

# **Group Venus**

## Project outcome and process report

Joar Heimonen      Iselin Skorpen      Salim Said  
`contact@joar.me`

Mostafa Mohammadi      Ibrar Hussain  
Hassan Ali Bokhari

June 10, 2024

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Technical background</b>	<b>3</b>
2.1	Agile development[2] . . . . .	3
2.1.1	Scrum[13] . . . . .	3
2.1.2	Scrumwise[14] . . . . .	3
2.1.3	Scrum master . . . . .	3
2.2	Development sprint[15] . . . . .	4
2.3	Backlog[4] . . . . .	4
2.4	React[12] . . . . .	4
2.4.1	Component based development[6] . . . . .	4
2.5	Vite[16] . . . . .	4
2.5.1	Pull Request[1] . . . . .	4
2.6	Burndown graph[5] . . . . .	5
2.7	Large Language Models[3] . . . . .	5
2.8	Mermaid[10] . . . . .	5
<b>3</b>	<b>Project outcome</b>	<b>5</b>
3.1	Desing sprint . . . . .	6
3.2	Development sprints . . . . .	6
3.2.1	Sprint 1 . . . . .	6
3.2.2	Sprint 2 . . . . .	6
3.3	Testing . . . . .	7
3.4	Software documentation . . . . .	7
<b>4</b>	<b>Project process</b>	<b>7</b>
4.1	Methodology . . . . .	7
4.2	Scrum . . . . .	7
4.3	Git[8] . . . . .	7

## 1 Introduction

This report aims to document the outcome of this project and the processes used to achieve this outcome.

## 2 Technical background

This section aims to describe the following terms:

### 2.1 Agile development[2]

Agile software development is software that is developed according to the ideas and values presented in the Manifesto for Agile Software Development[9]. The manifesto presents the following core ideas:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

The manifesto is based on ideas created by the Agile Alliance in 2001, this is a group of 17 software developers.

#### 2.1.1 Scrum[13]

Scrum is a type of agile methodology. The methodology was created in 1986 and widely popularized after the manifesto for agile software development was published.

#### 2.1.2 Scrumwise[14]

Scrumwise is a service for managing a scrumboard. Scrumwise claims it is "The easiest Scrum tool you'll find", while this has not been substantiated in any meaningful way the tool is used by many.

#### 2.1.3 Scrum master

The leader of a scrum project is often referred to as the scrum master. It is the scrum master's task to remove obstacles and streamline the team's development processes.

## **2.2 Development sprint[15]**

A development sprint, also known as a Timebox is the process of allocating a time constraint to reach a goal. These time constraints usually consists of a week to a month of time. Timeboxes are great for mitigating risk as per Parkinson's law[11]

## **2.3 Backlog[4]**

A backlog usually refers to an accumulation of unfinished work.

## **2.4 React[12]**

React is JS[7] framework for component based development of websites.

### **2.4.1 Component based development[6]**

Component based development also known as Component based software engineering is a method of software development that aims for components of software to be loosely coupled and reusable. Component based development is an essential part of any agile workflow.

## **2.5 Vite[16]**

Vite is a local development server aimed at developing web applications.

## **Git[8]**

Git is a free and open source version control system designed by Linus Torvalds in 2005. Torvalds created Git since he assesed that there were no existing version control tools fitting the size and activity found in Linux kernel development.

### **2.5.1 Pull Request[1]**

A Pull Request is a proposal to merge one branch with another, usually from different forks of the same repository. This project heavily relies on pull requests as this allows the scrum master to maintain the intergrity of the project.

## 2.6 Burndown graph[5]

A burndown graph is a graph that visualizes the remaining tasks in a backlog over time. This graph is used as a simple status indicator during development sprints. see *Figure ??* for an illustration of a burndown graph.

## 2.7 Large Language Models[3]

A large language model also known as an LLM is a machine learning model trained on predicting the next word in a series of words. There are many different LLMs (ChatGTP, Mixtral 7x8b, LLAMA...)

## 2.8 Mermaid[10]

The mermaid desing tool is a diagram and charting tool that was integrated into github in 2022, see *Figure 1*.

```
'''mermaid
graph TD
A[User] -- Login information --> B{Is Login information valid?}
B -- Username --> C((DATABASE))
C -- Hash + Salt --> B
B -- Valid --> D[JWT-generator]
B -- Invalid: 500 --> A
D -- Fetch JWT key --> E(secret_key.txt)
E -- Key --> D
D -- Valid JWT --> A
'''
```

Figure 1: An example of a mermaid diagram

# 3 Project outcome

During this project our group has participated in two development sprints and one design sprint.

