

# **Project Dissertation Semester I,II,III AND IV TEMPLATE**

## **I. OBJECTIVES**

- Describe the Systems Development Life Cycle (SDLC).
- Evaluate systems requirements.
- Complete a problem definition.
- Evaluate a problem definition.
- Determine how to collect information to determine requirements.
- Perform and evaluate feasibility studies like cost-benefit analysis, technical feasibility, time feasibility and Operational feasibility for the project.
- Work on data collection methods for fact finding.
- Construct and evaluate data flow diagrams.
- Construct and evaluate data dictionaries.
- Evaluate methods of process description to include structured English, decision tables and decision trees.
- Evaluate alternative tools for the analysis process.
- Create and evaluate such alternative graphical tools as systems flow charts and state transition diagrams.
- Decide the S/W requirement specifications and H/W requirement specifications.
- Plan the systems design phase of the SDLC.
- Distinguish between logical and physical design requirements.
- Design and evaluate system outputs.
- Design and evaluate systems inputs.
- Design and evaluate validity checks for input data.

- Design and evaluate user interfaces for input.
- Design and evaluate file structures to include the use of indexes.
- Estimate storage requirements.
- Explain the various file update processes based on the standard file organizations.
- Decide various data structures.
- Construct and evaluate entity-relationship (ER) diagrams for RDBMS related projects.
- Perform normalization for the unnormalized tables for RDBMS related projects
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
- Perform various systems testing techniques/strategies to include the phases of testing.
- Systems implementation and its key problems.
- Generate various reports.
- Be able to prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
- To decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.
- Decide the various processing systems to include distributed, client/server, online and others.
- Perform project cost estimates using various techniques.
- Schedule projects using both GANTT and PERT charts.
- Perform coding for the project.
- Documentation requirements and prepare and evaluate systems documentation.
- Perform various systems testing techniques/strategies to include the phases of testing.
- Systems implementation and its key problems.
- Generate various reports.
- Be able to prepare and evaluate a final report.
- Brief the maintenance procedures and the role of configuration management in operations.
- To decide the future scope and further enhancement of the system.
- Plan for several appendices to be placed in support with the project report documentation.

- Work effectively as an individual or as a team member to produce correct, efficient, well-organized and documented programs in a reasonable time.
- Recognize problems that are amenable to computer solutions, and knowledge of the tool necessary for solving such problems.
- Develop of the ability to assess the implications of work performed.
- Get good exposure and command in one or more application areas and on the software
- Develop quality software using the software engineering principles
- Develop of the ability to communicate effectively.

## II. Type of the Project

The majority of the students are expected to work on a real-life project preferably in some industry/ Research and Development Laboratories/Educational Institution/Software Company. Students are encouraged to work in the areas listed below. However, it is ***not mandatory*** for a student to work on a real-life project. The student can formulate a project problem with the help of her/his Guide and submit the project proposal of the same. **Approval of the project proposal is mandatory.** If approved, the student can commence working on it, and complete it. Use the latest versions of the software packages for the development of the project.

## III. SOFTWARE AND BROAD AREAS OF APPLICATION

<b>FRONT END / GUI Tools</b>	.Net Technologies,Java
<b>DBMS/BACK END</b>	Oracle, SQL Plus, MY SQL, SQL Server,
<b>LANGUAGES</b>	C, C++, Java, VC++, C#, R,Python
<b>SCRIPTING LANGUAGES</b>	PHP,JSP, SHELL Scripts (Unix), Tcl/Tk,
<b>.NET Platform</b>	F#,C#. Net, Visual C#. Net, ASP.Net
<b>MIDDLE WARE (COMPONENT) TECHNOLOGIES</b>	COM/DCOM, Active-X, EJB
<b>UNIX INTERNALS</b>	Device Drivers, RPC, Threads, Socket programming
<b>NETWORK/WIRELESS TECHNOLOGIES</b>	-

<b>REALTIME OPERATING SYSTEM/ EMBEDDED SKILLS</b>	LINUX, Raspberry Pi, Arduino, 8051
<b>APPLICATION AREAS</b>	Financial / Insurance / Manufacturing / Multimedia / Computer Graphics / Instructional Design/ Database Management System/ Internet / Intranet / Computer Networking-Communication Software development/ E-Commerce/ ERP/ MRP/ TCP-IP programming / Routing protocols programming/ Socket programming.

