Software Requirements Specification

for

C^3 (Crafting Cost Calculator)

Version 1.0

Prepared by

Alif Muqri Bin Hazmi	**2018413956**
Nurul Farhanah Binti Fazid	**2018439772**
Syakir Zufayri Bin Azhar	**2018223854**
Mohammad Amirul Luqman Bin Shamshi	**2018298154**

Instructor: Dr Nur Atiqah Sia Binti Abdullah

Course: CSC577

Date: 28/4/2020

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Code Crunchers	28/4/20	Initial version created for review by mentor.	1.0
Code Crunchers	10/5/20	First review of full system requirement specification	1.0

1. Introduction

1.1 Purpose

The purpose of this document is to provide a detailed and complete specification of Crafting Cost Calculator (C^3). It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. This document is intended for users and shareholders.

1.2 Document Conventions

This document was created based on the IEEE template for System Requirement Specification Documents.

1.3 Intended Audience and Reading Suggestions

This document is intended for Team CodeCrunchers, future user of C^3 and instructor for CSC577 – Software Engineering : Theories and Principles. This document will have information that may be deem pertinent for each member of the intended audience.

This document is arranged in 6 sections. The first section is introduction, which summarizes about the whole document. The second section is overall description, which tells about the functionality of the software. The third section is external requirements, which gives reader information about user interfaces. The fourth section is system features, which tells about the functional requirements that are in the system. The fifth section is other non-functional requirements, which includes any non-functionals that are in the system. The last section is analysis model, which provides that use case diagram of the system.

1.4 Product Scope

C^3 is a software developed to help software developers to calculate and determine the cost of their projects. The program aims to help developers to decide the cost that needed to be charged to their clients, how many members required to deliver the system, and how many months required to complete it.

It is a daily struggle for developers, especially freelancers to decide how much they want to charge their clients. Sometimes they charge it low expecting that it can be delivered easily or maybe they charge it too high until their clients feel that it is not affordable. With this system, users can reduce the risk of being paid lower or not enough than the overhead cost and time used developing.

This system also can calculate how many members that you actually need to deliver the system efficiently. This can prevent developers from being over-worked and can lead to delivering a poor system that need to be fixed.

In addition, the system can also tell the time required to complete it. This can avoid developer from giving dateline that is not possible to achieve thus causing unwanted delays which makes client feel that you are not serious in developing. Not all clients understand that developing a system takes a lot of time. Therefore, having a proper time estimation is crucial.

1.5 References

- https://eu-smartcities.eu/sites/default/files/201709/EIP_RequirementsSpecificationGLA_%20V2-5.pdf
- http://groups.umd.umich.edu/cis/course.des/cis375/projects/fp99/main.html
- https://www.javatpoint.com/cocomo-model
- https://ufuture.uitm.edu.my

2. Overall Description

2.1 Product Perspective

C³ was developed for everyone who are working on a project that require a number of months and cost. It can calculate the cost of the production required to make the project and the duration for the project to be done. Other than that, it also will estimate the number person that required to develop the project.

C³ also is an open source project and have a very active developer team to support and provide feedback to users and to ensure that it will run smoothly and effectively. It was developed to run on Windows.

2.2 Product Functions

Main Page

- Start button – It will lead the user to the next section.

Function Point Calculator Page

- Home button – It will lead the user to the Main Page
- Reset button – It will clear the table if the user enter the wrong input and want to replace the values
- Next button – It will lead the user to the next section
- Help button – To help users to understand the meaning of each meaning of the input.
- Error message – Will pop-up went the user did not enter any value in the table.

Counting Cost Calculator Page

- Home button – It will lead the user to the Main Page

- Next button – It will lead the user to the next section
- Range button – User need to press the right button exactly to their project

Salary and Type of Application.

- Home button – It will lead the user to the Main Page
- Next button – It will lead the user to the next section
- Help button – To help users to understand the meaning of each meaning of the input.
- Selection button – User need choose their corresponding type of application

Result Page

- Exit button - The result will be saved in file and the user will go back to the welcome section.

2.3 User Classes and Characteristics

• General users, such as students, especially for the Final Year Student that have Final Year Project (FYP) that need them to make a project. The project may be an application or something else. Thus, C³ will help them for the project.

• Professional users, such as Software Engineer. C³ will help them with their work. It will give all the details such as required worker, estimated time needed and more.

• Programmers that are interested on making their own application or project. Will help them manage their time needed to finish their project or application.

2.4 Design and Implementation Constraints

C³ is developed in Java and has been developed on the NetBeans platform. It will use Java Web extension to build it.

2.5 User Documentation

A help button will be at all the certain page to clarify the user about the instruction given. Other than that, all of the function buttons are easy to understand as we made the application to be user-friendly. This will make it easier for other people to understand and continue development on our product if needed.

