A drawing of a face

Description automatically generated

ICT2101/2201 Introduction to Software Engineering

Software Requirements Specifications (SRS)

Milestone 1

AY2021/2022, Trimester 1

<Project Title>

PLAUSIBLE REFERENCES:

<https://elc.polyu.edu.hk/inscribe/wp-content/uploads/2017/11/i3Project-Proposal.pdf>

Lab group P3-8

<Team’s Name>

[Following Roles are TBC]

Team Leader: Loh Hui Qi (2001718)

<Role 2>: Leong Jin Kuan, Joshua (2002133)

<Role 3>: Muhammad Alam Sah Bin Syed Hamzah (2001776)

<Role 4>: Tse Kin Ping (2001568)

# Introduction

1. Guide user (kids) to design and build their own challenges while teaching critical problem-solving and computational thinking skills.
   1. Is this up to us? What kind of computational thinking skills are they looking for?
2. A web-based portal in a language of your choosing
   1. Dashboard that reflects the various data from the robot’s sensors.
   2. Should the users be able to customise the dashboard?
   3. What is the design theme?
3. User should be able to observe the robotic car actions reflect in the portal.
   1. Realtime?
   2. What sort of actions should the users be able to perform?
4. User should be able to provide a list of commands to the robotic car.
   1. Provide a list of commands…? As in Block Based Programming? Or what is the intended vision?

## 1.1 Product Scope

*<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. TO DO: 1-2 paragraphs describing the scope of the product. Make sure to describe the benefits associated with the product.>*

To teach users of various ages computational thinking through gamification and subsequently, block-based programming. In learning to manipulate the robot car, users will be taught to think and plan in a logical manner. It will enable users to be better problem solvers as they navigate challenges.

## 1.2 Intended Audience and Document Overview

*<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers (In your case it would probably be the “client” and the professor). Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>*

## 1.3 Definitions, Acronyms and Abbreviations

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.*

*TO DO: Please provide a list of all abbreviations and acronyms used in this document sorted in alphabetical order.>*

## 1.4 Document Conventions

*<In general, this document follows the IEEE formatting requirements. Use Arial font size 11, or 12 throughout the document for text. Use italics for comments. Document text should be single spaced and maintain the 1” margins found in this template. For Section and Subsection titles please follow the template.*

TO DO: Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. Sometimes, it is useful to divide this section to several sections, e.g., Formatting Conventions, Naming Conventions, etc.>

# Overall Description

This is the paragraph content of your section. Include any background research or literature research you have done here.

Other various robots

|  |  |
| --- | --- |
| Cozmo Robot by Anki | Visual Programming mode to teach kids how to code  Kids are allowed to drag and drop functions on Code Lab to get Cozmo to:   * Forward & Backward movement * Identify Faces * Pick up blocks   Basically a lot of preprogramming. The app offers different challenges to teach users the foundations of basic programming.  Eg. When SMILE is SEEN, DRIVE one side of a SQUARE and TURN  Then it shows the challenge answers in a visual format to the children    It is built off Scratch Programming language, using Python.  Its capabilities include facial recognition, name recognition, color, count items, path mapping.  How do you provide the user of feedback of whatever the toy is seeing? That allows for further interaction between the user and the device.  Games that are available include, quick tap, keepaway, memory match, etc. |
| Fisher-price - Code a Pillar Twist | Twist the dials on the caterpillar’s segments to create a sequence to send the bot along the programmed path.  Teaching points:   * REASONING & PROBLEM SOLVING   Programming a path for the bot to go will teach the children how to figure out how to get their pal to go wherever they want.   * PLANNING AND SEQUENCING   Twisting the segments will teach the children that if they program it in the correct sequence, they can get the bot to move the way they want to.   * CRITICAL THINKING   Trying again and again… teaches them how to think critically..? It allows the bot to make music and such. |
| SPHERO Mini Robot Ball | Allows facilitation by parents to play STEM games via a Play App:  Scratch Block Based Coding or JavaScript  Drive Mode: Control the robot using a joystick in the Sphero Play App   * Parents Guide * Account Creation * Class Code   Programming Levels: |

