Bilkent University

Department of Computer Engineering

CS413 Course Project

Jobify

Software Project Management Plan

Emre Tolga Ayan, Cemal Arda Kızılkaya, Enes Merdane, Alptekin Önder

Instructor: Haluk Altunel
Teaching Assistant(s): Saeed Karimi

December 3, 2020

Contents

1. Overview

This section of the document consists of a brief introduction to the web-based career platform Jobify. In this section, first, the purpose, scope, and the objectives of the project will be described. Then, the assumptions and constraints related to the project will be explained. Finally, the project deliverables will be presented.

1.1. Project Summary

Jobify is a web-based career portal with mobile application support. The main point that sets Jobify apart from other competitors in the market is its ability to collect hundreds of job advertisements on other career platforms, such as LinkedIn, under a single platform. In this way, the users won't have to spend too much time searching for open positions on dozens of different platforms. This also makes the companies' work easier since they won't have to post job advertisements on different platforms to reach as many candidates as possible. Instead, they can simply post their job advertisements on Jobify, where the job seekers can see their job advertisements easily. The portal also provides auto-fill by reading a CV, where the users can upload their CVs and the system processes the document to fill in the information required by the user as accurately as possible. Also, Jobify will have a team dedicated entirely to account verification to increase the reliability of the system.

1.1.1 Purpose, Scope, and Objectives

The purpose of Jobify is to make the job searching process less cumbersome for job seekers by collecting job advertisements from different platforms into one page. The scope of this project can be listed as:

- Creating a profile and listing past experiences
- Matching candidates with relevant job advertisements
- Automatic fill-in system at sign up by uploading a CV
- Collecting relevant job advertisements from other career platforms
- Messaging system for both the job seekers and enterprises
- Job advertisement posting system for enterprises
- Verification of the users to ensure that the platform is reliable

The objectives of the project can be listed as:

- Deliver the system, described in the project charter and SRS document, on time within the budget provided by the sponsor
- Comply with the legal regulations regarding usage and handling of user data

1.1.2 Assumptions and Constraints

The assumptions and constraints regarding Jobify were described in the project charter and the SRS document. The assumptions regarding the project can be summarized as:

- Searching for open positions is already a tedious project for job seekers to the extent that it might cause them to not search for a job at all.
- Users can apply to multiple positions if the job advertisements are collected in a single application rather than being on different platforms.
- The users will have a device that is capable of handling a web application.

The constraints regarding the project can be summarized as:

- The most crucial constraint is the time constraint since we have to complete the project by the deadline specified by the sponsor.
- The budget provided for the project is another constraint that must be taken into account when allocating resources for the project.
- The workforce is also an important constraint since most of the team members don't have experience with relevant technologies that will be used for the implementation of the project.

1.1.3 Project Deliverables

The project deliverables include the documents that are shared with the sponsor during different stages of the project as well as the internal deliverables which will be used by the developers for the maintenance of the project in the future. The deliverables are as follows:

- Project charter
- SRS document
- Software project management plan
- Software analysis report
- Software design report
- The software
- User manual

1.2 References and Definitions

1.2.1 References

[1] Standards.ieee.org. 2020. IEEE SA - The IEEE Standards Association - Home. [online]
Available at: <https://standards.ieee.org/> [Accessed 2 December 2020].
[2] Chef.io. [online]
Available at: <https://www.chef.io> [Accessed 2 December 2020].

1.2.2 Definitions

- The term "User" includes the job seekers and recruiters.