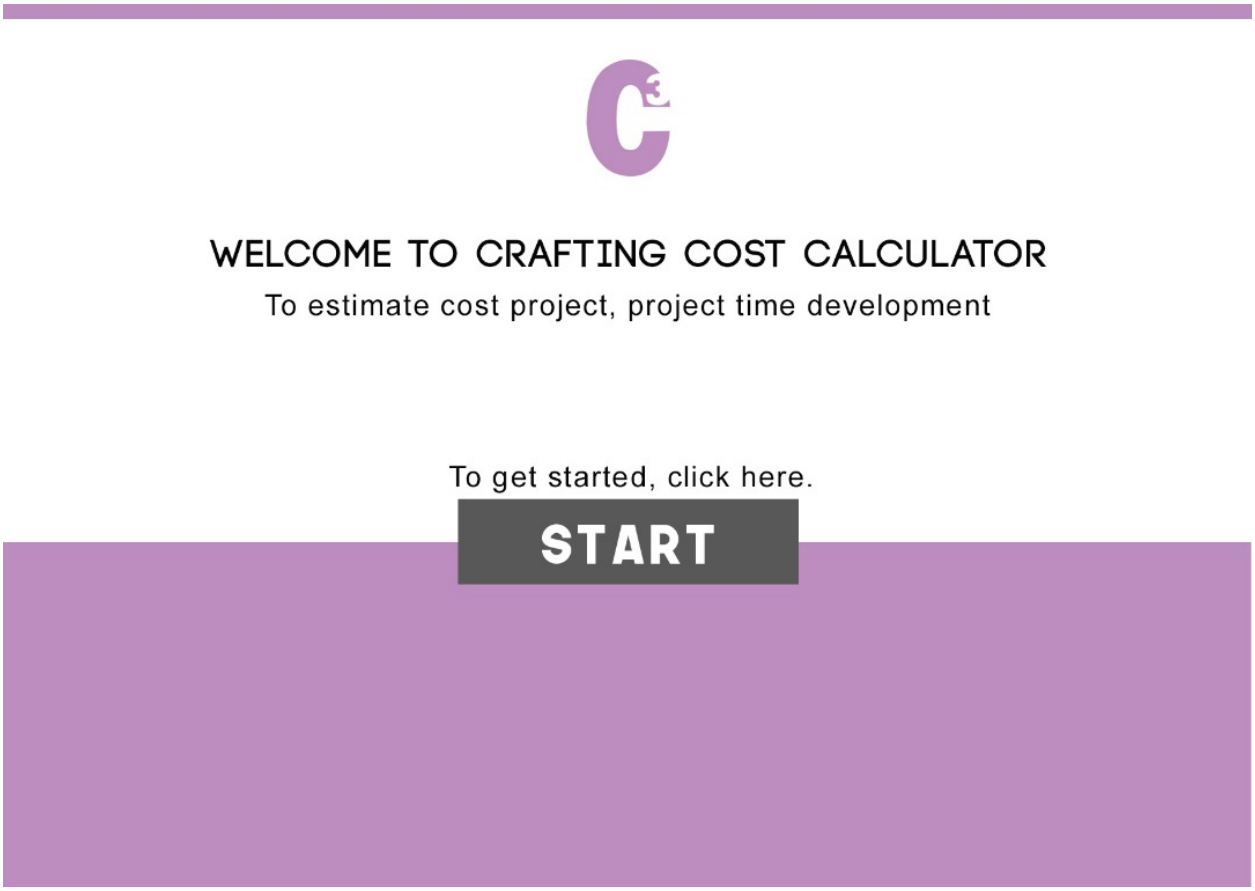
2.6 Assumptions and Dependencies

We are assuming that users have the data that required for fields that are requried to be filled in the system. We are also assuming that the user has some hardware to run our module, which will consume storage and CPU to store and analyze data. We are also assuming that all user have the basic windows such as Windows & to run our application. Finally, we are also assuming that all the user have enough RAM to run this application smoothly.