## 2.1 Product Overview

* *<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. In this part, make sure to include a simple diagram that shows the major components of the overall system, subsystem interconnections, and external interface. In this section it is crucial that you will be creative and provide as much information as possible.*
* *TO DO: Provide at least one paragraph describing product perspective. Provide a general diagram that will illustrate how your product interacts with the environment and in what context it is being used. This is not a formal diagram, but rather something that is used to illustrate the product at a high level.>*

## 2.2 Product Functionality

* *<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary is needed here. These can be at the level given in the project description.>*
* *TO DO:*
* *1. Provide a bulleted list of all the major functions of the system*

# Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

* *<Describe the logical characteristics of each interface between the software product and the users. For your project, you only need to be concerned with the main thermostat (not the mobile app) and can use the graphic from the project description as the basis for your user interface..*
* *TO DO: Provide the graphic for the thermostat user interface and provide a basic description as to how users will interact (e.g., tough screen, menus, etc.).>*

### 3.1.2 Hardware Interfaces

* *<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware. You are not required to specify what protocols you will be using to communicate with the hardware, but it will be usually included in this part as well.*
* *TO DO: Please provide a short description of the different hardware interfaces. This can simply be a list of the devices you must interact with at this point. For sensors, you can assume that each has a basic “read” interface to get the current values. Each sensor in this system uses English units.>*

### 3.1.3 Software Interfaces

* *<Describe the connections between this product and other specific software components (in your case, just the mobile app that can send commands).>*

## 3.2 Functional Requirements

< Functional requirements capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. This section is the direct continuation of section 2.2 where you have specified the general functional requirements. Here, you should list in detail the different product functions. >

**F1: The system shall …**

**F2: The system shall…**

…

## 3.3 Use Case Model

* *TO DO: Provide a use case diagram that will encapsulate the entire system and all actors.*

### 3.3.1 Use Case #1 (use case name and unique identifier – e.g. U1)

*TO DO: Provide a specification for each use case diagram*

**Author –** Identify team member who wrote this use case

**Purpose** - What is the basic objective of the use-case. What is it trying to achieve?

**Requirements Traceability –** Identify all requirements traced to this use case

**Priority** - What is the priority. Low, Medium, High. Importance of this use case being completed and functioning properly when system is depolyed

**Preconditions** - Any condition that must be satisfied before the use case begins

**Post conditions** - The conditions that will be satisfied after the use case successfully completes

**Actors** – Actors (human, system, devices, etc.) that trigger the use case to execute or provide input to the use case

**Extends –** If this is an extension use case, identify which use case(s) it extends

**Flow of Events**

* 1. Basic Flow - flow of events normally executed in the use-case
  2. Alternative Flow - a secondary flow of events due to infrequent conditions
  3. Exceptions - Exceptions that may happen during the execution of the use case

**Includes** (other use case IDs)

**Notes/Issues** - Any relevant notes or issues that need to be resolved

### 3.3.2 Use Case #2

…

# Other Non-functional Requirements

## 4.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.

TODO: Provide performance requirements based on the information you collected from the client/professor. For example, you can say “P1. The secondary heater will be engaged if the desired temperature is not reached within 10 seconds”>

## 4.2 Safety and Security Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied. Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements.

TODO:

• Provide safety/security requirements based on your interview with the client - again you may need to be somewhat creative here. At the least, you should have some security for the mobile connection.

# Work Distribution & Plan

You can include the type of SDLC your team uses here. Remember to justify why your team employ such a SDLC approach.

# Conclusion

Summarise the project and highlight the key features and timeline required.

# Individual Reflections

# References

Some references here

# Appendix A – Data Dictionary

<Data dictionary is used to track all the different variables, states and functional requirements that you described in your document. Make sure to include the complete list of all constants, state variables (and their possible states), inputs and outputs in a table. In the table, include the description of these items as well as all related operations and requirements.>

# Appendix B