

**Blekinge Institute of
Technology
Institution of Computer Science
PA1414
Project Plan**



Ali Reza Nazari
2022-10-28

Table of Contents

1. Project Overview	3
The project.....	3
Method	3
Budget.....	3
Timeline	6
Procedures and Routines	6
2. Requirements.....	9
Product Backlog	9
Sprint Backlog: Sprint 1	10
Sprint Backlog: Sprint 2	12
.....	12
Sprint Backlog: Sprint 3	14
.....	14
Technical Solution	15
3. Risk management.....	16
4. Test.....	17

1. Project Overview

The project

The project is about developing/creating a task scheduler as a web application that allows the user to schedule, search, and see an overview of the scheduled tasks in a single web application. Its main purpose is to simplify and make the process of task scheduling easier and more manageable. The customer for this project is a Ph.D. student at Blekinge institute of technology by the name of Waleed.

Moreover, the intended end user for the application is anyone who wishes/wants to improve their task scheduling experience and wants to put/devote their energy to finishing the tasks rather than worrying about managing them. The application is supposed to be used for scheduling, searching, and seeing an overview of the tasks. Since the application will be a web-based app, the user will be able to access it through a web browser using the customer's domain of choice. First, a task should be scheduled by typing/inserting the required information such as description, category, starting time, and deadline before the app's other features such as task overview and so on will be of any use.

Method

The agile project model with the scrum method was used to complete this project which implies that the project is adaptable/flexible, divided into sub-projects, and customer-oriented where the customer is a part of the development. To make a long story short, it was used to make the development more flexible to new changes, bringing down both the workload and making it adaptable. Furthermore, it has the customer's needs at the center which means that the customer will be a part of the project and will be able to influence the result. This will lead to a better product because by knowing the customer and their needs you can develop the product that they want.

Budget

The total budget for this project was 140 hours.

Activity	Planned hours/Effort	Actual hours/Effort	Comments
Pre-study	8	8	Reviewing lectures and external researching
Requirement Handling	4	5	Reading the requirements, creating a questionnaire, sending it to the customer, and updating the requirement list
Development Environment Set Up	1	2	Creating the file system, installing the node modules
WBS and Time Plan	4	3	Breaking the task into smaller tasks and prioritizing them
Sprint Backlogs	6	4	N/A
Customer Meetings	1	1.5	Going through the requirements for clarifications and feedback on existing development
HoD Meetings	6	6	V.37-V.42
Weekly Reports	9	7	N/A
Commitment planning and submissions	3	3	N/A
Sprint 1	30	32	Developing the application
Sprint 2	30	36	Developing the application

Sprint 3	30	30	Developing the application
Demo	3	5	V.37, V.39, and V.41
Writing Project Plan	5	6	N/A
Total	140	148.5	N/A

Activity	W36					W37					W38				
Date	2022-09-05	2022-09-06	2022-09-07	2022-09-08	2022-09-09	2022-09-12	2022-09-12	2022-09-12	2022-09-14	2022-09-15	2022-09-19	2022-09-20	2022-09-21	2022-09-22	2022-09-23
Pre-study															
Requirement Handling															
Send Questionnaire to Customer															
HoD Meeting															
Weekly Status Report															
Documenting the Progress															
Commitment 1															
Sprint 1															
Demo 1															
Commitment 2															
Sprint 2															
Demo 2															
Commitment 3															
Final Sprint															
Final Demo															
Project Plane															
Final Report															
Table 2: Timeline week36-week38 year 2022.															

Activity	W39					W40				
Date	2022-09-26	2022-09-27	2022-09-28	2022-09-29	2022-09-30	OCT 03	OCT 04	OCT 05	OCT 06	OCT 07
Pre-study										
Requirement Handling										
Send Questionnaire to Customer										
HoD Meeting										
Weekly Status Report										
Documenting the Progress										
Commitment 1										
Sprint 1										
Demo 1										
Commitment 2										
Sprint 2										
Demo 2										
Commitment 3										
Final Sprint										
Final Demo										
Project Plane										
Final Report										

Table 2: Timeline week39-week40 year 2022.