3. External Interface Requirements

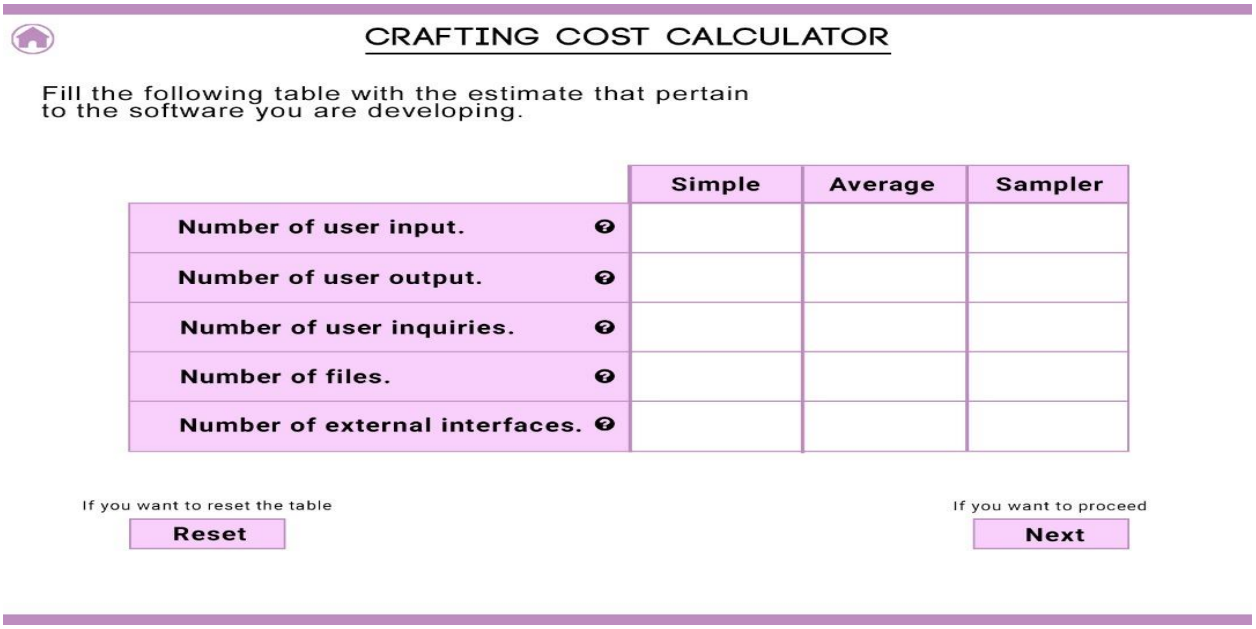
3.1 User Interfaces

1.Crafting Cost Calculator(C^3) welcome section

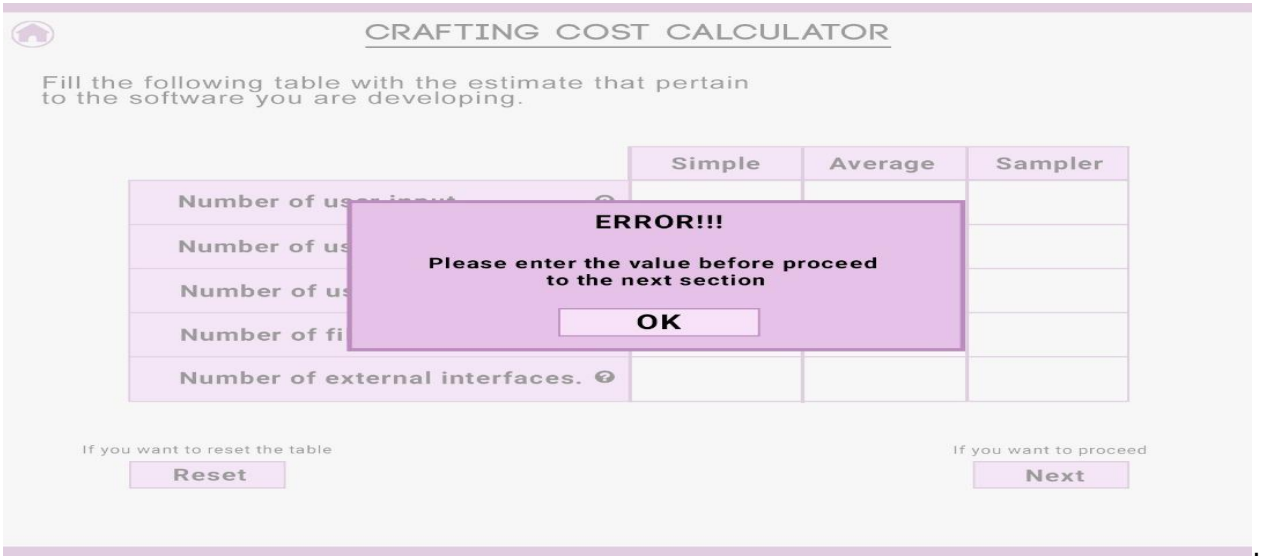


Start button - Used to proceed to the next section and start the calculation.

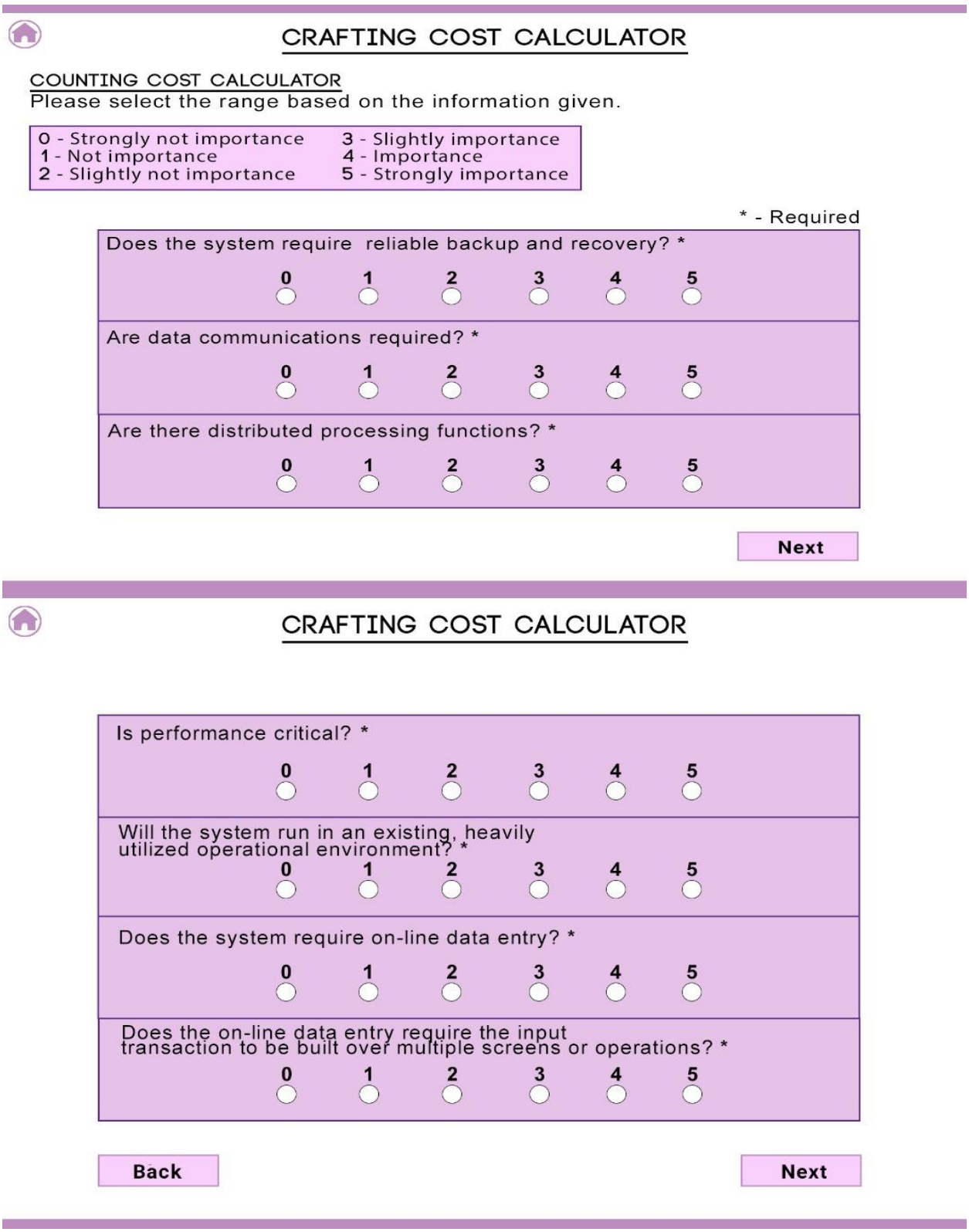
2. User needs to input for unadjusted function point.



