



COMSATS UNIVERSITY ISLAMABAD

DEPARTMENT OF COMPUTER SCIENCE

# **UNDERGRADUATE FINAL YEAR PROJECT HANDBOOK**

Guidelines, Procedures and Regulations

Document Issued: SPRING 2019

Version: 2.0

Prepared By: FYP Committee

## **TABLE OF CONTENTS**

<b>1. Introduction .....</b>	<b>04</b>
1.1.Objectives.....	04
1.2. Assigning Students to Supervisors and Projects .....	04
<b>2. Guidelines .....</b>	<b>05</b>
2.1. Guidelines for Students .....	05
2.2. Role of Convener, Coordinator and Moderator.....	06
2.3. Role of Supervisor .....	07
2.4. Role of Evaluation Team .....	08
2.5. Penalties .....	09
2.6. Formatting and Layout.....	10
<b>3. Road map for FYP.....</b>	<b>11</b>
<b>4. Milestone and Activities .....</b>	<b>12</b>
4.1. 06 <sup>th</sup> Semester Students .....	12
4.2. 07 <sup>th</sup> Semester Students .....	12
4.3. 08 <sup>th</sup> Semester Students .....	12
<b>5. Appendix A- Documentation Templates .....</b>	<b>14</b>
<b>6. Appendix B- Evaluation Rubrics .....</b>	<b>62</b>
<b>7. Appendix C- FYP Forms .....</b>	<b>69</b>
<b>8. Appendix D- Meeting &amp; Evaluation Log .....</b>	<b>76</b>

# 1. INTRODUCTION

The Final Year Project (FYP) is a mandatory prerequisite for award of degree in BS (CS) and BS (SE) programs. At the department of Computer Science, CUI Islamabad campus, it is spread over the final three semesters as follows:

- **6<sup>th</sup> Semester:** Submission of FYP Scope document,
- **7<sup>th</sup> Semester:** Presentation and submission of Software Requirement Specification (SRS) and Software Design Document (SDD)
- **8<sup>th</sup> Semester:** Implementation.

## 1.1. Objectives

FYPs are considered as unitary of the core components for computer science and software engineering discipline students at the undergraduate levels. These projects play a key role in forming the students' mind-set towards performing real life projects. In summation, the FYP aims to encourage the student to integrate and use almost all core modules/forms they have examined during their undergraduate journey. The objectives of FYP include:

- To identify and formulate a computer-based system according to agreed requirements
- To use and implement latest tools and technologies to meet the requirement of the software industry.
- To apply software engineering theory and practices in the modelling and design of computer-based systems.
- To select and utilize the knowledge, techniques and skills of their respective discipline to produce a mandatory system.
- To work effectively in a team environment.
- To apply the professional, ethical, legal, and social issues and responsibilities.
- To present effectively to a range of audiences

## 1.2 Assigning Students to Supervisors and Projects

1. Faculty of Department of Computer Science, with designation of lecturer and above is eligible for supervision of FYP. Rest of the faculty can co-supervise.
2. Each faculty member eligible for supervision, must supervise or co-supervise **at least one** project and Maximum **THREE** FYPs every semester.

## 2. GUIDELINES

- The student who is eligible for FYP is encouraged to conform to the guidelines provided by FYP committee.
- In case of any complain/request, student should submit their issues in the prescribed Performa (See Appendix C) to their respective coordinators: FYP Coordinator-1, FYP Coordinator-II.
- If a student has any issue with their supervisor, evaluator or the respective FYP Coordinator, he or she can submit their complaint within three days of the incidence to the FYP Moderator. The FYP Convener will be the final authority to resolve the issue.
- For any updates and notifications regarding FYP, it is mandatory for every FYP student to join their respective FYP-FB groups. Link will be given to respective CRs.
- If a student fails to follow the SOPs/Guideline and show non-compliance on the deliverables assigned by FYP committee, he / she can be penalised according to SECTION 2.5.
- Research based projects must be based on substantial coding/implementation, GUI and must be in the form of a product.

### 2.1. Guidelines for Student

#### 2.1.1. Registration Guidelines

- **Registration:**
  - Each student doing FYP must register on CU online during registration week in 7<sup>th</sup> and 8<sup>th</sup> semester according to their Scheme of Studies.
- **Formation and number of students in an FYP Group:** There can be minimum 2 and maximum 3 members in an FYP group, depending on the scope of the project.

#### 2.1.2. Evaluation Folder:

- Each FYP will have an evaluation folder that include:
  - i- Meeting log with Supervisor (See Appendix D)
  - ii- Evaluator's comments/recommendations/suggestions at each milestone
  - iii- Supervisor remarks on the evaluators' comments /recommendations/suggestions
  - iv- All documentation related to FYP (e.g. Scope, SRS, and SDD etc.)

- Students must bring their evaluation folder complete in every aspect to the presentation venue for all milestone.
- Documentation at each milestone is to be punched and placed in the evaluation folder
  - Separate binding, file cover etc. are not allowed.

### **2.1.3. Meeting with Supervisor:**

1. Students should meet regularly with their respective supervisor and record their meeting log (duly signed by the supervisor) in the evaluation folder.
2. Each FYP group must submit minimum four meeting logs before evaluation of each milestone.
3. Students must incorporate all suggestion/comments given by evaluation committee and get it verified and signed by their supervisor.

**Note: Students will not contact evaluators after the evaluation slot. In case of any query contact respective supervisor.**

## **2.2.Role and Responsibilities of FYP Committee.**

Department of Computer Science at CUI, Islamabad Campus constituted a dedicated FYP committee to smoothly execute the FYP process. The committee consists of the following members and report to Head of Department:

1. Prof. Dr. Sohail Asghar and Dr. Manzoor Ilahi Tammimy (Convener)
2. Mr. Muhammad Haris (Moderator)
3. Mr Umar Iqbal(FYP Coordinator-1)
4. Mr. Jawed Tanveer (FYP Coordinator-2)

FYP Office is Room 248 at Second Floor of AB-II. Role of each stakeholder is defined as:

### **2.2.1 Role of Convener**

- Constitute an evaluation committee with the consensus of HoD.
- Control overall project related activities. (Direct whole FYP process)
- Coordinate with FYP Coordinators for all FYP events and activities.

### **2.2.2 Role of FYP Coordinator**

- Prepare and share (with students and faculty) FYP semester schedule with the approval of convener, before start of the semester.
- Organize and manage students FYP evaluation activities.
- Coordinate with convener, evaluator, supervisor and students.
- Process all cases regarding FYP Funding.
- Resolve any query related to all stakeholders.
- Coordinate with Account, audit, exam, transport and security offices.
- Submit Final FYP result to Exam department.

### **2.2.3 Role of Moderator**

FYP Moderator will be responsible/contact person for following activities

- Monitoring of FYP process.
- Contact person for conflict of interest among stakeholders.
- Recruitment Drives
- Recruitment Seminars
- Inter & Intra departmental activities
- FYP Industrial Collaboration & Participation in National Level Events

## **2.3. Role of Supervisor**

The role of supervisor includes: (may not be limited to this only)

- Discussing the project plan and suggesting changes if necessary.
- Ensure regular and effective meetings with students.
- Advising on the appropriate materials and methods/design/technology to use.
- Offering advice on sources of information for respective project.
- Advising on issues related to FYP report write up.
- Reading drafts of each chapter of FYP report.
- Ensure that their groups are following required standard SRS, SDD, and Presentation etc.
- Grade their FYP students after every milestone as per defined rubrics.

## 2.4.Role of Evaluation Committee

Each FYP group will be evaluated by a team of faculty members (referred as evaluators). They are collectively responsible for both:

- Critique and evaluate different stages of FYP (milestone) according to predefined rubrics (See Appendix B) along with the convener.
- Submit evaluations as per evaluation plan/ rubrics to Final Year Projects Office through prescribed method.

### SOPs for Evaluation Committee

- The evaluation team assigned at the 1<sup>st</sup> milestone (i.e. Scope Document Evaluation) will evaluate project throughout the year.
- The evaluator should make the presentation venue before time to avert any unnecessary delay.
- Evaluators will be assigned evaluation slots as per timetable.
- It is the evaluator's responsibility to ensure;
  - The smooth conduct of all evaluations according to announced FYP schedule.
  - If any evaluator is not available during the evaluation, he or she inform the respective coordinator and must provide their substitute.
  - Any issue regarding substitute arrangement, should be reported to the convener with CC to coordinators.
- Ensure completion of evaluation of milestone within allocated time (30 minutes).
- There will be 05 minutes for Q/A between evaluators and students at the end of the demonstration.
- After evaluation of project, there will be 05 minutes (if required) for discussion between evaluators and supervisor(s).
- Evaluators are required to complete the evaluation result sheets within the evaluation week and submit it to respective the FYP Coordinator.
- **Evaluators should not discuss evaluation matters with students after the evaluation slot.**

**Note: Evaluators are requested to follow announced schedule and time for smooth conduct of evaluation process.**

**Important Note:**

- i. The Evaluation Committee will decide whether to accept or reject the Project Proposal. If the proposal is rejected or some changes are suggested by the committee then the student will resubmit the Project Proposal in the next week.
- ii. Each group member will be graded individually depending on the performance of each student.
- iii. The external evaluator can be either from industry or academia depending on the nature of project.

