Parul University

Faculty of Engineering & Technology
Subject-Name: Software Engineering
Subject-Code: 303105254
B.Tech AI & DS Year: 2rd year/4th

Parul University

FACULTY OF ENGINEERING AND TECHNOLOGY

Artificial Intelligence & Data Science

Software Engineering

(303105254)

4th SEMESTER

Laboratory Manual

Enrollment No:

Parul University

Faculty of Engineering & Technology
Subject-Name: Software Engineering
Subject-Code: 303105254
B.Tech AI & DS Year: 2rd year/4th

CERTIFICATE

This is to certify that **Mr. ----------** with enrolment no.

---------- and **4th Semester/ AIDS (Batch No)** has successfully

completed her laboratory experiments in the **Software Engineering (303105254)** from the department of Artificial Intelligence & Data Science **during the academic year 2025-26**

योगः कर्मसु कौशलम्
PARUL UNIVERSITY

Date of Submission: .....

Staff In charge: .....

Head Of Department: .....

Enrollment No:

Parul University

Faculty of Engineering & Technology
Subject-Name: Software Engineering
Subject-Code: 303105254
B.Tech AI & DS Year: 2rd year/4th

Table of Content

SR. NO.	TITLE	Page no From	To	Date of Start	Date of completion	Sign	Marks
1	Project definition and objective of the specifiedmodule and perform requirement EngineeringProcess						
2	Identify Suitable Design and Implementation model from the different software engineering models.						
3	Prepare a Software Requirement Specification(SRS) for the selected module						
4	Develop a Software project management planning (SPMP) for the specific module						
5	Do Cost Estimation use Different Software Cost Estimation Models.						
6	Prepare System Analysis and System Design of Identified Requirement specification using structure design as DFD with Data Dictionary and Structure chart for the specific module.						
7	Designing the module using Object						

Enrollment No:

	Oriented approach including Use case Diagram with scenarios, Class Diagram and State Diagram, Sequence Diagram and Activity Diagram.						
8	**Defining Coding Standardsand walk through.**						
9	**Write the test cases for theidentified module.**						
10	**Demonstrate the use of different Testing Tools with comparison.**						
11	**Define security and qualityaspects of the identified module.**						

Enrollment No:

Parul University

Faculty of Engineering & Technology
Subject-Name: Software Engineering
Subject-Code: 303105254
B.Tech AI & DS Year: 2rd year/4th

Practical 1

Project Title: Attendance Management System

Definition: Attendance Management keeps track of your employee hours. It is the system you use to document the time your employees work and the time they take off. Attendance Management can be done by recording employee hours on paper, using spreadsheets, punching time cards, or using online attendance software for your company.

Information: Attendance management is the act of managing attendance or presence in a work setting to minimize loss due to employee down time. Attendance control has traditionally been approached using time clocks, timesheets, and time tracking software, but attendance management goes beyond this to provide a working environment which maximizes and motivates employee attendance. Recently it has become possible to collect attendance data automatically through using real-time location systems, which also allow for cross-linking between attendance data and performance.
Attendance management takes place in all educational campuses be they university

Objectives: It facilitates to access the attendance information of a particular student in a particular class. ... This system will also help in evaluating attendance eligibility criteria of a student. Purpose The purpose of developing attendance management system is to computerized the tradition way of taking attendance.

Functions:

i. Administrator
ii. User (Teacher & Student)

Administrator: Administrator have the rights to manage student details, add a new student, provide register number for all students, assign each student a course etc. Administrator can update his profile, and also can give help to teachers and students.
User: There are 2 users

a. **Student:** Student do the login and see profile. Attendance Details etc.
b. **Teacher:** Add student, view the student details and take attendance student.

Features

- Manage attendance of all the students and staff from one place.
- Manage basic attendance, day boarding attendance, lunch attendance and snacks attendance for students if the school provides day boarding facilities.
- Simple and easy interface for filling attendance of all staff and students.
- The administrator can keep a note of staff attendance and keep a record of the same.
- As staff attendance is linked with the payroll module, it allows automatic calculation of working days and generate pay slips for all the staff members.

Enrollment No:

- Cumulative term, mid-term and annual attendance report can be generated.
- Integrated with holiday management module so that holidays are already assigned in the attendance.
- Facility to import and export the student and staff attendance for a specific periods.

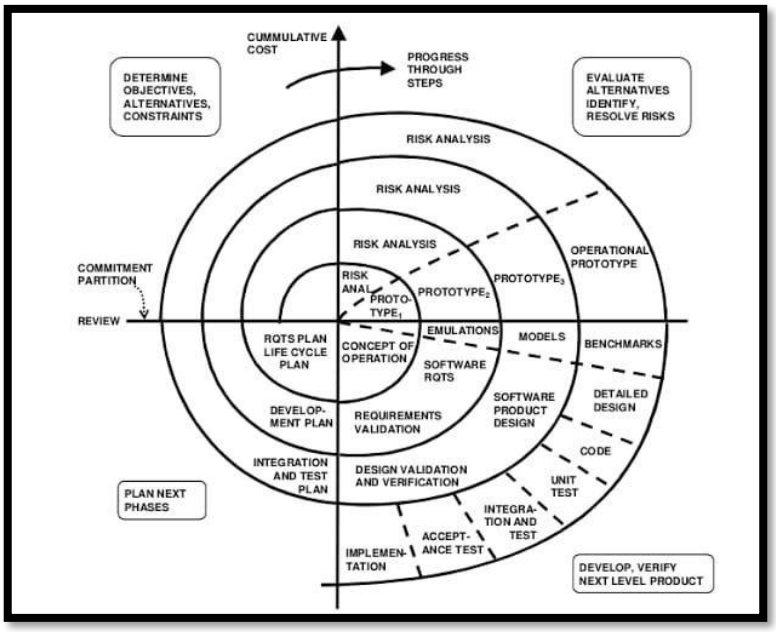
Practical 2

Aim: Identify Suitable Design and Implementation model from the different software engineering model

Spiral Model

In each phase of the Spiral Model, the features of the product dated and analysed, and the risks at that point in time are identified and are resolved through prototyping. Thus, this model is much more flexible compared to other SDLC models.

Spiral model is one of the most important Software Development Life Cycle models, which provides support for Risk Handling. In its diagrammatic representation, it looks like a spiral with many loops. The exact number of loops of the spiral is unknown and can vary from project to project. Each loop of the spiral is called a Phase of the software development process. The exact number of phases needed to develop the product can be varied by the project manager depending upon the project risks. As the project manager dynamically determines the number of phases, so the project manager has an important role to develop a product using the spiral model.



Justification for choosing Spiral Model

The spiral model is a systems development lifecycle (SDLC) method used for risk management that combines the iterative development process model with elements of the Waterfall model. The spiral model is used by software engineers and is favoured for large, expensive and complicated projects.

Enrollment No:

Justification for not choosing Waterfall Model

- Once an application is in the testing stage, it is very difficult to go back and change something that was not well-thought out in the concept stage.
- No working software is produced until late during the life cycle.
- High amounts of risk and uncertainty.
- Not a good model for complex and object-oriented projects.
- Poor model for long and ongoing projects.
- Not suitable for the projects where requirements are at a moderate to high risk of changing.

Justification for not choosing Incremental Model

- Needs good planning and design.
- Needs a clear and complete definition of the whole system before it can be broken down and built incrementally.
- Total cost is higher than waterfall.