#### **IV. Introduction**

The project report should be documented with scientific approach to the solution of the problem that the students have sought to address. The project report should be prepared in order to solve the problem in a methodical and professional manner, making due references to appropriate techniques, technologies and professional standards. The student should start the documentation process from the first phase of software development so that one can easily identify the issues to be focused upon in the ultimate project report. The student should also include the details from the project diary, in which they will record the progress of their project throughout the course. The project report should contain enough details to enable examiners to evaluate the work.

The documentation should use tools like star UML, Visuo for windows, Rational Rose for design as part of Software Project Management Practical Course. The documentation should be spiral bound for semester I,II,II and the entire documentation should be hard bound during semester IV.

#### **V. EXPLANATION OF CONTENTS**

Title Page

Sample format of Title page is given in this block. Students should follow the given format.

Certificate of Authenticated work

Sample format of Certificate of Authenticated work is given in Appendix 3 of this block.

Students should follow the given format.

## Abstract

This should be one/two short paragraphs (100-150 words total), summarising the project work. It is important that this is not just a re-statement of the original project outline. A suggested flow is background, project aims and main achievements. From the abstract, a reader should be able to ascertain if the project is of interest to them and, it should present results of which they may wish to know more details.

## Acknowledgements

This should express student's gratitude to those who have helped in the preparation of project.

**Table of Contents:** The table of contents gives the readers a view of the detailed structure of the report. The students would need to provide section and subsection headings with associated pages. The formatting details of these sections and subsections are given below.

**Table of Figures:** List of all Figures, Tables, Graphs, Charts etc. along with their page numbers in a table of figures.

## Chapter 1: Introduction

The introduction has several parts as given below:

**Background:** A description of the background and context of the project and its relation to work already done in the area. Summarise existing work in the area concerned with the project work.

**Objectives:** Concise statement of the aims and objectives of the project. Define exactly what is going to be done in the project; the objectives should be about 30 /40 words.

**Purpose, Scope and Applicability:** The description of Purpose, Scope, and Applicability are given below:

- **Purpose:** Description of the topic of the project that answers questions on why this project is being done. How the project could improve the system its significance and theoretical

framework.

- **Scope:** A brief overview of the methodology, assumptions and limitations. The students should answer the question: What are the main issues being covered in the project? What are the main functions of the project?
- **Applicability:** The student should explain the direct and indirect applications of their work. Briefly discuss how this project will serve the computer world and people.

**Achievements:** Explain what knowledge the student achieved after the completion of the work. What contributions has the project made to the chosen area? **Goals achieved** - describes the degree to which the findings support the original objectives laid out by the project. The goals may be partially or fully achieved, or exceeded.

**Organization of Report:** Summarizing the remaining chapters of the project report, in effect, giving the reader an overview of what is to come in the project report.

## Chapter 2: System Analysis

In this chapter Survey of Technologies should demonstrate the students awareness and understanding of Available Technologies related to the topic of the project. The student should give the detail of all the related technologies that are necessary to complete the project. The should describe the technologies available in the chosen area and present a comparative study of all those Available Technologies. Explain why the student selected the one technology for the completion of the objectives of the project.

**Problem Definition:** Define the problem on which the students are working in the project.

Provide details of the overall problem and then divide the problem in to sub-problems. Define each sub-problem clearly.

**Requirements Specification:** In this phase the student should define the requirements of the system, independent of how these requirements will be accomplished. The Requirements Specification describes the things in the system and the actions that can be done on these things. Identify the operation and problems of the existing system.