- Home button - If the user wants to cancel the process and go back to the welcome section.
- Help button - To help users to understand the meaning of each row with a small pop message.
- Reset button - To reset the input number in table.
- Next button - Used to proceed to the next section. If the user didn't enter any value in the table, the user cannot proceed to the next section and will receive an error message (example error message below)



3. User needs to input for adjusted function point.



CRAFTING COST CALCULATOR

Are conversion and installation included in the design? *

0 1 2 3 4 5

Is the system designed for multiple installations in different organizations? *

0 1 2 3 4 5

Is the application designed to facilitate change and ease of use by the user? *

0 1 2 3 4 5

Next

CRAFTING COST CALCULATOR

Are the master files updated on-line? *

0 1 2 3 4 5

Are the inputs, outputs, files, or inquiries complex? *

0 1 2 3 4 5

Is the internal processing complex? *

0 1 2 3 4 5

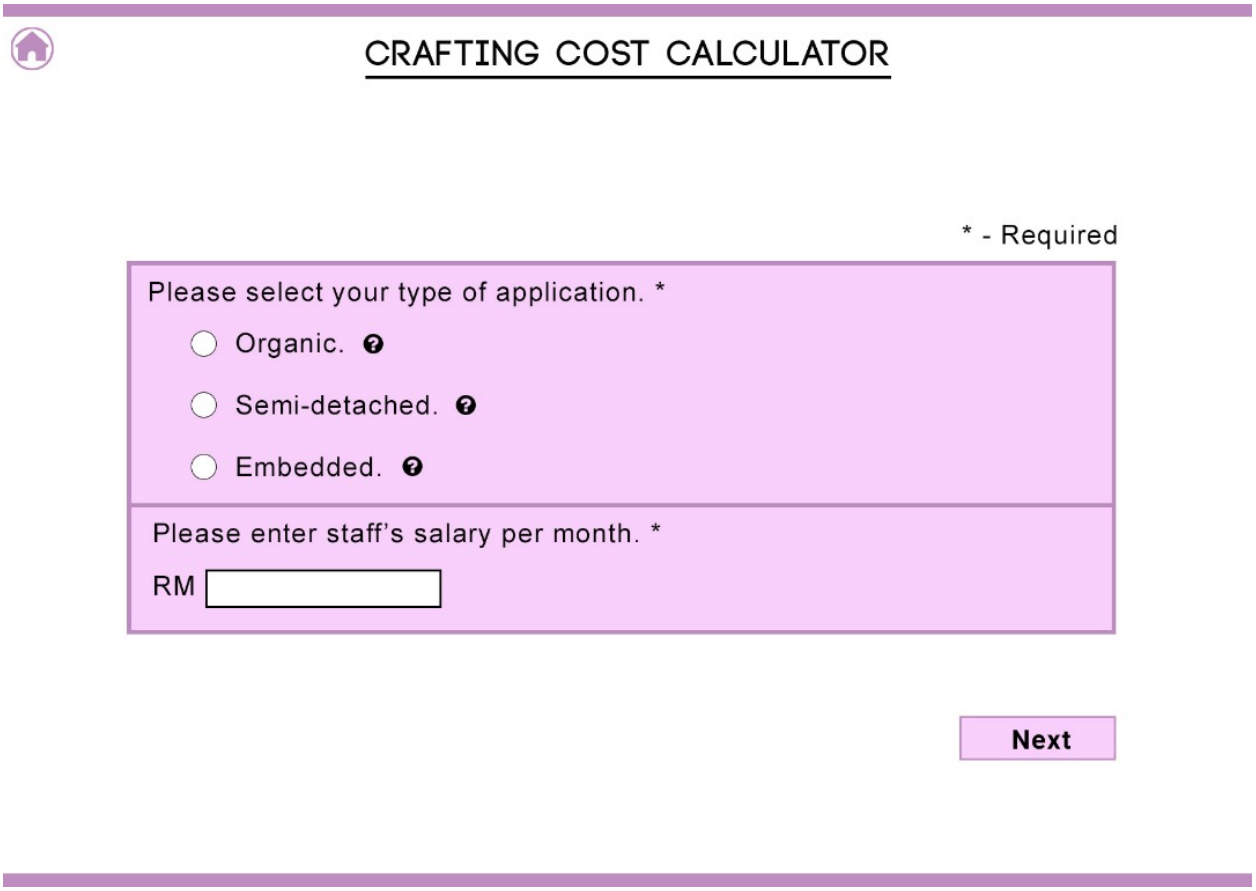
Is the code designed to be reusable? *

0 1 2 3 4 5

Back Next

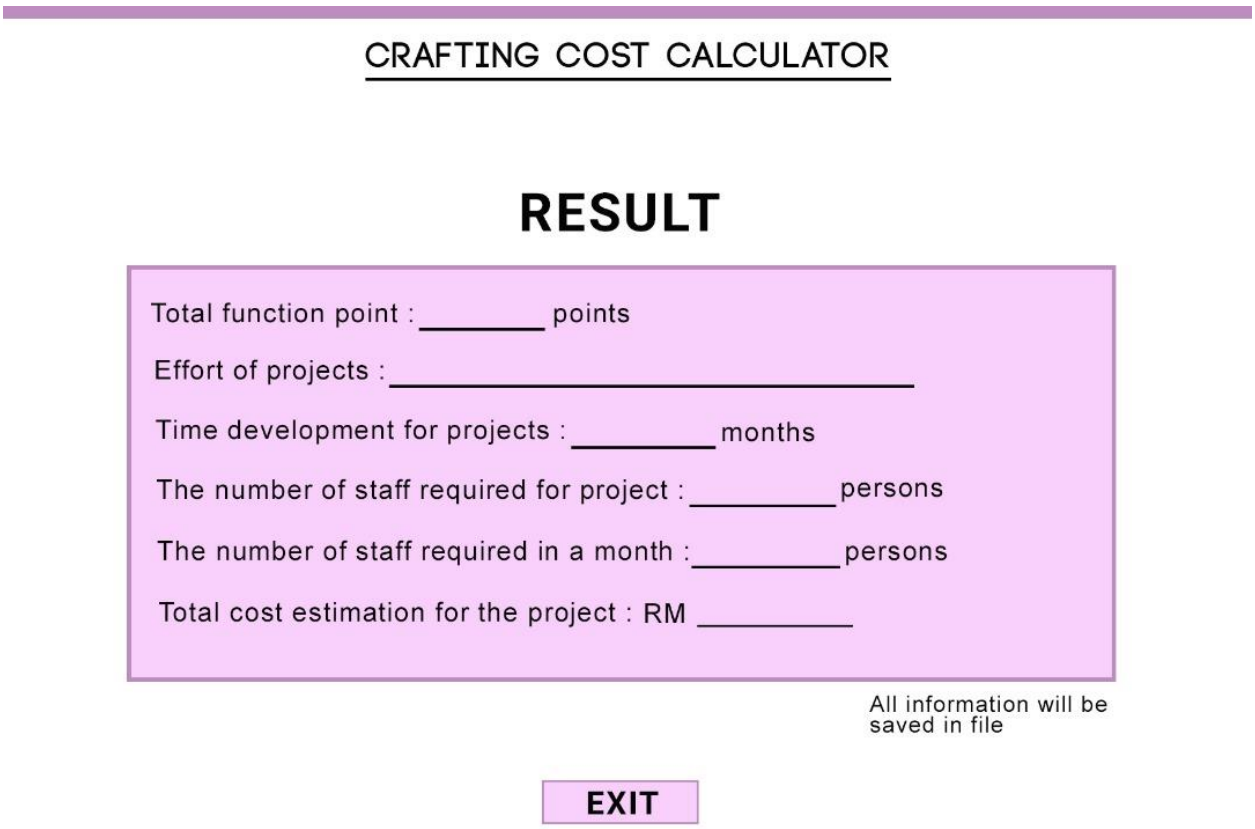
- Users need to select the range value for each question. The default value for each question is 0.
- Home button - If the user wants to cancel the process and go back to the welcome section.
- Next button - Used to proceed to the next section.

4. Users need to input the type of application and salary staff per month.



- Users need to select the type of application and salary staff per month. The default value for the type of application is organic.
- Home button - If the user wants to cancel the process and go back to the welcome section.
- Next button - Use to proceed to the next section. If the user didn't put the value on the salary of staff per month it will be an error.

5. Display the result to the user.



It will display total function point, effort of projects, time development, the number of staff required in a month , the number of staff required for a project and total cost estimation for the project.
Exit button - The result will be saved in file and the user will go back to the welcome section.

3.2 Use Case View

3.2.1 Enter inputs for the Unadjusted Function Point (UFP) Use Case

Use Case: Enter 5 required inputs for UFP.
ID: UC01
Description: User can enter values for inputs in “simple”, “average”, and “complex” category.
Actors: System user
Precondition: User has to open the software.
Main Flow:

1. The user enters the value for simple, average and complex in the number of user inputs(EI).
2. The user enters the value for simple, average and complex in the number of user outputs(EO).

3. The user enters the value for simple, average and complex in the number of user inquiries(EQ).
4. The user enters the value for simple, average and complex in the number of internal logic files(ILF).
5. The user enters the value for simple, average and complex in the number of external interfaces(EIF).

Exception Flow:

1. If there is parameter with no value or wrong input (non numerical value).