**Note:** Students are advised not to contact the evaluators directly before or after the evaluation.

**2.5. Penalties**

Students are required to respect all due dates. In case of any violation following penalties will be given to students:

<b>Violation</b>	<b>Penalty</b>
<b>Late submission of Registration document</b>	<ul style="list-style-type: none"><li>• 5% marks will be deducted if required document is submitted after due date.</li><li>• Cancellation of project if required document is not submitted till first presentation</li></ul>
<b>Individual not attending Presentation</b>	<ul style="list-style-type: none"><li>• Zero marks will be awarded to the individual for that particular mile stone.</li></ul>
<b>Late submission of Spiral Binding</b>	<ul style="list-style-type: none"><li>• Up to one grade reduction.</li><li>• If not submitted till due date, FYP may be postponed till next semester</li></ul>
<b>Late submission of Final Book Binding</b>	Up to one grade reduction. If not submitted till due date, FYP may be postponed till next semester
<b>Plagiarism:</b>	Plagiarism in any part(s) of project will result in “F” in the FYP and case shall be forwarded to disciplinary committee as per University’s regulations.



<b>NON-COMPLIANCE</b>	<ul style="list-style-type: none"> <li>• For non-compliance from a student on the tasks assigned by FYP committee; <ul style="list-style-type: none"> <li>○ <b>For First Time: 30% Marks</b> will be deducted from the current mile stone.</li> <li>○ <b>Regular Non-Compliance: 10% Marks</b> deduction to a degradation grade on recommendation of coordinator, to be decided by convener FYP committee.</li> </ul> </li> </ul> <p><b>Probation:</b> If 02 probations are assigned by the evaluation team then the student(s) will present their FYP in next semester.</p>
-----------------------	--

## 2.6.FORMATTING AND LAYOUT

- Students will be provided template files for FYP document and presentation of each milestone through their respective FYP Facebook groups.
- FYP Template Files are appended. (See Appendix A)

### 3. Road Map for BS

#### 2. Supervisor Selection and Project Registration

- Student can only register for project if he/she successfully completed the core courses
- Project is worth 6 credit hours and ideally completed in two semesters
- Student should select supervisor according to the his/her working of interest

#### 4. Project Evaluation – I and SDS Submission

- Student will present his/her project implementation (30%) with SDS in front of FYP committee.
- SDS can be accepted with minor / major revision after incorporating any changes suggested by FYP committee.
- If project is plagiarised / behind the time line / not according to the scope, students will get conditional I.P.

#### 6. Project Evaluation - III

- Student will present his/her project implementation (100%) in front of FYP committee with complete project report.
- If project is plagiarised / behind the time line / not according to the scope, students will get permanent I.P and they will appear in next semester.
- If approved, submit 2 copies after incorporating changes

1

2

3

4

5

6

7

#### 1. Courses Completion

- Student can register for Final Year Project after completing the prerequisite courses as prescribed in the scheme of studies.

#### 3. Project Scope and SRS Submission and Defence

- Submission of scope document will be in 6<sup>th</sup> semester. Pre-Screening team will decide if the project idea is i) accepted ii) accepted with minor changes iii) accepted with major changes iv) Not accepted
- After acceptance, student will present his/her project scope with SRS in front of FYP committee.
- If approved, resubmit scope document after incorporating any changes suggested by FYP committee.
- Students will not be allowed to appear in front of the evaluation committee without evidence of at least four meetings with their supervisor before every milestone.

#### 5. Project Evaluation - II

- Student will present his/her project implementation (60%) in front of FYP committee.
- If project is plagiarised / behind the time line / not according to the scope, students will get conditional I.P.

#### 7. External Defence

- Student will present his/her work in front of nominated external examiner.
- Submit final approved report along with CD (containing complete project data) after incorporating changes suggested by external to the FYP secretaries.

## **4. Milestones and Activities**

### **4.1. 06<sup>th</sup> Semester Students**

- Collection of FYP proposals from faculty
- Introductory Seminar
- Registration of Students (Google form)
- Collection of Scope Documents
- Distribution of scope documents to evaluation team
- Collection of evaluated documents
- Sharing of evaluated scope docs to students
- Resubmission of rejected (Major change/ Rejected) FYPs
- Re-evaluation of modified scope docs

### **4.2. 07<sup>th</sup> Semester Students**

- Verification of FYP Credit Hours on Portal as per batch SOS
- Seminar of SRS
- Presentation of Scope Document and SRS document
- Compilation and sharing of result with students and supervisors
- Compilation and sharing of result with students and supervisors
- Seminar of SDS
- Presentation of SDS Document + 20% Implementation
- Compilation and sharing of result with students and supervisors
- Collection of Supervisor Marks
- Compilation of FYP-I Complete Result (Scope + SRS + SDS)
- Submission of FYP-I result to Exam Department
- Resolving queries of Exam & Students

### **4.3. 08<sup>th</sup> Semester Students**

- Verification of FYP Credit Hours on Portal as per batch SOS
- Introductory Seminar Regarding Deliverables, Implementation, Rubrics
- Evaluation of 60% Implementation
- Compilation and sharing of result with students and supervisors
- Evaluation of 100% Implementation + System Testing & Debugging
- Compilation and sharing of result with students and supervisors
- Consent of External Evaluators  
(Updated Address, Vehicle No, Contact Details, CNIC, NTN)
- Nomination of External Evaluators list to Exam Department
- Collection of Supervisor Marks
- Compilation and sharing of result with students and supervisors

- Collection of 02 Spiral Bindings from students
- Verification & Submission of Spiral Bindings to exam department
- Resolving queries of Exam related to External Evaluator & Spiral Binding
- Contact External Evaluator for availability (Date & Time)
- External Evaluation Schedule
  - External Availability
  - Students Availability
  - Resolving student terminal exam clashes
- Preparation of External Evaluation Sheets
- Conduct External evaluations
- Compilation of combined result (FYP-I & II)
- Compilation of complete Result Sheets  
(Committee + Supervisor + External marks)
- Verification of complete result
- Making Back up of complete result for record
- Collection of FYP Report Book Binding + DVD from students
- Verification of collected Book Bindings
- Submission of Result Sheets & Book Binding to Exam Department
- Resolving queries of Exam & Students
- Compilation and submission of ICEF form

## **APPENDIX A**

### **(DOCUMENTATION TEMPLATES)**



**COMSATS University Islamabad,  
Park Road, Chak Shahzad, Islamabad, Pakistan**

# **Project Proposal**

**(SCOPE DOCUMENT)**

**for**

**<PROJECT NAME>**

Version 1.0

*By*

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

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

*Supervisor*

**Supervisor Name**

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

## SCOPE DOCUMENT REVISION HISTORY

No.	Comment	Action

**Supervisor Signature**

**Date:**

## Table of Contents

<b>1. Introduction .....</b>	<b>17</b>
<b>2. Problem Statement.....</b>	<b>18</b>
<b>3. Problem Solution for Proposed System .....</b>	<b>18</b>
<b>4. Related System Analysis/Literature Review .....</b>	<b>18</b>
<b>5. Advantages/Benefits of Proposed System .....</b>	<b>18</b>
<b>6. Scope.....</b>	<b>19</b>
<b>7. Modules.....</b>	<b>19</b>
7.1     Module 1: Module Name.....	19
7.2     Module 2: Module Name.....	19
<b>8. System Limitations/Constraints .....</b>	<b>19</b>
<b>9. Software Process Methodology .....</b>	<b>19</b>
<b>10. Tools and Technologies.....</b>	<b>20</b>
<b>11. Project Stakeholders and Roles .....</b>	<b>20</b>
<b>12. Team Members Individual Tasks/Work Division.....</b>	<b>21</b>
<b>13. Data Gathering Approach.....</b>	<b>21</b>
<b>14. Concepts.....</b>	<b>21</b>
<b>15. Gantt chart .....</b>	<b>22</b>
<b>16. Mockups.....</b>	<b>23</b>
<b>17. Conclusion .....</b>	<b>24</b>
<b>18. References .....</b>	<b>24</b>
<b>19. Plaragism Report .....</b>	<b>24</b>



**Project Category:** (Select all the major domains of proposed project)

☐ **A**-Desktop Application/Information System ☐ **B**-Web Application/Web Application based Information System  
☐ **C**- Problem Solving and Artificial Intelligence ☐ **D**-Simulation and Modeling ☐ **E**- Smartphone Application ☐ **F**-  
Smartphone Game ☐ **G**- Networks ☐ **H**- Image Processing ☐ Other (specify category) \_\_\_\_\_

## **Abstract**

Provide a one to two paragraph summary of your project. The abstract should give an idea of what your project is trying to achieve. Think of your abstract as a condensed version of your whole project. By reading it, the reader should understand the nature of your project. It should be comprehensive, and concise.