2. Project Organization

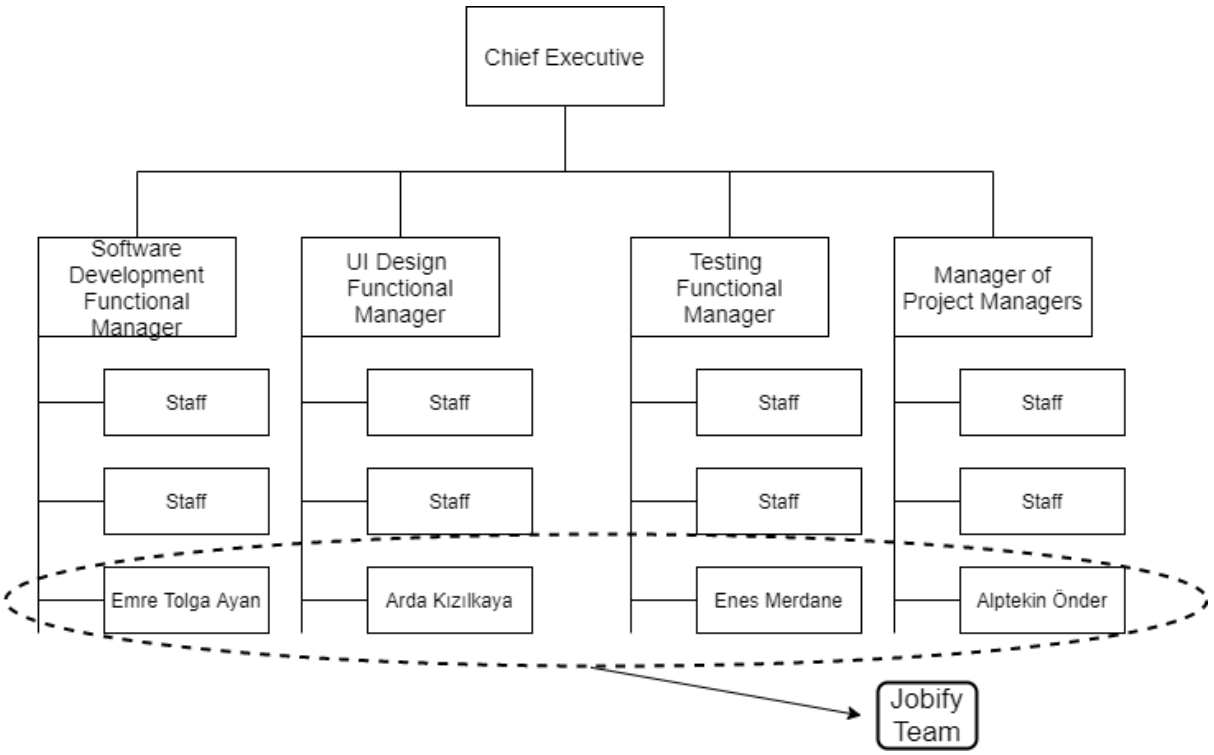
2.1 Internal Structure

For the Jobify project, the company has chosen to use a composite organizational structure. In this structure a team is going to be assembled from different functional units of the company. Then a project manager is going to be assigned to this team from the project managers group. Project manager is responsible for providing communication channels inside the team and making sure that the assigned engineers are sufficient for the project. If the team requires more engineers, the project manager should contact the responsible functional group's manager in the company.

Project manager is responsible for configuration management inside the team and between the team and other units. For that purpose CHEF configuration management tools are going to be used.

For quality assurance of the Jobify project, all the code is going to be checked by another worker. Therefore, at least two engineers should approve the code, this assures the quality of software.

All software developers are from the software development functional group in the company. When a team like Jobify needs engineers, people from desired experience are chosen from that functional group.



2.2 Roles and Responsibilities

2.2.1 Major work Activities

Website Development: This part includes work activities that involve website development. Unit that is responsible will decide which technologies should be used for the website.

Database Management: This part includes work activities that involve database management. Unit that is responsible will decide which database should be used for the website and mobile app. They will implement the database side of the project.

Mobile App Development: This part includes work activities that involve mobile app development. Unit that is responsible will decide which technologies should be used for the mobile app.

Frontend Development: In this part the UI design unit and software development unit works together.

UI Design: UI design unit will design an UI for website and mobile application. These UIs will be designed according to the environments that they will be used.

Testing Software Activities: This work activity involves the all test processes of the software development unit. After a completion or a progress of each of the software development processes, it will be tested under this activity.

Coordination and Communication of Project: Management, coordination and communication of the project will be held under this activity. Plans and workflow of the project will be controlled.

Organizational Unit → Work Activity↓	Software Development Unit	UI Design Unit	Testing Unit	Project Managing Unit
Website Development	X			
Database Management	X			
Frontend Development	X	X		
Mobile App Development	X			

Testing Software Activities			X	
UI Design		X		
Coordination and Communication of Project				X

3. Managerial Process Plan

3.1 Start-up Plan

3.1.1 Project Effort, Cost, and Schedule Estimation

Schedule:
Jobify team will use the agile work model scrum as a software management plan. In the work schedule there are going to be sprints each month. Developers will try to complete the work of that sprintin this month. Overall structure of the management plan has phases, as explained below. In this work model phases will be in the following order: requirements, design, implementation, testing, installation and maintenance. Schedule is going to be divided into scrum work packages of two weeks. When necessary work packages of a phase are completed, the phase will be over. In each of these phases we may need to add or subtract features and change the plan of the project. Because of such cases we choose an agile work model.

