

MGM's COLLEGE OF ENGINEERING AND TECHNOLOGY

Navi Mumbai-410209 DEPARTMENT OF COMPUTER ENGINEERING

Regulations: 2021

Batch: 2024-25

Year: TE

Semester: V

CSL501 – SOFTWARE ENGINEERING



Prepared by Prof. Vilas

Jadhav

LAB INCHARGE

HOD

CONTENTS

Sr. No.	Topic	Page No.
1	Vision & Mission	ii
2	Study and Evaluation Scheme	iii
3	Departmental Program Outcome	iv
4	Departmental Program Outcome	v
5	Course Objective, Course Outcomes and Experiment Plan CO-PO Mapping	vi
6	CO-PO Mapping, CO-PSO mapping	vii
7	List of Experiments	viii
8	Additional Experiments	ix



MGM's College of Engineering and Technology

Vision:

To become one of the outstanding Engineering Institute in India by providing a conducive and vibrant environment to achieve excellence in the field of Technology.

Mission:

To empower the aspiring professional students to be prudent enough to explore the world of technology and mould them to be proficient to reach the pinnacle of success in the competitive global economy.

Department of Computer Engineering

Vision:

To emerge out as a prominent department offering a programme in its pursuit for academic excellence in order to develop professionally competent and socially responsible engineers capable of meeting industry demands and social obligations in a vibrant global environment.

Mission:

To strive towards building an atmosphere that will be a catalyst for innovative ideas and learning, providing students with various opportunities and experiences that can help them to thrive and prosper through a blend of academics, practical exposure and research programs to pursue successful careers in a global environment.

Study and Evaluation Scheme

Course Code	Course Name	Teaching Scheme			Credits Assigned			
CSC502	Software Engineering	Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
		03	02	--	03	01	--	04

Course Code	Course Name	Examination Scheme		
CSC502	Software Engineering	Term Work	Oral	Total
		25	25	50

Term Work:

Internal Assessment consists of two tests. Test 1, an Institution level central test, is for 20 marks and is to be based on a minimum of 40% of the syllabus. Test 2 is also for 20 marks and is to be based on the remaining syllabus. Test 2 may be either a class test or assignment on live problems or course project

Practical & Oral:

Oral exam will be based on CSC502 and CSL501 syllabus.

Oral examination is to be conducted by pair of internal and external examiners based on the syllabus.

Departmental Program Outcomes (POs)

1. Engineering Knowledge: Apply the knowledge of mathematics science engineering fundamentals, and an engineering specialization to the solution of complex engineering problem.

2. Problem Analysis: Identify, formulate, review research, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, naturalizing sciences, and engineering sciences.

3. Design/Development of Solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The Engineer and Society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and Sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and individual and as a member or leader in diverse teams and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, give and receive clear instructions.

11. Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work as a member and leader in a team to manage projects and in multidisciplinary environments.

12. Life-Long Learning: Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1	Acquire skills to design, analyze and develop algorithms and implement them using high-level programming languages
PSO2	Contribute their engineering skills in computing and information engineering domains like network design and administration, database design and knowledge engineering.
PSO3	Develop strong skills in systematic planning, developing, testing implementing and providing IT solutions for different domains which helps in the betterment of life.

Course Objectives & Course Outcome, Experiment Plan

Course Objectives:

1.	To provide knowledge of software engineering discipline.
2.	To apply analysis, design and testing principles to software project development
3.	To demonstrate and evaluate real time projects with respect to software engineering principles

Course Outcomes:

CO1	Students will Understand and demonstrate basic knowledge in software engineering.
CO2	Students will be identify requirements, analyze and prepare models.
CO3	Students will be able to plan, schedule and track the progress of the projects
CO4	Students will have Design & develop the software projects.
CO5	Student will have students will be able to Identify risks, manage the change to assure quality in software projects. .
CO6	Students will be Apply testing principles on software project and understand the maintenance concepts

Mapping of Course Outcomes (COs) to Program outcomes (POs):

Name of the Course	CO Code														
		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	MAP	

Software Engineering	CO1	√	√	√										Weak
	CO2			√	√	√				√		√		Weak
	CO3			√	√	√								Weak
	CO4						√	√	√	√	√		√	Medium
	CO5						√	√	√	√	√		√	Medium
	CO6						√	√	√	√	√		√	Medium

CO & PSO Mapping

Name of the course	CO Code	PSO1	PSO2	PSO3	MAP
Software Engineering	CO1		✓	✓	medium
	CO2		✓	✓	medium
	CO3	✓	✓	✓	strong
	CO4	✓	✓	✓	strong
	CO5	✓	✓	✓	strong
	CO6	✓	✓	✓	strong

SE Lab Manual

vii

List of Experiments

Sr. No	Experiments	CO	PO	PSO	Page No
--------	-------------	----	----	-----	---------

1	To write a Application of at least two traditional process models	CO1	PO2	PSO 1	01
2	To write a Application of the Agile Process models.	CO1	PO2	PSO 1	05
3	To preparation of software requirements specification(SRS) document in IEEE format.	CO2	PO3	PSO 1	08
4	To design structured data flow analysis.	CO3	PO4	PSO 3	13
5	To use of metrics to estimates the cost.	CO2	PO3	PSO 2	20
6	To Use to schedule and tracking of the project.	CO3	PO4	PSO 3	24
7	To write test cases for black box testing	CO3	PO4	PSO 3	28
8	To write test cases for white box testing	CO2	PO3	PSO 2	33
9	To preparation of Risk mitigation ,Monitoring and Management Plan (RMMM)	CO2	PO4	PSO 2	40
10	To Version controlling of the project.	CO3	PO4	PSO 3	46

MGM's COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

Modifications in SE Lab manual 2021-22

Following are the modifications in SE lab manual.

1. More Examples are added wherever required.
2. The updated version is more users friendly.

List of Experiments

Sr. No	Experiments	DOP	DOS	Page No.	Sign
1	To write a Application of at least two traditional process models				

2	To write a Application of the Agile Process models.				
3	To preparation of software requirements specification(SRS) document in IEEE format.				
4	To design structured data flow analysis.				
5	To use of metrics to estimates the cost.				
6	To Use to schedule and tracking of the project.				
7	To write test cases for black box testing				
8	To write test cases for white box testing				
9	To preparation of Risk mitigation ,Monitoring and Management Plan (RMMM)				
10	To Version controlling of the project.				

DOP	DOS	Experiment No 1	Remark	Signature

Experiment No: 1

Aim: Application of at least two traditional process models.

Software Used: NIL

Theory: Software Process Models

A software process model is a simplified representation of a software process. Each model represents a process from a specific perspective.

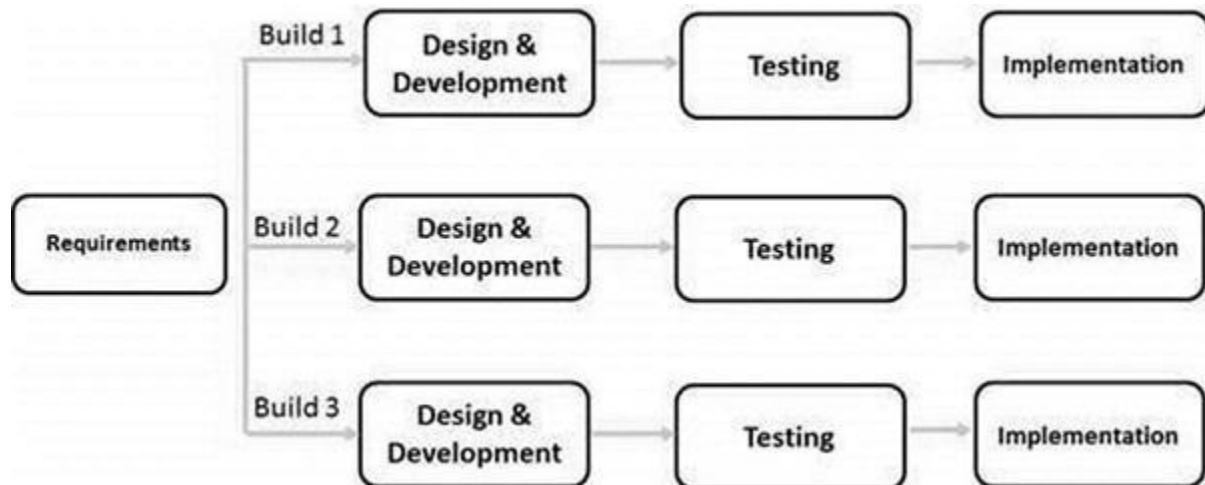
We're going to take a quick glance about very general process models. These generic models are abstractions of the process that can be used to explain different approaches to the software development. They can be adapted and extended to create more specific processes.

*Some methodologies are sometimes known as **software development life cycle (SDLC)** methodologies, though this term could also be used more generally to refer to any methodology.*

There are various software process models like:

- 🏠 Prototyping Model
- 🏠 RAD Model
- 🏠 The Spiral Model
- 🏠 The Waterfall Model
- 🏠 The Iterative Model

Of all these process models we've used **the Iterative model (Iterative Incremental Method)** for the development of our project. In this model each phase well defined starting and ending point, with identifiable deliveries to the next phase. This model is sometimes referred to as the linear sequential model or the software life cycle. The following illustration is a representation of the Iterative and Incremental model –



Iterative and Incremental development is a combination of both iterative design or iterative method and incremental build model for development. "During software development, more than one iteration of the software development cycle may be in progress at the same time." This process may be described as an "evolutionary acquisition" or "incremental build" approach."

In this incremental model, the whole requirement is divided into various builds. During each iteration, the development module goes through the requirements, design, implementation and testing phases. Each subsequent release of the module adds function to the previous release. The process continues till the complete system is ready as per the requirement.

The key to a successful use of an iterative software development lifecycle is rigorous validation of requirements, and verification & testing of each version of the software against those requirements within each cycle of the model. As the software evolves through successive cycles, tests must be repeated and extended to verify each version of the software.

Iterative Model – Application :

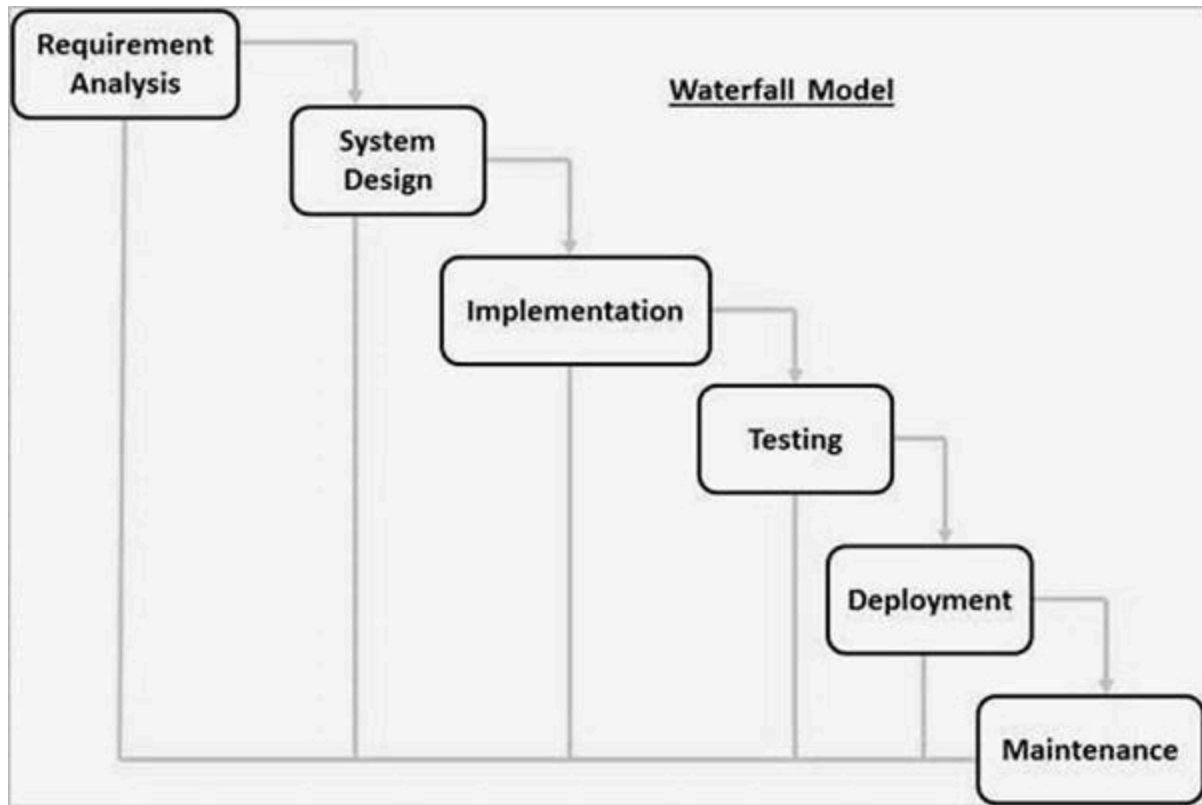
Like other SDLC models, Iterative and incremental development has some specific applications in the software industry. This model is most often used in the following scenarios –

- Requirements of the complete system are clearly defined and understood.
- Major requirements must be defined; however, some functionalities or requested enhancements may evolve with time.
- There is a time to the market constraint.
- A new technology is being used and is being learnt by the development team while working on the project.
- Resources with needed skill sets are not available and are planned to be used on contract basis for specific iterations.
- There are some high-risk features and goals which may change in the future.

Another model is **Waterfall method** :

In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.

The following illustration is a representation of the different phases of the Waterfall Model.



The sequential phases in Waterfall model are –

- **Requirement Gathering and analysis** – All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **System Design** – The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** – With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
- **Integration and Testing** – All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.