Procedures and Routines

I created and used a WBS by breaking down all the customer's requirements into user stories and tasks. Moreover, I also used a time plan for recording the total number of spent hours for each task. Finally, HoD-meeting was a weekly routine for evaluating my performance in reaching my commitments and helping me to adjust my effort and goals to reach fulfill my commitments and reach my goals.

2. Requirements

The customer had a total of 12 requirements of which 10 were implemented and delivered in three sprints. Below, you will see a detailed description of all the requirements, in a product backlog, and major deliveries in three sprint backlogs.

Product Backlog

ID	USER STORY	PRIORITY	Sprint	STATUS
RQ8	As a customer, I want the task to contain the following fields (field name: data type) so that I can export/import the tasks from/to other programs: <ul style="list-style-type: none"> • Id: integer • Description: string • Category: string (select from list) • Starting time: DATETIME • Deadline: DATETIME • Estimated duration: integer • Actual duration: integer 	1	1	Completed
RQ1	As a user, I want to schedule a task so that I can save time and simplify my TASK scheduling experience.	2	1	Completed
RQ 6	As a user, I want to keep track of the elapsed time for each task so that I prioritize and evaluate my progress.	3	1	Completed
RQ4	As a user, I want to read/see a list of all tasks and their current state (Open, in progress, done).	4	1	Completed
RQ 7	As a user, I want to have the ability to schedule tasks parallel to each other so that I increase my productivity.	5	1	Completed
RQ5	As a user, I want to have the ability to search the current task list by the following search keys: <ul style="list-style-type: none"> • Description • Category • Starting time • Deadline • Estimated duration • Actual duration so that I can access tasks quickly.	6	2	Completed
RQ 3	As a user, I want to see the tasks as a timeline so that I can get a visual aid that helps to see the start time and deadline for the tasks.	7	2	Completed
RQ 2	As a user, I want to schedule, start, and complete a task based on my capacity, current tasks, and task duration so that I can get my priorities straight.	8	3	Completed
RQ 11	As a user, I want to export the tasks as an excel sheet so that the data can be useable in different programs.	11	3	Discarded
RQ 12	As a user, I want to export the tasks as a pdf file so I can share them with others.	12	3	Discarded
RQ 9	As a customer, I want the software to process a request in under 2 seconds so that I can ensure a high-performance service to the end user.	9	3	Completed
RQ 10	As a customer, I want the software to be able to run on a system with minimum requirements: <ul style="list-style-type: none"> • a laptop with an Intel Core i7 processor (or equivalent) and 16GB of RAM 	10	3	Completed

Sprint Backlog: Sprint 1

In this sprint, requirements 8, 1, 6, 7, and 4 were committed and delivered. Note: Requirement 7 was a part of requirement 1.

ID	USER STORY	TASKS	STATUS	ESTIMATED EFFORT	DAY 1	D2	D3	D4	D5	D6	D7	D8	D9	D10
RQ8	As a customer, I want the task to contain the following fields (field name: data type) so that I can export/import the tasks from/to other programs: <ul style="list-style-type: none"> • Id: integer • Description: string • Category: string (select from list) • Starting time: DATETIME • Deadline: DATETIME • Estimated duration: integer • Actual duration: integer 	<ul style="list-style-type: none"> • Create an ER-Modell/Diagram with a table named task for the system, including the mentioned data fields. 	Completed	2	2	0	0	0	0	0	0	0	0	0
RQ1	As a user, I want to schedule a task so that I can save time and simplify my TASK scheduling experience.	<ul style="list-style-type: none"> • Create a database for storing all the task information. <ol style="list-style-type: none"> Write SQL code for the required tables. Write the relevant functions to store data in the created database • Create a file system where all the necessary files for the web app. 	Completed	20	2	4	4	4	0	4	2	0	0	0