First phase is the requirements phase. In that phase the customer explains what's their requests from the software, the project manager and the team creates a requirements file. This phase will take 1 work package of 2 weeks in order to have a good understanding of the project. Also the project manager can discuss the features with other project managers and developers in that period. Project managing unit is responsible.

Second phase is the design phase. In that phase which software and hardware technologies should be used in order to meet the requirements is decided. In that phase behavioural and structural diagrams and plans made. It is a preparation for the implementation phase. This phase will take 2 work packages of 2 weeks.

Third phase is the implementation phase. In that phase many different parts of the project will be handled. For example backend, frontend, UI, database etc. These parts will need different time periods. Overall implementation of all systems will take 6 work packages of 2 week.

Fourth phase is the testing phase. In this phase all implementations will be tested. Different parts of the project will be exposed to different kinds of tests. This phase will take 1 week.

After testing, installation and maintenance will begin. Installation will not take weeks, but only days and maintenance is not exactly a phase but a continuing process after the project ends.

EFFORT Estimation:
In effort computation for the Jobify project we will use Cocomo constructive cost model. Cocomo will be used to estimate the effort that is used in the implementation phase. Since the implementation phase is 16 weeks, its critical path is 12 weeks. The Kilo Line of Code that will be generated for this project should be around 8. Also we get the constants we need in Cocomo from the similar projects in this field.Constants are a,b,c,d.

a = 2, b = 1, c= 1, d=½ Loc = 2000

E = 2*(8) = 16 D =1*(16)^½ =4 R= 16/4 = 4 -> 4 is the number of developer

E is 16 man months of work and 4 developers are needed in order to finish in 4 months.

COST Estimation:
The cost estimation is done by considering different expenses of the system and salaries of company workers. The system expenses are server payment to amazon, hosting for the website.These expenses will be shown in the table below. Also, some of these expenses are going to be paid every month. These are for server costs and salaries of maintenance engineers.

The salaries of company workers are going to be as listed below:

- Software engineers 6500 turkish liras, there will be 4 engineers so 26000 in total,
- Software Architect 7000 turkish liras,
- Project Manager 7000 turkish liras,
- Tester 6500 turkish liras.
- System expenses are server cost from amazon aws which includes many hosting features and website supports in the package. It is 2000 turkish liras every month. Also there should be an SSL certificate cost of 200 turkish liras.

This cost plan is for until the system is ready to publish. After that company should assign software engineers that will always work on this project for future updates and necessary fixes. At this point, their salaries and amazon server costs will be added.

3.1.2 Staffing Plan

The Jobify project requires different types of experienced workers for different phases. First of all, it needs a project manager for managing the project, creating coordination between groups and companies. Project manager will work in all phases. Then one software architect is required. Software architect works in the design phase and makes the necessary software design for the project. The architect has to be experienced in order to make sure that design will not fail the project from foundation. We estimated that we need 4 developers for the project for the given time period. These developers are 1 frontend developer, 1 backend developer, 1 database engineer, 1 web developer. These developers should be experienced in the field they work but it is not necessary for them to be worked on a project like Jobify.

These developers will work in the implementation phase. 1 tester is required for the testing phase. Tester will test all features of the app and website. Other than these phases, installation and maintenance phase requires 1 software engineer.

Roles	Phase	Number
Project Manager	All Phases	1
Software Architect	Design Phase	1
Frontend Developer	Implementation Phase	1
Backend Developer	Implementation Phase	1
Database Engineer	Implementation Phase	1
Web Developer	Implementation Phase	1
Tester	Testing Phase	1
Software Engineer	Installation and Maintenance	1

3.1.3 Training Plan

In the Jobify project, the entire staff can go through training if it seems necessary. There are 2 qualification criteria for training. First one is developers have to work in the area that they are hired. They should have experience in at least 2 projects. Second criteria is, developers have to work on the specific technologies that are decided to be used in Jobify. Same rules will apply to managers, software architects and testers. The ones who need training will go through a training process of 2 weeks before the project.

