weekly discussing with our group about project	Meetings(group members)	Lounge/Çatı Restaurant	Every Thursday between 17.00 and 18.30
Every Team Member and Customer	keeping in touch with customer and getting information about their needs and requests	Meetings(with customer)	A4 Lab	Every time needed or depends on customer's request
Communicator and Customer	keeping in touch with customer	Email(with customer)	Mobile/PC	Every time needed or depends on customer's request

Every Team Member	discussing any issue about project	Email(with group members)	Mobile/PC	Every time needed or
-------------------	------------------------------------	---------------------------	-----------	----------------------

After that we should create a general plan for giving tasks to different people. In meeting 3.1 we discuss our abilities and experiences and then we arrange a division of labour according to that. At last we document the project plan with using project libre and E-Table.

These are the links for them:

Project Libre:

<https://github.com/bounswe/bounswe2018group8/blob/master/352-%20Project%20Plan-v1.2.pdf>

E- Table:

https://docs.google.com/spreadsheets/d/1uUG5mU9WokJ2TqfvCEcZNPEbG_aYLMoZJf8PpfA14QQ/edit?ts=5a91acdb&pli=1#gid=1493953838.

10 Software Development Infrastructure(API Assignment)

This part of the report is about the assignment of Twitter API. It is not directly related to our Freelancer Project but it was like a demo for real project. We have learned a lot.

1 Plan

Meeting 9 was about API assignment. We tried to decide working environment of the project. Due to the lack of experience about that kind of projects, deciding environment was hard for us. So at the first meeting we could just decided to the language which is Python. We decided to make a little research about twitter api.

In the second meeting, we could decided the topics of our work then assigned each of them to a team member but working environment can't be clarified.

2 Execution

Every team member created its own branch on git and team started to work. Jupyter Notebook seemed to be a good and easy environment to start so some of the team members implemented their own work on it and committed it to the main repository. However it was not a useful environment in terms of user interface and deployment. So the project stuck.

Then we tried to create a Django project. I took a while to learn but a very easy api call was made. Then we used Digital Ocean to deploy it. Everybody tested its own work.

3 Outcome

Outcome is available at our repository. It consists of some discrete python codes written in Jupyter Notebook and a little Django project available at <http://159.65.192.31/>.

4 What Was Learned

First of all “Making a bad decision is better than making no decision at all.” We lost most of our time to deciding the working environment however deciding on something and starting to work would be better for our project.

Time management and group communication was fundamental for a good quality software project. Our communication was not very good so it costed us a lot. At the end we could not manage to combine our work.

We learned about Twitter API and using APIs. It was a good experience. We have meet with Django and we learned how to deploy a web project. After all we took our lesson from this assignment. We decided to improve our coding skills at summer time to make better work at the next semester.

11 Evaluation of Tools and Project Management

11.1 Tools

Up to the present, we used some tools to manage our project.

- **Github** : It was a base for our project. One can see all of our work done in our repository and wiki page. It also helped to our communication via issues part.
- **Git** : We used git for development part of twitter api. It was a super useful tool for collaborate software development.
- **Project Libre**: We used this for creating project plan. However it was not easy to use. First we can not work collaborate. We first write to a Google Doc to out plan using Excel then one of our team member had to transfer them into Project Libre. It was time consuming. Its looking is also not very well designed.
- **Google Docs**: It is not a project management tool but we have used it a lot. It allowed us to see other team members work and contribute to them.
- **Lucid Chart**: We used this for creating sequence diagrams. It was very nice and easy to use tool.
- **NinjaMock**: We used this for making our Android and Web mockups.
- **Jupyter Notebook**: We used this for developing our Twitter API assignment.
- **Django**: We used this Python Framework to develop our Twitter API assignment for both backend and frontend side.
- **Pycharm**: Python IDE for development of any kind of python code.

- **Digital Ocean:** A platform for web project deployment.

11.2 Project Management

Project management is a hard process, it includes so many factors. We tried to manage that hard process as a group. Each one gained new skills like managing time, communicating with other people and make things done on time.

Our team members background was near to each other and we didn't know much about web development and contributing to a long project. Some of us took responsibility and learned a lot and some of us not. Our communication was better than average. We tried to help each other. We tied to make our meetings on time and with as much people as possible so we were flexible about meeting times.

We took so many lessons in this semester. We hope it will make us a better team at the next semester.