## Introduction

Specify the purpose of this project proposal document along with description detailed background of the system. (Usually in 10-12 sentences).

## Problem Statement

**What** problem does your software solve? **Why** you are developing this system? **Does** the same system already exist? **If yes**, how will a re-implementation aid your learning? **What** skills do you expect to learn from this project? (Usually in 14-16 sentences)

## Problem Solution for Proposed System

Briefly explain how your system solves the problems mentioned in the problem statement. (Usually in 14-16 sentences)

## Related System Analysis/Literature Review

**Write** about the existing/similar systems related to your proposed project. **At least three** existing systems should be discussed.

Don't use more than 4 sentences for explaining a single system/application.

Briefly explain the related system analysis which help to specify the contribution of the proposed project.

(**Note:** Research based projects may provide literature review instead of related system analysis.)

**Table 1 Related System Analysis with proposed project solution**

<b>Application Name</b>	<b>Weakness</b>	<b>Proposed Project Solution</b>
The name of related application(s).	Weaknesses may include limited features, low quality functionality and processes.	The way the proposed project mitigates the weaknesses.

## Advantages/Benefits of Proposed System

Write down the advantages and benefits of the system you are developing. In other words, you have to mention here advantage of your proposed solution to the existing problem.

(Usually in 5-7 advantages)

## Scope

Write down the scope of your project in a paragraph. Briefly define the main functionalities of the proposed project. Scope defines the boundaries and range of the proposed solution, that what would be the part of your project and what will be not. Write down in logical flow with consistency.  
(Usually in 14-18 sentences)

## Modules

Write down the modules of the proposed project. Don't forget to mention special/new features. Briefly explain your one module in 6 to 8 sentences.  
(Note: Usually 5-6 Modules for 2 student's projects and 8-9 modules for 3 student's project)

**Explanation of a Module:** Module is a section of a program that performs a task. Programs consist of modules, each of which contains one or more routines. The term routine is synonymous with procedure, function, and subroutine.

Example:

*Enterprise resource planning (ERP) software - is comprised of several large modules (for example, finance, supply chain and payroll, etc.), which may be implemented with little or no customization.*

(Briefly explain each module with respect to major functionality in user context)

**Module 1: Module Name**

**Module 2: Module Name**

## System Limitations/Constraints

Write down the limitations and constraints of the proposed project.  
(Usually 2-4 constraints)

## Software Process Methodology

Write down your software methodology/ software process that will be used for project development. Also mention why you have chosen this methodology. (Usually 3-5 sentences)

1. You can use Object Oriented Methodology, or Procedural methodology.
2. Choice of methodology will affect choice of tools and technologies
3. Choice of methodology will affect nature of design (SDS)
4. Choice should be made depending on your expertise and your needs e.g. most simulation and device level software can only be programmed in procedural languages.

## Tools and Technologies

Mention all the hardware/software tools and technologies with version number which will be used in implementation of the project. Write about the APIs, language(s), SDK(s) etc. which you will use for implementation.

Example:

**Table 2 Tools and Technologies for Proposed Project**

<b>Tools And Technologies</b>	<b>Tools</b>	<b>Version</b>	<b>Rationale</b>
	MS Visual Studio	2015	IDE
	MS SQL Server	2015	DBMS
	Adobe Photoshop	CSC 6	Design Work
	MS Word	2015	Documentation
	MS Power Point	2015	Presentation
	Pencil	2.0.5	Mockups Creation
	<b>Technology</b>	<b>Version</b>	<b>Rationale</b>
	C#	6.0	Programming language
	SQL	2013	Query Language
	Html	5	Web Development

## Project Stakeholders and Roles

Write down the project stakeholders and their roles.

**Table 3 Project Stakeholders for Proposed Project**

<b>Project Sponsor</b>	<p>All web applications and desktop applications should have real client. Mention your project sponsor. Default option will be: COMSATS University, Islamabad</p>
<b>Stakeholder</b>	<p>Mention your stake holders with their roles and responsibilities. Default option will be</p> <ul style="list-style-type: none"> <li>• Students names</li> <li>• Project Supervisor Name: Mr./Miss ...</li> <li>• Final Year Project Committee: Evaluation of project</li> </ul>

## Team Members Individual Tasks/Work Division

Table 4 Team Member Work Division for Proposed Project

Student Name	Student Registration Number	Responsibility/ Modules
Student 1 Name	Student 1 Registration Number	Describe the work division of each student along with modules E.g. Mr. Ali (Module1-Module3) Augmented reality and Databases tasks.

## Data Gathering Approach

Write down information and requirement gathering approaches for proposed project e.g. Interview, Questionnaire etc. (Usually 3-5 sentences)

## Concepts

Mention the concepts that you will learn while doing the proposed project.

For example: Augmented Reality, Virtual Reality, Algorithms, API's Code injection, Closures, VI technique etc.

Not more than 4 sentences for one concept. (Usually 3-5 concepts are briefly mentioned)

Example:

*Concept-1: Concept Name E.g. Augmented Reality (Briefly give the overview of concept with respect to your project)*

## Gantt chart

Create the Gantt Chart and provide estimated start and end dates of all proposed modules/tasks for each team member. Also identify the dependencies (which tasks cannot be started/completed, until the dependent task is completed). Gantt chart can be created using MS Project.

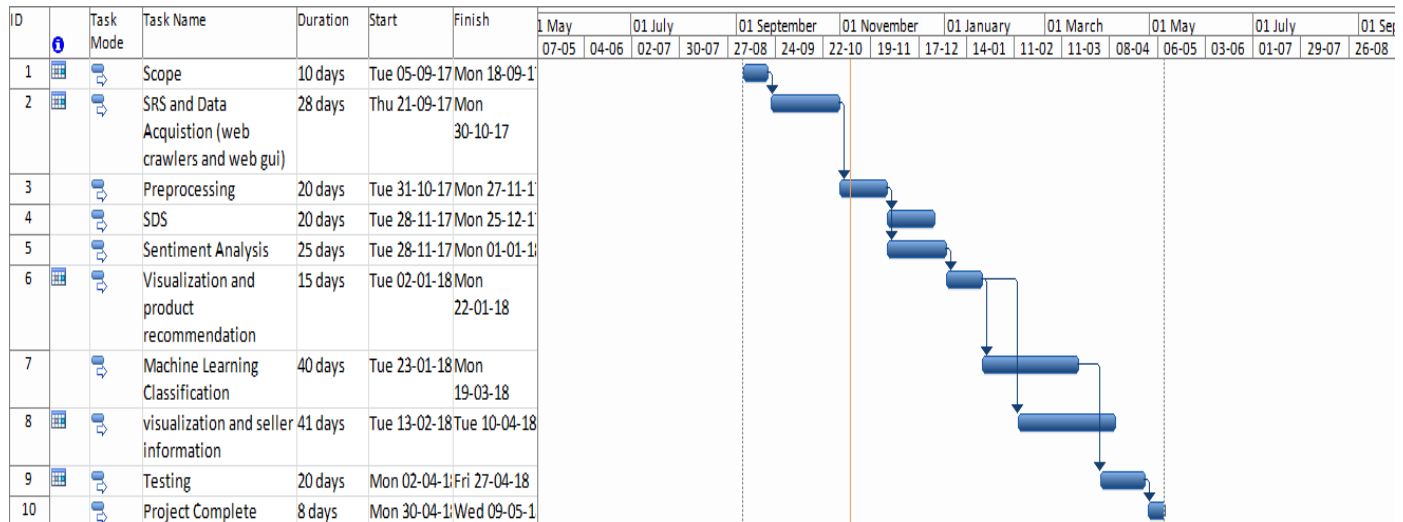
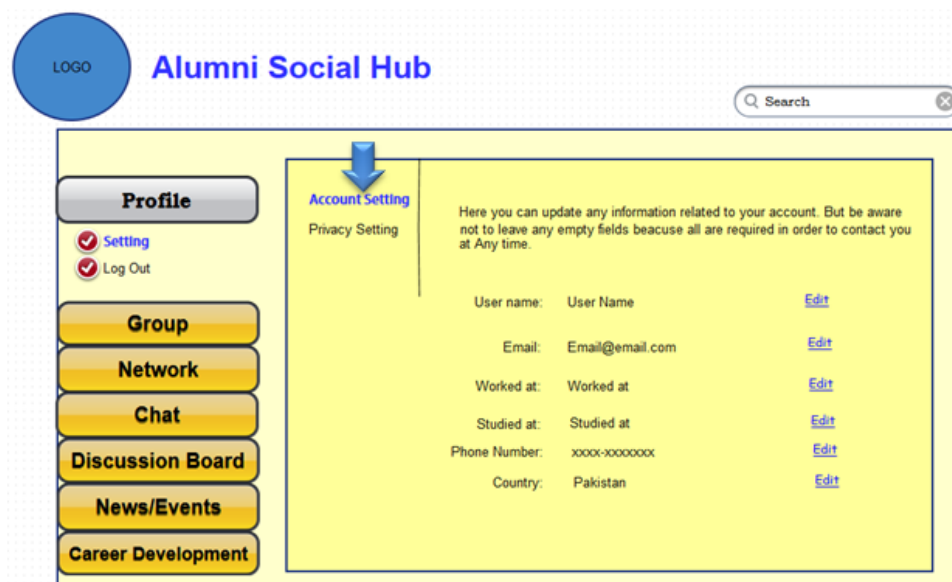