3.2 Work Plan

3.2.1 Main Activities (WBS)

In this project, we have five main phases which are analysis and planning phase, high-level and low-level design phase, development phase, testing phase and presentation or deployment phase.

The analysis and planning phase includes the analysis of the domain, analysis of the tools and understanding the customer expectations. This phase includes the deliverables project charter, scope and requirements analysis report and software project management plan which is this document.

The high-level and low-level design phase is the phase that the software development plan is done. In this phase, we will design the software application in high-level and low-level, and deliver the high-level and low-level design reports.

In the development phase, we will be implementing the project based on the high-level and low-level design reports that we have delivered. In this phase, we will deliver the web based and mobile based platforms.

In the testing phase, we will be testing each modules' functionality, and integration of these modules. Then, we will be doing system-wide tests and we will take actions for any bug.

The last phase is the presentation phase which includes polishing the software and preparing a presentation. In this phase, we will deliver the final product presentation. Work packages for each of the phases and general WBS can be found in appendix A.

3.2.2 Schedule Allocation (Gantt Chart)

We have assigned each work package to a member who will be responsible for the delivery of that work package. We have distributed the workload as fair as possible between the members such that no member will have a significant amount of extra workload than others. Also, we have designed the schedule such that different components of the project will be developed simultaneously, and every member will be working on a component in each instance of time. By this way, we will be using both our time and resources efficiently. The work schedule and corresponding gantt chart can be found in appendix B.

3.3 Control Plan

3.3.1 Requirements Control Plan

The control of requirements is an ongoing process that starts when the project starts and ends with the project's ending phase. Because how much you meet the requirements decides the quality of the product/service, it is a very crucial process that should be kept properly during the project lifetime.

To follow the requirements during the project we are going to use RTM (Requirements Traceability Matrix). RTM will contain Business Requirements/Business Use Cases with their IDs, Functional Requirements that are related to these use cases and their IDs, the priority of these requirements and Test Case IDs of corresponding test cases. That way, the tester will be able to trace the status of the requirements including their tests and related other requirements.

The priorities of the requirements will be decided by the project team by Scrum Poker which is a game that is created for that purpose.

The completed version of the RTM can be found on the appendix C.

3.3.2 Schedule Control Plan

The schedule control of this project will be maintained using the gantt chart that is created and shared in this document. That gantt chart is created according to the milestones that are defined in the project charter. The resources are allocated accordingly so that no one on the

project team does quite more work than other members. Prerequisites of the deliverables are also considered during the creation of the gantt chart.

For the control of the schedule, the project manager should make checks on the gantt chart properly to keep track of the project. And if the project team is behind the schedule, he should take necessary actions for the deliverables to be completed in time. To compare the schedule and the teamwork, project managers can use Earned Value Analysis (EVA) (see Appendix D). If EVA shows that we are behind the schedule, the project manager can allocate some extra budget to the project to keep up with the milestones.

3.3.3 Budget Control Plan

The budget estimation is made in this report. Since we have a limited budget that is provided by the sponsor for our project, we have to be very careful while using it. The Project manager should keep regular meetings to keep track of the budget usage of the project and should identify whether we are behind over or under budget. According to the situation, the project manager should take necessary actions.

For the control of the plan, the project manager should calculate the cost performance index (CPI) and report what is the situation of the project team in terms of budget. The project manager should consider the budget and schedule together. If needed, he should decide to make transactions between these two factors.

3.3.4 Quality Control Plan

Because we are trying to use Agile Methodologies in our project, the quality control of the product will be consistent during the lifetime of the project. At each phase of the project, the created deliverables should be sent to the sponsor for the validation of the requirements and according to the feedback, necessary actions should be taken by the project team to meet the customer’s expectations. According to the feedback, the project manager may decide to hold an urgent meeting with the project team or he may invite the sponsor or any other stakeholder to the meeting.

3.3.5 Reporting Plan

All the deliverables specified in this report will be reported to the sponsor with their time limit. These deliverables can be divided into three categories, which are the documentation, project implementation and demo. The documentation includes Project Specification Report, Project Charter, Software Project Management Plan, High Level and Low Level Design Reports. The project implementation includes only the end product and demo includes the presentation slides and preparation of the project demo.

During the lifetime of the project, the project manager should be in touch with the sponsor for any change in the requirements. In case of any change in the requirements, the project manager should gather the team for an urgent meeting and necessary changes on the documentation and the project should be done according to that.

