****

**COMSATS University Islamabad (CUI)**

**Project Name**

***By***

**Student Name 1 CIIT/SP09-BCS-xxx/ISB**

**Student Name 2 CIIT/SP09-BCS-xxx/ISB**

***Supervisor*Supervisor Name**

***Co-Supervisor*C0-Supervisor Name**

***Bachelor of Science in Computer Science (20xx-20xx)***

**The candidate confirms that the work submitted is their own and appropriate  
 credit has been given where reference has been made to the work of others**.

****

**COMSATS University Islamabad (CUI)**

**Project Name**

**A project presented to**

**COMSATS University Islamabad**

**In partial fulfillment**

**of the requirement for the degree of**

***Bachelor of Science in Computer Science (20xx-20xx)***

**By**

**Student Name 1 CIIT/SP09-BCS-xxx/ISB**

**Student Name 2 CIIT/SP09-BCS-xxx/ISB**

**DECLARATION**

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

|  |  |  |
| --- | --- | --- |
| Student Name1 | Student Name2 | Student Name3 |
| --------------------------- | --------------------------- | --------------------------- |

**CERTIFICATE OF APPROVAL**

It is to certify that the final year project of BS (CS) “Project title” was developed by   
**STUDENT 1 NAME (CIIT/FAXX-BCS-000)** and **STUDENT 2 NAME (CIIT/FAXX-BCS-000)** under the supervision of “SUPERVISOR NAME” and co supervisor “CO-SUPERVISOR NAME” and that in (their/his/her) opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences.

---------------------------------------

**Supervisor**

---------------------------------------

**Co-Supervisor**

---------------------------------------

**External Examiner**

---------------------------------------

**Head of Department**

**(Department of Computer Science)**

**Executive Summary**

In public places, there is often a need for monitoring people and different activities going on, which can be referred later for many reasons including security. Appointing humans for this task involves many problems such as increased employee hiring, accuracy problem, trust, no proof for later use, and also the fact that a human can remember things till a certain time limit. Talking about the current security system, they use dumb still cameras with a continuous recording facility irrespective of the fact that any event may happen or not. Moreover, they are usually pointing at a specific user defined location so more than one cameras are required to cover the entire region.

To prevent all these problems from prevailing, the CSCS is developed. It is a surveillance system, which provides solution to many of these problems. It is a stand-alone application which doesn’t require any computer to operate. It monitors different situations using a camera which is able to rotate intelligently based on sensor messages and captures the scene in the form of video or photos later reference as well.

**C**ustomizable **S**urveillance **C**ontrol **S**ystem **(CSCS)** is a surveillance system that can be assigned a sensor type as in our case a heat sensor is used, it works accordingly, rotates the camera upon event detection and perform user defined actions like capturing video and stores them, for the future use.

It is an embedded system consisting of Linux fox kit with embedded a running server application also a camera, USB storage device and a sensor node base station is attached with fox kit. LAN communication is used by user to download the videos and to operate the system manually.

**Acknowledgement**

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Dr. Majid Iqbal Khan” and our Co-Supervisor “Mr. Mukhtar Azeem”. Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are deeply indebted to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

|  |  |  |
| --- | --- | --- |
| Student Name1 | Student Name2 | Student Name3 |
| --------------------------- | --------------------------- | --------------------------- |

**Abbreviations**

|  |  |
| --- | --- |
| **SRS** | Software Requirement Specification |
| **PC** | Personal Computer |
|  |  |
|  |  |
|  |  |

**Table of Contents**

[**1 Introduction**](#_Toc520773734) 1

[1.1 Brief Overview 1](#_Toc520773735)

[1.2 Relevance to Course Modules 1](#_Toc520773736)

[1.3 Project Background 1](#_Toc520773737)

[1.4 Literature Review 1](#_Toc520773738)

[1.5 Analysis from Literature Review 1](#_Toc520773739)

[1.6 Methodology and Software Lifecycle for this Project 1](#_Toc520773740)

[1.6.1 Rationale behind the Selected Methodology 1](#_Toc520773741)

