

System Design for Software Eng 4G06

Team 2, Parnas' Pals

William Lee

Jared Bentvelsen

Bassel Rezkalla

Yuvraj Randhawa

Dimitri Tsampiras

Matthew McCracken

January 15, 2023

1 Revision History

Date	Version	Notes
January 13 2023	1.0	Initial Draft

2 Reference Material

This section records information for easy reference.

2.1 Abbreviations and Acronyms

symbol	description
Software Eng 4G06 [... —SS]	Explanation of program name [... —SS]

Contents

1	Revision History	i
2	Reference Material	ii
2.1	Abbreviations and Acronyms	ii
3	Introduction	1
4	Purpose	1
5	Scope	1
6	Project Overview	2
6.1	Normal Behaviour	2
6.2	Undesired Event Handling	2
6.3	Component Diagram	3
6.4	Connection Between Requirements and Design	3
7	System Variables	3
8	User Interfaces	3
9	Design of Hardware	3
10	Design of Electrical Components	4
11	Design of Communication Protocols	4
12	Timeline	4
A	Interface	5
B	Mechanical Hardware	5
C	Electrical Components	5
D	Communication Protocols	5
E	Reflection	5

List of Tables

List of Figures

1	System Context Diagram	2
---	----------------------------------	---

3 Introduction

[Include references to your other documentation —SS]

This document serves to illustrate and explain design decisions (in the context of alternatives) and the thought processes and considerations of the team that made said decisions.

4 Purpose

[Purpose of your design documentation —SS]

[Point to your other design documents —SS]

The purpose of this design documentation is to make clear current design choices, and to show why these designs were chosen over possible alternatives. This document also must prove that the current designs fulfill the requirements enumerated in the Software Requirements Specification (SRS) document.

5 Scope

[Include a figure that show the System Context (showing the boundary between your system and the environment around it.) —SS]