The preparation of the reports should be according to the IEEE standards[1] and all the references should be cited in the reports to avoid any kind of plagiarism.

3.4 Risk Management Plan

Risk estimation is done according to the probability impact matrix, which can be found in the Appendices.

1. **Misunderstanding the requirements and/or change in requirements**

   Probability: 0.8 (High)
   Impact: 0.1 (Moderate), since we are following the agile methodology
   Effect: 0.08 (Moderate risk)
   Assigned Person: Cemal Arda Kızılkaya
   Handling:
   - We will analyse the new requirements as quickly as possible and make the necessary changes. Thus, the next sprint of the project will be based on the new requirements → Mitigate

2. **Team member leaving the project**

   Probability: 0.5 (Moderate)
   Impact: 0.4 (High), due to the loss of work force and increased workload for rest of the team members
   Effect: 0.2 (High Risk)
   Assigned Person: Emre Tolga Ayan
   Handling:
   - The tasks will be redistributed among the remaining team members. → Mitigate
   - A new team member will be hired. → Mitigate
   - Some part of the project will be done by another company or team. → Transfer

3. **Choosing technologies that are difficult to integrate, outdated, or relatively new**

   Probability: 0.6 (Moderate)
   Impact: 0.8 (High)
   Effect: 0.48 (High risk)
   Assigned Person: Enes Merdane
   Handling:
   - At the beginning, team members will try to do small projects with the tools they have chosen. If they find out that the technologies they have chosen are outdated, new, or difficult to use together, they will choose new technologies. → Avoid

4. **Incorrect planning or estimation**

   Probability: 0.3 (High)
   Impact: 0.4 (High)
   Effect: 0.12 (Moderate risk)

Assigned Person: Alptekin Önder
Handling:

- Exchanging information with people experienced in developing similar types of projects can help us make better decisions during planning and estimation at the beginning. → Avoid
- New team members can be hired to decrease the workload. → Mitigate

3.5 Closure Plan

There are two different ways that we can reach project closure. One of these ways is cancelling or failing the project due to an unpredicted risk. The other way is a more desirable closure, where the project deliverables are handed to the sponsor and the project is maintained by the developers over time to make sure the system works in a desirable way and possibly enhancing the product by adding new features depending on the demand and interest of the users. As the system is being used by others, it will attract more and more users for some time, which will help us recover the overall cost of the project and possibly cover the current maintenance costs and allow us to make profit from the project. As long as this is the case, the project will be maintained. However, if the revenue generated by the platform is not even covering the maintenance costs, we will wrap up the project and the project will reach its closure.

4. Technical Process Plan

4.1 Process Model

4.2 Methods, Tools, and Techniques

- Because the project requirements and deliverables are not strictly defined and unchangeable, being agile in this project is very important. Therefore, SCRUM methodology will be used during the project lifetime.
- For the creation of the database, Firebase Database will be used.
- For the frontend development of the website, React ve Bootstrap wil be used.
- Backend services will be written with Javascript and the web scraping part will be implemented using Python.
- The collaboration within the team members and tracking of the requirements will be provided by Jira.
- The communication between the project team will be on the Discord channel. The meetings will also be held on that channel.
- The documentation will be created using Google Docs and the presentations will be created using Google Slides.
- For the version control of the project Github will be used.

4.3 Infrastructure Plan

- Because of the intense schedule that the project team members have other than that project, the daily SCRUM meetings will be held only on mondays, tuesdays and fridays.
- Each feature will be opened as an issue in Jira. Issues will be assigned to developers and they will be closed when the pull request for the issue is approved.
- Github CI/CD pipeline will be used for automatic testing.

4.4 Product Acceptance Plan

Communication is the key point to be accepted by the customer. After submitting each deliverable (reports, product and demo) to the customer, the feedback of the customer will be taken and discussed between the team members to fix the product according to customer's expectations. In that way, it is expected that the end-product will satisfy the customer.

5. Supporting Plans

5.1 Configuration Management Plan

The team members will use CHEF Configuration Management Tool to keep track of changes and component versions of Jobify in order to ensure a healthy development process. The main reason behind our choice to use CHEF is its strong version control capabilities which come from its tight integration with Git technologies. Additionally, CHEF offers great flexibility with its code-driven setup, which makes it highly customizable for different use cases and adaptable for various needs. Also, CHEF has a highly active community, which makes it easier for us to get support when we have difficulty using it.