Figure 1 Sample Gantt chart

# Mockups

Insert minimum mockups (Usually 4-6 mockups) which show the major modules mentioned in the scope section of the document. Do not include mockups for Login, Signup, Forgot Password, Contact Us, About Us etc. If the project is a Web or a Smartphone Application, then include at least three mockups from each part of the project. You can design mockup in any design tool for example pencil tool (<https://pencil.evolus.vn/>) or Balsamiq (<https://balsamiq.com/>)



## **Conclusion**

Conclude this document. (Usually 4-5 sentences)

## **References**

Mention the books, research papers, web links etc.

## **Plagiarism Report**

Attach the Plagiarism report of your project scope document from library staff of turnitin tool (<http://turnitin.com>)





**COMSATS University Islamabad,  
Park Road, Chak Shahzad, Islamabad, Pakistan**

# **SOFTWARE REQUIREMENTS SPECIFICATION**

**(SRS DOCUMENT)**

**for**

**<PROJECT NAME>**  
Version 1.0

***By***

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

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

***Supervisor***

**Supervisor Name**

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

# Table of Contents

<b>Revision History .....</b>	<b>27</b>
<b>1. Introduction.....</b>	<b>29</b>
1.1 Purpose .....	29
1.2 Scope .....	29
<b>2. Overall description.....</b>	<b>29</b>
2.1 Product perspective .....	29
2.2 Operating environment .....	29
2.3 Design and implementation constraints.....	29
<b>3. Requirement identifying technique .....</b>	<b>30</b>
3.1 Use Case Diagram .....	30
3.2 Use Case Description.....	30
<b>4. Specific Requirements .....</b>	<b>32</b>
4.1 System feature X .....	32
<b>5. Quality attributes .....</b>	<b>Error! Bookmark not defined.</b>
5.1 Usability .....	33
5.2 Performance.....	34
<b>6. External interface requirements.....</b>	<b>Error! Bookmark not defined.</b>
6.1 User interfaces .....	<b>Error! Bookmark not defined.</b>
6.2 Software interfaces .....	<b>Error! Bookmark not defined.</b>
6.3 Hardware interfaces .....	<b>Error! Bookmark not defined.</b>
6.4 Communications interfaces .....	<b>Error! Bookmark not defined.</b>
<b>7. Project Gantt chart .....</b>	<b>Error! Bookmark not defined.</b>
<b>8. References .....</b>	<b>34</b>

## Revision History

Name	Date	Reason for changes	Version

### Application Evaluation History

<b>Comments (by committee)</b> *include the ones given at scope time both in doc and presentation	<b>Action Taken</b>

**Supervised by**  
**<Supervisor's Name>**

Signature\_\_\_\_\_

# Introduction

The introduction presents an overview to understand how the SRS is organized and how to use it.

## Purpose

Identify the product or application whose requirements are specified in this document.

## Scope

Provide a short description of the software being specified and its purpose. You might provide a high-level summary of the major features the software contains or the significant functions that it performs.

# Overall description

## Product perspective

Describe the product's context and origin. Is it the next member of a growing product line, the next version of a mature system, a replacement for an existing application, or an entirely new product?

## Operating environment

Describe the environment in which the software will operate, which might include the hardware platform; operating systems and versions; geographical locations of users, servers, and databases; and organizations that host the related databases, servers, and websites.

Example:

*OE-1: The System shall operate correctly with the following web browsers: Windows Internet Explorer versions 7, 8, and 9; Firefox versions 12 through 26; Google Chrome (all versions); and Apple Safari versions 4.0 through 8.0.*

## Design and implementation constraints

There are times when a certain programming language must be used, a code library that has already had time invested to develop it needs to be used, and so forth. Describe any factors that will restrict the options available to the developers and the rationale for each constraint. Constraints are described further in Chapter 14<sup>1</sup>, "Beyond functionality."

Example:

---

<sup>1</sup>Karl Wiegers and Joy Beatty, Software Requirements 3<sup>rd</sup> edition.

Note: All the referenced chapters are selected from the above book

*CO-1: The system shall use the current corporate standard Oracle database engine*

## Requirement identifying technique

This section describes the requirements identifying technique(s) which further help to derive functional requirements specification. The selection of the technique(s) will depend on the type of project. For instance,

- **Use case** is an effective technique for interactive end-user applications
- **Event-response tables** is for real time system and
- **Story boarding** for graphically intensive applications.

In addition to above, the projects involving data warehouses, batch processes, hardware devices with embedded control software, and computationally intensive applications required to follow other suitable techniques. Such techniques are described further in Chapter 12, “A picture is worth 1024 words.” For documenting this section let consider identifying requirements through use case as an example.

### Use case diagram

Create a use case diagram using **MS Visio** for your system. For detail guideline to develop use case diagram, follow any of latest **UML book**]

### Use case description

The table below indicate a comprehensive use case template filled in with an example drawn from the Cafeteria ordering system (COS). (Appendix C) shows more sample use cases written according to this template. As with all templates, you don’t complete this from top to bottom, and you don’t necessarily need all the template information for every use case. The template is simply a structure in which to store the information you encounter during a use case discussion in an organized and consistent fashion. The template reminds you of all the information you should contemplate regarding each use case. For more detail see Chapter 8, “Understanding user requirements”

**Table 5 Show the detail use case template**

<b>Use Case ID:</b>	Enter a unique numeric identifier for the Use Case. e.g. UC-1
<b>Use Case Name:</b>	Enter a short name for the Use Case using an active verb phrase. e.g. Order a Meal
<b>Actors:</b>	[An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks.] e.g. Primary Actor: Patron Secondary Actors: Cafeteria Inventory System
<b>Description:</b>	[Provide a brief description of the reason for and outcome of this use case.] e.g. A Patron accesses the Cafeteria Ordering System from either the corporate intranet or external Internet, views the menu for a specific date, selects food

	items, and places an order for a meal to be picked up in the cafeteria or delivered to a specified location within a specified 15-minute time window.
<b>Trigger:</b>	[Identify the event that initiates the use case.]e.g. A Patron indicates that he wants to order a meal.
<b>Preconditions:</b>	[List any activities that must take place, or any conditions that must be true, before the use case can be started. PRE-1. Patron is logged into COS. PRE-2. Patron is registered for meal payments by payroll deduction.
<b>Postconditions:</b>	[Describe the state of the system at the conclusion of the use case execution. POST-1. Meal order is stored in COS with a status of “Accepted.” POST-2. Inventory of available food items is updated to reflect items in this order. POST-3. Remaining delivery capacity for the requested time window is updated.
<b>Normal Flow:</b>	[Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. 1.0 Order a Single Meal 1. Patron asks to view menu for a specific date. (see 1.0. E1, 1.0.E2) 2. COS displays menu of available food items and the daily special. 3. Patron selects one or more food items from menu. (see 1.1) 4. Patron indicates that meal order is complete. (see 1.2) 5. COS displays ordered menu items, individual prices, and total price, including taxes and delivery charge. 6. Patron either confirms meal order (continue normal flow) or requests to modify meal order (return to step 2). 7. COS displays available delivery times for the delivery date. 8. Patron selects a delivery time and specifies the delivery location. 9. Patron specifies payment method. 10. COS confirms acceptance of the order. 11. COS sends Patron an email message confirming order details, price, and delivery instructions. 12. COS stores order, sends food item information to Cafeteria Inventory System, and updates available delivery times.
<b>Alternative Flows:</b> [Alternative Flow 1 – Not in Network]	[Document legitimate branches from the main flow to handle special conditions (also known as extensions). For each alternative flow reference the branching step number of the normal flow and the condition which must be true for this extension to be executed. e.g. 1.1 Order multiple identical meals 1. Patron requests a specified number of identical meals. (see 1.1. E1) 2. Return to step 4 of normal flow. 1.2 Order multiple meals 1. Patron asks to order another meal. 2. Return to step 1 of normal flow. Note: Insert a new row for each distinctive alternative flow. ]
<b>Exceptions:</b>	1.0. E1 Requested date is today and current time is after today’s order cutoff time 1. COS informs Patron that it’s too late to place an order for today. 2a. If Patron cancels the meal ordering process, then COS terminates use case. 2b. Else if Patron requests another date, then COS restarts use case.