### **3.1 Desing sprint**

During the design print we managed to develop the basic outline for the application, it was also during this design sprint that we came up with the show-it logo and name. During this sprint we faced one of the issues that would percist troughout the rest of development, this was low group participation. We ended up in a situatuion were a minority of group members were overworked while the rest were abscent. The desing sprint ended up beeing a tiring affair for the remaining group members as the sprint in total lasted for around 16 hours.

### **3.2 Development sprints**

#### **3.2.1 Sprint 1**

The goal with our first development sprint was to develop a functioning outline of the application. We started the sprint by organizing a sprint planning meeting. The sprint consisted of multiple story items needed to create the basic outline of an application that were derived during the sprint organizing meeting. Due to the fact that most of the groups members had not worked with either Git or React progress was slow. After a couple of days some of the team members managed to get quite good at Git usage and the project quickly picked up steam. Another issue during this first sprint was the lack of fitting story items during the start of the sprint, this resulted in more story items being added to the backlog during the sprint. As a result of this the burndown graph flattened out and stagnated. Even though we did not quite meet our sprint goal we were still content with the work we managed to get done.

#### **3.2.2 Sprint 2**

Due to the problems faced during the first sprint we decided on changing the goal of this sprint to producing documentation and designs for the application. This was a measure that would allow for the less technically enclined group members to also contribute, sadly this had little to no effect as group participation during this sprint was lower than the first. During this sprint we also managed to get basic navigation working on the application, a considerable time was spent ensuring that this was done in a way that did not introduce any technical debt.

### **3.3 Testing**

Due to the lack of resources experienced by this group testing had to be omitted.

### **3.4 Software documentation**

During the second sprint we were able to produce two documents detailing the project design and how access control should be implemented.

## **4 Project process**

### **4.1 Methodology**

The group methodology was simple. A sprint would be created with relevant story items and some wildcard story items (Create a report...). Certain story items were assigned to specific members, The thought was that this would ensure that everyone had atleast some manageable and simple story items to work on. When a story item was finished a pull request would be submitted to the scrum master of the project. The scrum master would then perform a code review before deciding if the pull request should be accepted or not. If a pull request was not accepted the story item would be moved from finished back to in progress. The team communicated through teams and announcements were published on scrum wise.

### **4.2 Scrum**

To manage our storyboard the team used a tool known as Scrumwise [14]. Scrumwise gave all team members great insight over what tasks were being worked on, what tasks were reserved and what tasks were finished.

### **4.3 Git[8]**

The group utilized Git as our version control system. Each time a new commit was added to the main branch a github action would compile and publish the application to github pages.

## References

- [1] *About Pull Requests*. <https://docs.github.com/en/pull-requests/collaborating-with-pull-requests/proposing-changes-to-your-work-with-pull-requests/about-pull-requests>. (Visited on 06/02/2024).
- [2] “Agile Software Development”. In: *Wikipedia* (May 2024). (Visited on 06/02/2024).
- [3] Mistral AI. *Mixtral of Experts*. <https://mistral.ai/news/mixtral-of-experts/>. Dec. 2023. (Visited on 06/09/2024).
- [4] “Backlog”. In: *Wikipedia* (Oct. 2022). (Visited on 06/02/2024).
- [5] “Burndown Chart”. In: *Wikipedia* (Apr. 2024). (Visited on 06/02/2024).
- [6] “Component-Based Software Engineering”. In: *Wikipedia* (May 2024). (Visited on 06/02/2024).
- [7] *ECMA-262*. (Visited on 06/02/2024).
- [8] *Git*. <https://www.git-scm.com/>. (Visited on 06/10/2024).
- [9] *Manifesto for Agile Software Development*. <http://agilemanifesto.org/>. (Visited on 06/02/2024).
- [10] *Mermaid — Diagramming and Charting Tool*. <https://mermaid.js.org/>. (Visited on 06/09/2024).
- [11] “Parkinson’s Law”. In: *Wikipedia* (Apr. 2024). (Visited on 06/02/2024).
- [12] *React*. <https://react.dev/>. (Visited on 06/02/2024).
- [13] “Scrum”. In: *Wikipedia* (Dec. 2023). (Visited on 06/02/2024).
- [14] Scrumwise. *Scrum Tools / Scrum Software*. <https://www.scrumwise.com>. (Visited on 06/02/2024).
- [15] “Timeboxing”. In: *Wikipedia* (Apr. 2024). (Visited on 06/02/2024).
- [16] *Vite*. <https://vitejs.dev>. (Visited on 06/02/2024).