**Planning and Scheduling:** Planning and scheduling is a complicated part of software development. Planning, for our purposes, can be thought of as determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules, known as constraints, which, control when certain tasks can or cannot happen. Scheduling can be thought of as determining whether adequate resources are available to carry out the plan. The student should show the Gantt chart and Program Evaluation Review Technique (PERT).

Software and Hardware Requirements: Define the details of all the software and hardware needed for the development and implementation of the project.

- Hardware Requirement: In this section, the equipment, graphics card, numeric co-processor, mouse, disk capacity, RAM capacity etc. necessary to run the software must be noted.
- Software Requirements: In this section, the operating system, the compiler, testing tools, linker, and the libraries etc. necessary to compile, link and install the software must be listed.

Preliminary Product Description: Identify the requirements and objectives of the new system. Define the functions and operation of the application/system the students are developing as project.

Conceptual Models: The student should understand the problem domain and produce a model of the system, which describes operations that can be performed on the system, and the allowable sequences of those operations. Conceptual Models could consist of complete Data Flow Diagrams, ER diagrams, Object-oriented diagrams, System Flowcharts etc.

### Chapter 3: System Design

Describes desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudocode and other documentation.

Basic Modules: The students should follow the divide and conquer theory, so divide the overall problem into more manageable parts and develop each part or module separately. When all modules are ready, the student should integrate all the modules into one system. In this phase, the student should briefly describe all the modules and the functionality of these modules.

Data Design: Data design will consist of how data is organised, managed and manipulated.

- Schema Design: Define the structure and explanation of schemas used in the project.
- Data Integrity and Constraints: Define and explain all the validity checks and constraints provided to maintain data integrity.

Procedural Design: Procedural design is a systematic way for developing algorithms or procedurals.

- Logic Diagrams: Define the systematical flow of procedure that improves its comprehension and helps the programmer during implementation. e.g., Control Flow Chart, Process Diagrams etc.
- Data Structures: Create and define the data structure used in procedures.
- Algorithms Design: With proper explanations of input data, output data, logic of processes, design and explain the working of algorithms.

User Interface Design: Define user, task, environment analysis and how to map those requirements in order to develop a “User Interface”. Describe the external and internal components and the architecture of user interface. Show some rough pictorial views of the user interface and its components.

Security Issues: Discuss Real-time considerations and Security issues related to the project and explain how the student intends avoiding those security problems. What are the security policy plans and architecture?

Test Cases Design: Define test cases, which will provide easy detection of errors and mistakes with in a minimum period of time and with the least effort. Explain the different conditions in which the students wish to ensure the correct working of the project.

#### Chapter 4: Implementation and Testing

Implementation Approaches: Define the plan of implementation, and the standards the students have used in the implementation.

Coding Details and Code Efficiency: Students not need include full source code, instead, include only the important codes (algorithms, applets code, forms code etc). The program code should contain comments needed for explaining the work a piece of code does. Comments may be needed to explain why it does it, or, why it does a particular way.

The student can explain the function of the code with a shot of the output screen of that program code.



- **Code Efficiency:** The student should explain how the code is efficient and how the students have handled code optimisation.

**Testing Approach:** Testing should be according to the scheme presented in the system design chapter and should follow some suitable model – e.g., category partition, state machine-based. Both functional testing and user-acceptance testing are appropriate. Explain the approach of testing.

- **Unit Testing:** Unit testing deals with testing a unit or module as a whole. This would test the interaction of many functions but, do confine the test within one module.
- **Integrated Testing:** Brings all the modules together into a special testing environment, then checks for errors, bugs and interoperability. It deals with tests for the entire application. Application limits and features are tested here.

**Modifications and Improvements:** Once the students finish the testing they are bound to be faced with bugs, errors and they will need to modify your source code to improve the system. Define what modification are implemented in the system and how it improved the system.

## Chapter 5: Results and Discussion

**Test Reports:** Explain the test results and reports based on the test cases, which should show that the project is capable of facing any problematic situation and that it works fine in different conditions. Take the different sample inputs and show the outputs.

