Quantified Student

Analysis document

Jelle Maas (479146)

Koen Janssen (451394)

Jasper van den Meiracker (362948)

Neal Geilen (4173996)



# Table of Contents

[**Table of Contents**](#_i32y0w5okmr8) **2**

[**Version History**](#_bfopbgwi7tmg) **3**

[**1 Introduction**](#_mrdcpr1iu5bw) **4**

[**2 Requirements**](#_opzo5t9dtge4) **5**

[**2.1 User stories**](#_s8mfk2f7zxti) **5**

[**2.2 Non-functional**](#_h8lhac219h9) **6**

[**3 Definition of Done**](#_684qvwsbhjc7) **7**

[3.1 User stories](#_y4e56stfp46g) 7

[3.2 Sprints](#_vrqhcs95xqeb) 7

[**Conclusion**](#_fl254sla1nsf) **8**

[**Glossary**](#_vq96kva3glf0) **9**

# Version History

| Date | Version | Author | Comment |
| --- | --- | --- | --- |
| 14.04.2022 | 0.1 | J. Maas | Initial document setup |
| 19.04.2022 | 0.2 | K. Janssen | Wrote user stories |
| 19.04.2022 | 0.2 | J. Maas | Wrote user stories |
| 19.04.2022 | 0.3 | J. vd Meiracker | Wrote feedback |
| 19.04.2022 | 0.3 | J. Maas | Processed feedback |
| 20.04.2022 | 0.3 | N. Geilen | Processing feedback after talking with stakeholders |
| 21.04.2022 | 0.3 | J. Maas | Processed and wrote new feedback |
| 22.04.2022 | 0.4 | J. Maas | Wrote Definition of Done |
| 26.04.2022 | 0.4 | J. Maas | Improve introduction, write conclusion and add items to glossary |
| 26.04.2022 | 0.4 | N. Geilen | Processed feedback |
| 30.05.2022 | 0.5 | J. Maas | Processed feedback from stakeholders |
| 13.06.2022 | 0.6 | J. Maas | Clean-up document and add initial prioritisation |
| 13.06.2022 | 0.6 | K. Janssen | Reviewed the document |

# 1 Introduction

The Quantified Student (QS for short) project focuses on helping students with their development and optimising their performance with the help of collected data. The collected data will be shown in a dashboard where the student can see it. After which, the student can conclude where and how to improve their workflow. For example, the system can show when it is the best time to work for the student.

# 2 Requirements

The requirements are prioritised using the MoSCoW method:

* **Must have:** These requirements must be returned, without these requirements, the product cannot be used.
* **Should have:** These requirements are highly desirable, but the product can be used without them.
* **Could have:** These requirements will not be addressed in this project, but may be addressed in the future.
* **Will not/would not have:** These demands will not be processed.

## 2.1 User stories

The functional requirements are written down in the form of user stories. The user stories and acceptance criteria will be described according to the following template:

* **User Story:** “As a <role>, I want <feature>, so I <value>.”
* **Acceptance criteria:** “Given <input situation>, when <event>, then <outcome>.”

The acceptance criteria are only specified for reviewed and picked user stories.

| **ID** | **Priority** | **Description** |
| --- | --- | --- |
| 220601 | Must | As a student, I want to be able to view my study performance over different timeframes (e.g., week, month, year, to be discussed), so I can see how I did during that time frame. |
| 220602 | Could | As a student, I want to be able to see the biometric data in a Canvas LTI dashboard, so I have better insight in biometric data such as stress level in correlation with study events. |
| 220603 | Must | As a student, I want to be able to add and remove available data sources to and from my dashboard, so I have control of which data sources are used to determine my performance. |
| 220604 | Must | As a student, I want to be able to share my study performance with other students, so I can compare my performance with my peers. |
| 220605 | Could | As an educator, I want the application to be able to send nudges to the students, so that they can assist the student in creating better habits. |
| 220606 | Could | As a student, I want to control what notifications or suggestions are sent to me, so I can stay engaged with the dashboard and not receive unwanted notifications. |
| 220607 | Could | As a student, I want to be able to get suggestions based on my data, so I can improve my study performance in a targeted manner. |
| 220608 | Must | As a student, I want to be able to export my data, so I can get a transparent view of what data is collected about me, and so I am able to use it for my own purposes. |
| 220609 | Could | As a student, I want to be able to see which factors influence my study performance score, so I know how to improve it. |
| 220610 | Could | As a student, I want to be able to request permission from other peer students, so I can compare my study performance with my peer students. |
| 220611 | Could | As a student, I want to be able to see the screen time of my mobile devices within the dashboard, so I have a better understanding of how study performance and screen time are related to each other. |
| 220612 | Could | As a student, I want to be able to bring my own peripheral (e.g., smartwatch) for collecting relevant data for the dashboard. |
|  |  |  |
|  |  |  |

## 2.2 Non-functional

| *ID* | *Priority* | *Name* |
| --- | --- | --- |
| 220601 | Must | The application will need to be compliant with the GDPR rules. |
| 220602 | Should | The dashboard where the student can see his data must be clear and uncluttered and have a good user experience. |
| 220603 | Could | There must be a brand book with all the used colours, fonts, font sizes and images for further development in the future. |
|  |  |  |

# 

# 3 Definition of Done

A Definition of Done provides a clear description of what a delivered product backlog item should look like. It is a way to guarantee the quality of delivered products within the Scrum development methodology and provides transparency and clarity. Developers should commit to achieving the Definition of Done. For more information, please refer to <https://www.productplan.com/learn/agile-definition-of-done/>

## 3.1 User stories

* Project builds without errors
* Produced code for presumed functionalities
* Assumptions of user story is met
* Unit tests written or existing modified and passing
* Feature is tested against acceptance criteria
* Feature approved by the relevant product owners
* Browser compatibility is confirmed
* Documentation is up-to-date

## 3.2 Sprints

* Definition of Done is met for each user story included in the sprint
* All to-do items are completed
* All unit tests are passing
* All performance tests are passing
* All known bugs are fixed
* Product backlog is updated with the latest information

# Conclusion

Based on this analysis documentation and the associated knowledge acquisition, the next step will be the technical design of the software system based on the criteria laid down in this document.

Furthermore, the system is being tested by the participants of the Quantified Student. If any adjustments need to be made, it will be relatively straightforward to make changes due to the extensive documentation and planning process carried out by the team of students in charge of this document.

# Glossary

| *Term* | *Explanation* |
| --- | --- |
| QS | Quantified Student |
| DoD | Definition of Done |
| FR | Functional requirements |
| NFR | Non-functional requirements |
| PO | Product owner |
| Backlog | A build-up of work that has not been completed in a timely fashion. |
| User story | An informal, general explanation of a software feature written from the perspective of the end user or customer. |