- **Deployment of system** – Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** – There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

All these phases are cascaded to each other in which progress is seen as flowing steadily downwards (like a waterfall) through the phases. The next phase is started only after the defined set of goals are achieved for previous phase and it is signed off, so the name "Waterfall Model". In this model, phases do not overlap.

Waterfall Model – Application :

Every software developed is different and requires a suitable SDLC approach to be followed based on the internal and external factors. Some situations where the use of Waterfall model is most appropriate are –

- Requirements are very well documented, clear and fixed.
- Product definition is stable.
- Technology is understood and is not dynamic.
- There are no ambiguous requirements.
- Ample resources with required expertise are available to support the product.
- The project is short.

1. **Conclusion:** The understand Traditional Process are use for project was written successfully.
2. **Industrial Applications:** used to collect the user requirements & assess the feasibility of software.

Questionnaire:

1. What does RAD stand for?

.....
.....
.....
.....

2. What software development model is not suitable for accommodating any change?

.....
.....
.....
.....

3. What is software engineering?

.....
.....
.....
.....

4. What is problem statement?

.....
.....
.....
.....

5. What is the most important phase of spiral model ?

.....
.....
.....
.....

6. What is incremental model?

.....

.....

.....

.....

7. What is agile methodology?

.....

.....

.....

.....

8. What is prototype model?

.....

.....

.....

.....

9. What is evolutionary model?

.....

.....

.....

.....

10. Which development model is best when user requirements are not clear?

.....

.....

.....

.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGowan Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

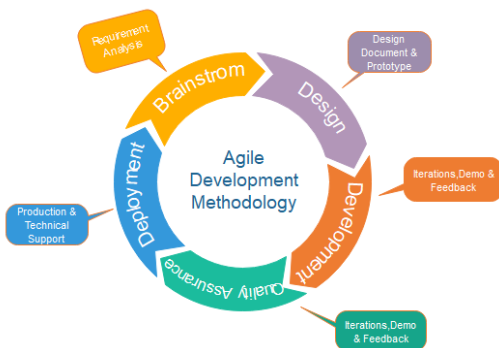
DOP	DOS	Experiment No 2	Remark	Signature

Experiment No: 2

Aim: Application of the Agile process models.

Software Used: JIRA

Theory: The meaning of Agile is swift or versatile. "**Agile process model**" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance. Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team working through a full software development life cycle including planning, requirements analysis, design, coding, and testing before a working product is demonstrated to the client.



Every iteration involves cross functional teams working simultaneously on various areas like –

Planning

Requirements Analysis

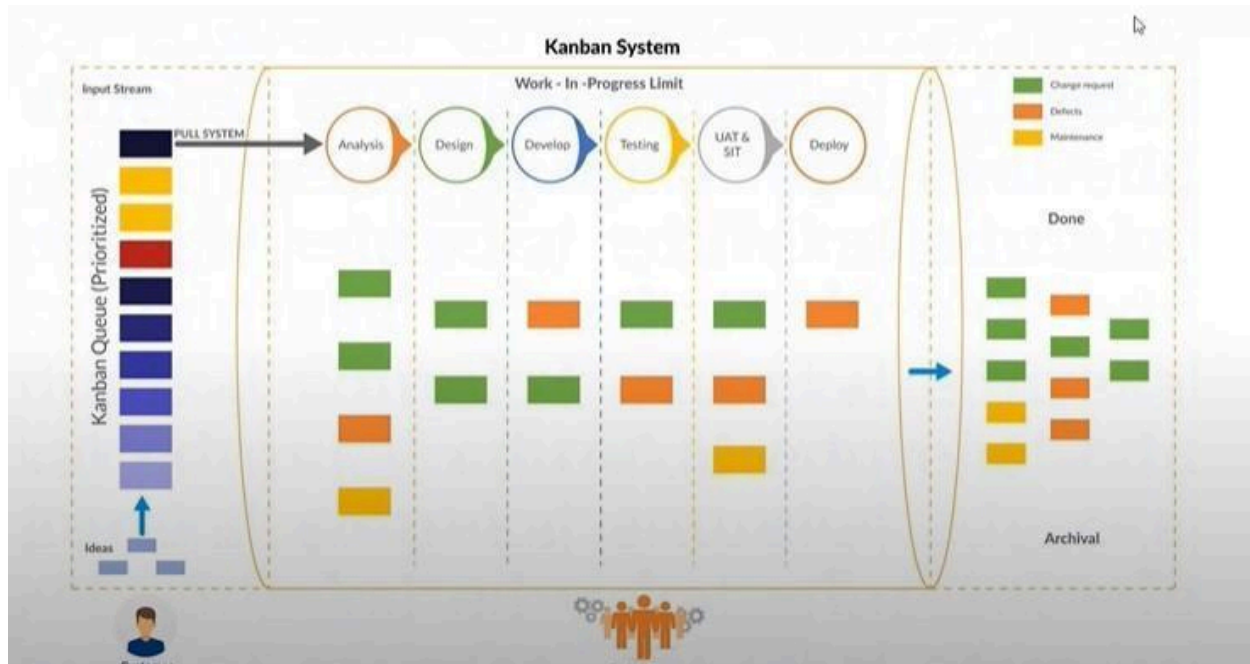
Design

Coding

Unit Testing and

Acceptance Testing.

At the end of the iteration, a working product is displayed to the customer and important stakeholders.



1. Results:

.....
.....
.....

2. **Conclusion:** Thus Agile process model (KANBAN Board) for the assigned project is designed successfully.
3. **Industrial Applications:** To make an agreement between client & software development company.

Questionnaire:

1. What is Agile Processes ?

.....
.....
.....

2. What are the principles of Agile methodology ?

.....
.....
.....

3. What is advantages of Agile Process ?

.....
.....
.....

4. What is XP ?

.....
.....
.....

5. What are different between Prescriptive process model and Agile process model ?

.....
.....
.....

6. What is SCRUM ?

.....
.....
.....

7. What is KANBAN Model?

.....
.....
.....

8. What are the difference between Scrum & kanban ?

.....
.....
.....
.....

9. What is kanban Boards and kanban cards?

.....
.....
.....

10. What is Kanban for Software teams ?

.....
.....
.....
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

<https://www.atlassian.com/software/jira>

https://www.youtube.com/watch?v=uM_m6EzMg3k

DOP	DOS	Experiment No 3	Remark	Signature

Experiment No: 3

Aim: Preparation of Software Requirement Specification (SRS) document in IEEE format

Software Used: NIL

Theory:

The purpose of this document is to give a detailed description of the requirements for the software. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

Table of Contents

Introduction

1.1 Purpose

1.2 Scope

1.3 Definitions, acronyms, and abbreviations

1.4 References

2. Overall description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Constraints

2.5 Assumptions and dependencies

2.6 Apportioning of requirements

3. Specific requirements

3.1.1 User interfaces

3.1.2 Hardware interfaces

3.1.3 Software interfaces

3.1.4 Communications interfaces

3.2 Functional requirements

3.2.1 User Class 1

3.2.2 User Class 2

3.2.3 User Class 3

3.3 Performance requirements

3.4 Design constraints

3.5 Software system attributes

4. Prioritization and Release Plan

4.1 Choice of prioritization method

Appendix

Conclusion:

Thus we have prepared the SRS for our project .

Software Requirements Specification

For

FACE RECOGNITION BASED ATTENDENCE SYSTEM

Prepared by

1. KARN SHRUTI SHAILESH KUMAR – 62
GUPTA ROSHNI SANTOSH – 45
LANKE SHITAL RATAN – 72
KADAM CHAITALI GANPAT – 55

**Guide Name :
MR. VILAS JADHAV**

**MGM'S COLLEGE OF ENGG. & TECH.
KAMOTHE , NAVI MUMBAI
UNIVERSITY OF MUMBAI.**

2TH August, 2022

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1 Purpose	1
1.2 Intended Audience	1
1.3 Project Scope	1
1.4 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Project Plan	3
2.3 Product Features	3
2.4 User Classes and Characteristics	6
2.5 Operating Environment	6
2.6 Data Model Design and	7
3. System Features	9
4. External Interface Requirements	10
4.1 User Interfaces	10
4.2 Hardware Interfaces	15
4.3 Software Interfaces	15
5. Other Nonfunctional Requirements	15
5.1 Security Requirements	15
5.2 System Architecture	18
Appendix A: Use Case model	19

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The main purpose of this specification is to help people who will work on this system to maintain the objectives and get started working in this project. This specification will direct people who will work on this project step by step through the process until they finish it successfully. This statement will describe specific details into every step of this project that workers will immediately locate the needs of this system to understand the purpose of doing any of the following steps into the system.

1.2 Intended Audience

The audience of this system will be:

1. Students
2. Faculty members
3. Registration office.

This project will be managed by registration office, created and developed by the IT staff and other specialized people in the technology, such as programming, web design and others.

1.3 Project Scope

The scope of the system is to have a high-tech environment in the Dominican university community. That means by using the automatic attendance system, the community will transfer to the technical environment that they already have the Canvas system to help them manage the courses they have in the whole semester. This system will add some features in the automatic attendance system to Canvas by using fingerprint device in every classroom at Dominican University. That will help the community use the technology in effective ways:

1. Make the attendee process easier and effective.
2. Help faculty in the attendance process every time.
3. Mange and organize the attendance page through Canvas.

1.4 References

1. Instructure: <http://www.instructure.com/>
2. Dominican University Canvas: dominicanu.instructure.com

2. Overall Description

2.1 Product Perspective

At Dominican University, instructors manually take attendance in every class each day. They spend time to do that during class time. The Automatic Attendance System will help them do this process in an easy way. The main scope of this project is to make attendance process more organized in every class. This project will help instructors take the attendance automatically without spending some time during the class. It will provide the instructor who is/isn't present an early-warning of high levels of non-attendance through the Canvas page. There are also many benefits for students: they can manage their attendance, absences, and late walk-ins by checking the Canvas site. They will also know the current grade in their reports. It makes it easier to have a clear picture of every student's attendance throughout the academic year.

The system is about to modify an existing system to develop the project. This system comes from Instructure. Instructure is a new company that has 200 employees. This company is an educational origination that works with technology to help the education community in an effective way. This company provides Canvas. The Canvas system is about a website page, which contains classes managed by instructors. It has management tools for courses. These tools play a significant role in the educational models these days, which are to organize the educational level using technology to achieve the educational goals easily. Instructors have the control panel for every class they have. The control panels allow them to create and develop the course's page that all students can see. They may have a Home Page, Syllabus, Discussion, Grade, Assignments, People, Files, and more. All of these components are available and controlled by the faculty member to make any changes.

Definitions:

Users: This means students who will get the most benefits of the system.

Faculty: Also, who has the top priority to get benefit for the system and they are the target actors of the system.

The Registration Office: This is for the system management, and it will be presented by the employees.

2.2 Project Plan

This project has six phases to be completed within the time line. They are initiating, project plan, components, process model, testing, and feedback. The expected time for the project will take around six months.

The project plan is in a PDF document.

2.3 Product Features

There are two kinds of process models for this system. There is the overview process model and the conditional process model. Starting with the first one. The first step of this process is to have a fingerprint capture device. That will do the following steps:

- 1- Students enter their fingerprints into the device.
- 2- Every fingerprint has a special code number for every recode. This code number takes the other step, which is matching.
- 3- The system checks on the fingerprint and sends to the server and the student database.
- 4- In this database file, the system checks this print for the identification. Also, the database sends it to the registration office data file to check if this code exists or not. If the code number for the fingerprint is in both databases, the code number will continue for other steps. And if the code is not there, it will give you a false result. Then, it will send you to the registration office for the identification and look for your record to modify it if there is any issue. Then, students will try again.

There is another step after the general identification.

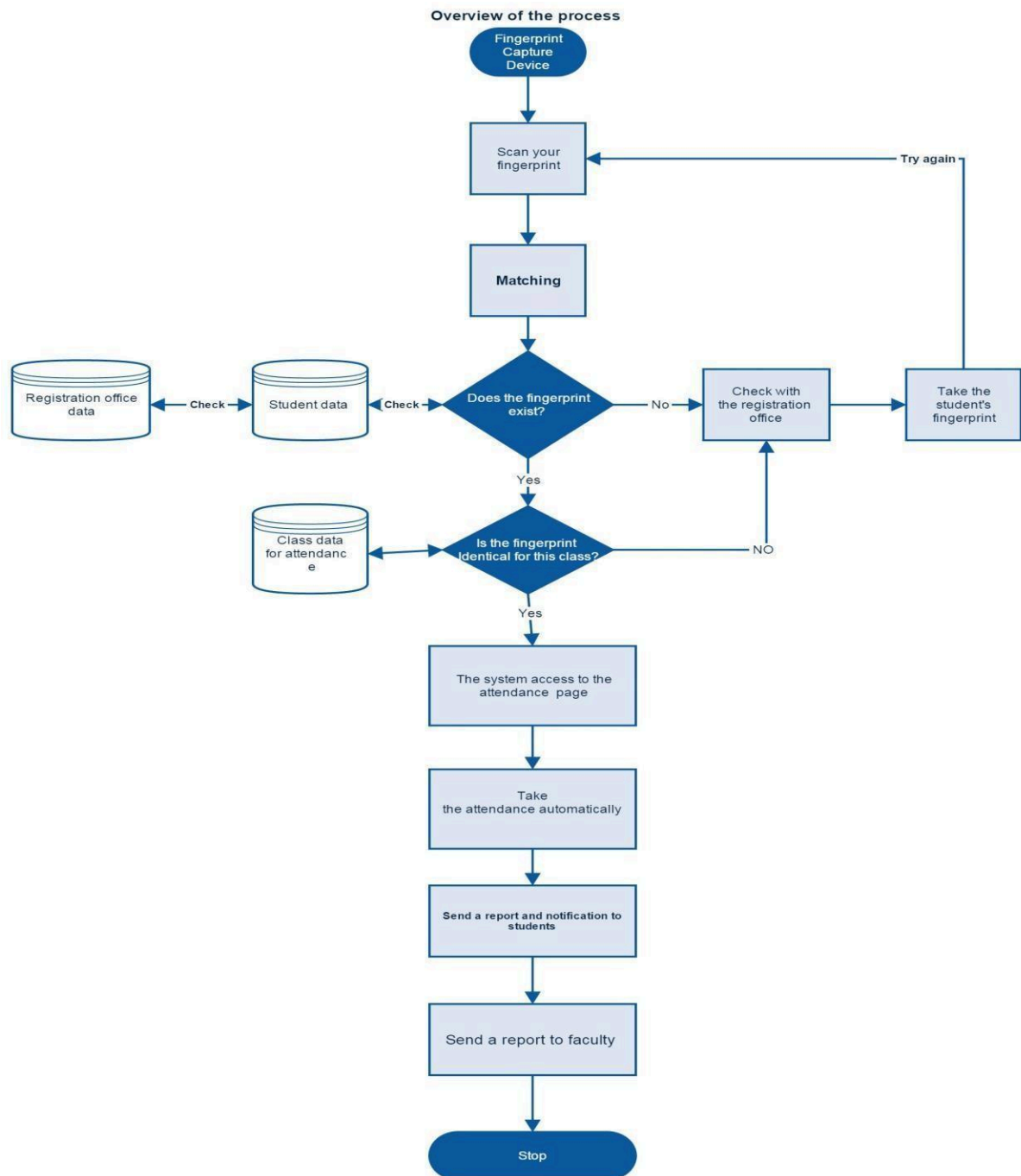
- 5- Checkpoint, which will check if the student data is enrolled in the particular class or not. If yes, the process will continue, and if not, the system will send you to the registration office to check.