**User Documentation:** Define the working of the software; explain its different functions, components with screen shots. The user document should provide all the details of the product in such a way that any user reading the manual, is able to understand the working and functionality of the document.

## Chapter 6: Conclusions

**Conclusion:** The conclusions can be summarised in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that would have made in the other chapters.

**Limitations of the System:** Explain the limitations encountered during the testing of the project that the students were not able to modify. List the criticisms accepted during the demonstrations of the project.

**Future Scope of the Project** describes two things: firstly, new areas of investigation prompted by

developments in this project, and secondly, parts of the current work that was not completed due to time constraints and/or problems encountered.

Chapter 7: Challenges and Learning

Chapter 8: Sustainability Goals

1. **Include the diagram** of the 17 Sustainable Development Goals (SDGs).  
(You can use the official UN SDG image: [Click here](#))



2. **Explain all 17 SDGs briefly** in your own words.
3. **Identify and justify the SDGs relevant to your project.**  
For example, if your project involves renewable energy, justify how it supports **SDG 7 – Affordable and Clean Energy**, **SDG 9 – Innovation**, or **SDG 13 – Climate Action**, etc.
4. Make sure this chapter is clear, concise, and directly linked to your work.

## REFERENCES

It is very important that the students acknowledge the work of others that they have used or adapted in their own work, or that provides the essential background or context to the project. The use of references is the standard way to do this. Please follow the given standard for the references for books, journals, and online material. The citation is mandatory in both the reports.

E.g:

Linhares, A., & Brum, P. (2007). Understanding our understanding of strategic scenarios: What role do chunks play? *Cognitive Science*, 31(6), 989-1007.  
<https://doi.org/doi:10.1080/03640210701703725>

*Lipson, Charles (2011). Cite right : A quick guide to citation styles; MLA, APA, Chicago, the sciences, professions, and more (2nd ed.). Chicago [u.a.]: University of Chicago Press. p. 187. ISBN 9780226484648.*

Elaine Ritchie, J Knite. (2001). *Artificial Intelligence, Chapter 2* , p.p 23 - 44. Tata McGrawHill.

## GLOSSARY

If you the students any acronyms, abbreviations, symbols, or uncommon terms in the project report then their meaning should be explained where they first occur. If they go on to use any of them extensively then it is helpful to list them in this section and define the meaning.

## APPENDICES

These may be provided to include further details of results, mathematical derivations, certain illustrative parts of the program code (e.g., class interfaces), user documentation etc.

In particular, if there are technical details of the work done that might be useful to others who wish to build on this work, but that are not sufficiently important to the project as a whole to justify being discussed in the main body of the project, then they should be included as appendices.

## VI. SUMMARY

Project development usually involves an engineering approach to the design and development of a software system that fulfils a practical need. Projects also often form an important focus for discussion at interviews with future employers as they provide a detailed example of what the students are capable of achieving. In this course the students can choose your project topic from the lists given in Unit 4: Category-wise Problem Definition.

## VII. FURTHER READINGS

1. Modern Systems Analysis and Design; Jeffrey A. Hoffer, Joey F. George, Joseph,S. Valacich; Pearson Education; Third Edition; 2002.

2.ISO/IEC 12207: Software Life Cycle Process

(<http://www.software.org/quagmire/descriptions/iso-iec12207.asp>)

3. IEEE 1063: Software User Documentation (<http://ieeexplore.ieee.org>).

4. ISO/IEC: 18019: Guidelines for the Design and Preparation of User Documentation for

Application Software.

5. <http://www.sce.carleton.ca/squall>.

6. <http://en.tldp.org/HOWTO/Software-Release-Practice-HOWTO/documentation.html>.