Justification for not choosing Prototype Model

- This model is costly.
- It has poor documentation because of continuously changing customer requirements.
- There may be too much variation in requirements
- Customers sometimes demand the actual product to be delivered soon after seeing an early prototype

Justification why Spiral Model is better than any other model

- In each phase of the Spiral Model, the features of the product dated and analysed, and the risks at that point in time are identified and are resolved through prototyping.
- Thus, this model is much more flexible compared to other SDLC models.

Practical 3

Aim: Study Software Requirement engineering. Student should include SRS document for current semester project. SRS (Software requirement specification)

ASSUMPTION:

1. Users are already registered.

Requirement 1: Registration

Req1.1: Enter the details of user (as admin)
Input: details of user (external).
Output: save the details in system (internally)

Req1.2: Register of members done by admin
Input: Enter the details of members by admin
Output: save the details in system (internally)

Requirement 2: Show information of admin

Req 2.1: User information
Input: user id and password (internal)
Output: username, email, phone no., address personal information

Req 2.2: Account information
Input: account information
Output: name, date, personal information

Requirement 3: Show information of user

Req 3.1: User information
Input: user id and password (internal)
Output: username, email, phone no., address personal information attendance

Req 3.2: Firm information
Input: account information
Output: name, date, personal information, attendance

Enrollment No:

Requirement 4: Delete user (only by admin).
Input: username and password (external).
Output: Deletion status.

Requirement 5: Modification of the user attendance by admin

Req5.1: Marking daily attendance
Input: user name (External).
Output: user status.

Requirement 6: View status of every user under admin control
Requirement 7: Managing the user according to their specific role.
Requirement 8: User are not allowed to make changes in some categories

Requirement 9: User can make changes in personal information

Req 9.1: Add new changes when required
Input: new changes save (external).
Output: addition status.

Req 9.2: Change the new details already present.
Input: Changes to be updated (external).
Output: updating status.

SOFTWARE QUALITY ATTRIBUTES

Accuracy :

The level of accuracy in the proposed system will be higher . All operation would be done correctly and it ensures that whatever information is coming from the center is accurate.

Reliability :

The reliability of the system will be high due to the above stated reason. The reason for the increased reliability of the system is that now there would be proper storage of the information

No Redundancy :

In the system the utmost care would be that no information is repeated anywhere, in the storage or otherwise. This would assure economic use of storage space and consistency in the data stored.

Immediate retrieval of the information :

The main objective of the system is to provide for a quick and efficient retrieval of information. Any type of information would be available whenever the user requires.

Easy to operate :

The system should be easy to operate and should be such that it can be developed within a short period of time and fir in the limited budget of the user.

3) External Interfaces:

3.1) Hardware Interfaces:

All the components able to be executed on personal computer with Windows OS and Mac Os. Following are some minimum hardware requirement for the system which will be compatible with the os-

Hard disk- 1TB
RAM- 2GB
Processor : Intel core i3 7thgen
Wi-fi
Printer

3.2) Software Interfaces:

All the Interfaces will be ASPX pages running within the internet Browser. The system will be hosted in web server and languages and database used are-

1)C++
2) MongoDB
3) Os- Windows 7 or above and for Mac os Mavericks

3.3) Connection Protocols and interfaces:

This project can compatible with all platforms. Connections to the system will be over TCP/IP connection , project supports all types of web browsers .

3.4) User Interface –

The user interface screen shall respond within 5 seconds

Practical 4

Aim: Study Software project management planning. Student should write SPMP document for current semester project.

Scheduling in project management is the listing of activities, deliverables, and milestones within a project.
Seven Principle of Software Scheduling:

1) Compartmentalization –
   The project must be compartmentalized into a number of manageable activities ans tasks.
   To accomplish compartmentalization, both the product and process are decomposed.

2) Inderdependency-

   The interdependency of each compartmentalized activity or task must be determined.
   Some tasks must occur insequence while others can occur in parallel

3) Time allocation-

   Each task to be scheduled must be allocated some number of work units

4) Effort validation-

   Every project has a defined number of staff members. As time allocation occurs
   The project manager must ensure that no more than the allocated number of
   People has been scheduled at any given time.

5) Defined resposibilites-

   Every task should be assingned to a specific team member.

6) Defined Outcomes –

   Every task should have a defined outcome, normally a work product

7) Defined milestones-

   Every task or group of tasks should be associated with a project

Scheduling methods:-

Two project scheduling methods that can be applied to software development.

- **Program Evaluation and Review Technique (PERT)**
- **Critical Path Method (CPM)**

Both techniques are **driven** by **information already** developed in **earlier project planning** activities:

- estimates of effort
- a decomposition of the product function
- the selection of the appropriate process model and task set
- decomposition of the tasks that are selected

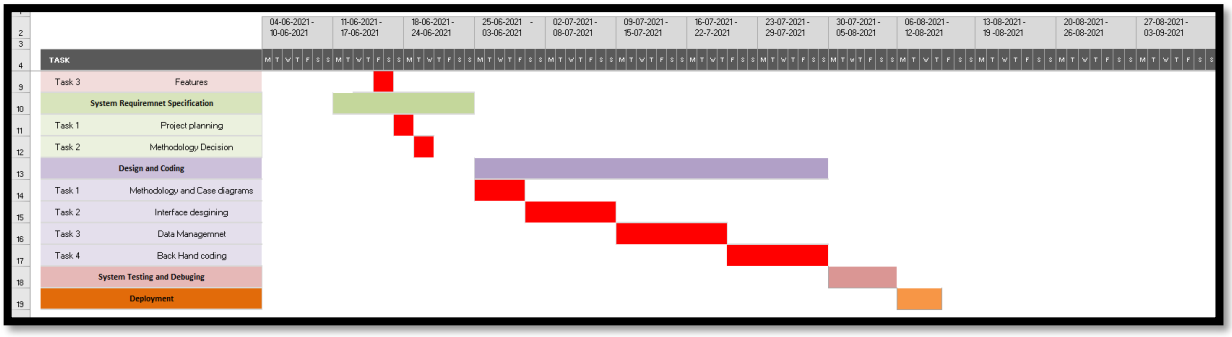
PERT and **CPM** provide quantitative tools that allow you to:

- **Determine the critical path**—the chain of tasks that determines the duration of the project
- **Establish "most likely" time estimates** for individual tasks by applying statistical models
- **Calculate "boundary times"** that define a "time window" for a particular task

WHAT IS GANTT CHART:-Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity. This allows you to see at a glance:

- What the various activities are
- When each activity begins and ends
- How long each activity is scheduled to last
- Where activities overlap with other activities, and by how much
- The start and end date of the whole project

Gantt Chart



Practical 5

Aim: Do cost and Effort Estimation using Software Cost Estimation Model
Objectives: To make use of COCOMO model to find out the cost of software development.