[**2** **Problem Definition** 3](#_Toc520773742)

[2.1 Problem Statement 3](#_Toc520773745)

[2.2 Deliverables and Development Requirements 3](#_Toc520773746)

[2.3 Current Systems 3](#_Toc520773747)

[**3** **Requirement Analysis** 4](#_Toc520773748)

[3.1 Use Cases Diagram(s) 4](#_Toc520773750)

[3.2 Detailed Use Cases 4](#_Toc520773751)

[3.3 Functional Requirements 4](#_Toc520773752)

[3.4 Non-Functional Requirements 4](#_Toc520773753)

[**4 Design and Architecture** 5](#_Toc520773754)

[4.1 System Architecture 5](#_Toc520773759)

[4.2 Data Representation 5](#_Toc520773760)

[4.3 Process Flow/Representation 5](#_Toc520773761)

[4.4 Design Models 5](#_Toc520773762)

[**5** **Implementation** 7](#_Toc520773763)

[5.1 Algorithm 7](#_Toc520773765)

[5.2 External APIs 7](#_Toc520773766)

[5.3 User Interface 7](#_Toc520773767)

[**6 Testing and Evaluation** 8](#_Toc520773768)

[6.1 Manual Testing 8](#_Toc520773771)

[6.1.1 Unit testing 8](#_Toc520773772)

[6.1.2 Integration Testing 9](#_Toc520773773)

[6.1.3 Functional Testing 9](#_Toc520773774)

[6.1.4 System Testing 10](#_Toc520773775)

[6.2 Automated Testing 10](#_Toc520773776)

[**7** **Conclusion and Future Work** 12](#_Toc520773778)

[7.1 Conclusion 12](#_Toc520773780)

[7.2 Future Work 12](#_Toc520773781)

[**8** **References** 13](#_Toc520773782)

**List of Figures**

Fig 1.1 Block Diagram 1

Fig 2.1 Use Case Diagram 2

# 

# Introduction

This chapter provides the overview of the project. The first paragraph of every chapter should provide the chapter summary. You are required to use a Grammarly or some other automated tool to avoid the grammar and language mistakes.

## Brief Overview

A very brief introduction of project work, outcome of your work, tools, methodology used & highlights of discussions in various chapters of report.

## Relevance to Course Modules

A brief explanation of how your project is related to various courses studied during BCS degree program.

## Project Background

It includes explanation of the idea behind the project. For example, if the project is related to VoIP then this section describes that what is voice over IP & how it works.

## Literature Review

This section will describe current trends/ research/ products etc. related to your project.

## Analysis from Literature Review

This section will provide an analytical discussion in the context of your targeted work with compared to the literature review. You may add a Table to clearly depict the analysis from literature review.

## Methodology and Software Lifecycle for this Project

A brief discussion of methodology and SDLC model selected for this project. You may add a suitable Figure to highlight the methodology.

### Rationale behind the Selected Methodology

Explicitly mention the rationale of your selected methodology. In other words, why you have selected the methodology (as discussed in Section 1.6) such as structural and Object-Oriented and software life cycle for this project?

#### Example for adding fourth level heading

It is the example of last level heading. Please do not insert further levels in numbers. Use different format style e.g. italic to highlight the important text.

The following list style is the sample to consistently follow in the whole report.

* List items 1
* List items 2

# Problem Definition

This chapter would discuss the targeted problem in a precise manner. It should also need to clearly discuss the outcome of the targeted project.



## Problem Statement

Problem statement goes here. It is important to re-check the problem statement by consulting with your supervisor/co-supervisor.

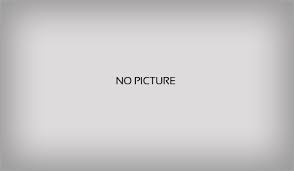
## Deliverables and Development Requirements

This section would need to discuss about the deliverables and the development requirements.

## Current System

A brief description of existing system(s) should be provided in this section. You may add a Table to succinctly discusses the strengths and limitations of current system(s) in comparison with the targeted system.