7. <http://www.sei.cmu.edu/cmm/>

--

(All the text in the report should be in times new roman)

# **TITLE OF THE PROJECT**

**(NOT EXCEEDING 2 LINES, 24 BOLD,  
ALL CAPS)**

**A Project Report (12 Bold)**

Submitted in partial fulfillment of the  
Requirements for the award of the Degree of (size-12)

**MASTER OF COMPUTER APPLICATIONS (12  
Bold)**

Name of The Student (size-15, title  
case) Seat Number (size-15)

**Under the esteemed guidance of (13 bold)**

**Mr./Mrs. Name of The Guide (15 bold, title case)**

**Designation (14 Bold, title case)**



**DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS (12  
BOLD, CAPS)**

**COLLEGE NAME (14 BOLD, CAPS)**

(Autonomous Institute Affiliated to the University of Mumbai, Approved by AICTE & Recognised by Govt. of Maharashtra)

*NAAC accredited with 'A grade*

**CITY, PIN CODE(12 bold, CAPS) MAHARASHTRA (12  
bold, CAPS)**

**ACADEMIC YEAR (12 bold)**

**COLLEGE NAME (14 BOLD, CAPS)**

**Autonomous Institute Affiliated to the University of Mumbai, Approved by AICTE & Recognised by Govt. of Maharashtra)**

*NAAC accredited with 'A grade  
(13, bold, italic)*

**CITY-MAHARASHTRA-PINCODE(13 bold, CAPS)**

**MASTER OF COMPUTER APPLICATIONS (14 BOLD, CAPS)**



**CERTIFICATE (14 BOLD, CAPS, underlined, centered)**

This is to certify that the project entitled, "**Title of The Project** ", is bonafied work of **NAME OF THE STUDENT** bearing Seat.No: (**NUMBER**) submitted in partial fulfillment of the requirements for the award of degree of MASTER OF COMPUTER APPLICATIONS from University of Mumbai. (12, times new roman, justified)

PRINCIPAL

HEAD OF DEPARTMENT

PROJECT GUIDE / INTERNAL EXAMINER

EXTERNAL EXAMINER

**COMPANY CERTIFICATE (if applicable)**

(Project Abstract page format)

# **Abstract (20bold, caps, centered)**

Content (12, justified)

**Note: Entire document should be with 1.5  
line spacing and all paragraphs should start with 1 tab space.**



# ACKNOWLEDGEMENT

(20, BOLD, ALL CAPS, CENTERED)

The acknowledgement should be in times new roman, 12 font with 1.5 line spacing, justified.

I Would like to thank all those people whose support and cooperation has been an invaluable asset during the course of this project. It would have been impossible to complete the project without their support, valuable suggestions, encouragement and guidance.

I am greatly indebted to express my immense pleasure and sense of gratitude towards my project guide <Internal Teacher Name> for her constant and valuable encouragement. I would also convey my gratitude towards <HOD's Name>, Head of Department for her constant encouragement. From <Internship's Firm Name>., I would like to thank my manager, mentor and other associates for being very cooperative and supportive throughout the project. The guidance provided from time to time during the internship period has been a great learning experience. They have been an amazing knowledge support for me during my entire training period.

I am also grateful for all other teaching and non-teaching staff members of M.C.A for directly or indirectly helping us for the completion of the project and the resources provided.

<Your Name>  
(<Roll No>)

Declaration page format)

## **DECLARATION (20 bold, centered, allcaps)**

**Content (12, justified)**

I the undersigned, solemnly declare that the report of the project work entitled “<**Project**>” is based on my own work carried out during the course of my study under the supervision of <**Teacher’s Name**> (Internal Project Guide)

I assert that the statements made, and conclusions drawn are an outcome of the project work. I further declare that to the best of my knowledge and belief the project report does not contain any part of any work which has been submitted for the award of any other degree/diploma/certificate in this University or any other University.

**Name and Signature of the Student**

# **TABLE OF CONTENTS (20bold, caps, centered)**

Should be generated automatically using word processing software.

<b>Chapter 1: Introduction</b>	01(no bold)
<b>1.1 Background</b>	02(no bold)
<b>1.2 Objectives</b>	....
<b>1.3 Purpose and Scope</b>	....
1.3.1 Purpose	....
1.3.2 Scope	

.....  
.....