	<p>1.0. E2 No delivery times left</p> <p>1. COS informs Patron that no delivery times are available for the meal date.</p> <p>2a. If Patron cancels the meal ordering process, then COS terminates use case.</p> <p>2b. Else if Patron requests to pick the order up at the cafeteria, then continue with normal flow, but skip steps 7 and 8.</p> <p>1.1. E1 Insufficient inventory to fulfill multiple meal order</p> <p>1. COS informs Patron of the maximum number of identical meals he can order, based on current available inventory.</p> <p>2a. If Patron modifies number of meals ordered, then return to step 4 of normal flow.</p> <p>2b. Else if Patron cancels the meal ordering process, then COS terminates use case.</p>
<b>Business Rules</b>	<p>Use cases and business rules are intertwined. Some business rules constrain which roles can perform all or parts of a use case. Perhaps only users who have certain privilege levels can perform specific alternative flows. That is, the rule might impose preconditions that the system must test before letting the user proceed. Business rules can influence specific steps in the normal flow by defining valid input values or dictating how computations are to be performed e.g.</p> <p>BR-1 Delivery time windows are 15 minutes, beginning on each quarter hour.</p> <p>BR-2 Deliveries must be completed between 11:00 A.M. and 2:00 P.M. local time, inclusive.</p> <p>Note: If you are maintaining the business rule in a separate table in SRS then only mention here their IDs.</p>
<b>Assumptions:</b>	<p>[List any assumptions.</p> <p>1. e.g. Assume that 15 percent of Patrons will order the daily special (Source: previous 6 months of cafeteria data).</p>

## Functional Requirements

This section describes the functional requirements of the system expressed in natural language style. This section is typically organized by feature as system feature name and specific functional requirements associated with this feature. It is just one possible way to arrange them. Other organizational options include arranging functional requirements by use case, process flow, mode of operation, user class, stimulus, and response depend what kind of technique which has been used to understand functional requirements. Hierarchical combinations of these elements are also possible, such as use cases within user classes. For further detail see Chapter 10 “Documenting the requirements”. Let consider feature scheme as an example.

### Functional Requirement X

Itemize the specific functional requirements associated with each feature. These are the software capabilities that must be implemented for the user to carry out the feature’s services or to perform a use case. Describe how the product should respond to anticipated error conditions and



to invalid inputs and actions. Uniquely label each functional requirement, as described earlier. You can create multiple attributes for each functional requirement, such as rationale, source, dependencies etc. The following template is required to write functional requirements. For further detail see Chapter 11” Writing excellent requirements”.

**Table 6 Show the functional requirement template**

<b>Identifier</b>	Requirement ID
<b>Title</b>	Title of requirement
<b>Requirement</b>	<p>Description of requirement which may be written either from user or system perspective e.g.</p> <p>If written in <b>user perspective</b></p> <p>The [user class or actor name] shall be able to [do something] [to some object] [qualifying conditions, response time, or quality statement].</p> <p>If written in <b>system perspective</b></p> <p>[optional precondition] [optional trigger event] the system shall [expected system response]</p>
<b>Source</b>	Where this requirement is come from (who originate it)
<b>Rationale</b>	Motivation behind the requirement
<b>Business Rule (if required)</b>	Any restriction, policy, rule that the particular requirement must be fulfilled through its functional behavior
<b>Dependencies</b>	Requirements ID that are dependent on this requirement
<b>Priority</b>	High/Medium/Low

## Non Functional Requirements

This section specifies nonfunctional requirements other than constraints, which are recorded in section 2.3, and external interface requirements, which will appear in section 7. These quality requirements should be specific, quantitative, and verifiable. Chapter 14 “beyond functionality” presents more information about these quality attribute requirements and many examples. Following are some example for documenting guideline.

### Usability

Usability requirements deal with ease of learning, ease of use, error avoidance and recovery, efficiency of interactions, and accessibility. The usability requirements specified here will help the user interface designer create the optimum user experience.

Example:

*USE-1: The COS shall allow a user to retrieve the previous meal ordered with a single interaction.*

## Performance

State specific performance requirements for various system operations. If different functional requirements or features have different performance requirements, it's appropriate to specify those performance goals right with the corresponding functional requirements, rather than collecting them in this section.

Example:

*PER-1: 95% of webpages generated by the COS shall download completely within 4 seconds from the time the user requests the page over a 20 Mbps or faster Internet connection.*

## References

List any documents or other resources to which this SRS refers, if any. These might include user interface style guides, standards, system requirements specifications, interface specifications, or the SRS for a related product.



**COMSATS University Islamabad,  
Park Road, Chak Shahzad, Islamabad, Pakistan**

# **SOFTWARE DESIGN DESCRIPTION**

**(SDD DOCUMENT)**

**for**

**<PROJECT NAME>**

Version 1.0

***By***

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

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

***Supervisor***

**Supervisor Name**

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

# Table of Contents

<b>Revision History .....</b>	<b>37</b>
<b>1. Introduction.....</b>	<b>39</b>
<b>2. Design Methodology and software process model .....</b>	<b>39</b>
<b>3. System Overview .....</b>	<b>39</b>
3.1 Architectural Design.....	39
3.2 Process Flow/Representation.....	39
<b>4. Design Models [along with descriptions] .....</b>	<b>39</b>
<b>5. Data Design.....</b>	<b>40</b>
5.1 Data Dictionary .....	40
<b>6. Algorithm &amp; Implementation .....</b>	<b>40</b>
<b>7. Software Requirements Traceability Matrix .....</b>	<b>40</b>
<b>8. Human Interface Design.....</b>	<b>41</b>
8.1 Screen Images.....	41
8.2 Screen Objects and Actions .....	41
<b>9. Appendix I .....</b>	<b>41</b>

## Revision History

Name	Date	Reason for changes	Version

### Application Evaluation History

<b>Comments (by committee)</b> *include the ones given at scope time both in doc and presentation	<b>Action Taken</b>

**Supervised by**  
**<Supervisor's Name>**

Signature\_\_\_\_\_

## Introduction

Briefly explain scope of the project covered till now including modules.

## Design methodology and software process model

Explain and justify the choice of design methodology being followed. (OOP or procedural). Also explain which process model are you following and why.

## System overview

Give a general description of the functionality, context and design of your project. Provide any background information if necessary.

## Architectural design

Develop a modular program structure and explain the relationships between the modules to achieve the complete functionality of the system. This is a high-level overview of how the system's modules collaborate with each other in order to achieve the desired functionality.

Don't go into too much detail about the individual subsystems. The main purpose is to gain a general understanding of how and why the system was decomposed, and how the individual parts work together.

Provide a diagram showing the major subsystems and their connections. **Use a simple Line-Box-Diagram for simpler systems and detailed diagrams (MVC, Client-Server, Layered, Multi-tiered) for complex systems.**

## Process flow/Representation

Provide a representation of the flow of **MAJOR processes** of your system in the form of an activity diagram. **DO NOT CREATE ACTIVITY DIAGRAMS FOR LOGIN OR SIGN-UP UNLESS THEY INVOLVE SIGNIFICANT COMPLEXITY.** Include only the major processes.

## Design models [along with descriptions]

The applicable models may include:

- Class Diagram
- Sequence Diagram
- State Transition Diagram
- Data Flow Diagram
- Schematic diagram (Hardware projects only)
- Timing diagram (Hardware projects only)

Insert applicable system models here.

You should be clear about all the concepts used in your diagrams for example for class diagram you should know about aggregation, composition, inheritance/generalization. Also ensure visibility of all diagrams.

Class diagram and associated models shall only be necessary for object-oriented approach. In case of procedural, create a DFD. Data flow diagram should be extended to 2-3 levels. It should clearly list all processes, their sources/sinks and data stores.

**Note: System design should be complete in all aspects. Create any/all diagrams if you need to. A DFD can also be supplemented by a State Transition Diagram depending on the nature of the project.**

**Hardware projects can include Schematic diagram, System block diagram, timing diagram, Flow charts as replacement of sequence diagram/ Data flow diagram AFTER CONSULTATION WITH THEIR SUPERVISORS. Choice of models must be properly justified.**

## Data design

Explain how the information domain of your system is transformed into data structures. Describe how the major data or system entities are stored, processed and organized.

List any databases or data storage items.

## Data dictionary

Alphabetically list the system entities or major data along with their types and descriptions. If you provided a functional description, list all the functions and function parameters. If you provided an OO description, list the objects and its attributes, methods and method parameters.