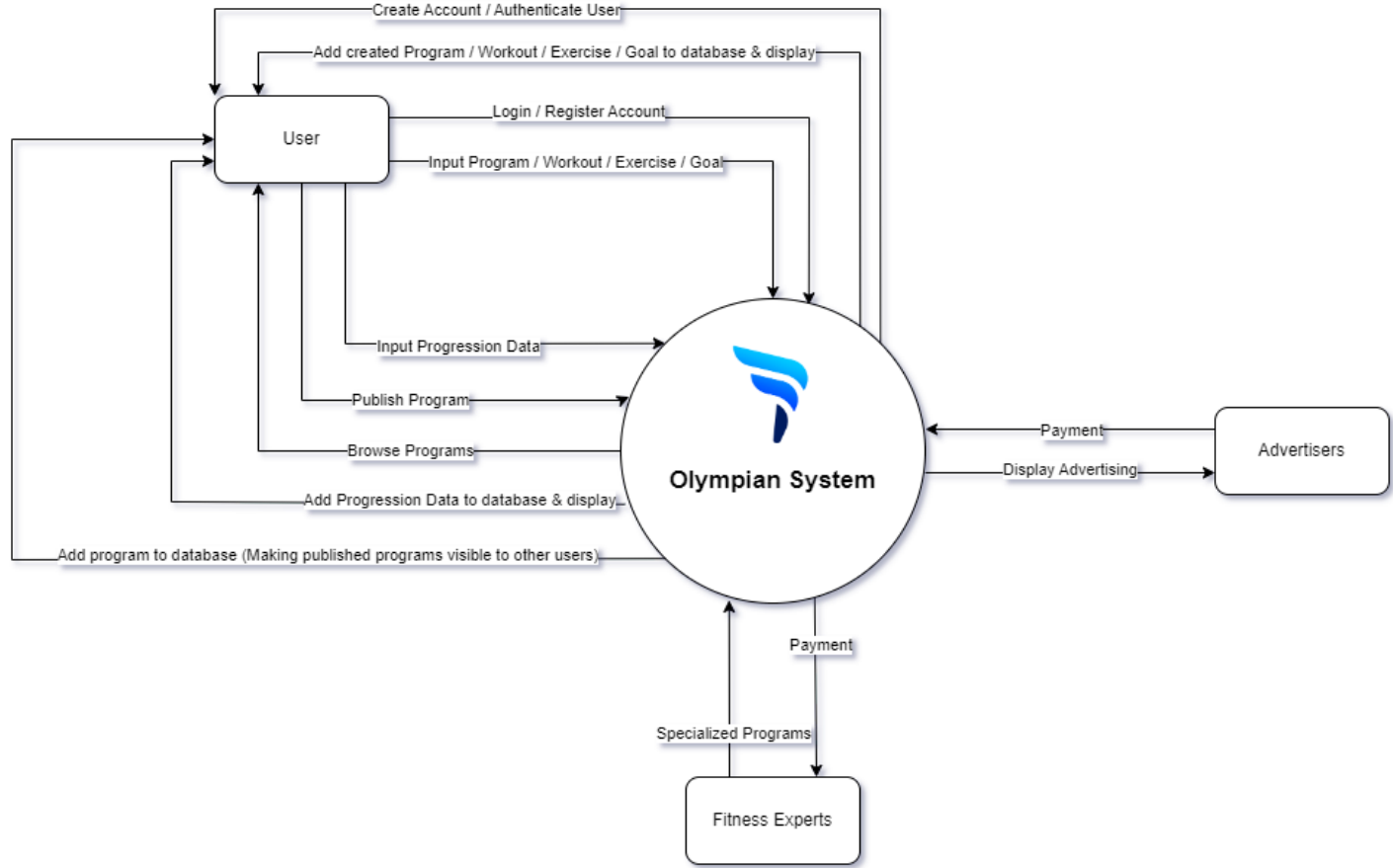


Figure 1: System Context Diagram

6 Project Overview

6.1 Normal Behaviour

Normal behaviour for the application can be defined as the state of the available program when all FRs and NFRs are fulfilled. If an FR or NFR is not fulfilled, this is abnormal behaviour, and indicative of the occurrence of an undesired event.

6.2 Undesired Event Handling

[\[How you will approach undesired events —SS\]](#)

An ‘undesired event’ can comprise anything from incorrect user input to database failure to connectivity issues.

Invalid User Input: Whenever a user provides incorrect information to a form (e.g. an invalid email address during sign-up), an informative error message is displayed beneath the incorrectly filled field, and a visual queue is given in the form of highlighting the field

in red to indicate error. To prevent annoying the user with error messages before they have actually made an error, forms will only display error messages after the first submit attempt. After an initial submit attempt, the error message will remain until the contents of the field are valid. This is to give the user information as to whether or not their entry is valid before having to submit again.

Connection/Database Failure: If the application is unable to connect to the internet, and subsequently the server and database, an informative error message is displayed, informing the user to verify their internet connection. Note that a stretch goal for the future is to have the Olympian application save certain information locally to enable the user to perform certain functions offline. Changed local information would be synced with the database once connection is re-established. However, this is a stretch goal, and current requirements and designs list the database as the only information store, no data (other than an authentication token) is stored locally. If the database/server itself is failing, an appropriate message is displayed to the user. The message will indicate that the failure is not the fault of the user, but rather an internal failure outside of their control.

Error messages are kept consistent and concise in format, to make errors easy for the user to recognize and act upon.

6.3 Component Diagram

6.4 Connection Between Requirements and Design

[The intention of this section is to document decisions that are made “between” the requirements and the design. To satisfy some requirements, design decisions need to be made. Rather than make these decisions implicit, they are explicitly recorded here. For instance, if a program has security requirements, a specific design decision may be made to satisfy those requirements with a password. —SS]

7 System Variables

[Include this section for Mechatronics projects —SS] N/A

8 User Interfaces

[Design of user interface for software and hardware. Attach an appendix if needed. Drawings, Sketches, Figma —SS]

9 Design of Hardware

[Most relevant for mechatronics projects —SS] [Show what will be acquired —SS] [Show what will be built, with detail on fabrication and materials —SS] [Include appendices as appropriate, possibly with sketches, drawings, CAD, etc —SS]

Olympian will run on mobile devices (e.g. iPhone, Samsung Galaxy) not designed or manufactured by the team.

10 Design of Electrical Components

[Most relevant for mechatronics projects —SS] [Show what will be acquired —SS] [Show what will be built, with detail on fabrication and materials —SS] [Include appendices as appropriate, possibly with sketches, drawings, circuit diagrams, etc —SS]

N/A - See Section 9.

11 Design of Communication Protocols

[If appropriate —SS]

12 Timeline

[Schedule of tasks and who is responsible —SS]

A Interface

[Include additional information related to the appearance of, and interaction with, the user interface —SS]

B Mechanical Hardware

N/A

C Electrical Components

N/A

D Communication Protocols

The project employs Hyper Text Transfer Protocol (HTTP), enabling the client to communicate with the server and database through HTTP requests.

E Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Problem Analysis and Design. Please answer the following questions:

1. What are the limitations of your solution? Put another way, given unlimited resources, what could you do to make the project better? (LO_ProbSolutions)
2. Give a brief overview of other design solutions you considered. What are the benefits and tradeoffs of those other designs compared with the chosen design? From all the potential options, why did you select documented design? (LO_Explores)