Organic: A development project can be considered of organic type, if the project deals with developing a well understood application program, the size of the development team is reasonably small, and the team members are experienced in developing similar types of projects

Model	Project Size	Nature of Project	Innovation	Dead Line	Development Environment
Organic	Typically 2-50 KLOC	Small Size Project, Experienced developers in the familiar environment, E.g. Payroll, Inventory projects etc.	Little	Not Tight	Familiar & In-house

COCOMO (Constructive Cost Estimation Model) was proposed by **Boehm**
According to Boehm, software cost estimation should be done through three stages:

- Basic COCOMO,
- Intermediate COCOMO, and
- Complete COCOMO

We are going for basic **COCOMO Model**

Basic COCOMO Model: The basic COCOMO model gives an approximate estimate of the project parameters. The basic COCOMO estimation model is given by the following expressions:
EFFORT = a1 * (KLOC)^a2 PM
Tdev =b1 x (EFFORT)^b2 Months

- KLOC is the estimated size of the software product expressed in Kilo Lines of Code,
- a_1, a_2, b_1, b_2 are constants for each category of software products,
- Tdev is the estimated time to develop the software, expressed in months,
- Effort is the total effort required to develop the software product, expressed in person months (PMs).
- Every line of source text should be calculated as one LOC irrespective of the actual number of instructions on that line
- If a single instruction spans several lines (say n lines), it is considered to be nLOC

Enrollment No:

- The values of a_1, a_2, b_1, b_2 for different categories of products (i.e., organic, semidetached, and embedded) as given by Boehm.

Cost Estimation of Software

Requirements:

External Input = 7

External Output = 2

No of Inquires – Status, Member information, Profit/loss

No of External Files- Data bases

No of External Interfaces = Client Machine

EI = 7 * 3 = 21

EO = 2 * 4 = 8

EQs = 3 * 3 = 9

IFS = 1 * 7 = 7

EIFS= 2 * 5 = 10

Count Total = 55

Value Adjustment Factors (VAF)

F1: Does the system require reliable backup & recovery?
5

F2: Are data communications required?
5

F3: Are there distributed processing functions?
1

F4: Is performance critical?
3

F5: Will the system run in an existing, heavily utilized operational environment?
0

F6: Does the system require online data entry?
2

F7: Does the online data entry require the input transaction to be built over multiple screens or operation? 3

F8: Are the master files updated online?
4

F9: Are the inputs, outputs, files or inquiries complex?
1

F10: Is the internal processing complex?
1

F11: Is the code designed to be reusable?
2

F12: Are the conversion and installation included in design?
1

F13: Is the system design for multiple installations in different organizations?
2

F14: Is the application designed to facilitate change and ease of use by the user?
4

∑FI=5+5+1+3+2+3+4+1+1+2+1+1+2+4=34

Function Point = Count Total* [0.65*0.01(∑Fi)]

Function Point = 55*[0.65*0.01(34)]

Function Point = 54.45 FP

Now 1 FP = 75 LOC in C++

So, for 54.45 FP = 54.45 *75 = 4083.750 LOC

Which is approximately

So, by the Basic COCOMO model the Effort and Development time required for our Organic type Software is

EFFORT = a1 * (KLOC)^a2 PM

=2.4 x (40) ^1.05

=115 PM

Tdev =b1 x (EFFORT)^b2 Months

= 2.5 x (115) ^0.38

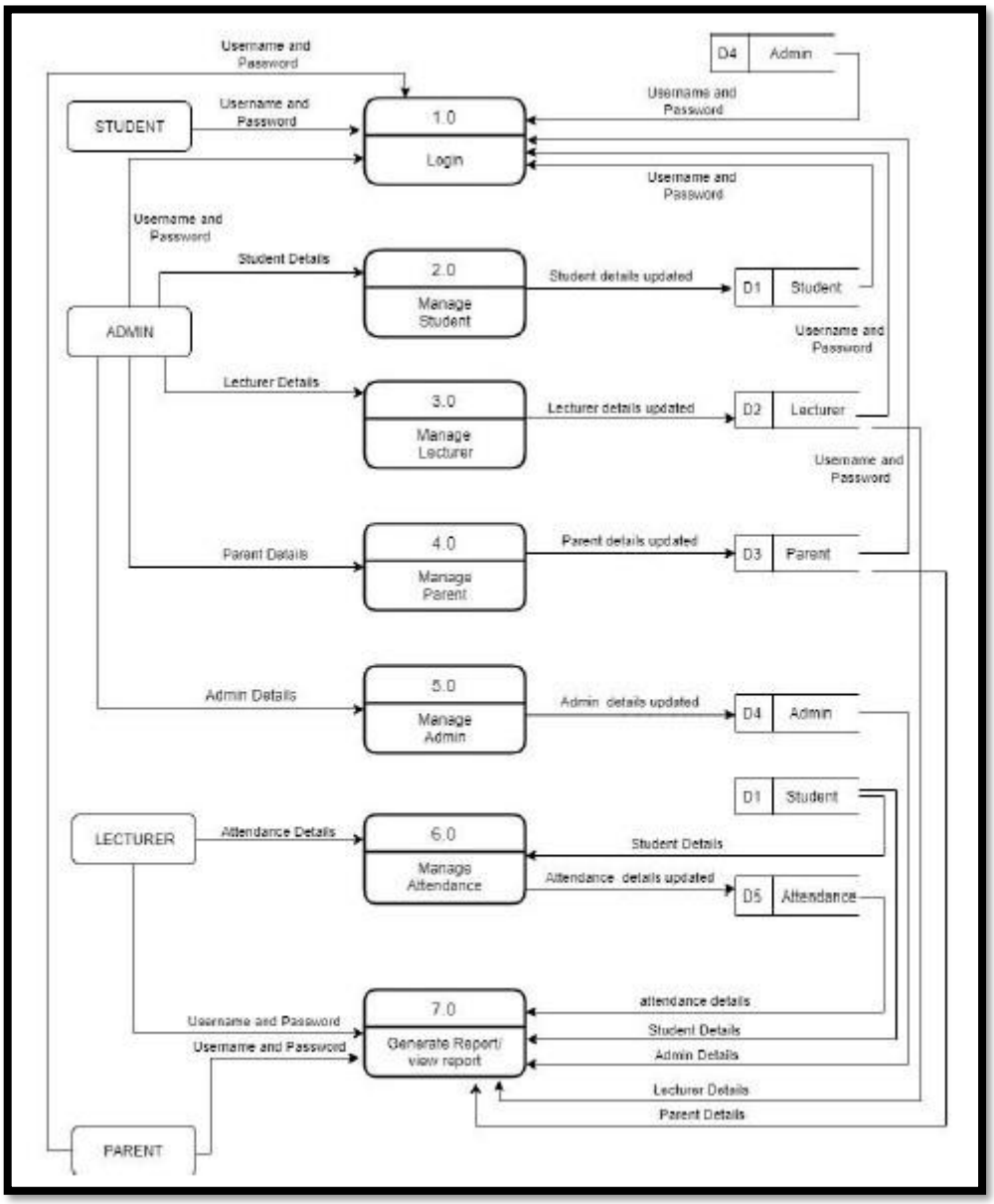
= 15 Months

If the salary of the developer is Rs. 10,000/- then the cost of development will be Rs. 1,15,000/-