2. user would not be able to continue to next section.

3.2.2 Enter Input For Adjusted Function Point (AFP) Use Case

Use Case: Rate 14 General System Characteristics (GSC)
ID: UC02
Description: User can rate 14 General System Characteristic (GSC) on a scale of 0 to 5.
Actors: System user
Precondition: UC01
Main Flow:

1. User can rate from 0 to 5 for data communication.
2. User can rate from 0 to 5 for distributed data processing.
3. User can rate from 0 to 5 for performance.
4. User can rate from 0 to 5 for heavily used configuration.
5. User can rate from 0 to 5 for the transaction rate.
6. User can rate from 0 to 5 for online data entry.
7. User can rate from 0 to 5 for end user efficiency.
8. User can rate from 0 to 5 for online updates.
9. User can rate from 0 to 5 for complex processing.
10. User can rate from 0 to 5 for reusability.
11. User can rate from 0 to 5 for installation ease.
12. User can rate from 0 to 5 for operational ease.
13. User can rate from 0 to 5 for multiple sites.
14. User can rate from 0 to 5 for faciliation change.

Exception Flow:

1. User did not tick all the 14 questions.
2. user would not be able to continue to next section.

3.2.3 Calculate For Unadjusted Function Point (UFP) Use Case

Use Case: Calculation of Unadjusted Function Point (UFP)
ID: UC03
Description: System calculates for unadjusted function point from the input in UC01. The values of simple, average and complex in UC01 will be multiply by their each weight factor and the values are added to determine the UFP.
Actors: System
Precondition: UC01
Main Flow:

1. The system calculates the number of user inputs (EI) in simple multiply with weight factor.
2. The system calculates the number of user inputs (EI) in average multiply with weight factor.
3. The system calculates the number of user inputs (EI) in complex multiply with weight factor.
4. The system calculates the number of user outputs (EO) in simple multiply with weight factor.
5. The system calculates the number of user outputs (EO) in average multiply with weight factor.
6. The system calculates the number of user outputs (EO) in complex multiply with weight factor.
7. The system calculates the number of user inquiries (EQ) in simple multiply with weight factor.
8. The system calculates the number of user inquiries (EQ) in average multiply with weight factor.
9. The system calculates the number of user inquiries (EQ) in complex multiply with weight factor.
10. The system calculates the number of external interface (EIF) in simple multiply with weight factor.
11. The system calculates the number of external interface (EIF) in average multiply with weight factor.
12. The system calculates the number of external interface (EIF) in complex multiply with weight factor.
13. The system calculates the number of files (ILF) in simple multiply with weight factor.
14. The system calculates the number of files (ILF) in average multiply with weight factor.
15. The system calculates the number of files (ILF) in complex multiply with weight factor.