5.2 Quality Assurance Plan

In the test phase, all the submodules and the system will be checked for any errors or any disqualified part or module. After detecting all the problematic parts, they will be assigned to a developer immediately and they will be solved in a given time interval. Besides, not to create these errors in the first place, every developer's pull request will be approved by another developer in the team, and if there is a problem in terms of code quality or functionality, the pull request will be rejected and sent back for revision. Additionally, every developer will write its unit tests before starting to code.

With these precautions, our first aim is not to create a disqualified module in the first place. However, if they are created and merged into the project, they will be detected in the testing phase and will be solved immediately. For all the tests and reviews, functional, non-functional, and pseudo requirements will be the main guideline.

5.3 Process Improvement Plan

When the Jobify system encounters a problem or malfunction in the website or in the app, process improvement plan will be applied.

First of all, when a problem occurred, the engineer who is responsible for that part of the project should be informed about the error. For example, if a part of a website crashes on certain occasions, a web developer is informed and assigned this job to take care of the problem. When an engineer is assigned, the engineer should analyze the foundation of the problem. When the problem is analyzed and well understood, the engineer prepares a plan for it. This plan includes necessary changes in software and systems. When the analyzing part is over, the engineer starts to implement the solution that is planned. If this solution requires a change in other parts of the system which is not his responsibility, he should inform the engineers of that part of the system. After the implementation part, the engineer evaluates the solution with the other engineers of the project. In this way it is understood that the problem is solved and the solution does not affect the other parts of the project. If the problem is not solved or other problems are occured the process improvement plan should be repeated. Moreover, if any part of the system works in underperformance than expected, the process improvement plan should be applied.

6. Additional Plans

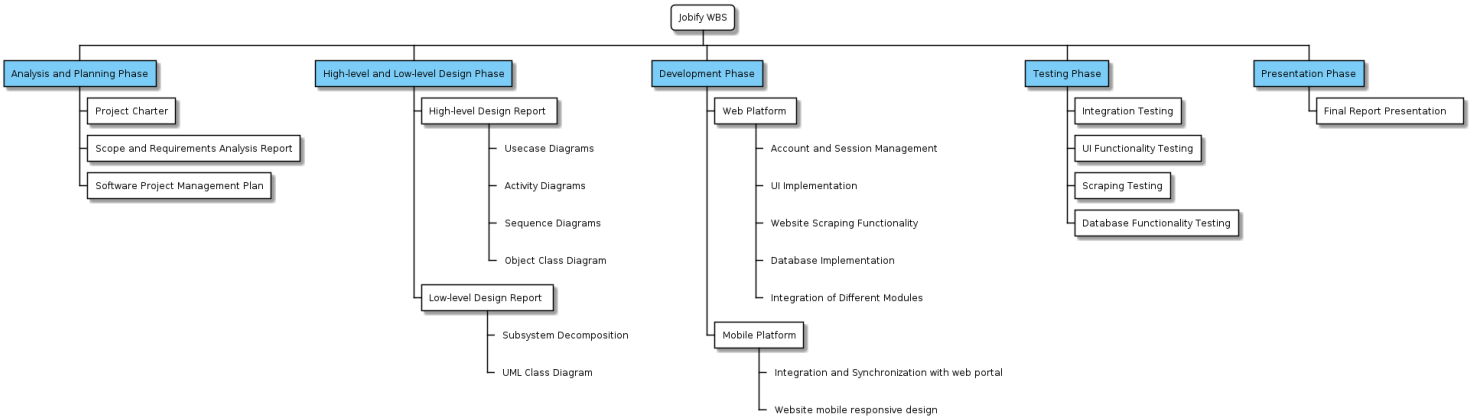
In the future, it is planned that all job searching websites will be scrapped in real-time and Jobify will become the Google of job searching. People will use Jobify instead of using Google to scan job searching websites. Therefore, marketers and SEO experts are planned to be employed to be the top and most well-known website on google.

Because Jobify deals with candidate and company information, being ready to legal actions should be a priority in the short run. The rivals of Jobify may accuse Jobify for several things not depending on being right or not. Therefore, a lawyer should be hired to protect the company against such actions.

Finally, because scraping from other websites and displaying at Jobify is a risky application to some extent due to being dependent on other websites stability, Jobify may decide to maintain the information scrapped in a larger database. Therefore in the future there may be a need to hire an expert and a place to build our own servers.