The following figure is a sample figure, Figure 2.1. You are required to follow the same style of numbering and caption for the whole report. The discussion should be like “ Figure 2.1 presents the…”



**Figure 2.1: Sample Figure**

The following table (Table 2.1) is sample table; You are required to follow the same style of numbering and caption for the whole report.

**Table ‎2.1: Sample Table**

|  |  |  |
| --- | --- | --- |
| **Header 1** | **Header 2** | **Header 3** |
| Text | Text | Text |
|  |  |  |

# Requirement Analysis

The following parts of Software Requirements Specification (SRS) report should be included in this chapter.



## Use Cases Diagram(s)

This section will present the use case diagram. It is also suggested to discuss the use case diagram from a technical perspective.

## Detailed Use Case

This section will present the detailed use case.

## Functional Requirements

The Functional Requirements needs to be mentioned in this section.

## Non-Functional Requirements

Non-Functional requirements are required to be explicitly mention in this section.

# Design and Architecture

The following parts of Software Design Description (SDD) report should be included in this chapter.



## System Architecture

[diagram + description]

## Data Representation

[diagram + description]

## Process Flow/Representation

[diagram + description]

## Design Models

In this section, present the UML diagrams (for object-oriented methodology). Sections 4.4.1 and 4.4.2 present the UML diagrams for OO methodology.

Note, in case of structured (procedural) methodology, you are required to present the respective set of diagrams, including Data-Flow Diagram, Event-Flow Diagram, Stat Diagram, and Activity Diagram.

### Structural Diagrams

This section would present the static structure of the system, its parts on different abstraction and implementation levels, and how they are related to each other. The elements in a structure diagram represent the meaningful concepts of a system, and may include abstract, real world and implementation concepts.

#### Class diagram

#### Object diagram

#### Component diagram

#### Package diagram

#### Deployment diagram

### Behavioral Diagrams

This section would present the behavior diagrams that show the dynamic behavior of the objects in a system, which can be described as a series of changes to the system over time.

#### Activity diagram

#### State machine diagram

#### Sequence diagram

#### Communication diagram

#### Interaction Overview diagram

#### Timing Overview diagram

Note this diagram can be used only for hardware related projects.

# Implementation

This chapter will discuss implementation details supported by UML diagrams (if applicable). You will not put your source code here. Any of the following sections may be included based on your project.



## Algorithm

Mention the algorithm(s) used in your project to get the work done with regards to major modules. Provide a pseudocode **OR** a natural language explanation regarding the functioning of main features. Be sure to use the correct syntax and semantics for algorithm representations.

## External APIs

Describe the APIs used in the following table.

**Table 5.1: Details of APIs used in the Project**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of API** | **Description of API** | **Purpose of Usage** | **List down the function/class name in which it is used** |
|  |  |  |  |
|  |  |  |  |

## User Interface

Details about user interface with descriptions will be presented in this section.

# Testing and Evaluation

This chapter may include the following sections. (Students are required to perform the testing both manually and automatically).



## Manual Testing

### Unit Testing

**Unit Testing 1:** Login as FYP Committee

**Testing Objective:** To ensure the login form is working correctly.

**Test Case Id:** BU\_001

**Test Case Description:** Test the login functionality.

**Test Scenario:** Verify on entering valid username and password, the user can login.

**Table 6.1: Test Cases for Login**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Test Case/Test Script** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail/Not Executed/ Suspended** |
| 1. | Verify user login after click on the ‘Login’ button on login form with correct input data | Username:  L001  Password:  1234 | Successfully log into the main page of the system as FYP Committee member. | As Expected | Pass |
| 2. |  |  |  |  |  |

**Unit Testing 2:** Edit Profile

**Testing Objective:** To ensure the edit profile form is working properly.

**Test Case Id:** BU\_002

**Test Case Description:** Test the edit profile functionality.

**Test Scenario:**

**Table 6.2: Test Cases for Edit Profile**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Test Case/Test Script** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail/Not Executed/ Suspended** |
| 1. | Verify user login after click on the ‘Login’ button on login form with correct input data | Username:  L001  Password:  1234 | Successfully log into the main page of the system as FYP Committee member. | As Expected | Pass |
| 2. |  |  |  |  |  |

### Functional Testing

The functional testing will take place after the unit testing. In this functional testing, the functionality of each of the module is tested. This is to ensure that the system produced meets the specifications and requirements.