3.2.4 Calculate for Adjusted Function Point (AFP) Use Case

Use Case: Calculation of Adjusted Function Point (AFP)
ID: UC04
Description: System calculated the AFP from the rating entered by user in each of 14 GSC in UC02. The formula is AFP = 0.65 + (Value Adjusted Factor*0.01). Value adjusted factor is sum of all rating in GSC questions.
Actors: System
Precondition: UC02
Main Flow:

1. System calculates the value adjusted factor which is sum of all rating in 14 GSC.
2. System calculates the Adjusted Function Point using the formula of AFP = 0.65+(VAF*0.01).

3.2.5 Calculate Function Point Analysis (FPA) Use Case

Use Case: Calculation of Function Point Analysis (FPA)
ID: UC05
Description: System calculates the value of Function Point Analysis using the value obtain from UC03 and UC04.
Actors: System
Precondition: UC03,UC04
Main Flow:

1. System calculates the function point analysis.
2. Formula = FP = UFP * AFP.

3.2.6 Select application types Use Case

Use Case: Selecting application types
ID: UC06
Description: User can select type of application that they developing. Types of application are organic, semi-detached and embedded.
Actors: System user
Precondition: UC05
Main Flow:

1. User selects the type of application between Organic, Semi-detached and embedded.

Exception Flow:

1. User did not select any of the type of application.
2. System display error.
3. User would not be able to continue until at least one is selected.

3.2.7 Calculate effort based on type of application Use Case

Use Case: Calculation for effort based on type of application
ID: UC07
Description: Calculate effort based on type of application choosen in UC06. Different type brings different value of a and b.
Actors: System
Precondition: UC05, UC06
Main Flow:

1. System calculates the effort based on type of application with formula, Effort = a(FPA)*(b).

3.2.8 Calculate time development based on type of application Use Case

Use Case: Calculate the time of development based on type of application
ID: UC08
Description: Calculate time development based on type of application choosed in UC05. Different type brings different value of c and d.
Actors: System
Precondition: UC06
Main Flow:

1. System calculates the time of development based on type of application with formula, time development = c(Effort)*(d).

3.2.9 Calculate Number of Staff Required In a Month Use Case

Use Case: Calculate the number of staff required in a month
ID: UC09
Description: Calculate the number of staff required in a month to complete the system. This use case uses value from UC07 and UC08.
Actors: System
Precondition: UC07, UC08
Main Flow:

1. System calculates the number of staff required in a month with formula of, number of staff required in a month = effort/time development.

3.2.10 Enter Salary For a Person Per Month Use Case

Use Case: Enter the salary for a person per month.
ID: UC10
Description: User enters the salary for a person in a month.
Actors: System user

Precondition: None
Main Flow:

1. User enters the salary for a person in a month.

Exception Flow:

1. User did not enter any value for salary.
2. System ran into an error.
3. User did not enter numerical value for salary.
4. System ran into an error.

3.2.11 Calculate Number Staff Required In The Project Use Case

Use Case: Calculate the number of staff required for the whole project
ID: UC11
Description: System calculates the number of staff required for the whole project using value from UC09 and UC08.
Actors: System
Precondition: UC09, UC08
Main Flow:

1. System calculates the number of staff required to complete the project with formula, number of staff required in the project = number of staff required in a month / time development.

3.2.12 Calculate Cost Estimation For The Project Use Case

Use Case: Calculate the cost estimation for the project
ID: UC12
Description: System calculates the cost estimation for the project using values from UC08 and UC10.
Actors: System
Precondition: UC08, UC10
Main Flow:

1. System calculates the cost estimation for the project using formula, cost estimation = number of staff required in the project * time development.

3.2.13 Display Result Of Calculation Use Case.

Use Case: View the result of calculation for the project.
ID: UC13
Description: System displays the result of all the calculations to the user in a file.
Actors: System
Precondition: UC04, UC06, UC07, UC08, UC10, UC11
Main Flow :

1. System displays the function points for the project.
2. System displays value of efforts for the project.
3. System displays the total time development required for the project.

4. System display the number of staff required in a month.
5. System display the number of staff required for the project.
6. System displays value of cost estimation to complete the project.

3.2.14 Save Result of Calculations In File Use Case

Use Case: Save all the informations in a file.
ID: UC14
Description:
Actors: System
Precondition: UC05, UC07, UC08, UC09, UC11, UC12
Main Flow:

1. System saves the information obtain from calculations in the file.

4. System Features

4.1 Enter Inputs for Unadjusted Function Point

4.1.1 Description and Priority

System shall be able to allow the user to input the simple, average and complex for each 5 required parameters.The priority to input 5 parameters is high, where the user must input for 5 requirements else it will be error.

4.1.2 Stimulus/Response Sequence

a. The user can enter the value for simple, average and complex in the number of user inputs(EI).
b. The user can enter the value for simple, average and complex in the number of user outputs(EO).
c. The user can enter the value for simple, average and complex in the number of user inquiries(EQ).
d. The user can enter the value for simple, average and complex in the number of files(ILF).
e. The user can enter the value for simple, average and complex in the number of external interfaces(EIF).

4.1.3 Functional Requirement

a. REQ-1: The user shall be able to enter the number of user inputs(EI) in simple, average and complex categories.
b. REQ-2: The user shall be able to enter the number of user outputs(EO) in simple, average and complex categories.
c. REQ-3: The user shall be able to enter the number of user inquiries(EQ) in simple, average and complex categories.
d. REQ-4: The user shall be able to enter the number of files(ILF) in simple, average and complex categories.
e. REQ-5: The user shall be able to enter the number of external interfaces(EIF) in simple, average and complex categories.

4.2 Enter input for Adjusted Function Point(AFP)

4.2.1 Description and Priority

System shall be able to allow the user to rate 14 General system characteristics(GSC) of the system or application that defines the types of application characteristics and is rated on a scale of 0 to 5.The priority to rate each question of GSC is high, where the user must rate for 14 GSC else it will be error.