7. Appendices

Appendix A:

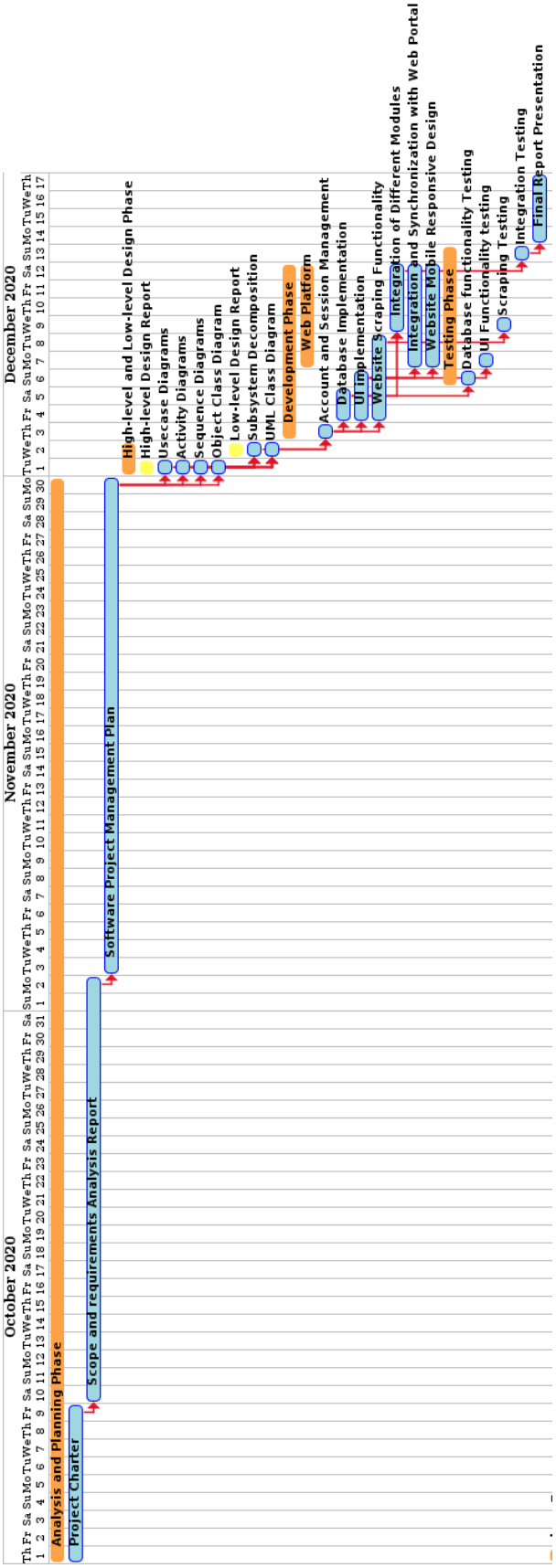


Appendix B:

	Name	Duration	Start	Finish	Leader
1	**Jobify**	77 days	01.10.2020	17.12.2020	
2	**Analysis and Planning Phase**	63 days	01.10.2020	01.12.2020	
3	Project Charter	10 days	01.10.2020	09.10.2020	All members
4	Scope and Requirements Analysis Report	25 days	10.10.2020	02.11.2020	All members
5	Software Project Management Plan	28 days	03.11.2020	01.12.2020	All members
6	**High-level and Low-level Design Phase**	2 days	01.12.2020	02.12.2020	
7	High-level Design Report	1 day	01.12.2020	01.12.2020	
8	Use case Diagrams	1 day	01.12.2020	01.12.2020	Alptekin Önder
9	Activity Diagrams	1 day	01.12.2020	01.12.2020	Arda Kızılkaya
10	Sequence Diagrams	1 day	01.12.2020	01.12.2020	Emre Tolga Ayan
11	Object Class Diagram	1 day	01.12.2020	01.12.2020	Enes Merdane
12	Low-level Design Report	1 day	02.12.2020	02.12.2020	
13	Subsystem Decomposition	1 day	02.12.2020	02.12.2020	Emre Tolga Ayan
14	UML Class Diagram	1 day	02.12.2020	02.12.2020	Enes Merdane
15	**Development Phase**	11 days	02.12.2020	12.12.2020	