After that, the system now has access to the Canvas system. The system will access the student's attendance page where it can take the attendance through the Canvas page automatically. The last step of this process is to send a notification message to students and faculty. Students can check on that and know their attendance grade. Faculty will have all students' attendance reports, and they know who is attending and who is not.

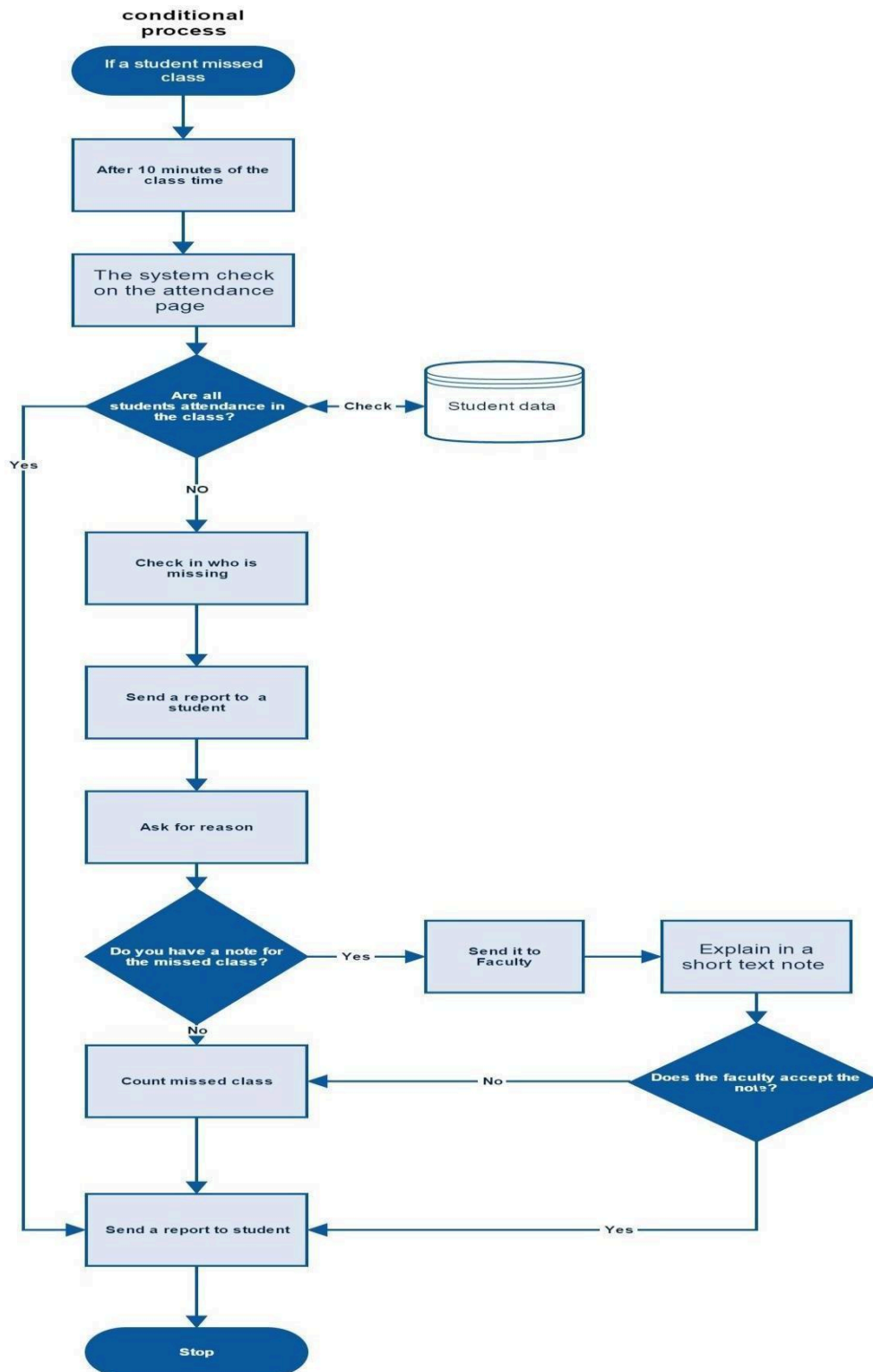
There is also another process for this project that if a student missed a class, the system would make a decision. The aim of this process is to contain every student's status and make sure the attendance for all students has already been taken. After ten minutes of the class time, the system will run automatically to check on the attendance page. If all students attended in that class, the system will send a report to them and stop.

However, if there are students who missed the class, the system would start some process. The system will check for who is missing the class and make a list of them. Every student of this list will receive a message that asks them for the reason of the missing the class. In this step, the system will wait for getting a response from each student separately. If the student answers with yes, and writes a note for it, the system will send this message to the faculty member. The faculty member has all the right to accept the excuse or not. If a student does not have a reason for the missed class, and checks on no, the system will count the missed class and send a report. Furthermore, if the

student has an acceptable reason that he/she provides to faculty, the system will automatically report them, and the system will be done.



Entity Relationship Diagram (ERD)



2.4 User Classes and Characteristics

There are three types of user classes in this community:

- 1) Students
- 2) Faculty
- 3) Registration Office

2.5 Operating Environment

This project will go through two steps:

The first step is to have the automatic attendance device in every classroom in the school. These devices will be connected to the computer and its system. Students have to put their fingerprints on file in the registration office on their first day to save their fingerprint data in the database.

The second step is to connect this system to the Canvas site. That is to connect the Canvas database to the system database to work as one system on the Canvas site. This step would complete the work, and the project will work in one system. That is because the attendance report will be updated all the time. Also, the Canvas site will control all the students' attendance reports not in a separate system or database.

This system has some requirements to be accomplished. It needs hardware and software.

Hardware requirements:

- 1) Biometric Fingerprint Scanners
- 2) Cables for the device

The current system work is already in existence. However, we need some system requirements:

- 1) Create new databases and indexes for students and class list by using mysql
- 2) Make connection to the current database
- 3) Design interfaces for the users
- 4) Design an attendance page on Canvas
- 5) Programming using JavaScript, PHP, and HTML

2.6 Data Model Design

This system contains many processes to be completed. One of the processes is the database design. It needs to present data that is the data understandable not only for the human being but also for computers. This step would organize the needed data on every side of this project to make the database relationship. In this project, I will use the Entities Relationship Diagram (ERD) to help this project make the database relationship.

The system has four entities, each with its own attributes: People, Class list, Courses and Canvas. The People entity has the student fingerprint ID, first name, last name, Faculty fingerprint ID, first name, last name. Student and faculty ID are based on the code of the fingerprint in the system. The class list entity has class list ID, class ID class data and class time. The course entity has course ID, course name, credits, room number, and class time. Also, the course entity has an index for the date and time. Finally, the Canvas entity has the class list ID, class ID, student fingerprint ID, class date, and attendance information.

The relationship in this entity relationship diagram has many ways to define the following:

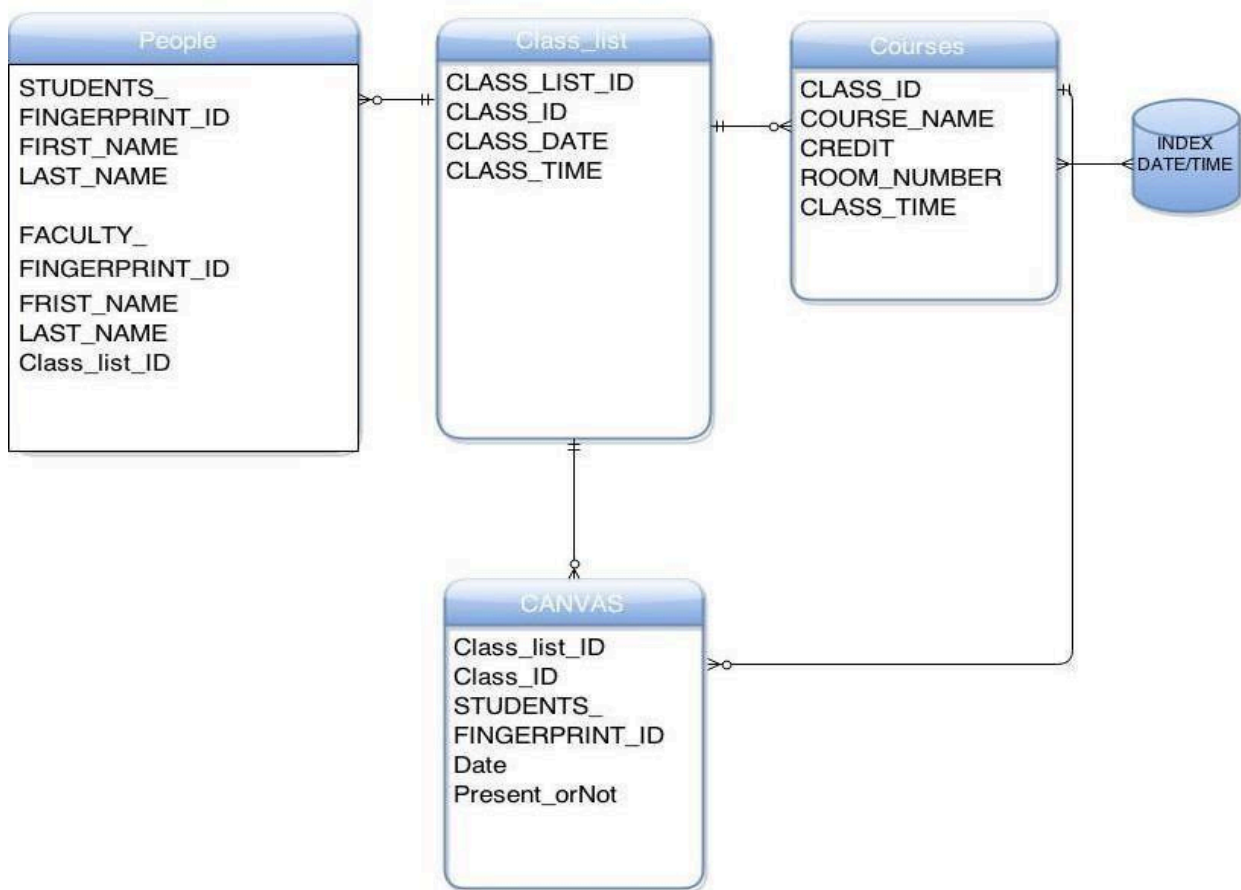
- Every student has zero to many classes (not all take classes this semester)
- Every faculty teaches zero to many classes (not all teach any class this semester, such as doing research)
- Every class has zero to many students
- Every class has only one faculty

The result of this relationship is connected with the others with the Canvas entity. The Canvas entity is the final result of this relationship. It has the student information, and the course information in one format. In the other words, it has the one to many relationships with each entity and it shows the result of the relationship.

- All entities related to the Canvas entity
- Classes have one to many relationships

The process starts with the class list entity. It must have at least one student in each class. Every faculty and student has a unique fingerprint ID. Also, every class has a unique ID too. Faculty can be signed for one mandatory class to many classes, and each class has only one faculty member, Students has zero to many optional classes. So, each class has many students with only one faculty. In the Canvas page, all results of the one to many relationships would end up on this page to give the outcome.

The whole process would be in this relationship as one part of the work, and it has everything we need to get the project started as planned. Entity relationship diagrams help this project to be more clear and understandable. This will continue working on the same steps that we use in this diagram. The overall benefit of this structure is to facilitate easy communication between humans and computers. Better communication will expedite the desired results.



3. System Features

Functional Requirements

3.1 Adding a New student

Function: Sign up a new student to the system.

Priority: Top (Required for first release)

Requirements: To add a new user to the system, all of them should have registered in the admission office before they can register in their classes. On the orientation day, all students must scan their thumbs in the input device for only one time to save the fingerprint data in the registration office to sign up.

