Your Project Title

Project Requirements and Specifications

Your Sponsor



Sponsor logo (if any)

**Your Team Name & Team Logo**

(Provide a list of team members)

[Date]

**Note**: Recall that this writing assignment says:

Length = 3 pages text + appendixes as needed.

Some materials do not count towards this 3 page minimum. These excluded parts include:  
 Cover page  
 Table of contents  
 Pictures  
 Images  
 Use-case UML diagrams

Posted as a single self‐contained file (no links to outside resources.)

Posted as a PDF file to both Blackboard and your team’s Assignments GitLab repository

Typed single‐spaced.

Typed with black text.

Typed with #11 font size.

Typed using Arial font.

Typed with one inch margins on sides, top and bottom.

**(Please erase this page in your final document.)**

**TABLE OF CONTENTS**

[**I.**](#_gjdgxs) **Introduction 4**

[**II.**](#_30j0zll) **System Requirements Specification 4**

[II.1.](#_1fob9te) Use Cases 4

[II.2.](#_3znysh7) Functional Requirements 4

[*II.2.1.*](#_2et92p0) *[The name of the module/component/part] 4*

[*II.2.2.*](#_tyjcwt) *[The name of the next module/component/part] 5*

[II.3.](#_3dy6vkm) Non-Functional Requirements 5

[**III.**](#_1t3h5sf) **System Evolution 5**

[**IV.**](#_4d34og8) **Glossary 5**

[**V.**](#_2s8eyo1) **References 5**

# Introduction

Provide a brief description of your project and summarize the objectives. Yes, this is somewhat lifted from the first writing assignment, but this document needs to be self contained and so a new reader needs a 1-2 paragraph introduction to know what the project is about.

# System Requirements Specification

In this section you will describe the features, functions and other specifications that are requirements for your product. You will also specify the client/stakeholder need(s) that requirement maps to. If you find a stakeholder that your first assignment did not identify, feel free to add them here as required.

Please refer to Section 4.4 in the CptS 322 &&/|| CptS 422 textbooks like “Object-Oriented Software Engineering” to refresh your knowledge on software requirements.

## Use Cases

If applicable, provide some major use-cases that illustrate scenarios for using your product[[1]](#footnote-1). Use cases tell a story about how an end user interacts with the system under a specific set of circumstances. You may illustrate the use-cases with UML diagrams.

For each use case, identify the related requirements (you may directly refer to the requirements listed in Section II.2).

(Note that the diagrams will not be counted for the 3 pages text length specified for this document.)

## Functional Requirements

List the functional requirements in this section[[2]](#footnote-2).

Include a subsection for each main part/module of your product and list the requirements for the module in that subsection. (*Please note that we are not considering any design issues yet. Each module (subsection) refers to a major part/functionality of the product, not to sub-section in the architecture. This classification of requirements is intended to improve the readability of the document.*)  
  
Generally, functional requirements are expressed in the form "system must do <requirement>”

Briefly describe each requirement and specify the client/stakeholder need(s) that requirement maps to. Each requirement should appear in ONLY ONE sub-section of the document.

Here is an example template for requirement specification: (*the requirement template is formatted with blue for readability. Please remove the color formatting in your document*.)

### [The name of the module/component/part]

List your requirements for this project module here. Each listed requirement MAY INCLUDE the following items:

**[Enter a Concise Requirement Name]:** [provide a concise description, in clear and easily understandable language to specify the requirement]

**Source**: [specify who/what originated this requirement, specify the client/stakeholder need(s) that requirement maps to]

**Priority**: specify the priority of this requirement, you may use the following scale to specify priority of the requirements:

Priority Level 0: Essential and required functionality

Priority Level 1: Desirable functionality

Priority Level 2: Extra features or stretch goals

Repeat the above for each requirement.

### [The name of the next module/component/part]

List your requirements for the next project module here based on the requirements template described above. Include a subsection for each part.

## Non-Functional Requirements

List the non-functional requirements in this section. Non-functional requirements define system properties (e.g. reliability, response time and storage requirements, etc.) and constraints (e.g. I/O device capability, system representations, etc.)[[3]](#footnote-3)   
  
Generally, non-functional requirements take the form "system shall be <requirement>."

Process requirements may also be listed here (e.g. specifying a particular programming language or development method.) This will include any general testing plans, but there is a later assignment that will go into much greater depth about testing the product.

Please refer to Section 4.4.7 in the book “Object-Oriented Software Engineering” for example categories of non-functional requirements.

You may use the following template for non-functional requirements (Please remove the color formatting in your final document):

**[Enter a Concise Requirement Name]:**

[provide a concise description, in clear and easily understandable language to specify the requirement]

# System Evolution

As our project progresses, it is important to keep in mind the fundamental assumptions that drives our development. We must also consider the potential changes that may come up due to evolving technology, changing user expectations, and unforeseen challenges. We will be able to make wise design choices that ensures the system’s longevity, scalability, and adaptability by understanding and anticipating these changes.