Practical 6

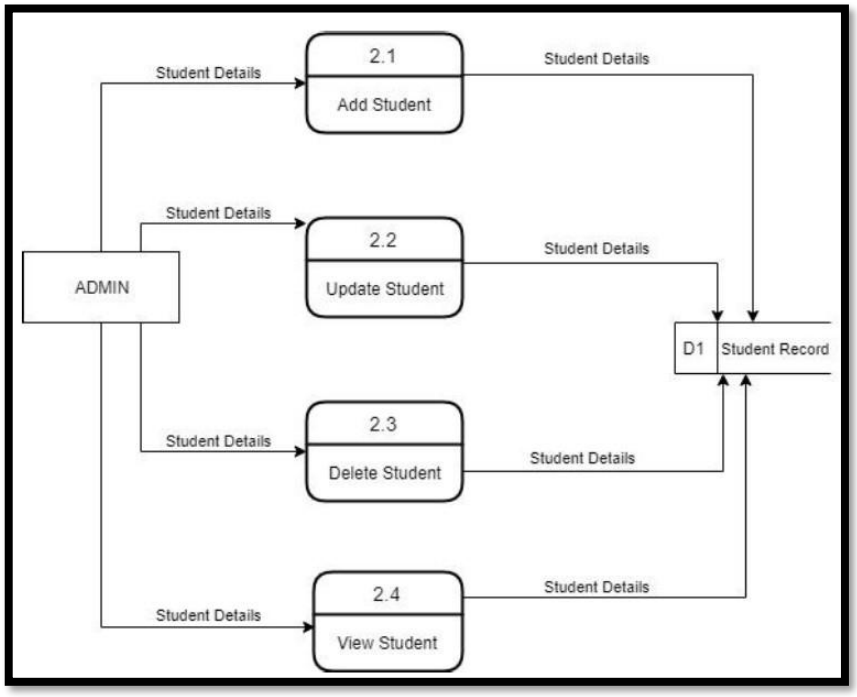
Aim: Prepare System Analysis and System Design of identified Requirement specification using structure design as DFD with data dictionary and Structure chart for the specific module

Data Flow Diagram

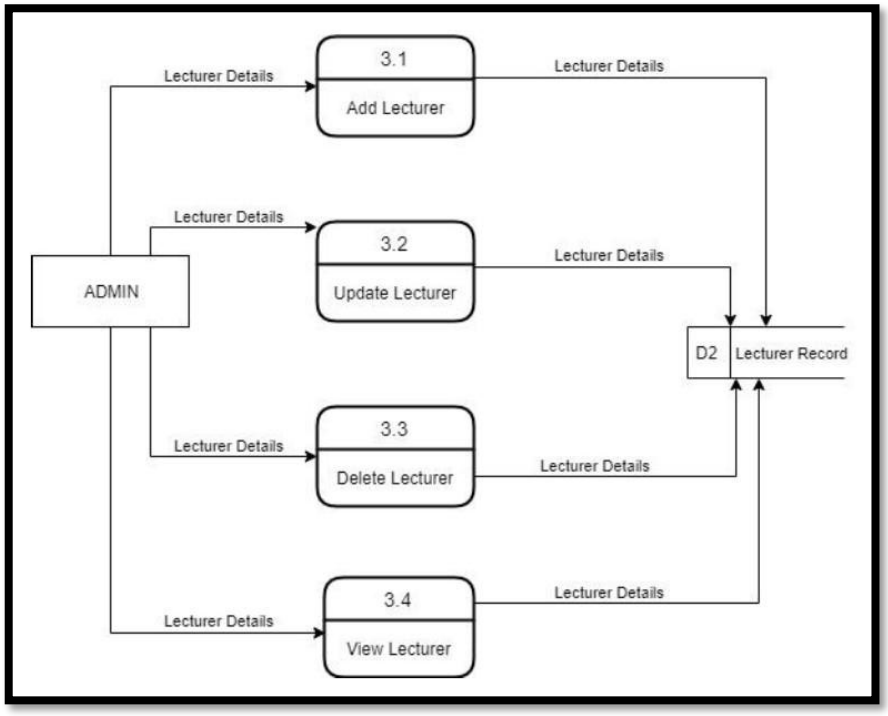


Enrollment No:

Level 1: For Manage Students

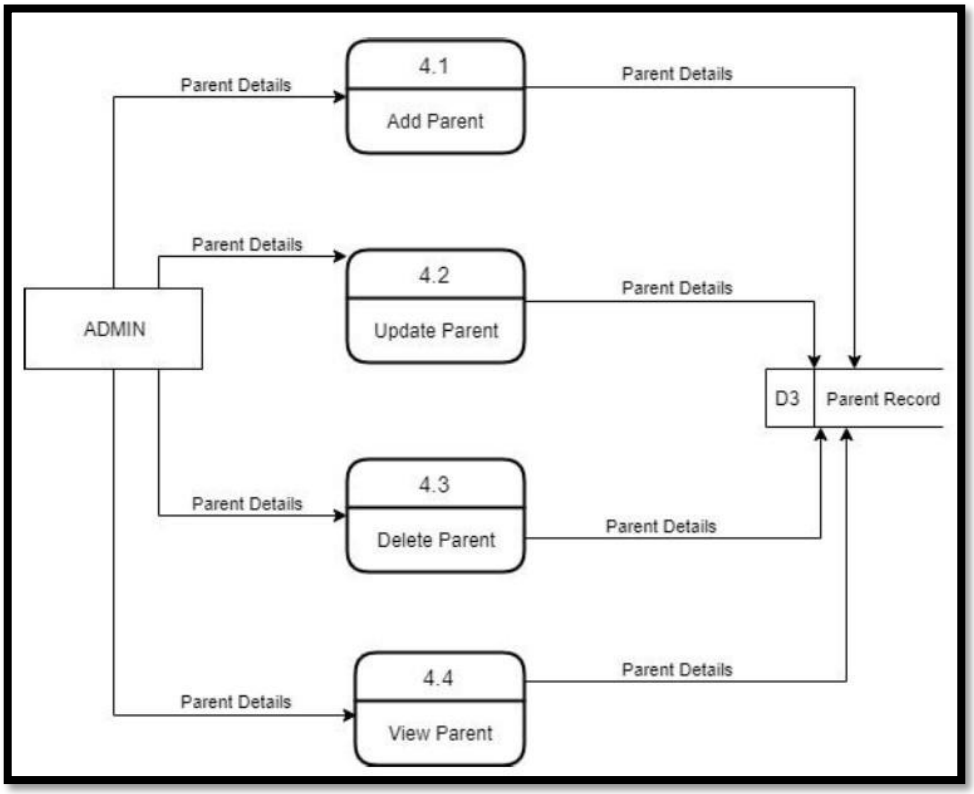


Level 1: For Manage Lecturer

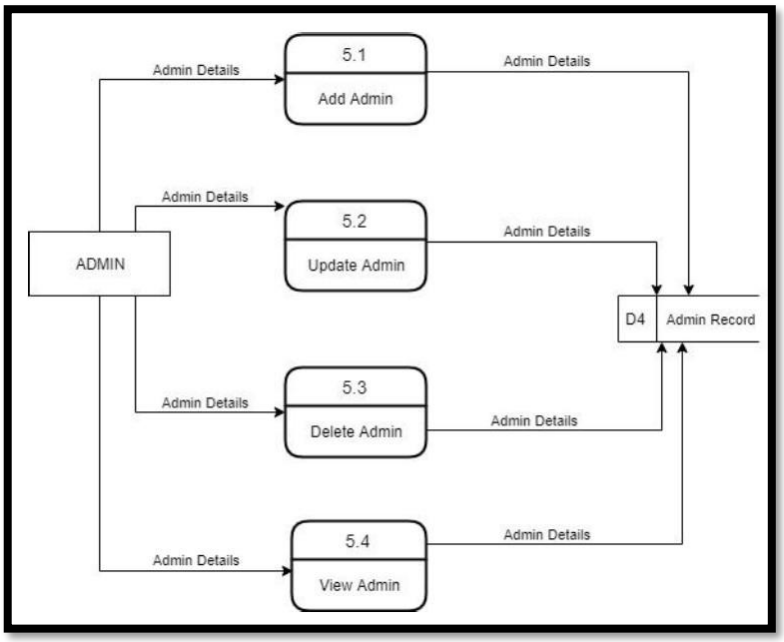


Enrollment No:

Level 1: For Manage Parent

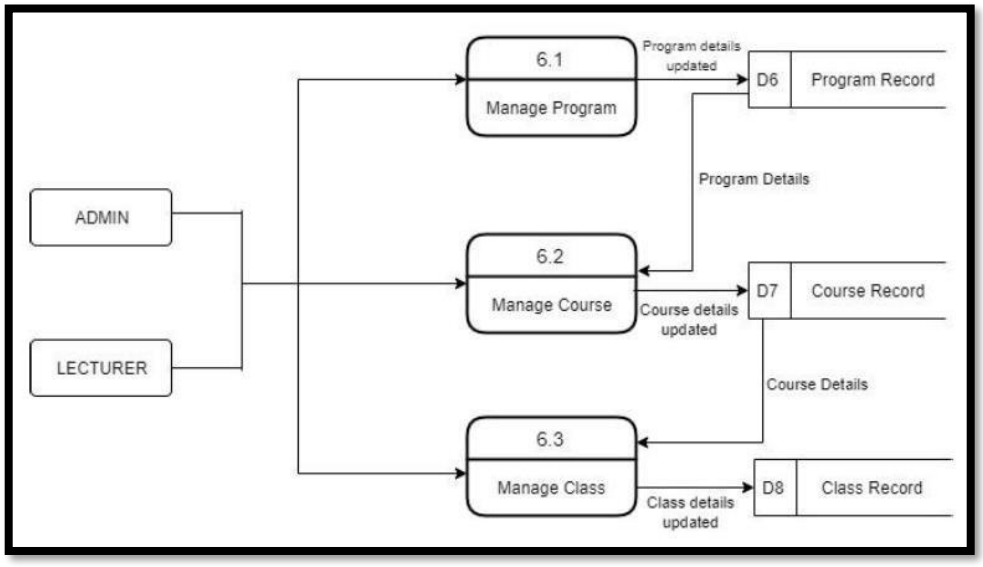


Level 1: For Manage Admin

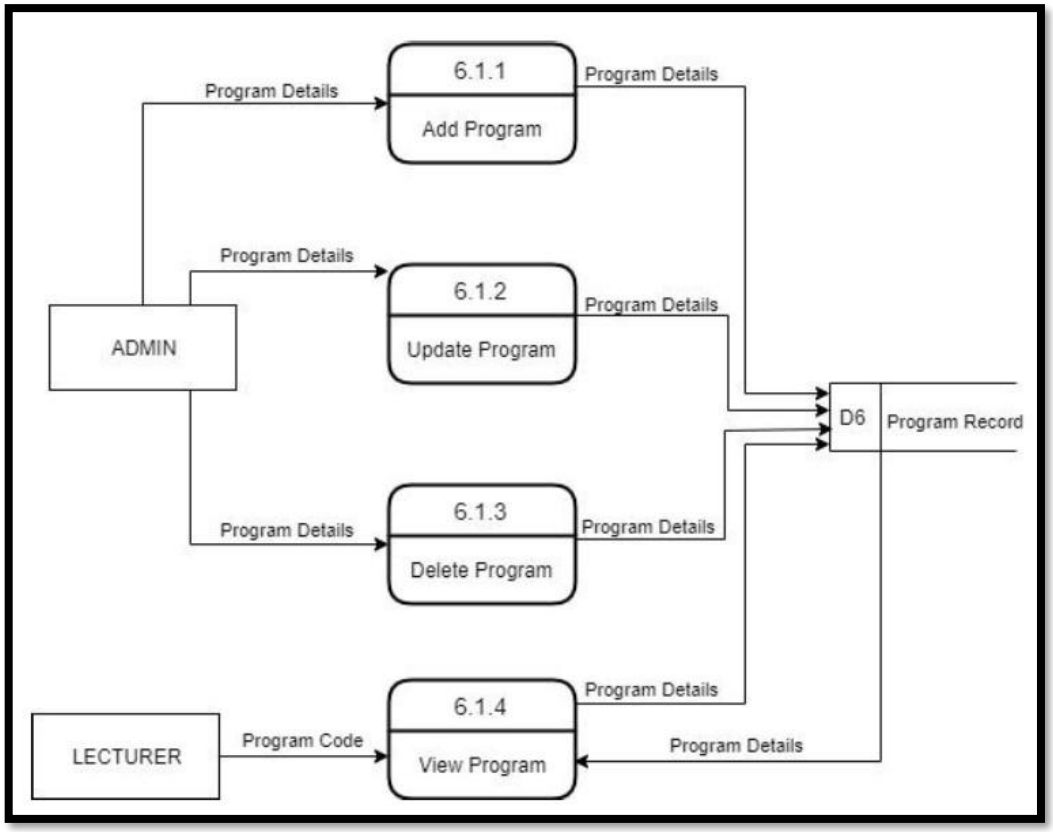


Enrollment No:

Level 1: For Manage Attendance

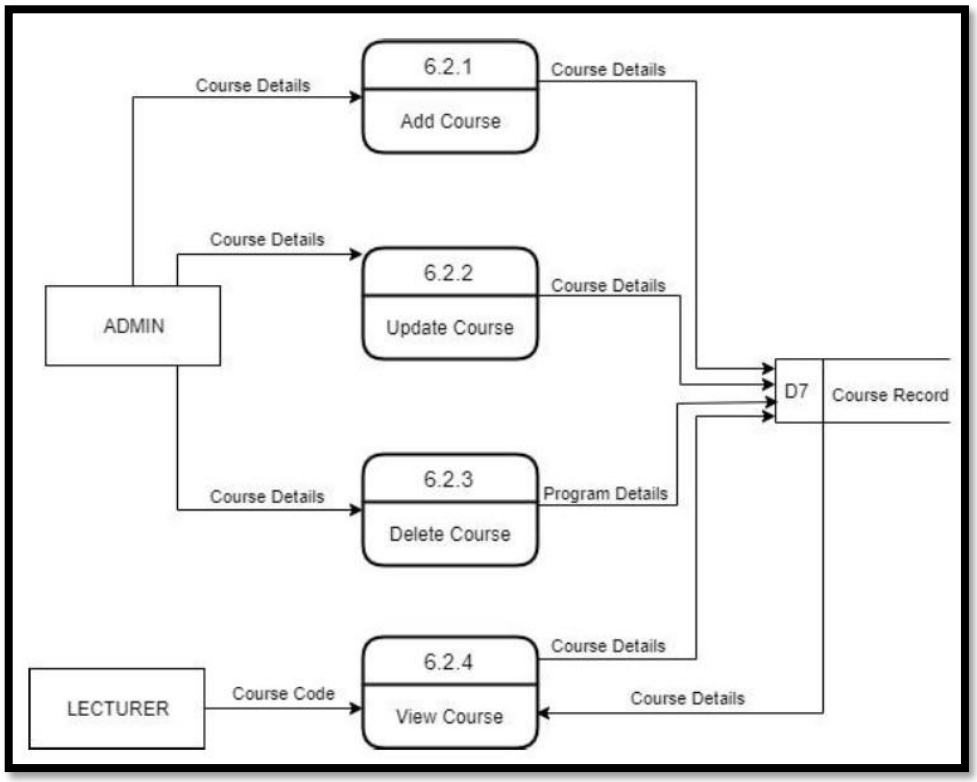


Level 2: For Manage Program

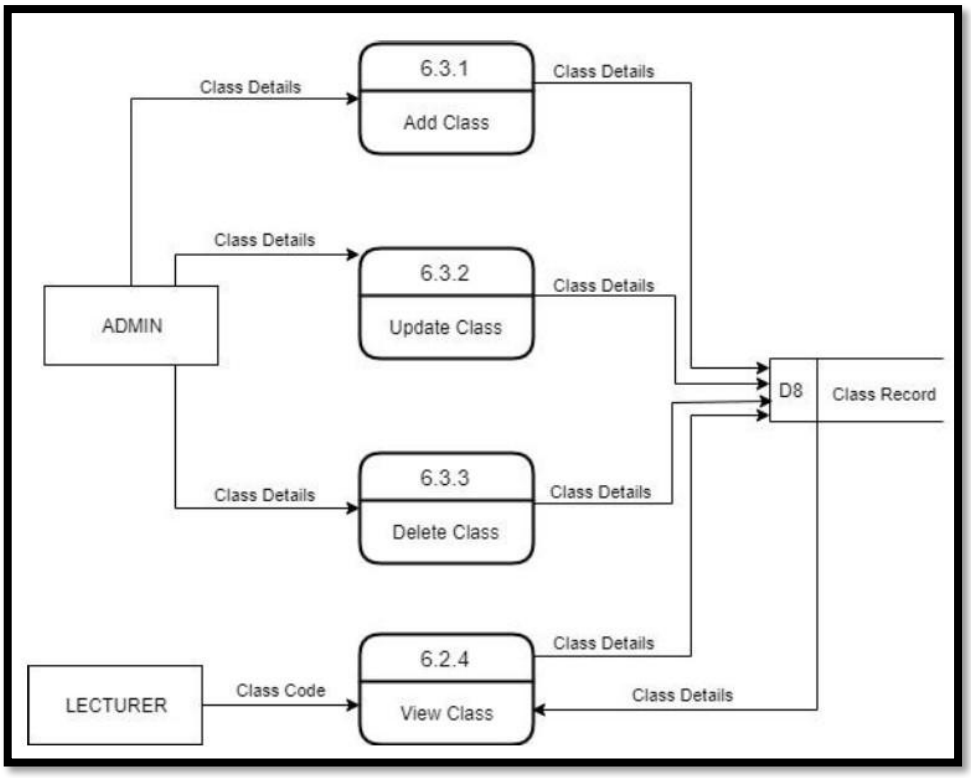


Enrollment No:

Level 2: For Manage Course



Level 2: For Manage Class

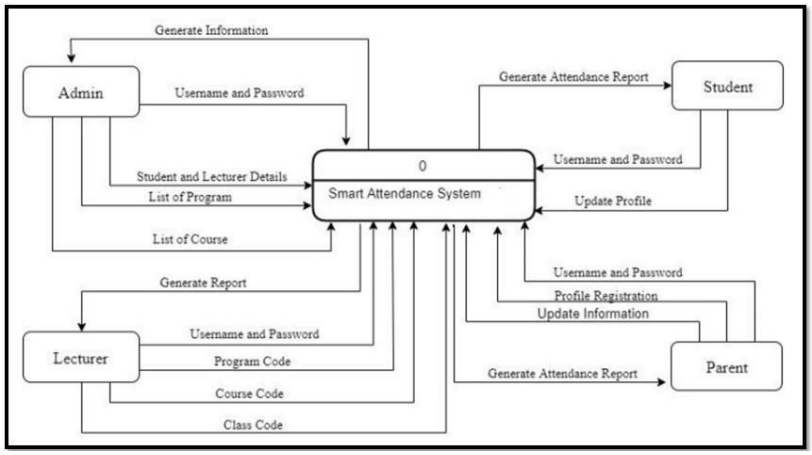


Enrollment No:

Practical 7

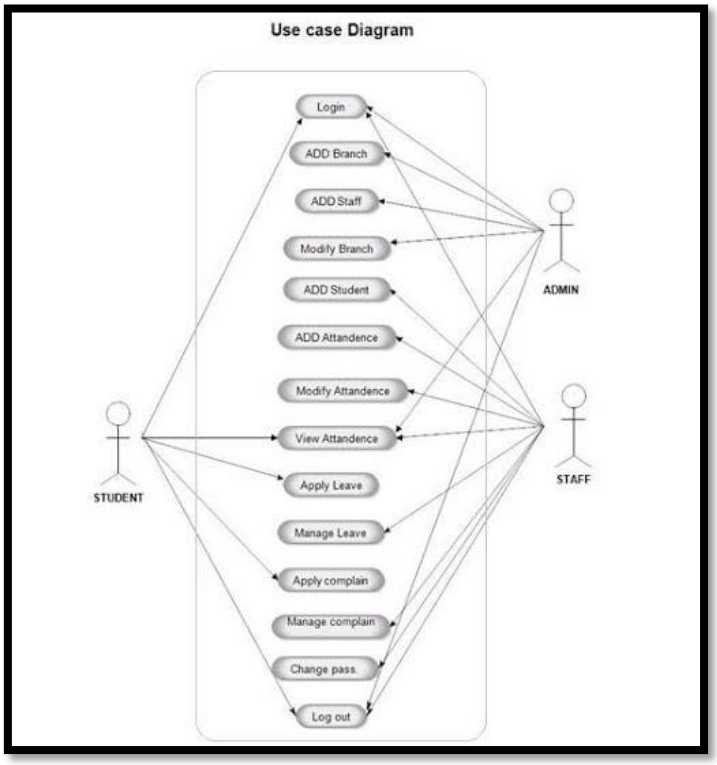
Aim: Designing the module using Object Oriented approach including Use case Diagram with scenarios, Class Diagram and State Diagram, Collaboration Diagram, Sequence Diagram and Activity Diagram

Class Diagram: A Class diagram which is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

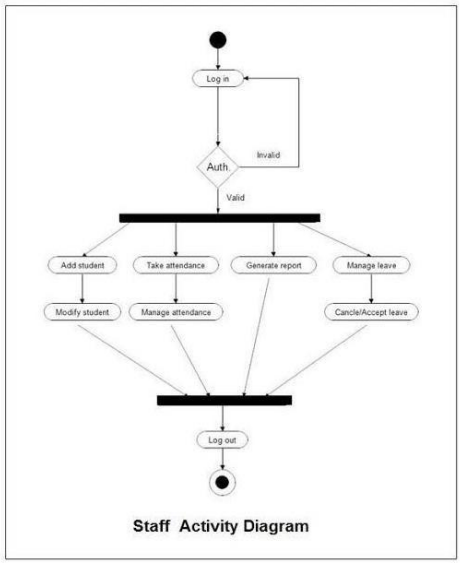


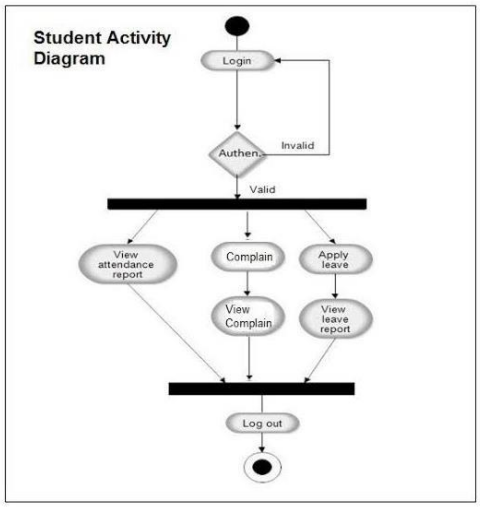
Use Case Diagram: A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses.