3.2 Use the system to attend to classes

Function: Attend to classes

Priority: Top (Required for every class attended)

Requirements: When students have a class, they must scan their thumbs in the fingerprint input device. If the scan matches, students can enter the class, and they will be checked on the Canvas page. If the scan does not match, the student must check with the registration office to figure out the checking device.

3.3 Report students

Function: The user will look at their reports for the current status.

Priority: Top (Required for first release)

Requirements: When the students have enrolled in the class, they are now able to check on their current attendance situation through the Canvas page. In the system, they will be shown a page that gives them the whole attendance status in the semester.

3.4 Faculty receive a report

Function: Faculty receives a current report every class. Priority: Top

Requirements: The system will send a message after ten minutes of the class time to the faculty. Faculty will have the all students' attendance reports in the particular class. Faculty can modify some of the attendance grades if he/she needs.

3.5 Students missed classes

Function: Students receive a message for missing class and have to submit a form.

Priority: High (Required for second release)

Requirements: When a student misses a class, he/she will receive a message via email and Canvas page. Students must log in to the Canvas page and go to the attendance page to write the reason for missing the class. Students have to submit the form to wait for the response of the faculty's decision.

3.6 Students missed two classes and more

Function: Students receive a warning message for missing class for the second time.

Priority: High (Required for second release)

Requirements: When students miss more than one class, students will receive a warning message for missing two classes or more. The warning message should be for the missed classes for the whole semester and their status in a danger level.

3.7 Faculty check the report and the attendance control

panel page Function: Users can provide feedback about search terms.

Priority: Medium (Second release if possible, mandatory for third release)

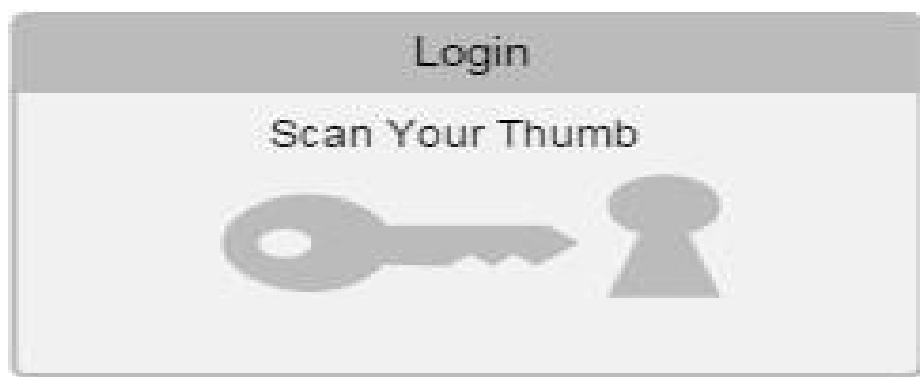
Requirements: Faculty has to check on the report and give the final submission. Faculty has full control of modifying any grades and looking at students who have excuses to modify their grades. Faculty receive messages from the system about students who missed classes. For students who submit the note for the missed class, faculty members could look at the note and give a decision on the student's grades.

4. External Interface Requirements

4.1 User Interfaces

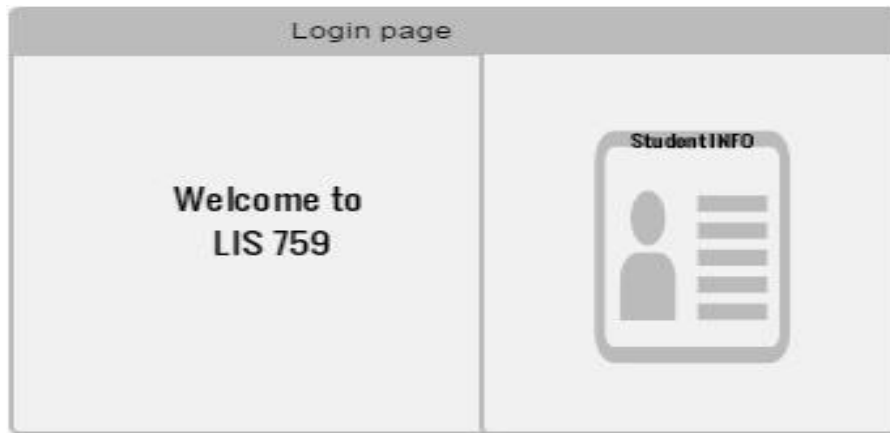
Login Display

This is the main login in the system which appears in the Biometric Fingerprint Scanners. This interface designed to be in the device view in every class. "Scan Your Thumb" is the login to the class and the system.



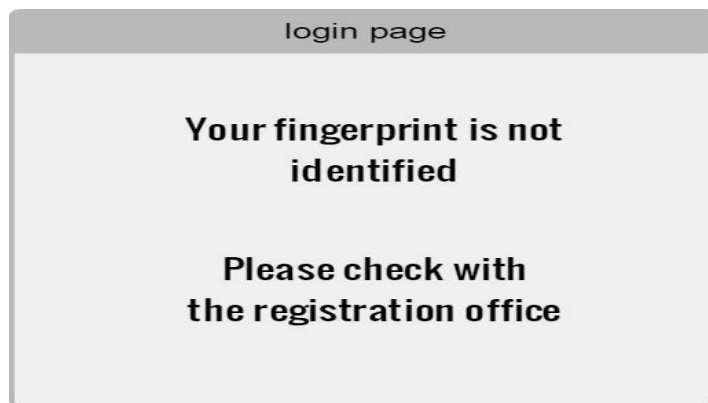
Welcome View page

This view will be also in the device screen. This page means the device accepts the fingerprint data and identify the person. So, in the screen, it will show the student information (name, and his picture). Also, it has the data for the particular class at the time.



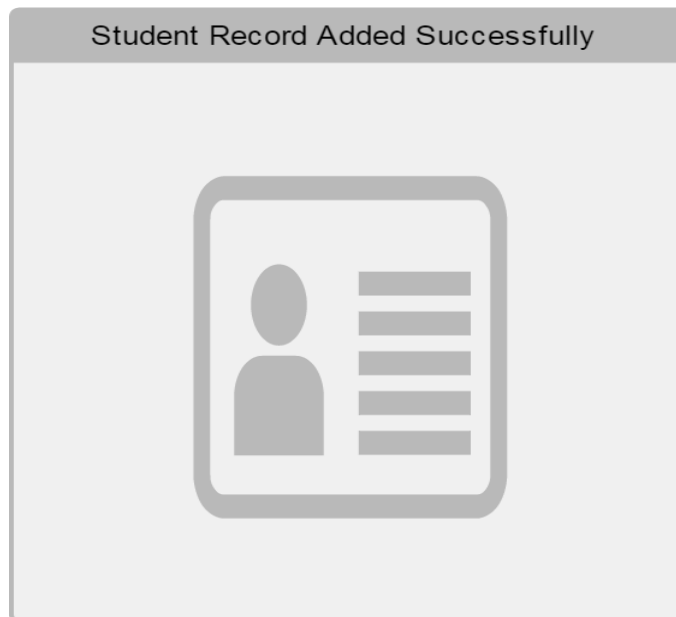
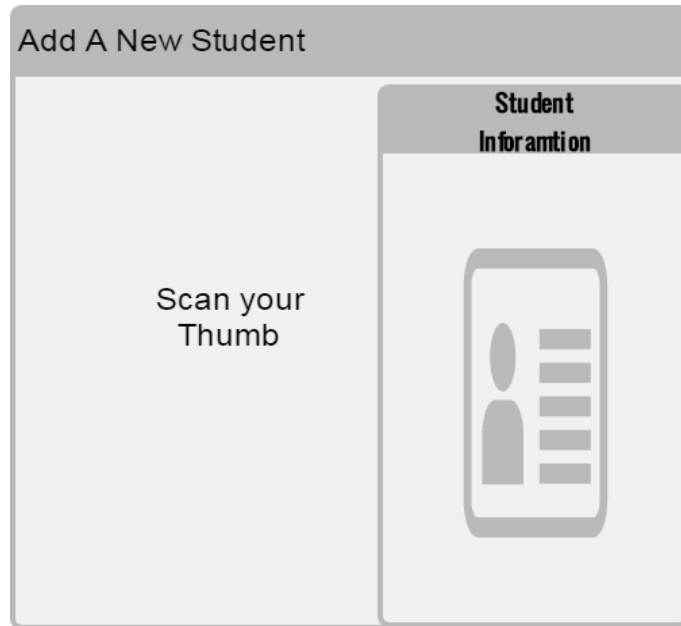
Error message view

If the student data is not identified with the device, the student has to check with the registration office to fix that. This means, this student does not have a record in the system. Or, the student is in the wrong class or wrong class time.



Add a New student

This page appears in the registration office when they add a new student. Students have to scan their fingerprint in the device. And then, their fingerprint record will be added by the employer to make sure the student has his record with his information; not anyone else's. The employer submits the information to the system and the system will give a message that this student with his fingerprint information has added to the system.



Student Report

This page will appear in a separate page in the system. It is a web page gives the students current report during the semester. It has the weekly report, the check in each class, and grade. It gives the student how many times he has been in the class and how many he missed. Also, it provides all grades that student makes during the semester.

Student Current Report		
Weeks	Check	Grade
Week1	<input checked="" type="checkbox"/>	1
Week2	<input checked="" type="checkbox"/>	1
Week3	<input type="checkbox"/>	0
Week4	<input checked="" type="checkbox"/>	1
Total	3	9/10

Faculty Attendance Report

This is a web page which has all student's reports in the class. This page is controlled by a faculty member. Faculty can modify in this report. This page has the name, the time of the class and the class room number. It has weekly reports with the average of the student's attendance in every class time. Also,

Faculty student's attendance page							
Class name : System analysis and Design		Class Time: 6:00 - 9:00 PM		Class Room: 330			
Weeks/ Students	Week1	Week2	Week3	Week4	Week5	Week6	Total grade
John	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	9/10
Paule	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	8/10
Ragheb	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	10/10
Sam	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	8/10
Total attendance each class	4	3	4	2	4	2	

it calculates the total grade for every student.

Missed class form

If a student missed any class, the system automatically will send to him this form. Students should write their note and submit it to the system. The system will send it to the faculty member to have the final decision.

Missed class form	
Student name: Ragheb	
Course name: System Analysis and Design	
Class Time: 6:00- 9:00	
write your note	
<div></div>	
<div>Submit</div>	

4.2 Hardware Interfaces

The hardware environment in this system will use the Biometric Fingerprint Scanners. These scanners will play a role in the system. This device must be available in every classroom in the school. Also, it must be in the registration office. The interfaces for the hardware part are the same in the registration office's interface. This part of the interface has also other components, such as student's information, faculty information, class's information, and other related information. All of these data are stored in the database and end with the device screen and web pages.

4.3 Software Interfaces

The system will use:

- 1) Biometric Fingerprint Devices display software
- 2) Web pages for the forms HTML, PHP
- 3) Server
- 4) Programing using JavaScript
- 5) Database uses with mysql

5. Other Nonfunctional Requirements

5.1 Security Requirements

The Current System Security

The current system, which is Canvas, has its policy on its site page. The current system builds upon a user name and password access. Students and faculty can access to his/her account through their page, and they can control it.

The system now has its own policy and security; however, the new feature we will add to the system will need some security requirements to the system. The new feature in the system will add some values to the current policy to maintain the security in the right way. It also provides proof of compliance.

The new policy in the system will deal with the security in many cases. The security will have more components on the system in a high control panel. The plan is to secure the outsider and insider community of misused the system (e.g. identification theft). Strong security is part of the policy's purpose.

User Access

Inside the community, there are students, faculty, and registration office staff who are going to use the system. The main actor of the users in the entity is students. Students will use the entity everyday by scanning their fingerprints by the beginning of each class. Faculty will only use the system through Canvas, and they can access to student information. The faculty member will use the entity to control the attendance page. Faculty's job is to add, edit, update and delete any record. Furthermore, registration office staff will check on every student's identification for security purposes. They check on students for identification in person before they add, edit, update and delete any information from their fingerprint records in the system. The staff will ask students for ID for identification and print their thumb in the device if needed to make sure the person is identified.

Threats to the system security.

This system may face many threats. Sometimes, it comes from a community insider. This could be someone who discloses the data form the database where it located, in the registration office. Another type of deception is false identification, such as a fake ID, when students present in the registration office. The system will reduce this kind of misuse because the fingerprint identification is more secure than others. Nobody can make up a fingerprint not related to them. However, staff in the registration office should check on the identification carefully before initiating any processes in the system. These records will be the official record for all students, since they begin school and until they graduate.

Levels of security:

- 1) **Hardware:** The fingerprint devices must be located in a secure location in every class. It should be behind the entrance that everyone can see the device inside and outside the class if the door is open.
- 2) **The operating system:** the security in this case will be in the same level of the Canvas security.
- 3) **The network:** it is part of the current system security.
- 4) **The data management system:**
 1. Students can access to their classes to the system check by fingerprint.
 2. Faculty access to Canvas would be the same as we have now, and they will control the attendance page/report.
 3. The purpose of the registration office is to make sure every student has the right fingerprint record and right information in the system on a consistent basis.

Level of access

Subject:

People level:

- 1) Users (students).
- 2) Faculty (Control on Canvas).
- 3) Registration office (control the system).

Computer level:

- 1) Hardware (fingerprint reader)
- 2) Software (the system)

Access request (operation)

The operation will be presented in the security matrix that provides every task in the system and everybody in all task responses.

Reference Monitor

The authorization and the access control present in the security matrix below:
Faculty can create, read, update and delete the “Report student,” “Report faculty,” and only read “process student check.”

Registration office can create, read, update and delete the “add a new record,” “Process student’s check,” and they can only read the “Check-in.”

Student are able to create and read the “Check-in,” and they can only read the “Receive student check” and “report student.”