## Algorithm & Implementation

In this section, we take a closer look at what each component does in a more systematic way. Provide a summary of your algorithm for each function listed in procedural description language (PDL) or pseudo code.

If you gave an OO description, summarize each object member function for all the objects listed in PDL or pseudocode. Describe any local data when necessary.

## Software requirements traceability matrix

This section should contain a table that summarizes how each software requirement has been met in this document. The tabular format permits one-to-one and one-to-many relationships to be shown.

**Table 7 Requirements Traceability Matrix**

Req. Number	Ref. Item	Design Component	Component Items
FR01	Class Diagram	ClassName	FunctionName(s)



OR			
FR01	DFD	DiagramNumber/Level	FunctionName(s)

## Human interface design

Describe the functionality of the system from the user's perspective. Explain how the user will be able to use your system to complete all the expected features and the feedback information that will be displayed for the user.

### Screen images

Display screenshots showing the interface from the user's perspective. These can be hand-drawn, or you can use an automated drawing tool. Just make them as accurate as possible. (Graph paper works well.)

### 8.2 Screen objects and actions

A discussion of screen objects and actions associated with those objects

## Appendix I

- How to design using UML (OOP): For guidance please follow the instructions mentioned in the link: <http://agilemodeling.com/artifacts/>
- How and when to design ER diagrams: For guidance please follow the instructions mentioned in the link: [http://people.inf.elte.hu/nikovits/DB2/Ullman\\_The\\_Complete\\_Book.pdf](http://people.inf.elte.hu/nikovits/DB2/Ullman_The_Complete_Book.pdf)
- Data flow diagrams: For guidance please follow the instructions mentioned in the link and book:
  - <http://www.agilemodeling.com/artifacts/dataFlowDiagram.htm>
  - Software Engineering –A Practitioner's approach by Roger Pressman
- Architecture diagram: For guidance please follow the instructions mentioned in the link and book:
  - Ian Sommerville – Software Engineering 9th Edition– Chapter 6



**COMSATS University Islamabad**  
**Islamabad, Pakistan**

**Project Name**

***By***

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

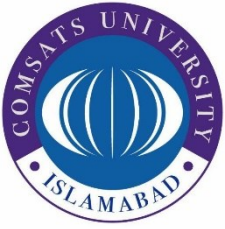
**Student Name 2      CIIT/SP09-BCS-xxx/ISB**

***Supervisor***

**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 Pakistan**

## **Project Name**

**A project presented to  
COMSATS Institute of Information Technology, 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 Name 1

-----

Student Name 2

-----

# 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/SE/TN-000)** and **STUDENT 2 NAME (CIIT/FAXX-BCS/SE/TN-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**

-----  
**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 locations 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.

**Customizable Surveillance Control System (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 Name 1

-----

Student Name 2

-----

# Abbreviations

<b>SRS</b>	Software Require Specification
<b>PC</b>	Personal Computer



# Table of Contents

<b>1</b>	<b><u>Introduction</u></b>	51
1.1	<u>Brief</u>	51
1.2	<u>Relevance to Course Modules</u>	51
1.3	<u>Project Background</u>	51
1.4	<u>Literature Review</u>	51
1.5	<u>Analysis from Literature Review (in the context of your project)</u>	51
1.6	<u>Methodology and Software Lifecycle for This Project</u>	51
1.6.1	<u>Rationale behind Selected Methodology</u>	51
<b>2</b>	<b><u>Problem Definition</u></b>	52
2.1	<u>Problem Statement</u>	52
2.2	<u>Deliverables and Development Requirements</u>	52
2.3	<u>Current System (if applicable to your project)</u>	52
<b>3</b>	<b><u>Requirement Analysis</u></b>	52
3.1	<u>Use Cases Diagram(s)</u>	53
3.2	<u>Detailed Use Case</u>	53
3.3	<u>Functional Requirements</u>	53
3.4	<u>Non-Functional Requirements</u>	53
<b>4</b>	<b><u>Design and Architecture</u></b>	54
4.1	<u>System Architecture</u>	54
4.2	<u>Data Representation [Diagram + Description]</u>	54
4.3	<u>Process Flow/Representation</u>	54
4.4	<u>Design Models [along with descriptions]</u>	54
<b>5</b>	<b><u>Implementation</u></b>	55
5.1	<u>Algorithm</u>	55
5.2	<u>External APIs</u>	55
5.3	<u>User Interface</u>	55
<b>6</b>	<b><u>Testing and Evaluation</u></b>	56
6.1	<u>Manual Testing</u>	56
6.1.1	<u>System testing</u>	56
6.1.2	<u>Unit Testing</u>	56
6.1.3	<u>Functional Testing</u>	57
6.1.4	<u>Integration Testing</u>	57
6.2	<u>Automated Testing:</u>	58
	<u>Tools used:</u>	58
<b>7</b>	<b><u>Conclusion and Future Work</u></b>	59
7.1	<u>Conclusion</u>	59
7.2	<u>Future Work</u>	59
<b>8</b>	<b><u>References</u></b>	60

# List of Figures

Fig 1.1 Block Diagram.....	8
Fig 2.1 Use Case Diagram .....	9

## **Introduction**

This chapter provides the overview of the project. The first paragraph of every chapter should provide the chapter summary.

### **Brief**

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.

### **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 describes current trends/ research/ products etc. related to your project.

### **Analysis from Literature Review (in the context of your project)**

This section provides an analytical discussion of your work in comparison with discussion in literature review.

### **Methodology and Software Lifecycle for This Project**

A brief discussion of methodology and SDLC model selected for this project.

### **Rationale behind Selected Methodology**

Why you selected above methodology (such as structural and Object Oriented) and software life cycle for this project?

### ***Rationale behind Selected Methodology***

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

## **Problem Definition**

This chapter discusses the precise problem to be solved. It should extend to include the outcome.

### **Problem Statement**

Problem statement goes here.

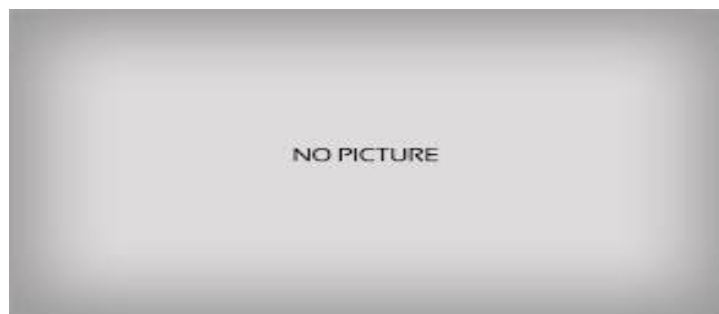
### **Deliverables and Development Requirements**

Deliverables and development requirements.

### **Current System (if applicable to your project)**

A brief description of an existing 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.



**Figure 2.1: Sample picture**

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 32.1: Sample Table**

Header 1	Header 2	Header 3
Text	Text	Text

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

- List items 1
- List items 2

## **Requirement Analysis**

The following parts of Software Requirements Specification (SRS) report should be included

in this chapter.

## **Use Cases Diagram(s)**

## **Detailed Use Case**

## **Functional Requirements**

## **Non-Functional Requirements**

## **Design and Architecture**

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

### **System Architecture**

### **Data Representation [Diagram + Description]**

### **Process Flow/Representation**

### **Design Models [along with descriptions]**

## 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 8 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.

## Testing and Evaluation

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

### Manual Testing

#### 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.

#### Unit Testing

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

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

No.	Test case/Test script	Attribute value and	Expected result	Result
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.	Pass
2.				

##### Unit Testing 2: Edit Profile

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

No.	Test case/Test script	Attribute value and	Expected result	Result
1.				
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

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

No.	Test case/Test script	Attribute and value	Expected result	Result
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	Pass
2.				

### Integration Testing

No.	Test case/Test script	Attribute and value	Expected result	Result
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	Pass
2.	Upload student record for Project 1	-	File successfully uploaded and return to the upload page. Student records are updated.	Pass
3.	View supervising student	-	The list of supervisees shown on the screen.	Pass

## **Automated Testing:**

### **Tools used:**

<b>Tool Name</b>	<b>Tool Description</b>	<b>Applied on [list of related test cases / FR / NFR]</b>	<b>Results</b>

## **Conclusion and Future Work**

This chapter concludes the project and highlights future work.