Our system’s development relies on several key assumptions about the hardware, software, and user requirements. To start, our current technology stack, which consists of primary Python as well as MongoDB, and Qualtrics, will continue to be maintained and developed in a manner that keeps it compatible with our project. Python Anywhere is a reliable platform for hosting applications and we expect this reliability to continue. Additionally, we anticipate that the end user’s devices will be able to meet the web-based application’s requirements in terms of processing speed and picture quality. The system is designed prioritizing ease of use, as we believe users will need intuitive and easy access to their personality assessment data and daily emotional tracking.

We anticipate several technological and user-driven changes over time that may have an impact on the system. One of which is that as technology evolves, so will cyber threats. Since our system handles sensitive personality assessment data that will be used for research, it must adhere and abide by strict data protection standards. As for the user side, we anticipate that user demands will change as more people use digital tools to manage their mental health. The use of artificial intelligence (AI) may make users demand more personalized and immediate feedback.

We must also consider several risk points that may come up throughout the development of our project. The integration of Qualtrics and the web-based app may not scale well as the number of users increases dramatically. The system now, uses relatively simple algorithms that process and display personalized data. As more users interact with the app, performance issues with data handling and processing could arise.

The app’s current clustering feature that uses a simple rule-based algorithm to determine “functionally equivalent” emotional events is another potential risk. As we explore the use of ChatGPT and other models that could be pre-trained and fine-tuned to fit this project, there is a chance that these approaches will not work as planned. They could also introduce complications and complexity that are difficult to handle in the current system. If there are changes in availability or cost, it will require us to adapt quickly or look for other options.

To address these risks, we are implementing several precautionary measures to mitigate these risks. First, we need to enhance and manage the system with scalability in mind. This is to ensure the system can handle increased user traffic without significant performance issues. As for integrating AI, we will monitor AI technologies and remain flexible in our approach. If ChatGPT or some other model becomes too costly or unavailable, we will investigate other alternative clustering methods. This may include refining the current clustering algorithm or implementing open-source AI tools. Finally, we will create modular system components that will allow us to upgrade or swap out certain features. Doing so will not require a complete redesign of the system.

Our proactive approach to anticipating and adjusting to these changes will be vital as our project progresses. By keeping a close eye of the assumptions underlying our system’s design, and continuously evaluating potential risks and opportunities for improvement, we ensure that we are well prepared to deliver a robust and secure application. This approach allows us to remain flexible and responsive which sets up the project for long-term success and stability.

Traceability Matrix

The table below aligns functional requirements with their corresponding use cases and user stories. This ensures that all requirements are accounted for and directly connected to specific user scenarios.

|  |  |  |  |
| --- | --- | --- | --- |
| Functional Requirements | Use Case | User Story | Priority |
| FR-1: PICA Assessment Results PDF | UC-3: Generate and Send PDF Report | US1: As a user, I want my PICA results PDF to be automatically sent to my email | Level 0 |
| FR-2: Automatic PDF Sending | UC-3: Generate and Send PDF Report | US1: As a user, I want my PICA results PDF to be automatically sent to my email | Level 0 |
| FR-3: Data Collection | UC-7: Collect User Feedback | US6: As an admin, I want to be able to keep user’s data in a database | Level 0 |
| FR-4: PICA Assessments Results in Phone App | UC-6: Import Personality Assessment Results | US5: As an admin, I want to be able to import a User’s PICA results into the phone app | Level 0 |
| FR-5: PICA-Based Questions | UC-4: Log Emotional Event | US4: As an admin, I want the PDF results of the PICA to be optimized | Level 0 |
| FR-6: Clustering Algorithm | UC-5: View Functionally Equivalent Situations | As a user, I want to see functionally equivalent situations | Level 2 |
| FR-7: Data Collection | UC-4: Log Emotional Event | US6: As an admin, I want to be able to keep user’s data in a database | Level 0 |

# Glossary

Define technical terms used in the document.

# References

Cite your references here.

For the papers you cite give the authors, the title of the article, the journal name, journal volume number, date of publication and inclusive page numbers. Giving only the URL for the journal is not appropriate.

For the websites, give the title, author (if applicable), date accessed, and the website URL.

Please use either IEEE or Chicago format for your references. Keep in mind that Google Scholar or BibTeX can help you easily format your citations for periodicals and journals.

1. https://en.wikipedia.org/wiki/Use\_case [↑](#footnote-ref-1)
2. https://en.wikipedia.org/wiki/Functional\_requirement [↑](#footnote-ref-2)
3. https://en.wikipedia.org/wiki/Non-functional\_requirement [↑](#footnote-ref-3)