The CRUD Security Matrix

	Add a new record	Check-in	Process student check	Receive student check	Report Student	Report Faculty
Faculty				R	CRUD	CRUD
Registration staff	CRUD	R	CRUD			
Student		CR		R	R	

Identity and Authentication

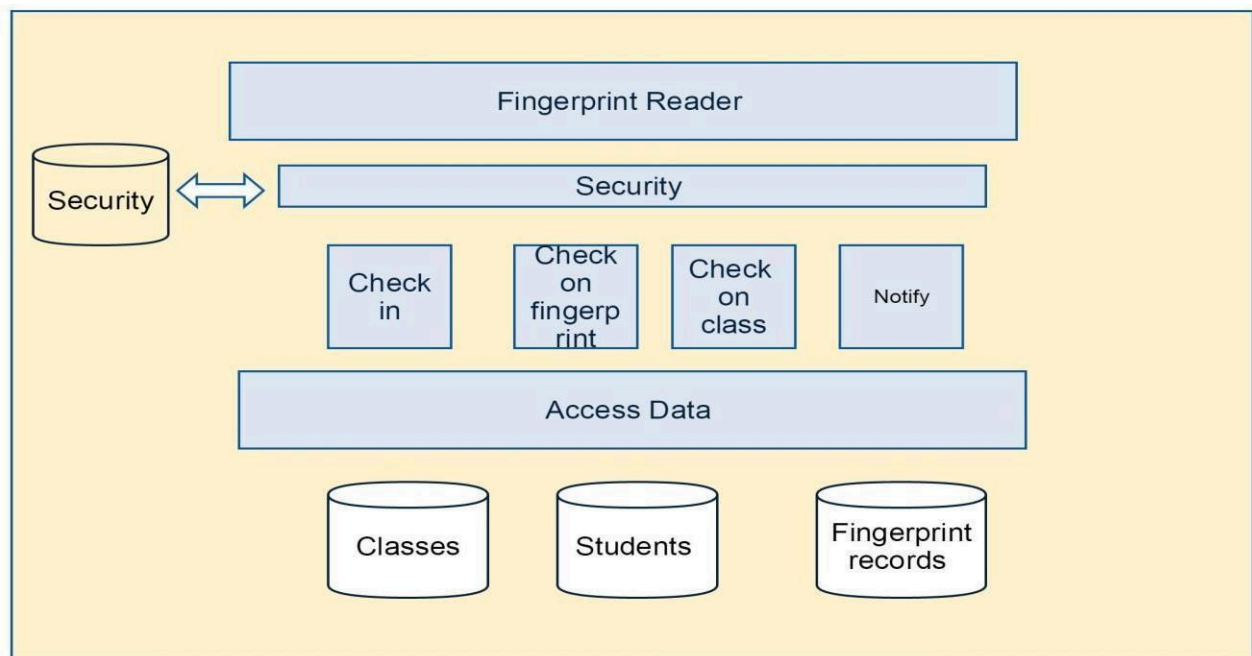
The system has the fingerprint entity. This entity has the student’s records that will play the role of student identification every class time. Students have to be identified to the system in order to access their record and complete the processes. However, faculty and registration staff will rely on the current system, which has a user name and password entity.

5.2 System Architecture

A system architecture is the ideational model that defines the structure, behavior, technology and other views of any system. In the fingerprint system, we have the whole structure to build the system. In this architecture, we will describe the formal definition and the representation of the system. This description is a high level that can show the relationship between the components induces software, hardware and the communication between them.

The first level of this architecture is the hardware component. The hardware here is the fingerprint reader which will be connected to the system and other components. Then, it will check with security level. The security level here has database for the security purposes. Next, the architecture level will go to the process task, which has four components in the software. They are check in for every class when student scan his/her fingerprint, check on fingerprint, this is kind of the security and to check into every class. Then check into the class, which is the class check list to make sure the student in the right class. Finally, notify step, this step is to report everyone involves in the system to receive a notification.

The last two level of this architecture are the access data and the databases. There are three databases for this step which are for the classes, students, and fingerprint records. This architecture is the whole structure for the Automatic Attendance System. So, all process here will complete the communication between all the components.



Appendix A: Use Case model

Develop a system that can help the Dominican university community to take the attendance automatically that will be connected with Canvas page on a new attendance page

The target actors are:

1. Students
2. Faculty
3. Registration Office

All the data will be gathered by the Biometric Fingerprint Scanners and Readers that will help faculty, students and, registration to reach the end user by the Canvas page.

We will create the separate page on Canvas to take the attendance. Fortunately, we do not need to create the login page, as we already have the Canvas site to log into the system, starting with the students, faculty, and the registration office staff.

1. Students:

- a. For the first time, students go to the registration for the fingerprint scanning.
- b. In the beginning of each class, students must scan their thumbs in the Biometric Fingerprint Scanner.
- c. Each student, receive the automatic attendance grad on his Canvas page.
- d. Students will receive the notification on his/her Canvas page for the attendance.
- e. Students can access to his page and look/print at his current attendance report and the final report.
- f. Student would receive a warning message if they miss more than two classes.

2. Faculty:

- a. To know who is in attendance automatically.
- b. On the class time, faculty will receive a report for this particular time.
- c. Faculty could know who is in attendance or missing that class.
- d. On the Canvas page, they have every student's report and grades.
- e. They have the percentage of the attendance for the whole semester.
- f. They can print the final attendance report for the class by the end of the semester.

3. Registration:

- a. Registration office has all the students' fingerprint records.
- b. They check the identification for every student.
- c. The fingerprint code will give access to the student account on Canvas (no username/password needed at this time).
- d. They use these records for any student who will come by the office as the identifier.
- e. They connect this information with the classes.
- f. Students would have a hold if he/she does not have a fingerprint record in the registration office.

Author	Ragheb Irqasous
Date	Feb,12, 2015
Version	1.0
Use-case name	Students fingerprint records for the registration
Use-case number	Unspecified
Priority	Students
Source	Dominican University
Primary business actor	Faculty
Other participating actors	Registration office
Interested stakeholders	Administration office
Description	The first step for student is to go to the registration office for a student's fingerprint record
Typical course of events	<ol style="list-style-type: none"> 1. Students go to the registration office to scan their thumbs on the fingerprint records. 2. Students must go to the registration office before the first class they have. 3. Students must provide the scan of their right and left thumb. 4. The right thumb will be the primary record for the students and the left one will be alternative access.
Alternate courses	Student would receive a warning message if they miss more than two classes
Conclusion	The aim is to save the class time.
Post-condition	No conditions required
Business rules	Fingerprint required for the class registration.
Implementation constraints and specifications	Every class must have at least one of the Biometric Fingerprint Scanners

Assumptions	
Open issues	

Author	Ragheb Irqasous
Date	Feb,12, 2015
Version	1.0
Use-case name	Students check for every class
Use-case number	Unspecified
Priority	Students
Source	Dominican University
Primary business actor	Faculty
Other participating actors	Registration Office
Interested stakeholders	Administration Office
Description	This use case is for every time students have class to check in to attend.
Typical course of events	<ol style="list-style-type: none"> 1. In the beginning of each class, students must scan their thumbs in the Biometric Fingerprint Scanner. 2. Students will stand by the device for matches. 3. All students must be correctly matched before entering the class. 4. Students now automatically take attendance. 5. Student will receive the notification on his Canvas page for the attendance. 6. Students can access this page and look at a current attendance report. 7. Student would receive warning messages if they miss more than two classes

Alternate courses	Student would receive warning message if they miss more than two classes
Conclusion	The aim is to save class time.
Post-condition	No conditions required
Business rules	Fingerprints are required for class registration.
Implementation constraints and specifications	Every class must have at least one of the Biometric Fingerprint Scanners
Assumptions	
Open issues	

Author	Ragheb Irqasous
Date	Feb,12, 2015
Version	1.0
Use-case name	Students missed classes
Use-case number	Unspecified
Priority	Students
Source	Dominican University
Primary business actor	Faculty
Other participating actors	Registration Office
Interested stakeholders	Administration Office
Description	This use case is for every time students miss one class and more. It is a sensitive case that that system will work automatically and send messages.
Typical course of events	1. Students who miss one class will receive a message to his/her email and the Canvas page.

	<p>2. Students have a chance to provide the reason for missing class.</p> <p>3. Students must fill out the note for the missed classes and submit it.</p> <p>4. Students will receive the feedback from the instructor to approve or deny.</p> <p>5. If students missed more than two classes, students will receive a warning message of the limit of the missing classes during the semester.</p>
Alternate courses	Students would receive warning messages if they miss more than two classes
Conclusion	Keep students in the current status all the time.
Post-condition	No conditions required
Business rules	
Implementation constraints and specifications	Every class must have at least one of the Biometric Fingerprint Scanners.
Assumptions	
Open issues	

Author	Ragheb Irqasous
Date	Feb.12.2015
Version	1.0
Use-case name	Know who is in attendance automatically
Use-case number	002
Priority	Faculty
Source	

Primary business actor	Faculty
Other participating actors	Registration Office
Interested stakeholders	Registration Office
Description	Attendance process will be taken automatically during class time.
Typical course of events	<ol style="list-style-type: none"> 1. During class time, faculty will receive a report for the particular class. 2. Faculty could know who is in attendance and misses that class. 3. On the Canvas page, faculty have every student's report and grades. 4. Faculty have the percentage of the attendance for the whole semester. 5. Faculty have a control panel for the attendance in the case when a student misses a class with excuses. 6. Faculty can modify any student's report.
Alternate courses	
Conclusion	No teacher needs to take the attendance manually in every class.
Post-condition	
Business rules	Log in to the Canvas site, then go to the attendance report page
Implementation constraints and specifications	
Assumptions	
Open issues	

Author	Ragheb Irqasous
Date	February 12, 2015
Version	1.0
Use-case name	Registration office fingerprint records
Use-case number	
Priority	Unspecified
Source	Dominican University
Primary business actor	Registration Office
Other participating actors	Students
Interested stakeholders	Faculty
Description	Registration Office has the database that has all fingerprint records.
Typical course of events	<ol style="list-style-type: none"> 1. The registration office has all the students' fingerprint records. 2. They check the identification for every student. 3. The fingerprint code will give access to the student account on Canvas (no username/password needed at this time). 4. They use this records for any student who will come by the office and ask for help as identifier. 5. They connect this information with the classes. 6. The registration office sets a hold if any student does not have a fingerprint record in the registration office.
Alternate courses	
Conclusion	The registration office has database control panel for all attendance in every class.

Post-condition	
Business rules	Manage and organize the process of adding students
Implementation constraints and Specifications	
Assumptions	
Open issues	

1.Results:

.....
.....
.....

2.Conclusion: Thus SRS(Software Requirement Specification) document in IEEE (Institute of Electrical and Electronics Engineers) format for the assigned project is designed successfully.

3.Industrial Applications: To make an agreement between client & software development company.

Questionnaire:

1 What is SRS document?

.....
.....
.....

2 What are the components of SRS document?

.....
.....
.....

3 What is functional requirement?

.....
.....
.....

4 What are non-functional requirements?

.....
.....
.....

5 What are different types of constraints in developing the software?

.....
.....
.....

6 What is the significance of SRS document?

.....
.....
.....

7 What are different requirement gathering techniques?

.....
.....
.....

8 What is the difference between requirement analysis & requirement specification ?

.....
.....
.....

9 Time limit of a software is functional requirement or non-functional requirement?

.....
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Sommerville, Software Engineering, 9th edition, Addison Wesley, 2011

DOP	DOS	Experiment No 4	Remark	Signature

Experiment No: 4

Aim: Structured Data Flow Analysis

Software Used: Star UML

Theory: Structured Analysis is a development method that allows the analyst to understand the system and its activities in a logical way.

It is a systematic approach, which uses graphical tools that analyze and refine the objectives of an existing system and develop a new system specification which can be easily understandable by user.

It has following attributes –

- It is graphic which specifies the presentation of application.

- It divides the processes so that it gives a clear picture of system flow.

- It is logical rather than physical i.e., the elements of system do not depend on vendor or hardware.

- It is an approach that works from high-level overviews to lower-level details.

Structured Analysis Tools

During Structured Analysis, various tools and techniques are used for system development. They are –

- Data Flow Diagrams

- Data Dictionary

- Decision Trees

- Decision Tables

- Structured English

- Pseudo code

Data Flow Diagrams (DFD)

It is a technique developed by Larry Constantine to express the requirements of system in a graphical form.

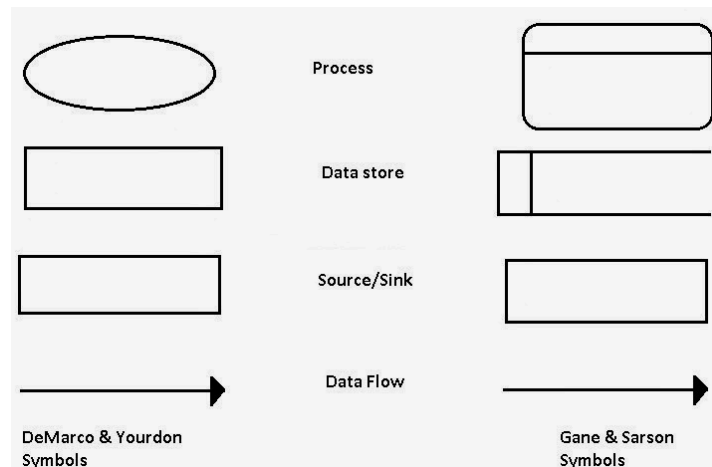
- It shows the flow of data between various functions of system and specifies how the current system is implemented.

It is an initial stage of design phase that functionally divides the requirement specifications down to the lowest level of detail.