Enrollment No:



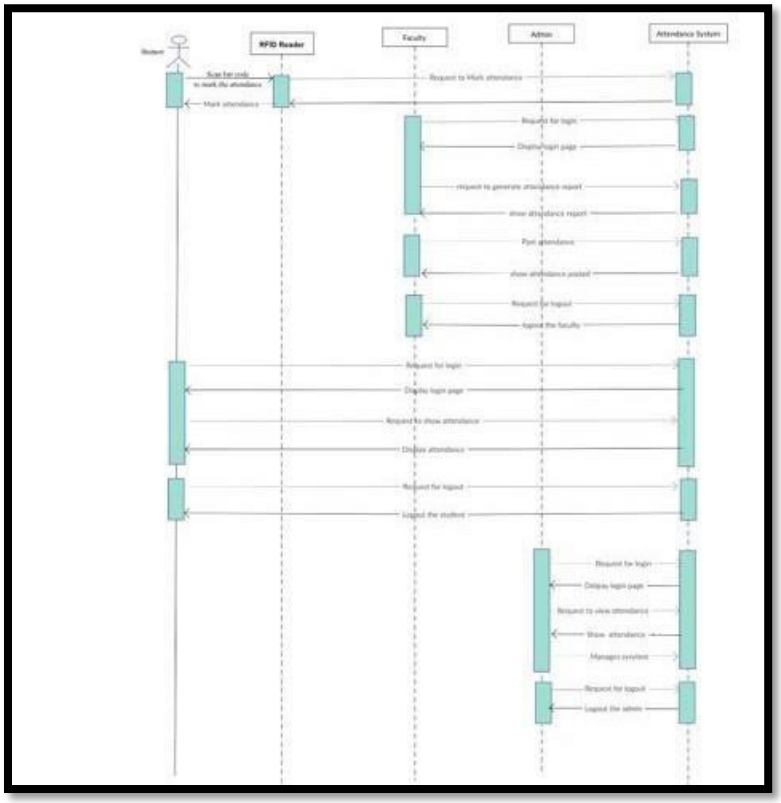
State Diagram: A state diagram is a type of diagram used in computer science and related fields to describe the behavior of systems. State diagrams require that the system described is composed of a finite number of states; sometimes, this is indeed the case, while at other times this is a reasonable abstraction.





Enrollment No:

Sequence Diagram: A sequence diagram is a Unified Modeling Language (UML) diagram that illustrates the sequence of messages between objects in an interaction. A sequence diagram shows the sequence of messages passed between objects. Sequence diagrams can also show the control structures between objects.



Practical 8

Aim: Defining Coding Standards and walk through.

Coding Standard :
Coding conventions are a set of guidelines for a specific programming language that recommend programming style, practices, and methods for each aspect of a program written in that language. These conventions usually cover file organization, indentation, comments, declarations, statements, white space, naming conventions, programming practices, programming principles, programming rules of thumb, architectural best practices, etc. These are guidelines for software structural quality. Software programmers are highly recommended to follow these guidelines to help improve the readability of their source code and make software maintenance easier. Coding conventions are only applicable to the human maintainers and peer reviewers of a software project. Conventions may be formalized in a documented set of rules that an entire team or company follows, or may be as informal as the habitual coding practices of an individual. Coding conventions are not enforced by compilers.

Why Are Coding Standards Important?

The reason why coding standards are important is that they help to ensure safety, security, and reliability Every development team should use one. Even the most experienced developer could introduce a coding defect — without realizing it. And that one defect could lead to a minor glitch. Or worse, a serious security breach. There are four key benefits of using coding standards:

1. Compliance with industry standards (e.g., ISO).
2. Consistent code quality — no matter who writes the code.
3. Software security from the start.
4. Reduced development costs and accelerated time to market.

Drawbacks of not having proper coding conventions for a team

Without predefined conditions that all team member should follow can result in the following:

- Reduced engineers motivation
- Increased development time
- Complex codebase structure

Enrollment No:

Common Aspects of Coding Standard :

- **Naming Conventions:** The naming convention is how your packages, classes methods, variables, etc. should be named. (eg. camelCase, PascalCase or snake_case)
- **File and folder Naming and Organization:** This is how your file and folder should be named and structured
- **Formatting and Indentation:** the code should be written in a standardized format and indentation
- **Commenting and Documenting:** This makes it easy for the reviewer of your code to better understand codes methods and declarations
- **Classes and Functions:** This specifies how classes and functions should behave.
- **Testing:** This specifies which approach and tools should be used to test the codes
- **Limited use of globals :** These rules tell about which types of data that can be declared global and the data that can't be.
- **Standard headers for different modules:** For better understanding and

maintenance of the code, the header of different modules should follow some standard format and information. The header format must contain below things that is being used in various companies like Name of module , Date of Module creation , Author of the Module , Modification History , Global variables accessed or modified by the module

- **Naming conventions for local variables, global variables, constants and functions**: Meaningful and understandable variables

name helps anyone to understand the reason of using it. Local variables should be named using camel case lettering starting with small letter (e.g. localData) whereas Global variables names should start with a capital

letter (e.g. GlobalData). Constant names should be formed using capital letters only (e.g.

CONSDATA). It is better to avoid the use of digits in

variable names.The names of the function should be written in camel case starting with small letters

The name of the function must describe the reason of using the function clearly and briefly.

- **Error return values and exception handling conventions:**

All functions that encountering an error condition should either return a 0 or 1 for simplifying the debugging. All braces should start from a new line and the code following the end of braces also start from a new line**.**

- **Avoid using a coding style that is too difficult to understand:**

Code should be easily understandable. The complex code makes maintenance and debugging difficult and expensive.

Google Code of Conduct

1. Serve Our Users
    1. Integrity
    2. Usefulness
    3. Privacy and Freedom of Expression
    4. Responsiveness
    5. Take Action
2. Respect Each Other
    1. Equal Opportunity Employment
    2. Positive Environment
    3. Drugs and Alcohol
    4. Safe Workplace
    5. Dog Policy
3. Avoid Conflicts of Interest
    1. Personal Investments
    2. Outside Employment and Inventions
    3. Outside Board Memberships
    4. Business Opportunities
    5. Friends and Relatives; Co-Worker Relationships
    6. Gifts, Entertainment and Payments
    7. Reporting
4. Preserve Confidentiality
    1. Confidential Information
    2. Google Partners
    3. Competitors; Former Employers
    4. Outside Communications and Research
5. Protect Google's Assets
    1. Intellectual Property
    2. Company Equipment
    3. The Network
    4. Physical Security
    5. Use of Google's Equipment and Facilities
    6. Employee Data

6. Ensure Financial Integrity and Responsibility
    1. Spending Google's Money
    2. Signing a Contract
    3. Recording Transactions
    4. Reporting Financial or Accounting Irregularities
    5. Hiring Suppliers
    6. Retaining Records
7. Obey the Law
    1. Trade Controls
    2. Competition Laws
    3. Insider Trading Laws
    4. Anti-Bribery Laws "

What is Code Review?
Code review is a software quality assurance process in which software's source code is analyzed manually by a team or by using an automated code review tool. The motive is purely, to find bugs, resolve errors, and for most times, improving code quality. Reviewing the codebase makes sure that every software or new feature developed within the company is of high quality. Code review is an essential process that every software company must follow, so we researched the best practices for reviewing code.

