

CS22120: Software Engineering Group Project 10

Design specification

Author: Juan [jmp16]

Conf. Ref: DesignGroup10_doc

Date: 16 February 2021

Version: 0.2

Status: Draft

Department of Computer Science Aberystwyth University Aberystwyth
Ceredigion

SY23 3DB

Copyright © of Aberystwyth University 2020

Contents

1 INTRODUCTION	1
1.1 PURPOSE OF THIS DOCUMENT	1
1.2 SCOPE	1
1.3 OBJECTIVES	1
2 DECOMPOSITION DESCRIPTION.....	1
2.1 PROGRAMS IN SYSTEM.....	1
2.2 SIGNIFICANT CLASSES IN EACH PROGRAM.....	1
2.2.1 <i>Significant classes in Program 1</i>	1
2.2.2 <i>Significant classes in Program 2</i>	<i>Error! Bookmark not defined.</i>
2.3 MODULES SHARED BETWEEN PROGRAMS	2
2.4 MAPPING FROM REQUIREMENTS TO CLASSES	2
3 DEPENDENCY DESCRIPTION.....	3
3.1 COMPONENT DIAGRAMS	3
3.1.1 <i>Component Diagram for program 1</i>	3
3.1.2 <i>Component Diagram for Program 2</i>	3
4 INTERFACE DESCRIPTION	3
4.1 CLASS 1 INTERFACE SPECIFICATION	3
4.2 CLASS 2 INTERFACE SPECIFICATION	3
5 DETAILED DESIGN	3
5.1 SEQUENCE DIAGRAMS	3
5.2 SIGNIFICANT ALGORITHMS.....	3
5.3 SIGNIFICANT DATA STRUCTURES	3
6 REFERENCES	3
7 DOCUMENT CHANGE HISTORY	3

1 Introduction

1.1 purpose of this document

The purpose of this document is to provide the Design Specification which describes the different system components and how they fit together to accomplish the application requirements.

1.2 Scope

This document provides an accurate translation of the requirements into a clear description of the program components and the relationships between them.

The document should be read by all the group members who have any relation to the Design Specification document [2]. It is assumed the reader is already familiar with the introductory QA document [1] and the Operating Procedures and Configuration Management Standards document [3].

1.3 Objectives

The objective of this document is to provide a system description of the following aspects:

- Decomposition description: Programs and modules that make up the system, along with requirements that meet
- Dependency description: Relationships between system components
- Interface description: Information required to use the facilities provided by a module
- Detailed design: Further details for the modules whose internal workings are not self-evident).

2 Decomposition description

2.1 Programs in system

The entire workout application will be developed within the homeExcerciseApp directory that can be found in the project repository.

One program or many programs for the entire app? is the homeExcerciseApp directory the one to be used? separate program for the XML thing?

2.2 Significant classes in each program

2.2.1 Significant classes in Program 1

The program will be made up using the following classes, which will be divided into three main categories:

Main functionality:

- **App:** It initialises the program and contains the main logic to start-up workouts. This class serves as a bridge between the persistent storage using XML and the classes modelling Exercises and Workouts.

- **WorkoutActivity:** Allows us to keep track of the user workout history. It binds some interesting performance facts as the rest time or the number of exercises with the date on which the workout was performed.

Exercises and workouts:

- **Exercise:** This class models a typical exercise with the required parameters. An enumeration will be used to specify to which stage of the workout this exercise can be part of (Warm-up, Working exercises, Cold Down).
- **WorkoutConfig:** Workouts with their required parameters will be represented using this class. An enumeration will be used to differentiate them into 4 categories (Beginner, Advanced, LIIT, HIIT and Custom). This classification will be used to attend to the different users described in the User Interface Specification [4].

Persistent storage:

- **ExerciseAndWorkouts**
- **WorkoutRead**
- **XMLio<abstract>**

2.3 Modules shared between programs

Any classes shared between programs?

2.4 Mapping from requirements to classes

FR4 to FR8 - should these requirements be mapped with the ConfWorkout class? any other classes needed?

<i>Requirement</i>	<i>Classes providing requirement</i>
FR1, FR2, FR3	App
FR4, FR5, FR6, FR7, FR8, FR9	WorkoutConfig
FR10	WorkouotActivity

3 Dependency description

3.1 Component Diagrams

3.1.1 Component Diagram for program 1

3.1.2 Component Diagram for Program 2

4 Interface description

4.1 Class 1 interface specification

4.2 Class 2 interface specification

5 Detailed design

5.1 Sequence diagrams

5.2 Significant algorithms

5.3 Significant data structures

6 References

[1] QA Document SE.QA.01 – Quality Assurance Plan.

[2] QA Document SE.QA.05 – Design Specification Standards.

[3] QA Document SE.QA.08 – Operating Procedures and Configuration Management Standards.

[4] Software Engineering Group Project 10 - User Interface Specification

7 Document Change History

<i>Version</i>	<i>CCF No.</i>	<i>Date</i>	<i>Changes made to document</i>	<i>Changed by</i>
0.1	N/A	23/02/21	Created layout; Intro section completed.	jmp16
0.2	N/A	10/03/21	Intro modified; Decomposition description nearly done.	jmp16