### **Conclusion**

### **Future Work**

## 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 Nishimura and Turu Ishida, (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)**

## **APPENDIX B (EVALUATION RUBRICS)**

## SCOPE

Criteria	Marginal	Adequate	Good	Excellent
<b>CLO-1(marks:5)</b> <b>Team coordination in terms of work division and collaboration</b>	Poor team coordination and poor work division	Poor team coordination and work division is satisfactory	good team coordination and work division is satisfactory	Excellent in team coordination and work division
<b>CLO-2(marks:8)</b> <b>1-Major Modules Completion status (atleast 30/60/100 percent of completion w.r.t to approved scope document)</b>	Major modules completion is at least 20% of the required percentage.	Major modules completion is at least 50% of the required percentage.	Major modules completion is at least 80% of the required percentage.	Major modules completion is at least 100% of the required percentage.
<b>CLO-5(marks:7)</b> <b>Software Testing according to required percentage</b> <b>1-Validation (Field Level, Code Level etc.) (2)</b>	Software validation has been applied on 20% of the implementation	Software validation has been applied on 50% of the implementation	Software validation has been applied on 80% of the implementation	Software validation has been applied on 100% of the implementation
<b>2-Test cases verification based on selected testing methodology (automated/manual) (3)</b>	Testing methodology has been defined and test case verification is applied on 20% of the implementation	Testing methodology has been defined and test case verification is applied on 50% of the implementation	Testing methodology has been defined and test case verification is applied on 80% of the implementation	Testing methodology has been defined and test case verification is applied on 100% of the implementation
<b>2-Look and Feel of User interfaces. (Visual objects and elements, visibility and flows of tasks etc) (2)</b>	Look and feel of user interface is poor according to HCI standards	Look and feel of user interface is satisfactory according to HCI standards	Look and feel of user interface is good according to HCI standards	Look and feel of user interface is excellent according to HCI standards
<b>CLO-4(marks:10)</b> <b>Followed proper coding standards/conventions (2)</b>	Very little understanding and usage of coding standards	Some understanding and usage of coding standards	Good understanding and usage of coding standards	Excellent understanding and usage of coding standards
<b>1-Understanding of implemented Algorithms or APIs or DB schema synchronization or Image processing technique(s) etc (6)</b>	Student has poor understanding of implementation	Student has satisfactory understanding of implementation	Student has good understanding of implementation	Student has excellent understanding of implementation
<b>2-Used suitable technologies to develop the software (2)</b>	Very little understanding of the suitability of the used technology	Some understanding of the suitability of the used technology	Good understanding of the suitability of the used technology	Excellent understanding of the suitability of the used technology
<b>CLO-3(marks:5)</b> <b>1-Is implementation according to proposed solution</b>	20% implementation is according to proposed solution	50% implementation is according to proposed solution	80% implementation is according to proposed solution	100% implementation is according to proposed solution
<b>CLO-6(marks:5)</b> <b>1-Is the student able to communicate effectively? (2)</b>	Answer at least one question correctly. Need clarification.	Answer most questions correctly. Need clarification sometimes.	Answer most questions correctly and concisely	Handle difficult questions with ease and confidence. Illustrative explanation.
<b>2-Is the document well formatted and grammatically correct? (3)</b>	The document is poorly formatted with many grammatical mistakes	The document is partially formatted with few grammatical mistakes	The document is well formatted with few grammatical mistakes	The document is well formatted with few grammatical correct
<b>Total = 40</b>				

## SRS

Criteria	Marginal	Adequate	Good	Excellent
<b>CLO-1(marks:2.5)</b> <b>Is the SRS documentation &amp; presentation free of Plagiarism? (1)</b>	The SRS document and presentation contain plagiarism between 21-30%	The SRS document and presentation contain plagiarism between 16-20%	The SRS document and presentation contain plagiarism between 6-15%	The SRS document and presentation contain plagiarism between 0-5%
<b>CLO-6(marks:10)</b> <b>1-Is the student wearing proper attire &amp; is he/she presentable? (3)</b>	Student's attire is barely acceptable	Student's attire is appropriate	Student's attire is good	Student's attire is excellent
<b>2-Is the student able to communicate effectively? (4)</b>	Answer at least one question correctly. Need clarification.	Answer most questions correctly. Need clarification sometimes.	Answer most questions correctly and concisely	Handle difficult questions with ease and confidence. Illustrative explanation.
<b>3-Is the student well prepared and organized? (3)</b>	Bare organization and preparation. Lack of confidence and familiarity in some parts of the presentation.	Basic organization and preparation. Confident in only some parts of the presentation	Good organization and preparation. Confident in most parts of the presentation.	Excellent organization and preparation. Confident and relaxed in the whole presentation
<b>CLO-5(marks:2.5)</b> <b>Is the SRS document well formatted and grammatically correct?</b>	The document is poorly formatted with many grammatical mistakes	The document is partially formatted with few grammatical mistakes	The document is well formatted with few grammatical mistakes	The document is well formatted with few grammatical correct
<b>CLO-3(marks:2.5)</b> <b>Requirements elicitation process defined and followed</b>	Unclearly defined and not properly followed	Suitable process is defined but not followed	Suitable process is defined but partially followed	Suitable process is defined and followed with evidence
<b>CLO-3(marks:6)</b> <b>Are user interactions clearly defined which may include detail</b> -Usecases, usecase diagram OR -Story boarding OR -Event Response Table	Incorrectly defined with low coverage.	Incorrectly defined with high coverage	Correctly defined with low coverage	Correctly defined with high coverage
<b>CLO-3(marks:6)</b> <b>Are FRs complete and defined correctly according to template?</b> Correct: Unambiguous, Complete, Verifiable and Consistent	Incorrectly defined with low coverage	Incorrectly defined with high coverage	Correctly defined with low coverage	Correctly defined with high coverage
<b>CLO-3(marks:3)</b> <b>Are the NFRs defined according to problems?</b> Correct: Unambiguous, Complete, Verifiable and Consistent	Incorrectly defined with low coverage	Incorrectly defined with high coverage	Correctly defined with low coverage	Correctly defined with high coverage
<b>CLO-3(marks:2.5)</b> <b>Are the Interfaces properly defined</b>	Incorrectly defined with low coverage	Incorrectly defined with high coverage	Correctly defined with low coverage	Correctly defined with high coverage
<b>Total = 35</b>				

**SDS**

<b>Criteria</b>	<b>Marginal</b>	<b>Adequate</b>	<b>Good</b>	<b>Excellent</b>
<b>CLO-1(marks:2.5)</b> <b>Is the SDD documentation &amp; presentation free of Plagiarism? (1)</b>	The SDD document and presentation contain plagiarism between 21-30%	The SDD document and presentation contain plagiarism between 16-20%	The SDD document and presentation contain plagiarism between 6-15%	The SDD document and presentation contain plagiarism between 0-5%
<b>CLO-6(marks:10)</b> <b>1-Is the student wearing proper attire &amp; is he/she presentable?(3)</b>	Student's attire is barely acceptable	Student's attire is appropriate	Student's attire is good	Student's attire is excellent
<b>2-Is the student able to communicate effectively?(4)</b>	Answer at least one question correctly. Need clarification.	Answer most questions correctly. Need clarification sometimes.	Answer most questions correctly and concisely	Handle difficult questions with ease and confidence. Illustrative explanation.
<b>3-Is the student well prepared and organized?(3)</b>	Bare organization and preparation. Lack of confidence and familiarity in some parts of the presentation.	Basic organization and preparation. Confident in only some parts of the presentation	Good organization and preparation. Confident in most parts of the presentation.	Excellent organization and preparation. Confident and relaxed in the whole presentation
<b>CLO-5(marks:2.5)</b> <b>Is the SDD document well formatted and grammatically correct?</b>	The document is poorly formatted with many grammatical mistakes	The document is partially formatted with few grammatical mistakes	The document is well formatted with few grammatical mistakes	The document is well formatted with few grammatical correct
<b>CLO-4(marks:1)</b> <b>Selection of Architectural style</b>	Architecture is not suitable	Architecture partially defined and represented	Suitable architectural pattern is defined and clearly represented	Suitable architectural pattern is defined and clearly represented with proper justification
<b>CLO-4(marks:1)</b> <b>Selection of Design methodology</b>	Not suitable without justification	Not suitable with justification	Suitable without justification	Suitable with justification
<b>CLO-4(marks:3)</b> <b>Data representation diagram (ERD, XML schema, SON schema etc.) with description</b>	Not suitable without justification	Not suitable with justification	Suitable without justification	Suitable with justification
<b>CLO-4(marks:3)</b> <b>Process flow (Activity Diagram)</b>	Incorrect without description	Incorrect with description	Correct without description	Correct with description
<b>CLO-4(marks:7)</b> <b>Design Models, the applicable models may include</b> -Class Diagram -Sequence Diagram -State Transition Diagram -Data Flow Diagram	Incorrect without description	Incorrect with description	Correct without description	Correct with description
<b>CLO-4(marks:3)</b> <b>Define Algorithm/pseudocode for major processes including external libraries/APIs</b>	Algorithms are incorrectly described and do not represent major processes	Most of the algorithms are correctly described but do not represent major processes	Most of the algorithms are correctly described and also represent major processes	All algorithms are correctly described and represent major processes
<b>CLO-4 (marks 2)</b> <b>User Interface design</b>	Look and feel of user interface is poor according to HCI standards	Look and feel of user interface is satisfactory according to HCI	Look and feel of user interface is good according to HCI standards	Look and feel of user interface is excellent according to HCI standards