Its graphical nature makes it a good communication tool between user and analyst or analyst and system designer.

It gives an overview of what data a system processes, what transformations are performed, what data are stored, what results are produced and where they flow.

Symbols used in DFD



Used Case Diagram

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. An effective use case diagram can help your team discuss and represent:

Scenarios in which your system or application interacts with people, organizations, or external systems

Goals that your system or application helps those entities (known as actors) achieve

The scope of your system

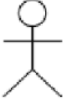

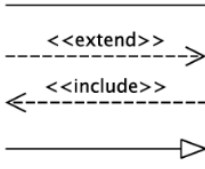
UML use case diagrams are ideal for:

Representing the goals of system-user interactions

Defining and organizing functional requirements in a system

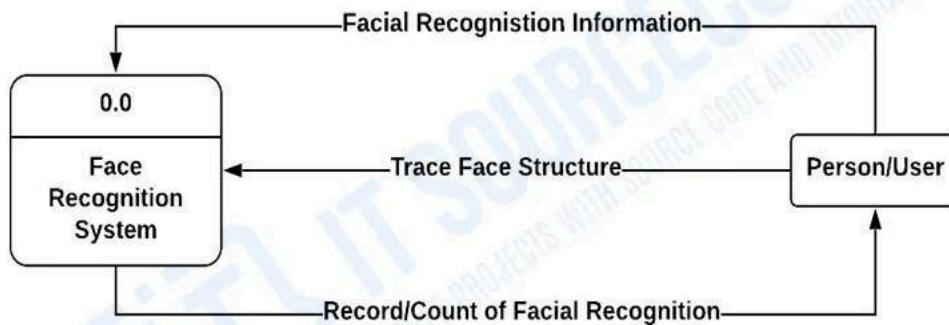
Specifying the context and requirements of a system

Modeling the basic flow of events in a use case

Symbol	Reference Name
	Actor
	Use case
	Relationship

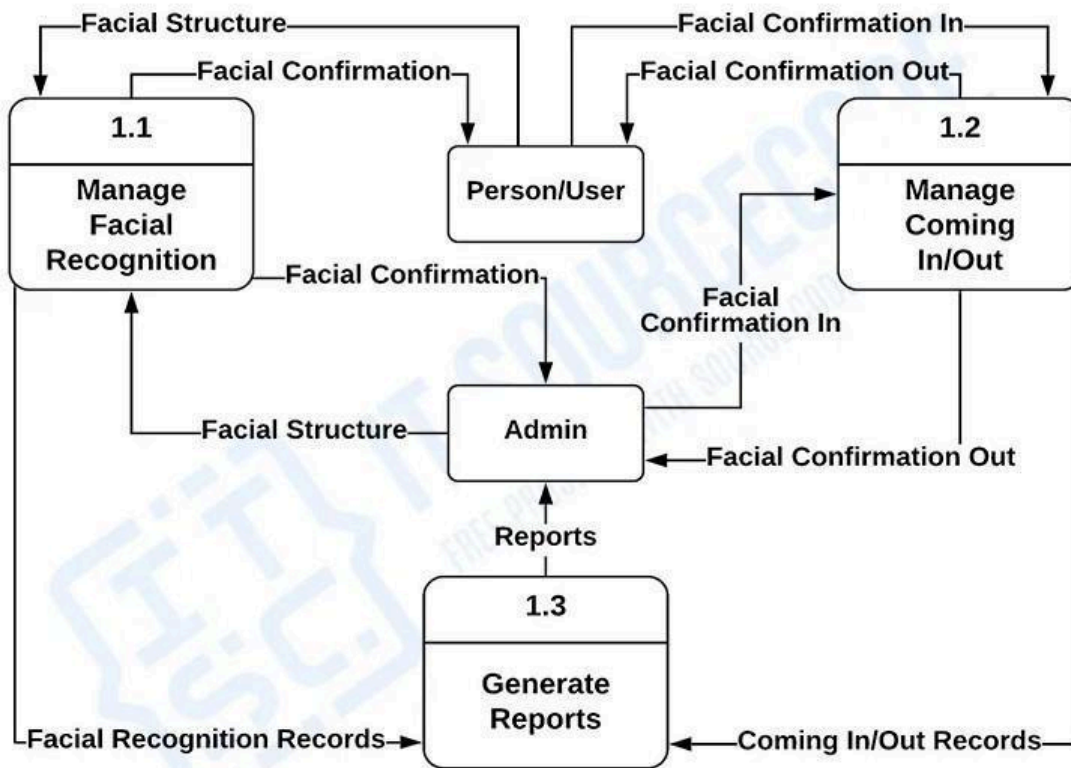
****Students draw the DFD (Level 0, Level 1, Level 2) and use case diagram for their mini project.****

FACE RECOGNITION SYSTEM



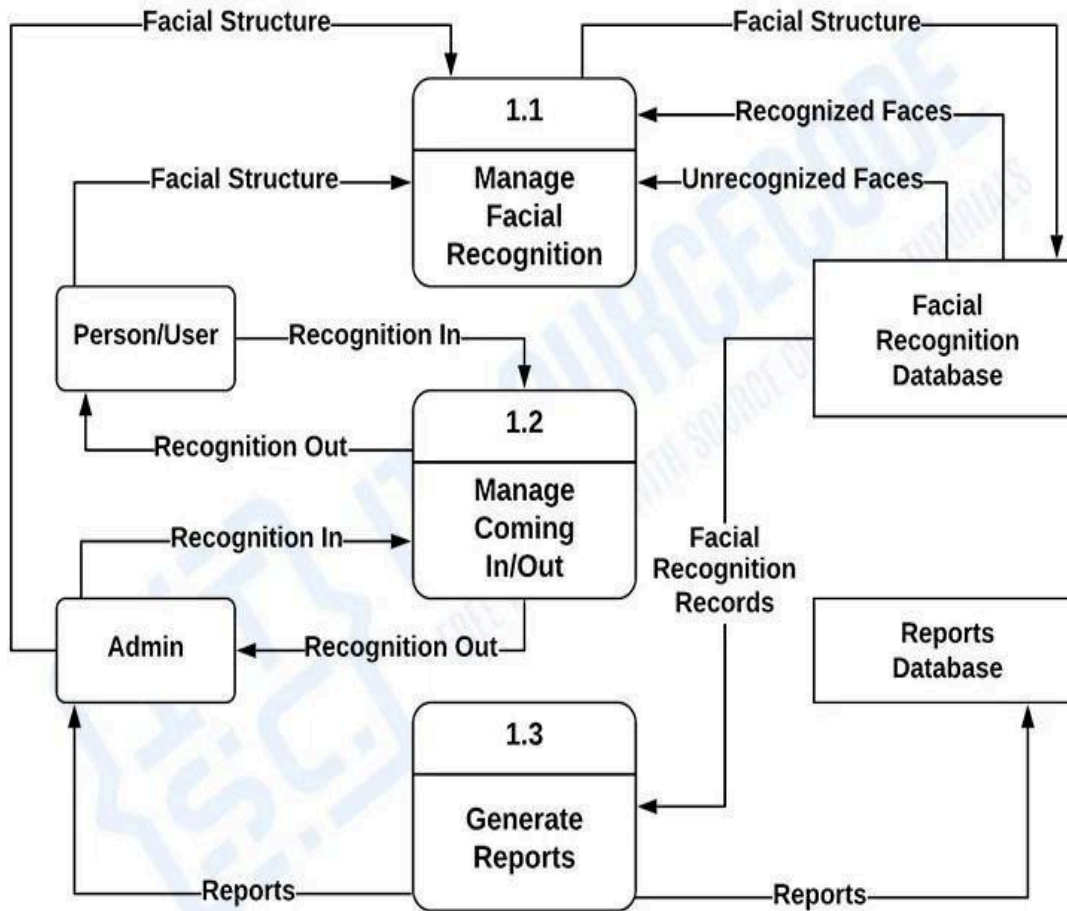
DATA FLOW DIAGRAM LEVEL 0

FACE RECOGNITION SYSTEM



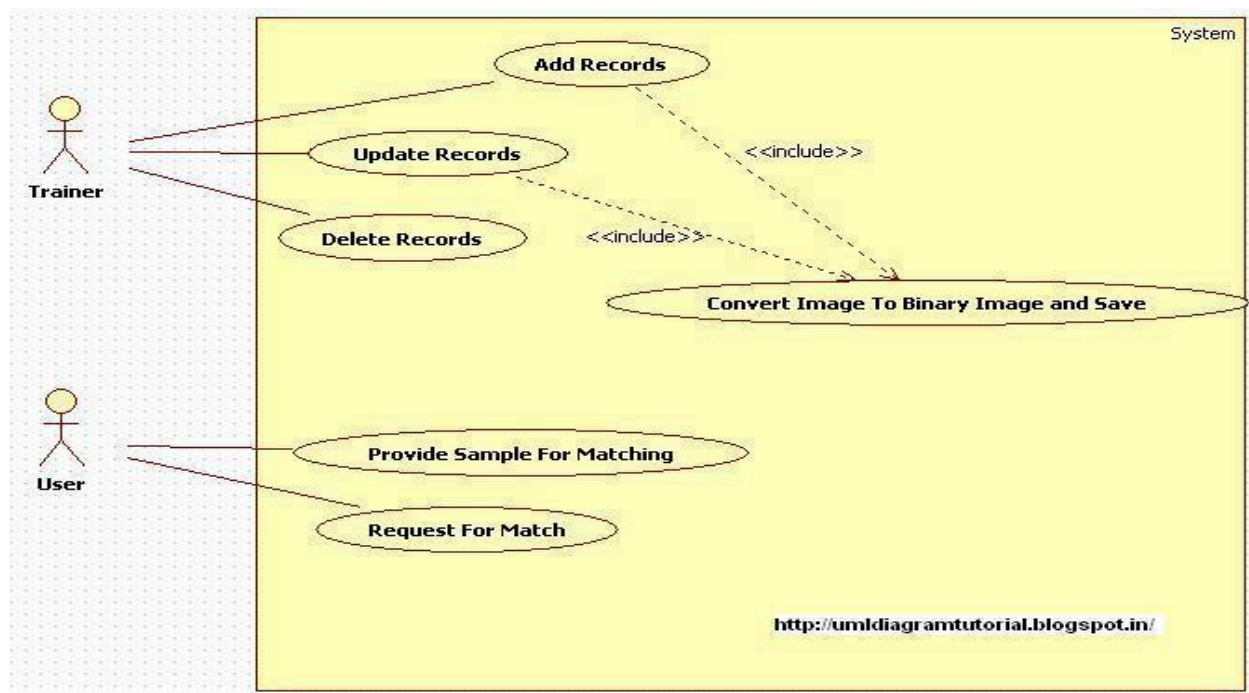
DATA FLOW DIAGRAM LEVEL 1

FACE RECOGNITION SYSTEM



DATA FLOW DIAGRAM LEVEL 2

USE CASE DIAGRAM :



1. Results:

.....
.....
.....

2. Conclusion: Thus Data Flow Diagram for the assigned project is drawn successfully.

3. Industrial Applications: Used to show the flow of data inside software

Questionnaire:

1. What is DFD?

.....
.....
.....

2. What is context diagram?

.....
.....
.....

3. Compare between data flow & control flow?

.....
.....
.....

4. What are different symbols used in DFD?

.....
.....
.....

5. What is Level 0 DFD?

.....
.....
.....

6. What is the difference between data flow & control flow?

.....
.....
.....

7. What is the significance of DFD?

.....
.....
.....

8. Which symbol is used to store data in DFD?

.....
.....
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

DOP	DOS	Experiment No 5	Remark	Signature

Experiment No: 5

Aim: Use of Metrics to estimate the cost

Software Used:

Metrics are measures of quantitative assessment commonly used for assessing, comparing, and tracking performance or production. Generally, a group of metrics will typically be used to build a dashboard that management or analysts review on a regular basis to maintain performance assessments, opinions, and business strategies.

Metrics have been used in accounting, operations, and performance analysis throughout history.

Software Cost Estimation :

For any new software project, it is necessary to know how much it will cost to develop and how much development time will it take. These estimates are needed before development is initiated, but how is this done? Several estimation procedures have been developed and are having the following attributes in common.

Project scope must be established in advanced.

Software metrics are used as a support from which evaluation is made.

The project is broken into small PCs which are estimated individually.

To achieve true cost & schedule estimate, several option arise.

Delay estimation

Used symbol decomposition techniques to generate project cost and schedule estimates.

Acquire one or more automated estimation tools.

Uses of Cost Estimation

During the planning stage, one needs to choose how many engineers are required for the project and to develop a schedule.

In monitoring the project's progress, one needs to access whether the project is progressing according to the procedure and takes corrective action, if necessary.

• Estimated LOC

User interface	2,300
2-D geometric analysis	5,300
3 D geometric analysis	6, 800
Database management	3,500
Graphic display facilities	4,950
I/O control function	2,100
Analysis function	8,400
Total estimated	33,350
Average productivity based on historical data 620 LOC/pm	
Rs. 8,000 per month/PP	
Rs. 12.90/LOC	
If the estimated project is 33,350 LOC,	

then the total estimated project cost is **Rs.432000**(33350 x 12.90) and the estimated effort is **54** (33350/620)person-months.

Consider a project with the following **functional units**

Number of user inputs = 50

Number of user outputs = 40

Number of user enquiries = 35

Number of user files = 06

Number of external interfaces = 04

Assuming all complexity adjustment factors and weighing factors as average

Calculate the function points for the project.

Thus function points can be calculated as:

= $(200 + 200 + 140 + 60 + 28) \times [0.65 + (0.01 \times (14 \times 3))]$ i.e average of 14 questions]

= $628 \times [0.65 + 0.42] = 628 \times (1.07)$

= 672

Thus the function points for the project will be 672.