4.2.2 Stimulus/Response Sequence

a. The user can enter the rate for data communication.
b. The user can enter the rate for distributed data processing.
c. The user can enter the rate for performance.
d. The user can enter the rate for heavily used configuration.
e. The user can enter the rate for the transaction rate.
f. The user can enter the rate for online data entry.
g. The user can enter the rate for end user efficiency.
h. The user can enter the rate for online updates.
i. The user can enter the rate for complex processing.
j. The user can enter the rate for reusability.
k. The user can enter the rate for installation ease.
l. The user can enter the rate for operational ease.
m. The user can enter the rate for multiple sites.
n. The user can enter the rate for the facilitate change.

4.2.3 Functional Requirement

a. REQ-1: The user shall be able to rate for data communication between 0 and 5.
b. REQ-2: The user shall be able to rate for distributed data processing between 0 and 5.
c. REQ-3: The user shall be able to rate for performance between 0 and 5.
d. REQ-4: The user shall be able to rate for heavily used configuration between 0 and 5.
e. REQ-5: The user shall be able to rate for the transaction rate between 0 and 5.
f. REQ-6: The user shall be able to rate for online data entry rate between 0 and 5.
g. REQ-7: The user shall be able to rate for end user efficiency between 0 and 5.
h. REQ-8: The user shall be able to rate for online update between 0 and 5.
i. REQ-9: The user shall be able to rate for complex processing between 0 and 5.
j. REQ-10: The user shall be able to rate for reusability between 0 and 5.
k. REQ-11: The user shall be able to rate for installation ease between 0 and 5.
l. REQ-12: The user shall be able to rate for operational ease between 0 and 5.
m. REQ-13: The user shall be able to rate for multiple sites between 0 and 5.
n. REQ-14: The user shall be able to rate for the facilitate change between 0 and 5.

4.3 Calculate for Unadjusted Function Point(UFP)

4.3.1 Description and Priority

System shall be able to calculate the UFP input from system feature 1. The functional complexities are multiplied with the corresponding weights against each function, and the values are added up to determine the UFP (Unadjusted Function Point) of the subsystem. The weighing factor will be simple, average, or complex for a measurement parameter type.The priority to calculate unadjusted function points is high.

4.3.2 Stimulus/Response Sequence

a. The system can calculate the number of user inputs(EI) in the simple of the weighing factor.
b. The system can calculate the number of user inputs(EI) in the average of the weighing factor.
c. The system can calculate the number of user inputs(EI) in the complex of the weighing factor.
d. The system can calculate the number of user outputs(EO) in the simple of the weighing factor.

e. The system can calculate the number of user outputs(EO) in the average of the weighing factor.
f. The system can calculate the number of user outputs(EO) in the complex of the weighing factor.
g. The system can calculate the number of user inquiries(EQ) in the simple of the weighing factor.
h. The system can calculate the number of user inquiries(EQ) in the average of the weighing factor.
i. The system can calculate the number of user inquiries(EQ) in the complex of the weighing factor.
j. The system can calculate the number of files(ILF) in the simple of the weighing factor.
k. The system can calculate the number of files(ILF) in the average of the weighing factor.
l. The system can calculate the number of files(ILF) in the complex of the weighing factor.
m. The system can calculate the number of external interfaces(EIF) in the simple of the weighing factor.
n. The system can calculate the number of external interfaces(EIF) in the average of the weighing factor.
o. The system can calculate the number of external interfaces(EIF) in the complex of the weighing factor.
p. The system can calculate unadjusted function point(UFP).

4.3.3 Functional Requirement

a. REQ-1: The system shall be able to calculate and multiply the number of user inputs(EI) in the low of the weighing factor with 3.
b. REQ-2: The system shall be able to calculate and multiply the number of user inputs(EI) in the average of the weighing factor with 4.
c. REQ-3: The system shall be able to calculate and multiply the number of user inputs(EI) in the high of the weighing factor with 6.
d. REQ-4: The system shall be able to calculate and multiply the number of user outputs(EO) in the low of the weighing factor with 4.
e. REQ-5: The system shall be able to calculate and multiply the number of user outputs(EO) in the average of the weighing factor with 5.
f. REQ-6: The system shall be able to calculate and multiply the number of user outputs(EO) in the high of the weighing factor with 7.
g. REQ-7: The system shall be able to calculate and multiply the number of user inquiries(EQ) in the low of the weighing factor with 3.
h. REQ-8: The system shall be able to calculate and multiply the number of user inquiries(EQ) in the average of the weighing factor with 4.

i. REQ-9: The system shall be able to calculate and multiply the number of user inquiries(EQ) in the high of the weighing factor with 6.
j. REQ-10: The system shall be able to calculate and multiply the number of files(ILF) in the low of the weighing factor with 7.
k. REQ-11: The system shall be able to calculate and multiply the number of files(ILF) in the average of the weighing factor with 10.
l. REQ-12: The system shall be able to calculate and multiply the number of files(ILF) in the high of the weighing factor with 15.
m. REQ-13: The system shall be able to calculate and multiply the number of external interfaces(EIF) in the low of the weighing factor with 5.
n. REQ-14: The system shall be able to calculate and multiply the number of external interfaces(EIF) in the average of the weighing factor with 7.
o. REQ-15: The system shall be able to calculate and multiply the number of external interfaces(EIF) in the high of the weighing factor with 10.
p. REQ-16: The system shall be able to calculate unadjusted function point(UFP) with the sum of all weighing factors for each parameter(EI, EO, EQ, EIF, ILF).

4.4 Calculate for Adjusted Function Point

4.4.1 Description and Priority

System shall be able to calculate the Adjusted Function Point that rate by user by each question based on GSC in system feature 2. $\sum(\mathbf{f_i})$ is the sum of all 14 questionnaires and show the complexity adjustment value or factor-VAF (where i ranges from question 1 to 14).The priority to calculate adjusted function points is high.

4.4.2 Stimulus/Response Sequence

a. The system can calculate the adjusted function point.