		standards		
<b>Total = 35</b>				

**100% IMPLEMENTATION**

<b>Criteria</b>	<b>Marginal</b>	<b>Adequate</b>	<b>Good</b>	<b>Excellent</b>
<b>CLO-1(marks:5)</b> <b>Team coordination in terms of work division and collaboration</b>	Poor team coordination and poor work division	Poor team coordination and work division is satisfactory	good team coordination and work division is satisfactory	Excellent in team coordination and work division
<b>CLO-2(marks:8)</b> <b>1-Major Modules Completion status (at least 30/60/100 percent of completion w.r.t to approved scope document)</b>	Major modules completion is at least 20% of the required percentage.	Major modules completion is at least 50% of the required percentage.	Major modules completion is at least 80% of the required percentage.	Major modules completion is at least 100% of the required percentage.
<b>CLO-5(marks:7)</b> <b>Software Testing according to required percentage</b> <b>1-Validation (Field Level, Code Level etc.) (2)</b>	Software validation has been applied on 20% of the implementation	Software validation has been applied on 50% of the implementation	Software validation has been applied on 80% of the implementation	Software validation has been applied on 100% of the implementation
<b>2-Test cases verification based on selected testing methodology (automated/manual) (3)</b>	Testing methodology has been defined and test case verification is applied on 20% of the implementation	Testing methodology has been defined and test case verification is applied on 50% of the implementation	Testing methodology has been defined and test case verification is applied on 80% of the implementation	Testing methodology has been defined and test case verification is applied on 100% of the implementation
<b>2-Look and Feel of User interfaces. (Visual objects and elements, visibility and flows of tasks etc) (2)</b>	Look and feel of user interface is poor according to HCI standards	Look and feel of user interface is satisfactory according to HCI standards	Look and feel of user interface is good according to HCI standards	Look and feel of user interface is excellent according to HCI standards
<b>CLO-4(marks:10)</b> <b>Followed proper coding standards/conventions (2)</b>	Very little understanding and usage of coding standards	Some understanding and usage of coding standards	Good understanding and usage of coding standards	Excellent understanding and usage of coding standards
<b>1-Understanding of implemented Algorithms or APIs or DB schema synchronization or Image processing technique(s) etc (6)</b>	Student has poor understanding of implementation	Student has satisfactory understanding of implementation	Student has good understanding of implementation	Student has excellent understanding of implementation
<b>2-Used suitable technologies to develop the software (2)</b>	Very little understanding of the suitability of the used	Some understanding of the suitability of	Good understanding of the suitability of the	Excellent understanding of the

	technology	the used technology	used technology	suitability of the used technology
<b>CLO-3(marks:5)</b> <b>1-Is implementation according to proposed solution</b>	20% implementation is according to proposed solution	50% implementation is according to proposed solution	80% implementation is according to proposed solution	100% implementation is according to proposed solution
<b>CLO-6(marks:5)</b> <b>1-Is the student able to communicate effectively? (2)</b>	Answer at least one question correctly. Need clarification.	Answer most questions correctly. Need clarification sometimes.	Answer most questions correctly and concisely	Handle difficult questions with ease and confidence. Illustrative explanation.
<b>2-Is the document well formatted and grammatically correct? (3)</b>	The document is poorly formatted with many grammatical mistakes	The document is partially formatted with few grammatical mistakes	The document is well formatted with few grammatical mistakes	The document is well formatted with few grammatical correct
<b>Total = 40</b>				

## **APPENDIX C**

### **(FYP FORMS)**



**COMSATS UNIVERSITY ISLAMABAD**  
**Department of Computer Science**  
**FYP COMPLAINT/QUERY FORM**

<b>Student Name</b>						
<b>Registration Number</b>			<b>6<sup>th</sup> Semester</b>		<b>FYP-I</b>	<b>FYP-II</b>
<b>Contact No.</b>		<b>Date</b>				
<b>Supervisor</b>						
<b>Description</b>						

**(FOR OFFICE USE)**

<b>Coordinator Remarks</b>	
<b>FYP Convener</b>	

**Semester: 00**

To  
Final Year Project Committee  
Department of Computer Science  
COMSATS University Islamabad

**Subject: Request of Fund for FYP “Project Title”**

Respectfully,

Paragraph 1 About what is the nature of project in 3-4 lines

Paragraph 2 What is benefit of solution being developed & why funding is required 2-3 Lines

Paragraph 3 Why funding is required. Clearly state the mentioned resources availability either Inland or will be imported. 3-4 Lines

**Supervisor:**

Name	Signature

**Student(s):**

No	Name	Registration No.	Signature
1)			
2)			
3)			
4)			

**Resource Required;**

S.No	Item	Cost (Rs.)	Source of Cost
1)	Samsung Galaxy Gear VR	9999.9/- ALL PRICES SHOULD BE IN PKRs	www.ciitish.edu.pk
2)			
3)			
4)			
5)			
6)			
7)			
8)			
TOTAL			

**Detail of Required Resources:**

S.No	Item Name	Specs
1)	Samsung Galaxy Gear VR	96° field of view 2560 x 1440 pixel Super AMOLED 60 Hz <b>Compatible</b> : Galaxy Note 5, Galaxy S6, S6 Edge, and S6 Edge+ <b>Sensors</b> : Accelerator, gyrometer, geomagnetic, proximity <b>Weight</b> : 201.9 x 116.4 x 92.6 mm 310 grams Frost White
2)		
3)		
4)		
5)		
6)		
7)		
8)		



## **UNDERTAKING FORM FOR FUNDING**

Final Year Project titled Project **Title** is been done under the supervision of **Supervisor Name**. Terms and condition for project funding;

- Funding will be utilized only for mentioned purpose
- If the funding for the project is approved the purchased equipment in working condition will be returned to FYP Committee.
- Receipts of purchased equipment along with utilization report will be submitted to the supervisor.

Failing to comply with the above-mentioned.

- Clearance Status of FYP will not be sent to Examination Department
- Penalties for Non-Compliance mentioned in FYP Rule Book will be imposed

**Supervisor:**

Name	Signature

**Student(s):**

No	Name	Registration No.	Signature
1)			
2)			
3)			
4)			



## COMSATS University Islamabad

### Department of Computer Science

#### Final Year Project Evaluation Performance

<b>Program: BS(CS)</b>	<b>Semester: Spring 2018</b>	<b>Project Course Code: CSC 499</b>
<b>Student Name: MOHAMMAD QASIM USMANI</b>	<b>Registration #: SP14-BCS-076</b>	
<b>Project Code: CSC 499</b>	<b>Project Title: Accident Detection System using Image processing</b>	

#### FYP-I Marks record:

Milestone	Total Marks	Obtain Marks	Eval.1 Sign	Eval.2 Sign	Eval.3 Sign
Scope					
SRS					
SDS					

FYP-I Total Marks [100]: \_\_\_\_\_

#### FYP-I Marks record:

Milestone	Total Marks	Obtain Marks	Eval.1 Sign	Eval.2 Sign	Eval.3 Sign
60% Implementation					
80% Implementation					
100% Implementation					

FYP-II Total Marks[100]: \_\_\_\_\_

Project Status: \_\_\_\_\_

Convener Sign: \_\_\_\_\_

## **APPENDIX D**

### **(Meeting and Evaluation Log)**



# COMSATS University Islamabad

## Department of Computer Science Final Year Project – Minutes of Meeting Fortnightly Progress Report

SECTION -1 (to be completed by the STUDENT prior to meeting)	
<b>Title of Project:</b>	
<b>Students' Names:</b> 1.	<b>Supervisor Name:</b>
2.	
3.	
<b>Date:</b>	<b>Date of previous meeting:</b>
<b>Work undertaken since last meeting/ Last milestone achieved:</b>	
<b>Issues/progress you would like to discuss in this meeting:</b>	
SECTION -2 (to be completed by the SUPERVISOR at the time of meeting)	
<b>Work student should undertake between now and next meeting(next meeting agenda points):</b>	
SECTION -3	
<b>Date of next meeting:</b>	<b>Student (Team Leader):</b>
<b>Signatures:</b>	<b>Supervisor:</b>

- **Evaluation 1(Scope & SRS) LOG:**

Evaluation Comments	Supervisor Remarks & Sig

- **Evaluation 2(SDS & 30% Implementation) LOG:**

Evaluation Comments	Supervisor Remarks & Sig

- **Evaluation 3(60% Implementation) LOG:**

Evaluation Comments	Supervisor Remarks & Sig

- **Evaluation 4(100% Implementation) LOG:**

Evaluation Comments	Supervisor Remarks & Sig