1. Results:

.....
.....

2.Conclusion: Thus Project Management tool has been used to estimate & schedule the assigned project.

3.Industrial Applications: used to estimate the requirements of software & plan accordingly.

Questionnaire:

- What is software project management?

.....
.....
.....

- What are different types of designs in a software project?

.....
.....
.....

- What are the activities of project management?

.....
.....
.....

- What is project breakdown structure?

.....

.....

.....

- What is risk analysis?

.....

.....

.....

- What is COCOMO model?

.....

.....

.....

- How to estimate efforts?

.....

.....

.....

- What are different cost estimation methods?

.....

.....

.....

- What is empirical estimation method?

.....

.....

.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

DOP	DOS	Experiment No 6	Remark	Signature

Experiment No: 6

Aim: Scheduling and tracking of the project

Software Used: Gantt Project

Theory:-

Project Scheduling

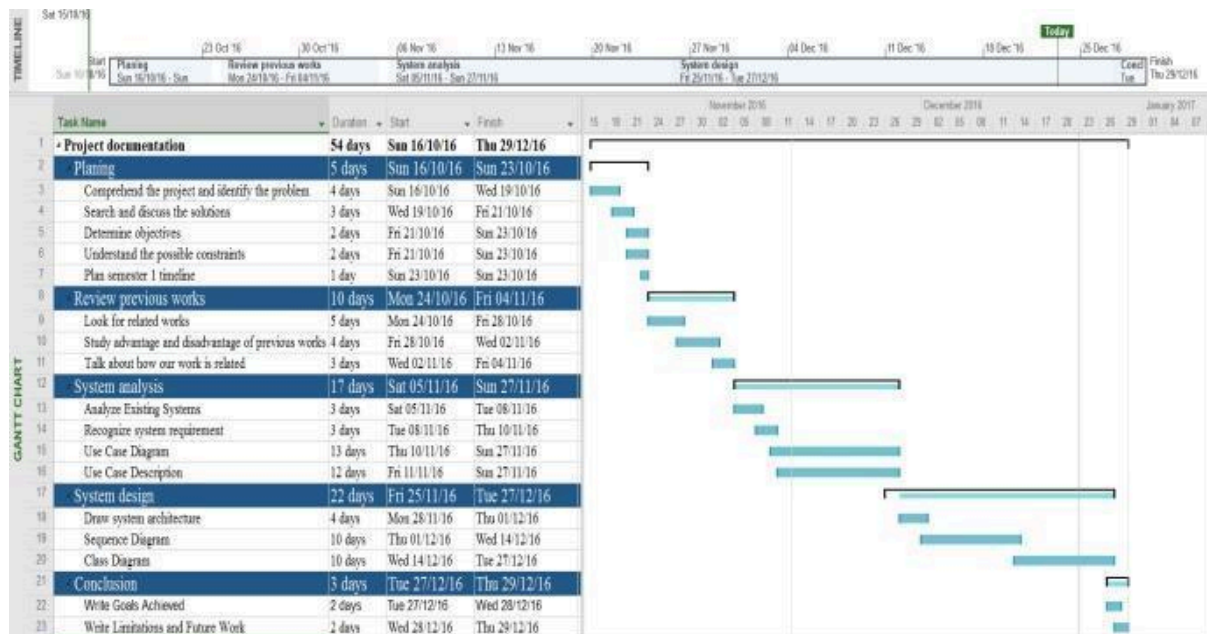
Project-task scheduling is a significant project planning activity. It comprises deciding which functions would be taken up when. To schedule the project plan, a software project manager wants to do the following:

1. Identify all the functions required to complete the project.
2. Break down large functions into small activities.
3. Determine the dependency among various activities.
4. Establish the most likely size for the time duration required to complete the activities.
5. Allocate resources to activities.
6. Plan the beginning and ending dates for different activities.
7. Determine the critical path. A critical way is the group of activities that decide the duration of the project.

The first method in scheduling a software plan involves identifying all the functions required to complete the project. A good judgment of the intricacies of the project and the development process helps the supervisor to identify the critical role of the project effectively. Next, the large functions are broken down into a valid set of small activities which would be assigned to various engineers. The work breakdown structure formalism supports the manager to breakdown the function systematically after the project manager has broken down the purpose and constructs the work breakdown structure; he has to find the dependency among the activities. Dependency among the various activities determines the order in which the various events would be carried out. If an activity A necessary the results of another activity B, then activity A must be scheduled after activity B. In general, the function dependencies describe a partial ordering among functions, i.e., each service may precede a subset of other functions, but some functions might not have any precedence ordering describe between them (called concurrent function). The dependency among the activities is defined in the pattern of an activity network.

Once the activity network representation has been processed out, resources are allocated to every activity. Resource allocation is usually done using a Gantt chart. After resource allocation is completed, a PERT chart representation is developed. The PERT chart representation is useful for program monitoring and control. For task scheduling, the project plan needs to decompose the project functions into a set of activities. The time frame when every activity is to be

performed is to be determined. The end of every action is called a milestone. The project manager tracks the function of a project by audit the timely completion of the milestones. If he examines that the milestones start getting delayed, then he has to handle the activities carefully so that the complete deadline can still be met.



1.Results:

.....

2.Conclusion: Thus Project Management tool has been used to schedule & tracking the assigned project.

3.Industrial Applications: used to schedule and tracking the requirements of software & plan accordingly.

Questionnaire:

- What is Scheduling in software Engineering ?

.....

.....

.....

- What is PERT?

.....

.....

.....

- What is CPM ?

.....

.....

.....

- What is work break down structure?

.....

.....

.....

- What is Tracking Schedule?

.....

.....

.....

- What is Time Line Charts ?

.....

.....

.....

- What is Earned Value Analysis in Project tracking?

.....

.....

.....

- What is feature of EVA?

.....

.....

.....

- What are need for EVA?

.....
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011

DOP	DOS	Experiment No 7	Remark	Signature

Experiment No: 7

Aim: Write test cases for black box testing

Software Used: Selenium

Theory:-

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing.

In this method, tester selects a function and gives input value to examine its functionality, and checks whether the function is giving expected output or not. If the function produces correct output, then it is passed in testing, otherwise failed. The test team reports the result to the development team and then tests the next function. After completing testing of all functions if there are severe problems, then it is given back to the development team for correction.



Generic steps of black box testing

- The black box test is based on the specification of requirements, so it is examined in the beginning.
- In the second step, the tester creates a positive test scenario and an adverse test scenario by selecting valid and invalid input values to check that the software is processing them correctly or incorrectly.
- In the third step, the tester develops various test cases such as decision table, all pairs test, equivalent division, error estimation, cause-effect graph, etc.
- The fourth phase includes the execution of all test cases.
- In the fifth step, the tester compares the expected output against the actual output.
- In the sixth and final step, if there is any flaw in the software, then it is cured and tested again.

Test procedure

The test procedure of black box testing is a kind of process in which the tester has specific knowledge about the software's work, and it develops test cases to check the accuracy of the software's functionality.

It does not require programming knowledge of the software. All test cases are designed by considering the input and output of a particular function. A tester knows about the definite output of a particular input, but not about how the result is arising. There are various techniques used in black box testing for testing like decision table technique, boundary value analysis technique, state transition, All-pair testing, cause-effect graph technique, equivalence partitioning technique, error guessing technique, use case technique and user story technique.

Test cases

Test cases are created considering the specification of the requirements. These test cases are generally created from working descriptions of the software including requirements, design parameters, and other specifications. For the testing, the test designer selects both positive test scenario by taking valid input values and adverse test scenario by taking invalid input values to determine the correct output. Test cases are mainly designed for functional testing but can also be used for non-functional testing. Test cases are designed by the testing team; there is not any involvement of the development team of software.

Techniques Used in Black Box Testing

<u>Decision Table Technique</u>	Decision Table Technique is a systematic approach where various input combinations and their respective system behavior are captured in a tabular form. It is appropriate for the functions that have a logical relationship between two and more than two inputs.
<u>Boundary Value Technique</u>	Boundary Value Technique is used to test boundary values, boundary values are those that contain the upper and lower limit of a variable. It tests, while entering boundary value whether the software is producing correct output or not.
<u>State Transition Technique</u>	State Transition Technique is used to capture the behavior of the software application when different input values are given to the same function. This applies to those types of applications that provide the specific number of attempts to access the application.
<u>All-pair Testing Technique</u>	All-pair testing Technique is used to test all the possible discrete combinations of values. This combinational method is used for testing the application that uses checkbox input, radio button input, list box, text box, etc.
<u>Cause-Effect Technique</u>	Cause-Effect Technique underlines the relationship between a given result and all the factors affecting the result. It is based on a collection of requirements.
<u>Equivalence Partitioning Technique</u>	Equivalence partitioning is a technique of software testing in which input data divided into partitions of valid and invalid values, and it is mandatory that all partitions must exhibit the same behavior.
<u>Error Guessing Technique</u>	Error guessing is a technique in which there is no specific method for identifying the error. It is based on the experience of the test analyst, where the tester uses the experience to guess the problematic areas of the software.
<u>Use Case Technique</u>	Use case Technique used to identify the test cases from the beginning to the end of the system as per the usage of the system. By using this

technique, the test team creates a test scenario that can exercise the entire software based on the functionality of each function from start to end.

Examples:**1- Boundary Value Analysis**

Example: Taking inputs for a test case data for an age section should accept a valid data of anything between 1-100. According to BVP analysis, the software will be tested against four test data as -1, 1, 100, and 101 to check the system's response using the boundary values.

2- Equivalence partitioning

Example: Taking inputs for a test case data for the example mentioned above will have three classes from which one data will be tested.

Valid class: 1 to 100 (any number), Invalid class: -1 (checking the lowest of lowest), Invalid class: 101(highest of highest).

3- State Transition Testing

Example: A login page will let you input username and password until three attempts. Each incorrect password will be sent the user to the login page. After the third attempt, the user will be sent to an error page. This state transition method considers the various states of the system and the inputs to pass only the right sequence of the testing.

4- Decision Table Testing

Example: A food delivery application will check various payment modes as input to place the order — decision making based on the table.

Case1: If the end-user has a card, then the system will not check for cash or coupon and will take action to place the order.

Case2: If the end-user has a coupon will not be checked for a card or cash and action will be taken.

Case3: if the end-user has cash, the action will be taken.

Case4: If the end-user doesn't have anything, then action will not be taken.

5- Graph-Based Testing:

It is similar to a decision-based test case design approach where the relationship between links and input cases are considered.

6- Error Guessing Technique:

This method of designing test cases is about guessing the output and input to fix any errors that might be present in the system. It depends on the skills and judgment of the tester.

7-Comparison testing

This method uses the two different versions of the same software to compare and validate the results.

Test case :

A simple login screen of software or a web application will be tested for seamless user login. The login screen has two fields, username and password as an input and the output will be to enable access to the system.

A black box testing will not consider the specifications of the code, and it will test the valid username and password to login to the right account.

This form of testing technique will check the input and output.

A user logged in when inputs a present username and correct password

A user receives an error message when enters username and incorrect password

The black box testing is also known as an opaque, closed box, function-centric testing. It emphasizes on the behavior of the software. Black box testing checks scenarios where the system can break.

1.Results:

.....
.....
.....

2.Conclusion: Thus test cases for Black box testing are developed for the assigned project.

3.Industrial Applications: Checking of functionality of an application without knowing its internal structure of code and various types of errors in software.

Questionnaire:

1. What is Software testing?

.....
.....

2. What is Advantages of testing?

.....
.....
.....

3. What is Types of software testing?

.....
.....
.....

4. What is the Principal of software testing?

.....
.....

.....
5. What is a types of Black box testing ?
.....
.....

.....
6. Compare Manual & Automaton testing?
.....
.....

.....
7. What is types System Testing ?
.....
.....

.....
8. What is objective of software testing ?
.....
.....

.....
9. What is software testing process?
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011 Pearson Education

DOP	DOS	Experiment No 8	Remark	Signature

Experiment No: 8

Aim: Write test cases for white box testing

Software Used: JSUnit.net

Theory:-

White Box Testing is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

White box testing which also known as glass box is **testing, structural testing, clear box testing, open box testing and transparent box testing**. It tests internal coding and infrastructure of a software focus on checking of predefined inputs against expected and desired outputs. It is based on inner workings of an application and revolves around internal structure testing. In this type of testing programming skills are required to design test cases. The primary goal of white box testing is to focus on the flow of inputs and outputs through the software and strengthening the security of the software.

The term 'white box' is used because of the internal perspective of the system. The clear box or white box or transparent box name denote the ability to see through the software's outer shell into its inner workings.

Developers do white box testing. In this, the developer will test every line of the code of the program. The developers perform the White-box testing and then send the application or the software to the testing team, where they will perform the [black box testing](#) and verify the application along with the requirements and identify the bugs and sends it to the developer.

The developer fixes the bugs and does one round of white box testing and sends it to the testing team. Here, fixing the bugs implies that the bug is deleted, and the particular feature is working fine on the application.

Here, the test engineers will not include in fixing the defects for the following reasons:

- Fixing the bug might interrupt the other features. Therefore, the test engineer should always find the bugs, and developers should still be doing the bug fixes.
- If the test engineers spend most of the time fixing the defects, then they may be unable to find the other bugs in the application.