4.4.3 Functional Requirement

a. REQ-1: The system shall be able to calculate adjusted function point(AFP) with sum of all questionnaires GSC , $\sum(\mathbf{f_i})$, add with 0.01 and multiply with 0.65. (AFP = [0.65 +(0.01 *$\sum(f_i)$]))

4.5 Calculate for Function Point Analysis(FPA)

4.5.1 Description and Priority

- System shall be able to calculate Function Point (FP) is thus calculated with the following formula.The priority to calculate FPA is high.

4.5.2 Stimulus/Response Sequence

a. The system can calculate the function point analysis(FPA).

4.5.3 Functional Requirement

a. REQ-1: The system shall be able to calculate function point analysis(FPA) with multiply unadjusted function point(UFP) from system feature 3 and adjusted function point(AFP) from system feature 4, (FP = UFP * AFP)**.**

4.6 Select application types

4.6.1 Description and Priority

Users shall be able to select the type of application depending on the project type.The priority to choose application type is high, where the user must select the type of application else it will be an error.

4.6.2 Stimulus/Response Sequence

a. The user can select the type of application.

4.6.3 Functional Requirement

a. REQ-1: The user shall be able to select the type of application between Organic, Semi-detached and Embedded.

4.7 Calculate effort based on type of application

4.7.1 Description and Priority

Calculate effort based on type application based on system feature 6. The priority to calculate effort based on type application is high.

4.7.2 Stimulus/Response Sequence

a. The system can calculate effort based on type of application from system feature 6.

4.7.3 Functional Requirement

Type Application	a_i	b_i
Organic	2.4	1.05
Semidetached	3.0	1.12
Embedded	3.6	1.20

a. REQ-1: The system shall be able to calculate the effort based on type of application from system feature 6 with multiply a_i (constant value based on type of application from system feature 6) and function point analysis(FPA) from system feature 5 and the power of b_i (constant value based on type of application from system feature 6), (Effort =a_i (FPA) *(b_i)).

4.8 Calculate time development based on type of application

4.8.1 Description and Priority

Calculate time development based on type application based on system feature 6. The priority to calculate time development based on type application is high.

4.8.2 Stimulus/Response Sequence

a. The system can calculate time development based on type of application from system feature 6.

4.8.3 Functional Requirement

a. REQ-1: The system shall be able to calculate the time development based on type of application from system feature 6 with multiply c_i (constant value based on type of application from system feature 6) and effort from system feature 7 and multiply with d_i (constant value based on type of application from system feature 6), (Time development $=c_i$ (Effort)d_i)

Type Application	c_i	d_i
Organic	2.5	0.38
Semidetached	2.5	0.35
Embedded	2.5	0.32

4.9 Calculate number of staff required in a month

4.9.1 Description and Priority

System shall be able to calculate the number of staff required in a month.The priority to calculate the number of staff required in a month is high.

4.9.2 Stimulus/Response Sequence

a. The system can calculate the number of staff required in a month.

4.9.3 Functional Requirement

a. REQ-1: The system shall be able to calculate the number of staff required in a month with effort from system feature 7 divided by time development from system feature 8, (number of staff required in a month = Effort / Time development).

4.10 Enter salary for a person per month

4.10.1 Description and Priority

Users shall be able to input salary for a person per month.The priority to input salary for a person per month is high, where the user must input salary for a person per month else it will be an error.

4.10.2 Stimulus/Response Sequence

a. The user can enter the salary for a person per month.

4.10.3 Functional Requirement

a. REQ-1: The user shall be able to enter the salary for a person per month.

4.11 Calculate number staff required in the project

4.11.1 Description and Priority

System shall be able to calculate the number of staff required.The priority to calculate number staff required in the project is high.

4.11.2 Stimulus/Response Sequence

a. The system can calculate the number of staff required in the project.

4.11.3 Functional Requirement

a. REQ-1: The system shall be able to calculate the number of staff required in the project with the number of staff required in a month system feature 9 divided by time development from system feature 8, (number of staff required in the project= number of staff required in a month / Time development).

4.12 Calculate cost estimation for the project

4.12.1 Description and Priority

System shall be able to calculate cost estimation for the project.The priority to calculate number staff required in the project is high

4.12.2 Stimulus/Response Sequence

a. The system can calculate the cost estimation for the project.

4.12.3 Functional Requirement

a. REQ-1: The system shall be able to calculate cost estimation for the project with the salary for a person per month from system feature 10 multiply with number of staff required in the project from system feature 11 and multiply with time development

from system feature 8). (Cost estimation = salary for a person per month * number of staff required in the project * time development)

4.13 Display result of calculations of the project for user

4.13.1 4.13.1 Description and Priority

System able to display the result to the user.The priority to display the result to the user is high.

4.13.2 Stimulus/Response Sequence

a. The system can display the function point in the project.
b. The system can display the effort for the project.
c. The system can display time development for the project.
d. The system can display the number of staff required in a month.
e. The system can display the number of staff required for the project.
f. The system can display cost estimation of the project.

4.13.3 Functional Requirement

a. REQ-1: The system shall be able to display the function point of the project from system feature 5.
b. REQ-2: The system shall be able to display effort for the project from system feature 7.
c. REQ-3: The system shall be able to display time development for the project from system feature 8.
d. REQ-4: The system shall be able to display the number of staff required in a month from system feature 9.
e. REQ-5: The system shall be able to display the number of staff required for the project from system feature 11.
f. REQ-6: The system shall be able to display cost estimation of the project from system feature 12.

4.14 Save result of calculations for user in file

4.14.1 Description and Priority

System able to save results in the file.The priority to save the result in the file is high.

4.14.2 Stimulus/Response Sequence

a. The system can save the information result in the file

4.14.3 Functional Requirement

a. REQ-1: The system shall be able to save the information resulting in the file.

5. Other Nonfunctional Requirements

5.1 Safety Requirements

To ensure that no one of C^3 users loses any data while using C^3 due to a crash or a bug of some kind, the developers team updates C^3 regularly.

5.2 Security Requirements

C^3 does not have any security requirements and thus any type of user can use it without any additional privileges

6. Other Requirements

6.1 Appendix A: Analysis Models