# Criteria 4 - Software Requirement Specification

|  |
| --- |
| **Instructions**  Criterion 4 assesses students’ skills in documenting a software requirements specification. Students will document the functional and non-functional requirements, constraints and scope as well as the technical environment and the intended audience of the software solution.  Students will document evidence of their critical and creative thinking through the identification, clarification and critical analysis of the data collected as part of the Analysis Stage in Criterion 3 and 4.  **Evidence**  Intended to be reviewed and signed off by your client, the following sub-sections outline what has to be completed as evidence for Criteria 4. |

## 4.1 - Client Brief

|  |
| --- |
| **Instructions**  Update your *Need or Opportunity Summary*, taking into account what you have discovered in your research for Criteria/Section 3.  **Evidence**  This should be a **formal** (150 - 200 word) outline of the software that is going to be supplied, it’s purpose, functionality and scope. In your outline, you should highlight the various user classes |

All data that was gathered from criteria 3 was the method of collection, users of the program and how the information is made into data then to the required data necessary for the solution. All data will be gathered from Interviews (To have a detailed idea of what the client is requesting for and collect data on the area’s information is needed), Observations (To observe and get a better understanding of the organization and note certain details that are important to the solution), and surveys (To get small bits of info to get the basic data). There will be 3 users that will be contributing to the use of the program, a new member will join and provide information then a admin will use this to create the data necessary for a account and then the new member will be assigned as a Staff member. With this account the admin will be able to view, data like the primary solution, hours done by each member. The function of being able to see the total hours done by a staff member will be the primary purpose, functionality, and scope.

*2 paragraphs, 150 words, 1004 bytes of Lorem Ipsum*

## 

## 4.2 - User Classes

|  |
| --- |
| **Instructions**  Identify and define the users of the software.  **Evidence**  Provide a summary of the expected end-users of the solution. Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on the frequency of use, a subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **User class** | **Frequency of Use** | **Product Functions Used** | **Technical Expertise** | **Security Level/Privileges** | **Educational Level** | **Experience/Skills** |
| Staff, Admin | Throughout the week (Mon-Fri) | To store data and collect and create data | Only expertise will be for the admin to be able to keep the program updated and updating files | Normal staff can access the certain functions of the program when ever and the admin can access all functions of the program | No education needed for staff to use. Admin will need experience in coding to be able to properly use the program | Staff will need non. Admin will have to be able to have an idea in coding to do any bug fixes. |

## 4.2 Items In Scope

|  |
| --- |
| **Instructions**  The boundaries or parameters of the solution. |

Must include features such as:

Program must be able to calculate the hours done

Program must be able to encrypt and decrypt data files

Program must be able to records the time staff come in and out

Program must be able to store and hold staff data

Program can be edited by a admin

Users can be added and removed

## 4.3 Constraints

|  |
| --- |
| **Instructions**  Describe any items or issues that will limit the options available to the developers. Because of access to restricted resources features like RFID will not be able to be implemented into the program and instead and password and username will be used. Also because of the coding language that is being used the program will not be able to be used on mobile iOS devices. |

Windows 10

Must be completed within 6 months

Must be coded using visual basic

Restricted access to resources such as RFID

## 4.4 Operating Environment

|  |
| --- |
| **Instructions**  Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must exist.  An updated, detailed diagram of the operating/network environment could be included. |

### 

No network diagram is necessary as it is on a standalone device

The program will be used in an environment that will have frequent use during the morning and evening by all staff. It will be displayed on a pc monitor, with windows 10, around the entrance of the Childcare centre for easy use.

## 4.5 Functional Requirements

|  |
| --- |
| **Instructions**  Itemize the detailed functional requirements. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Each requirement should be uniquely identified with a sequence number.  Add additional rows until all functional requirements are listed. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Requirement** | **Input** | **Process/Sequence** | **Output** | **Notes/Comments** |
| FR1 | Display hour done | Time in and out of work | Use time checked in and out to calculate hours done by each staff member | Hours done by each staff member |  |
| FR2 | Store Staff data | Staff Information | Use information to create account | Account is assigned to appropriate staff member for use |  |
| FR3 |  |  |  |  |  |

## 4.6 Non-functional Requirements

|  |
| --- |
| **Instructions**  Itemize the non-functional requirements such as user-friendliness, response rates, robustness, portability, reliability and maintainability. Write these to be specific, quantitative, and verifiable when possible. Each requirement should be uniquely identified with a sequence number.  Add additional rows until all non-functional requirements are listed. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Requirement** | **Input** | **Process/Sequence** | **Output** | **Notes/Comments** |
| NFR1 | User friendly | Easy to understand simple design | Little information to reduce clutter on screen | Easy to read instructions |  |
| NFR2 | Maintainability | Quick access to files | Files only have to be decrypted to be able access and edit | Data and information of files to be edited program |  |

# 