Techniques Used in White Box Testing

<u>Data Flow Testing</u>	Data flow testing is a group of testing strategies that examines the control flow of programs in order to explore the sequence of variables according to the sequence of events.
<u>Control Flow Testing</u>	Control flow testing determines the execution order of statements or instructions of the program through a control structure. The control structure of a program is used to develop a test case for the program. In this technique, a particular part of a large program is selected by the tester to set the testing path. Test cases represented by the control graph of the program.
<u>Branch Testing</u>	Branch coverage technique is used to cover all branches of the control flow graph. It covers all the possible outcomes (true and false) of each condition of decision point at least once.
<u>Statement Testing</u>	Statement coverage technique is used to design white box test cases. This technique involves execution of all statements of the source code at least once. It is used to calculate the total number of executed statements in the source code, out of total statements present in the source code.
<u>Decision Testing</u>	This technique reports true and false outcomes of Boolean expressions. Whenever there is a possibility of two or more outcomes from the statements like do while statement, if statement and case statement (Control flow statements), it is considered as decision point because there are two outcomes either true or false.

Let us consider the following code :

INPUT A & B

C = A + B

IF C>100

PRINT "ITS DONE"

Now in the first, line, we assign the value of A and B. Let us suppose A = 60 and B = 50. Moving on to the second line, now C is assigned a value of A+B, here A = 60 and B = 50, hence C = 110. Moving on to the third line, we will check if C > 100, here the condition is true and hence we should get our result as ITS DONE

Test case :

Step 1: Identify the feature, component, program to be tested

Zero in on what you want to test.

Step 2: Plot all possible paths in a flowgraph

Step 3: Identify all possible paths from the flowgraph

Step 4: Write Test Cases to cover every single path on the flowgraph.

Step 5: Execute, rinse, repeat

You're now ready to execute white box Testing for the identified system, component or module

1.Results:

.....
.....
.....

2.Conclusion: Thus test cases for white box testing are developed for the assigned project.

3.Industrial Applications: Checking of logical errors in software.

Questionnaire:

1.What is white box testing?

.....
.....

2.What is cyclomatic complexity?

.....
.....
.....
.....
.....

3.What is different techniques in white box testing ?

.....
.....

.....

4.What is the advantages of white box testing?

.....

.....

.....

5.What is disadvantages of white box testing?

.....

.....

.....

6.Compare White box & Black box testing?

.....

.....

.....

7.What is characteristics of good test ?

.....

.....

.....

8.What is condition testing ?

.....

.....

.....

.....

.....

9..What is data flow testing ?

.....

.....

.....

.....

.....

.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011 Pearson Education

DOP	DOS	Experiment No 9	Remark	Signature

Experiment No: 9

Aim: Preparation of Risk Mitigation, Monitoring and Management plan (RMMM.)

Software Used: Ms Word

Theory:-

RMMM Plan :

A risk management technique is usually seen in the software Project plan. This can be divided into Risk Mitigation, Monitoring, and Management Plan (RMMM). In this plan, all works are done as part of risk analysis. As part of the overall project plan project manager generally uses this RMMM plan.

In some software teams, risk is documented with the help of a Risk Information Sheet (RIS). This RIS is controlled by using a database system for easier management of information i.e creation, priority ordering, searching, and other analysis. After documentation of RMMM and start of a project, risk mitigation and monitoring steps will start.

Risk Mitigation :

It is an activity used to avoid problems (Risk Avoidance).

Steps for mitigating the risks as follows.

1. Finding out the risk.
2. Removing causes that are the reason for risk creation.
3. Controlling the corresponding documents from time to time.
4. Conducting timely reviews to speed up the work.

Risk Monitoring :

It is an activity used for project tracking.

It has the following primary objectives as follows.

1. To check if predicted risks occur or not.
2. To ensure proper application of risk aversion steps defined for risk.
3. To collect data for future risk analysis.
4. To allocate what problems are caused by which risks throughout the project.

Risk Management and planning :

It assumes that the mitigation activity failed and the risk is a reality. This task is done by Project manager when risk becomes reality and causes severe problems. If the project manager effectively uses project mitigation to remove risks successfully then it is easier to manage the risks. This shows that the response that will be taken for each risk by a manager. The main objective of the risk management plan is the risk register. This risk register describes and focuses on the predicted threats to a software project.

Steps for Risk Management

1. Identify possible risks and recognize what can go wrong
2. Analyse each risk to estimate the probability that it will occur and the impact (i.e., damage) that it will do if it does occur
3. Rank the risks by probability and impact. Impact may be negligible, marginal, critical, and catastrophic.
4. Develop a contingency plan to manage those risks having high probability and high impact

Risk Table

Risks ID	Risks	Category	Probability	Impact

Impact Values: 1 – Catastrophic 2 – Critical 3 – Marginal 4 – Negligible

Example:

Risks	Category	Probability	Impact
Computer Crash	TI	70%	1
Late Delivery	BU	30%	1
Technology will not Meet Expectations	TE	25%	1
End Users Resist System	BU	20%	1
Changes in Requirements	PS	20%	2
Lack of Development Experience	TI	20%	2
Lack of Database Stability	TI	40%	2
Poor Quality Documentation	BU	35%	2
Deviation from Software Engineering Standards	PI	10%	3
Poor Comments in Code	TI	20%	4

PS (Product Size) TE (Technical), CU (Customer) DE (Development) etc.

Risk: Computer Crash ·

Mitigation

The cost associated with a computer crash resulting in a loss of data is crucial. A computer crash itself is not crucial, but rather the loss of data. A loss of data will result in not being able to deliver the product to the customer. This will result in a not receiving a letter of acceptance from the customer. Without the letter of acceptance, the group will receive a failing grade for the course. As a result the organization is taking steps to make multiple backup copies of the software in development and all documentation associated with it, in multiple locations.

Monitoring

When working on the product or documentation, the staff member should always be aware of the stability of the computing environment they're working in. Any changes in the stability of the environment should be recognized and taken seriously.

Management

The lack of a stable-computing environment is extremely hazardous to a software development team. In the event that the computing environment is found unstable, the development team should cease work on that system until the environment is made stable again, or should move to a system that is stable and continue working there.

Risk: Late Delivery

Mitigation

The cost associated with a late delivery is critical. A late delivery will result in a late delivery of a letter of acceptance from the customer. Without the letter of acceptance, the group will receive a failing grade for the course. Steps have been taken to ensure a timely delivery by gauging the scope of project based on the delivery deadline.

Monitoring

A schedule has been established to monitor project status. Falling behind schedule would indicate a potential for late delivery. The schedule will be followed closely during all development stages.

Management

Late delivery would be a catastrophic failure in the project development. If the project cannot be delivered on time the development team will not pass the course. If it becomes apparent that the project will not be completed on time, the only course of action available would be to request an extension to the deadline for the customer.

5 minimum risk can be :

- 1) Its not easy**
- 2) Controlled environment**
- 3) Privacy**
- 4) New directions**
- 5) Audience participation**

This can be minimum five risk for our project .

RMMM PLAN:

Risk events
and their
relationships
are defined

Probabilities
and conse-
quences of
risk events
are assessed

1. Results:

.....

.....

.....

2. Conclusion: Thus Risk Mitigation, Monitoring and Management Plan has been prepared successfully for the assigned project.

3. Industrial Applications: Used to monitor the risks & mitigate in developing the software.

Questionnaire:

1. What is full form of RMMM?

.....

.....

.....

2. What is risk mitigation?

.....

.....

.....

3. What is risk management?

.....

.....

.....

4. What is risk monitoring?

.....

.....

.....

5. What is risk projection?

.....

.....

.....

6. What is slack time?

.....

.....

.....

7. What is cost slippage?

.....

.....

8. Which software development model does handle risks effectively?

.....
.....
.....
9. What are different risk mitigation technologies?
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison

Link for Download software <https://www.ganttproject.biz>

DOP	DOS	Experiment No 10	Remark	Signature

Experiment No: 10

Aim: Version control of the project

Software Used: [GitHub](#)

Theory:-

Version control systems are a category of software tools that helps in recording changes made to files by keeping a track of modifications done to the code.

As we know that a software product is developed in collaboration by a group of developers they might be located at different locations and each one of them contributes in some specific kind of functionality/features. So in order to contribute to the product, they made modifications in the source code(either by adding or removing). A version control system is a kind of software that helps the developer team to efficiently communicate and manage (track) all the changes that have been made to the source code along with the information like who made and what change has been made. A separate branch is created for every contributor who made the changes and the changes aren't merged into the original source code unless all are analyzed as soon as the changes are green signaled they merged to the main source code. It not only keeps source code organized but also improves productivity by making the development process smooth.

Benefits of the version control system:

- a)** Enhances the project development speed by providing efficient collaboration,
- b)** Leverages the productivity, expedite product delivery, and skills of the employees through better communication and assistance,
- c)** Reduce possibilities of errors and conflicts meanwhile project development through traceability to every small change,
- d)** Employees or contributor of the project can contribute from anywhere irrespective of the different geographical locations through this VCS,
- e)** For each different contributor of the project a different working copy is maintained and not

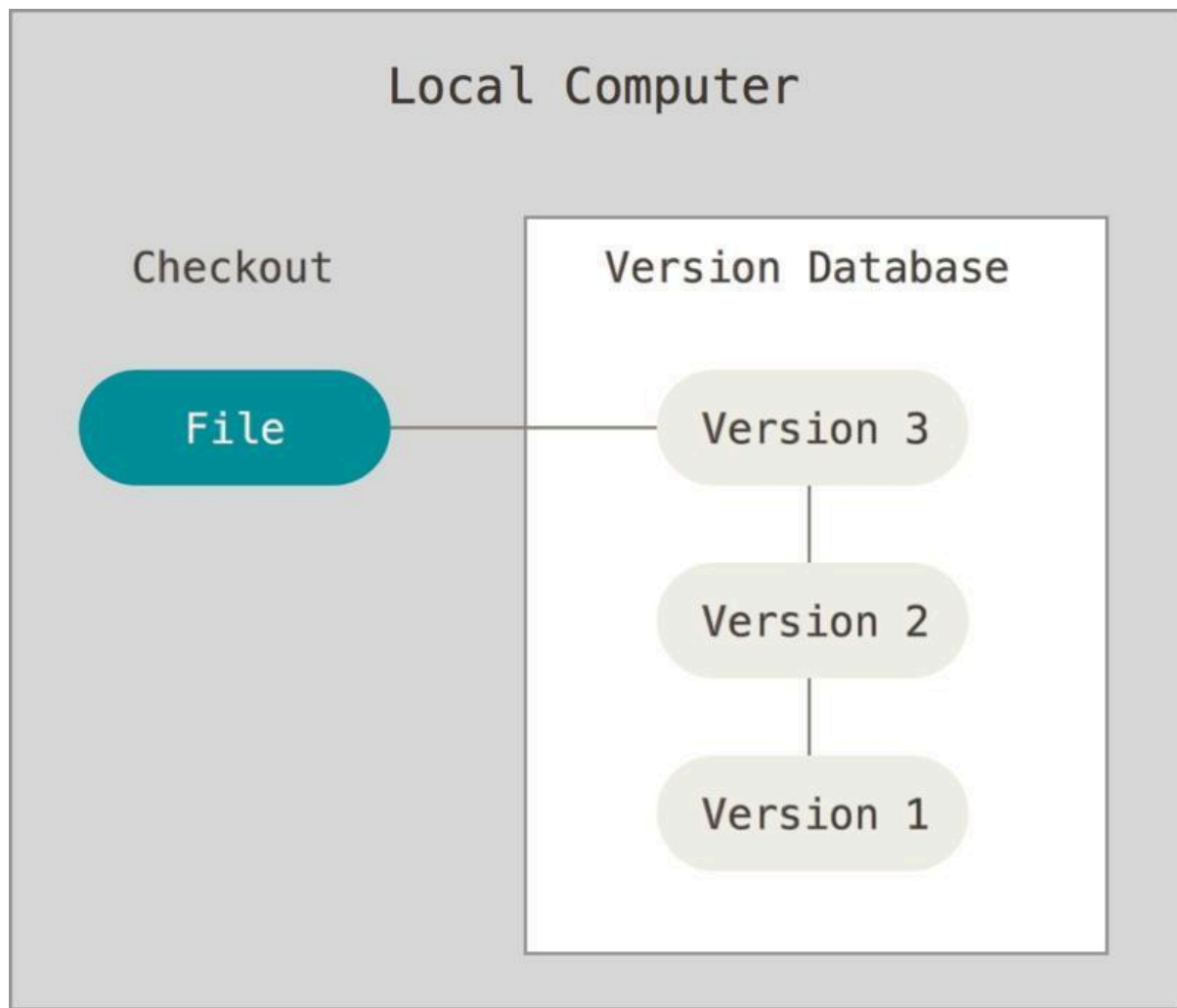
merged to the main file unless the working copy is validated. A most popular example is **Git**, **Helix core**, **Microsoft TFS**,

f) Helps in recovery in case of any disaster or contingent situation,

g) Informs us about Who, What, When, Why changes have been made.

Use of Version Control System:

- **A repository:** It can be thought of as a database of changes. It contains all the edits and historical versions (snapshots) of the project.
- **Copy of Work (sometimes called as checkout):** It is the personal copy of all the files in a project. You can edit to this copy, without affecting the work of others and you can finally commit your changes to a repository when you are done making your changes.



Results:

.....
.....

Conclusion:

Thus Change specifications are prepared and different versions are made using SCM (Software Configuration Management) tool.

Industrial Applications: maintenance of software after change of user requirements

Questionnaire:

1. What is software configuration management?

.....
.....
.....

2. What are different tools for SCM?

.....
.....
.....

3. What are different activities in SCM?

.....
.....
.....

4. How to control the proposed changes?

.....
.....
.....

5. What is the importance of SCM?

.....
.....
.....

6. What is software reliability?

.....
.....
.....

7. What are different software quality measurement metrics ?

.....
.....
.....

8. Compare verification vs validation?

.....
.....
.....

References:

1. Roger Pressman, Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill, 2010
2. Ian Somerville, Software Engineering, 9th edition, Addison Wesley, 2011
Pearson Education