<b>Chapter 2: System Analysis</b>	
<b>2.1 Existing System</b>	
<b>2.2 Proposed System</b>	
<b>2.3 Requirement Analysis</b>	
<b>2.4 Hardware Requirements</b>	
<b>2.5 Software Requirements</b>	
<b>2.6 Justification of selection of Technology</b>	

<b>Chapter 3: System Design</b>	
<b>3.1 Module Division</b>	
<b>3.2 Data Dictionary</b>	
<b>3.3 ER Diagrams</b>	
<b>3.4 DFD/UML Diagrams</b>	

## **Chapter 4: Implementation and Testing**

<b>4.1 Code (Place Core segments)</b>	
<b>4.2 Testing Approach</b>	
<b>4.2.1 Unit Testing (Test cases and Test Results)</b>	
<b>4.2.2 Integration System (Test cases and Test Results)</b>	

## **Chapter 5: Results and Discussions (Output Screens)**

**Chapter 6: Conclusion and Future Work**

**Chapter 7: Challenges and Learning**

**Chapter8: Sustainability Goals**

**References**

**NOTE:**

**Please note the following important instruction for your project report formatting:**

- 1. Page numbers should begin from Chapter 1 (i.e., no page numbers on the cover page, certificate, declaration, or abstract).**
- 2. Page numbers must be placed at the bottom center of each page starting from Chapter 1 onward.**
- 3. Ensure consistency throughout your report. Improper formatting may lead to deduction in evaluation.**

## **List of Tables (20 bold, centered, Title Case)**

Should be generated automatically using word processing software.

## **List of Figures (20 bold, centered, Title Case)**

Should be generated automatically using word processing software.

(Project Introduction page format)

# **Chapter 1**

## **Introduction (20 Bold, centered)**

**Content or text (12, justified)**

**Note: Introduction has to cover brief description of the project with minimum 4 pages.**

# Chapter 2

## System Analysis (20 bold, Centered)

Subheadings are as shown below with following format (16 bold, CAPS)

### 2.1 Existing System (16 Bold)

#### 2.1.1 (14 bold, title case)

##### 2.1.1.1 (12 bold, title case)

### 2.2 Proposed System

### 2.3 Requirement Analysis

### 2.4 Hardware Requirements

### 2.5 Software Requirements

### 2.6 Justification of Platform – (how h/w & s/w satisfying the project)

Table 2.1: Caption


# **Chapter 3**

## **System Design (20 bold, centered)**

Subheadings are as shown below with following format (16 bold, CAPS) Specify figures as Fig 11.1 – caption

### **3.1 Module Division**

### **3.2 Data Dictionary**

### **3.3 E-R Diagrams**

### **3.4 Data Flow Diagrams / UML**

Note: write brief description at the bottom of all diagrams

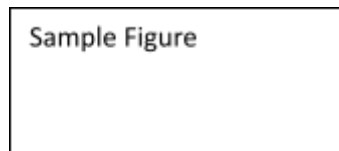


Fig. 3.1: Caption



# Chapter 4

## Implementation and Testing (20 bold, centered)

### 4.1 Code (Place Core segments)

Content includes description about coding phase in your project (Font-12)

(\* don't include complete code just description)

### 4.2 Testing Approach

Subheadings are as shown below with following format (16 bold, CAPS)

#### 4.2.1 Unit Testing

#### 4.2.2 Integration Testing

Note:

- Explain about above testing methods
- Explain how the above techniques are applied in your project  
Provide Test plans, test cases, etc relevant to your project

## **Chapter 5**

### **Results and Discussions (20 bold, centered)**

Note: Place Screen Shots and write the functionality of each screen at the bottom

## **Chapter 6**

### **Conclusion and Future Work (20 bold, centered)**

The conclusions can be summarized in a fairly short chapter around 300 words. Also include limitations of your system and future scope (12, justified)

## **Chapter 7**

### **Sustainability Goals (20 bold, centered)**

## **References (20 bold, centered)**

Content (12, LEFT)

[1] Title of the book, Author

[2] Full URL of online references

[3] \_\_\_\_\_