**Functional Testing 1:** Login with different roles

**Test Objective**: To ensure that the correct page with the correct navigation bar is loaded.

**Test Case Id:** BU\_003

**Test Case Description:**

**Test Scenario:**

**Table 6.3: Test Cases for Login with different Roles**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Test Case/Test Script** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail/Not Executed/ Suspended** |
| 1. | Login as a ‘FYP Committee’ member. | Username: L001  Password: 1234 | Main page for the FYP Committee member is loaded with the FYP Committee navigation bar | Login as a ‘FYP Committee’ member. | Pass |
| 2. |  |  |  |  |  |

### Integration Testing

Please specify the type of applied integration strategy, i.e. top down, bottom up. Also, elaborate the applied integration strategy.

**Table 6.4: Test Cases for Integration Testing of Unit X**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.** | **Test Case/Test Script** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail/Not Executed/ Suspended** |
| 1. | Login as “FYP Committee” member | Username: L001  Password: 1234 | Login successful and the FYP Committee page with its navigation bar is loaded and in the view profile page | As Expected | Pass |
| 2. | Upload student record for Project 1 | - | File successfully uploaded and return to the upload page. Student records are updated. | As Expected | Pass |
| 3. | View supervising student | - | The list of supervisees shown on the screen. | As Expected | Pass |
| 4. |  |  |  |  |  |

### System Testing

Once the system has been successfully developed, testing has to be performed to ensure that the system working as intended. This is also to check that the system meets the requirements stated earlier. Besides that, system testing will help in finding the errors that may be hidden from the user. There are few types of testing which includes the unit testing, functional testing and integration testing. The testing must be completed before it is being deploy for user to use.

## Automated Testing:

This section will discuss the testing tools used to automatically test the targeted project.

**Table 6.5: Tools employed for Automated Testing**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tool Name** | **Tool Description** | **Applied on [list of related Test Cases / FR / NFR]** | **Results** |
|  |  |  |  |
|  |  |  |  |

# Conclusion and Future Work

This chapter concludes the project and highlights future work.



## Conclusion

This section will summarize that how your project successfully solves the targeted problem. You may summarize your key project results in a few sentences and use this summary to support your conclusion. Also, include key facts from literature to explain your results as needed.

## Future Work

This section will help in explaining the readers where you think the results can lead you. What do you think are the next steps to take? Do you think certain paths seem to be more promising than others?

This section should be petty short. The goal should not be to go into a bunch of details, but instead just a sentence or two explaining each idea. It should just provide enough information as to a possible further development path and why the path may be important.

# References

References to any book, journal paper or website should properly be acknowledged. Please consistently follow the style. The following are few examples of different resources i.e. journal article, book, and website.

* 1. Lyda M.S. Lau, Jayne Curson, Richard Drew, Peter Dew and Christine Leigh, (1999), Use Of VSP Resource Rooms to Support Group Work in a Learning Environment, ACM 99, pp-2. **(Journal paper example)**
  2. Hideyuki Nakanishi, Chikara Yoshida, Toshikazu Nishmora and TuruIshada, (1996), FreeWalk: Supporting Casual Meetings in a Network, pp 308-314 **(paper on web)** http://www.acm.org/pubs/articles/proceedings/cscw/240080/p308-nakanishi.pdf
  3. Ali Behforooz& Frederick J.Hudson, (1996), Software Engineering Fundamentals, Oxford University Press. Chapter 8, pp255-235. **(book reference example)**
  4. Page Author, Page Title, http://www.bt.com/bttj/archive.htm, Last date accessed**. (web site)**