		<ul style="list-style-type: none"> • Watch lecture videos from the database course to refresh my knowledge. • Write the JavaScript, HTML, and CSS code and link them together to create a user interface where the user can add a task based on the information stated in RQ8. • Test to verify that a task can get scheduled. 													
RQ6	As a user, I want to keep track of the elapsed time for each task so that I prioritize and evaluate my progress.	<ul style="list-style-type: none"> • Write the necessary SQL, Node.js, HTML, and CSS code so that the elapsed time updates in real-time 	Completed	2	0	0	0	0	0	0	0	2	0	0	0
RQ4	As a user, I want to read/see a list of all tasks and their current state (Open, in progress, done).	<ul style="list-style-type: none"> • Write SQL code to retrieve all the tasks from the database • Write a function in NODE.js (JavaScript) that utilizes the SQL command to get the data. • Write the necessary HTML, and CSS code to present the result to the end user. • Test to verify that all the tasks are available as a list. 	Completed	4	0	0	0	0	0	0	0	4	0	0	0

Sprint Backlog: Sprint 2

In this sprint, requirements 5, and 3 were committed and delivered.

ID	USER STORY	TASKS	STATUS	ESTIMATED EFFORT	MON	TUE	WED	THU	FRI	MON	TUE	WED	THU	FRI
RQ5	<p>As a user, I want to have the ability to search the current task list by the following search keys:</p> <ul style="list-style-type: none"> Description Category Starting time Deadline Estimated duration Actual duration <p>so that I can access tasks quickly.</p>	<ul style="list-style-type: none"> Frontend: <ul style="list-style-type: none"> a. Create an HTML form where the user can select the search parameters and the search argument and click search b. Write Node.js code inside. post router which controls which function should be called. c. Create/write the necessary functions in helpers.js for each search parameter. d. Write the corresponding SQL code for search 	Completed	10	2	4	2	2	0	0	0	0	0	0

		e. Create the ejs page which will be rendered to show the search result. <ul style="list-style-type: none"> • Work on the styling of the page • Test the feature 												
RQ3	As a user, I want to see the tasks as a timeline so that I can get a visual aid that helps to see the start time and deadline for the tasks.	<ul style="list-style-type: none"> • Brainstorm on how this functionality could be implemented. • Brainstorm on design • Sketch my design • Write CSS and HTML code to create the sketch • Write Node.js code to make the sketch responsive • Write the necessary control flow code in Node.js • Write necessary functions needed to implement the timeline • Write corresponding SQL code to extract data from the database 	Completed	14	0	0	0	2	2	4	4	2		0

[illegible]

Technical Solution

My technical solution of choice is to create a local database that stores all the tasks and create Rest API to extract the stored data from the database using node.js, express, and ejs template motor. Moreover, I will use HTML and CSS to structure and design the frontend portion of the project. While brainstorming for a technical solution, I did not consider what is best for the customer but what technical expertise I had in hand.

Tech Stack, I had worked with before:

- Maria DB
- Express.js
- Node.js
- Ejs
- HTML
- CSS
- Npm
- Cygwin
- Nodemon

Tech Stack, I had not worked with before:

- Morgan

The technical solution that I had chosen worked perfectly; it met the customer's performance requirements as well. The following worked well:

- Data storage and retrieval
- Front-end design
- Usability

Moreover, implementing the timeline was a challenge because I used vanilla JavaScript. It took 1800 lines of code to implement that functionality. If I had used a front-end framework, I could have solved it by writing 10 lines of code. If I started over, I would use a front-end framework; It would have made my life much easier.

3. Risk management

I forgot to do a risk analysis when I started the project. But if I had done one, the following risks would have happened:

- Wrong choice of technical solution

I would not have missed any risks because I am very organized and critical when it comes to both project planning and execution.

4. Test

I only conducted manual testing. Under development.

- Unit Testing
- Component Testing
- Integration Testing

My goal for unit testing was to isolate my code and test it to determine if it works as intended. In other words, I conducted unit testing to detect early flaws in code which may be more difficult to find in later testing stages.

Moreover, my goal in component testing was to find bugs at a very early stage; to make sure that a group of code works well as a component. For example, a function that retrieves and categorizes all the different task details from a user input consists of many smaller parts such as function definition, local variables, for loops, conditional statements, and so on. To ensure that the function works as intended I conducted component testing.

My goal in integration testing was to make sure that different components worked and cooperated as a group to perform a certain task such as displaying all the tasks in a timeline that has many components such as information retrieval from the database, sending the retrieved data to the right route, and so on.