16	Web Platform	11 days	02.12.2020	12.12.2020	
17	Account and Session Management	1 day	03.12.2020	03.12.2020	Enes Merdane
18	UI implementation	3 days	04.12.2020	06.12.2020	Alptekin Önder, Enes Merdane
19	Website Scraping Functionality	5 days	04.12.2020	08.12.2020	Emre Tolga Ayan
20	Database Implementation	2 days	04.12.2020	05.12.2020	Arda Kızılkaya
21	Integration of Different Modules	4 days	09.12.2020	12.12.2020	Emre Tolga Ayan
22	Mobile Platform	5 days	07.12.2020	12.12.2020	
23	Integration and Synchronization with web portal	6 days	07.12.2020	12.12.2020	Enes Merdane
24	Website Mobile Responsive Design	6 days	07.12.2020	12.12.2020	Alptekin Önder
25	**Testing Phase**	8 days	06.12.2020	13.12.2020	
26	Database Functionality Testing	1 day	06.12.2020	06.12.2020	Arda Kızılkaya
27	UI Functionality Testing	1 day	07.12.2020	07.12.2020	Arda Kızılkaya
28	Scraping Testing	1 day	09.12.2020	09.12.2020	Arda Kızılkaya
29	Integration Testing	1 day	13.12.2020	13.12.2020	Enes Merdane
30	**Presentation Phase**	4 days	14.12.2020	17.12.2020	
31	Final Report Presentation	4 days	14.12.2020	17.12.2020	All members

Appendix C:


October 2020
November 2020
December 2020
Analysis and Planning Phase
Project Charter
Scope and requirements Analysis Report
Software Project Management Plan
High-level and Low-level Design Phase
High-level Design Report
Usecase Diagrams
Activity Diagrams
Sequence Diagrams
Object Class Diagram
Low-level Design Report
Subsystem Decomposition
UML Class Diagram
Development Phase
Web Platform
Account and Session Management
Database Implementation
UI implementation
Website Scraping Functionality
Integration of Different Modules
Integration and Synchronization with Web Portal
Website Mobile Responsive Design
Testing Phase
Database functionality Testing
UI Functionality testing
Scraping Testing
Integration Testing
Final Report Presentation

REQUIREMENTS TRACEBILITY MATRIX					
Project Name: Jobify					
Business Requirements Document (BRD)		Functional Requirements Document (FSD)			Test Case Document
Business Requirement ID	Business Requirement / Business Use Case	Functional Requirement ID	Functional Requirement	Priority	Test Case ID
BR001	Automatic CV scan and Registration	FR001	Support of differenf file formats	low	TC001 TC002 TC003
		FR002	Register Function		TC004 TC005
BR002	Gathering Job Advertisements	FR003	Scraping Websites	high	TC006 TC007
		FR004	Login Function		TC008
BR003	Checking Candidate Information	FR002	Register Function	high	TC009 TC010
BR004	Matching Candidates and Enterprises	FR005	Selecting Job Preferences	medium	TC011 TC012
		FR004	Login Function		TC013
BR005	Searching and Filtering Candidates /Companies	FR006	Search&Filter Function	medium	TC014 TC015 TC016
BR006	Sending Messages to Candidates /Companies	FR002	Register Function	low	TC017 TC018 TC019
		FR004	Login Function		TC020
		FR006	Search&Filter Function		TC021 TC022
BR007	Post Job Advertisements Requirement	FR002	Register Function	high	TC023 TC024 TC025
		FR004	Login Function		TC0026

Appendix D:

<u>Schedule Performance Index</u>

SPI = EV / PV

SPI < 1.0 → We are behind the schedule

SPI > 1.0 → We are front the schedule

Appendix E: Probability Impact Matrix

Probabilities	Threats				
	1	2	3	4	5
Very High / 0.9	0.05	0.09	0.18	0.36	0.72
High / 0.7	0.04	0.07	0.14	0.28	0.56
Moderate / 0.5	0.03	0.05	0.10	0.20	0.40
Low / 0.3	0.02	0.03	0.06	0.12	0.24
Very Low / 0.1	0.01	0.01	0.02	0.04	0.08
Impact	Very Low / 0.05	Low / 0.1	Moderate / 0.2	High / 0.4	Very High / 0.8

High Risk	Score > 0.14
Moderate Risk	0.05 < Score ≤ 0.14
Low Risk	Score ≤ 0.05

Probability x Impact Matrix