Best Code Review Technique

Every company has its own process of performing code review but we look into four of the best and most code review techniques.

1. Instant Code Reviewing Technique

The most direct form of reviewing code is the Instant Code review technique. In this, the developer is writing code while the reviewer sits beside reading the code simultaneously and correcting it on the go. Also known as pair programming, this process is best suited for highly complex programs where two minds can solve the problem much quicker and efficiently.

While this process looks favorable for companies but in reality, the time and workforce needed by this technique make it unfavorable. Two or more people working on code together means less average lines per developer. Interruption for corrections also halts the flow of work for the author of the code and the learning curve for a developer hinders if constant support or solution is presented right away by a reviewer for a complex problem.

2. Ad-hoc (synchronous) Code Reviewing Technique

Also known as "Over the Shoulder" code-review process. It is the most commonly used process with around 75% of companies participating in ad-hoc reviews. In this type of synchronous method, the coder produces the code

and then asks the reviewer to review the code. The reviewer joins the coder at the screen, reviews the code while discussing it, over the shoulder. It is implemented wisely because it is informal and spontaneous. The process is successful only if the reviewer is available at the time or it disrupts the coder's speed.

Three team members checking the code quality.

This method has a high probability of missing errors and glitches as most of the time, the reviewer lacks the knowledge of the goal of the task. Immediate review missed to bring out better results as a team would have in their refinement sessions together with tasks discussed upfront

The ad-hoc review usually results in only a developer knowing the goal of the project. The major problem of this process is forced context-switching. Imagine working on a complex software yourself, and then being called by your junior member for an ad-hoc review. You would have to leave your station immediately to review the code of your co-worker. Having to leave your work suddenly can create exhaustion and frustration.

3. Meeting Based Code Reviewing Technique

This is the least commonly used process with only 44% using it once a month. In meeting based code review, coders complete their work, and a meeting is called. The whole tech team sits, commenting, and attempting to improve the code together. It is a temporary process as it is highly unlikely to perform constantly considering the amount of time, loss of workforce for the time, decreased efficiency and inability to get the whole team together.

A team performing meeting based code review process

Meeting based code reviews make sense only when the whole team is inexperienced with the code review process. It involves assembling the entire team in a room, sharing ideas and solving problems for a few times. This helps every team member to understand the process much clearer. With just over half of the companies using this, this process is not adequate as a code quality assurance standard.

4. Tool Based Code Reviewing Technique

This process is not done by a team together, at least not on the same screen. It is also called an asynchronous code review. In this, once the code gets finished, the coder makes it available for others to review. The reviewer will review the code on their screen commenting, or even amending the errors in the codes. Then notifying the coder who on her agenda will improve it. When there are no changes, the code is marked with no comments for improvements and the software gets approved.

Practical 9

Aim: Write the test cases for the identified module.

The testing process cannot take place without prior communication with the programmers of the software. This is because the testers cannot start testing the software unless they know what the code does and how it works. Similarly, communicating with other software testers is also important. It helps to understand:

1. What to test?
2. What resources will be needed?
3. What will the schedule be?

The software test plan is the principal way through which software testers communicate their intent to the code developers. The test case is the heart of a test plan. A test case is a document that describes a set of data inputs and operating conditions required to run a test, together with the expected results of the run.

The tester should run the software according to the test case documentation and then compare the actual results with the expected results noted in the documents.

If the obtained results are in complete agreement with the expected results, then it indicates that no error has been identified. The result of such a test case is said to be ‘pass’.

A potential error is said to be identified when some or all of the results do not agree with the expected results. In such cases, the test case will have the status ‘fail’

Test Cases	Explanation	Result
LOGIN	To check if only admin and teacher and student will be able to login after filling in the correct details in the available fields. Check if it was successful	Pass
PREVIEW	To check if the admin and doctor will be able to visit their respective dashboard	Pass

CHECK ATTENDANCE	To check if the admin and teacher will be able to check the attendance system. Check if it was successful.	Pass
ADD MEMBERS	To check if the admin can add students of different classes and teachers of different classes. Check if it was successful.	Pass
CHECK EVENTS	To check if the admin, teacher or student can check event. If it was successful	Pass

Test Case ID	Test case Objective	Test Case Description	Input	Expected Output	Results
1	To check the interface link between the login page and the home page	Click on the LOGIN Button	Button Clicked	To be directed to the Home page	Pass
			Button not Clicked	Remains Unchanged	
2	To check the interface link between the home page and the dashboard page	Click on the Dashboard Button	Button Clicked	To be directed to Dashboard	Pass
			Button not Clicked	Remains Unchanged	
3	To check the interface link between the Dashboard Page and Attendance Page	Click on the Attendance Button	Button Clicked	To be directed to Attendance Page	Pass
			Button not Clicked	Remains Unchanged	
4	To check the interface link between the Attendance page and View/Add attendance	Click on View/Add	Button Clicked	To be directed to View/Add Attendance Window	Pass
			Button not Clicked	Remains Unchanged	

Practical 10

Aim: Demonstrate the use of different Testing Tools with comparison

Selenium IDE :

Selenium IDE (Integrated Development Environment) is the simplest tool in the Selenium Suite. It is a Firefox add-on that creates tests very quickly through its record-and-playback functionality. This feature is similar to that of QTP. It is effortless to install and easy to learn.

Because of its simplicity, Selenium IDE should only be used as a prototyping tool, not an overall solution for developing and maintaining complex test suites.

Selenium IDE: Why it is Good?

- Large community.
- Simultaneous tests.
- Mobile support.

Selenium IDE: Where's the catch?

- Not beginner-friendly.
- No image verification.

TestComplete: Why it is Good?

- Ease of use.
- Customization.
- Timely updates.
- Support of desktop apps.

TestComplete: Where's the catch?

- No Mac support.

Katalon Studio: Why it is Good?

- Good for both pros and non-techies.
- Unified bundle.
- Abundance of tutorials.

- Visualized reports.

Katalon Studio: Where's the catch?

- Poor language support.
- Small (although growing) community.
- Unified Functional Testing(UFT): Why it is Good?
- Automated tests from manual.
- Collaboration capabilities.

Unified Functional Testing(UFT): Where's the catch?

- Only one language supported.
- Price.
- Only Windows support.

Watir: Why it is Good?

- Choice of languages.

Watir: Where's the catch?

- May be too simple.
- Small community.

Comparison between different software testing tools

COMPARING QA AUTOMATION TOOLS

	Price	Platform	Supported languages	Tested apps	Coding skills required	Learning curve
Selenium	Free	Windows/ Mac/Linux	Java, Python, C#, PHP, JavaScript, Ruby, Perl	Web, mobile (with Appium)	Advanced skills	Steep
TestComplete	$4600/$9000	Windows	VB, JavaScript, Jscript, C++, C#, Delphi, Angular, Ruby on Rails, PHP	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Mild
Tricentis Tosca	Custom, high (according to online discussions)	Windows	JavaScript	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Mild
Katalon Studio	Free	Windows/Mac	Java/Groovy	Web, mobile	Minimum skills/Advanced skills for pro scripting	Mild
UFT	$2500/$3500	Windows	VBScript	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Moderate
Watir	Free	Windows	Ruby (Java/.Net alternatively)	Web	Minimum skills	Mild
Ranorex	$2800/$850	Windows	C#, VB.Net, Iron Python	Web, mobile, desktop	Minimum skills/Advanced skills for pro scripting	Moderate

altexsoft
software r&d engineering

Installation of Selenium:

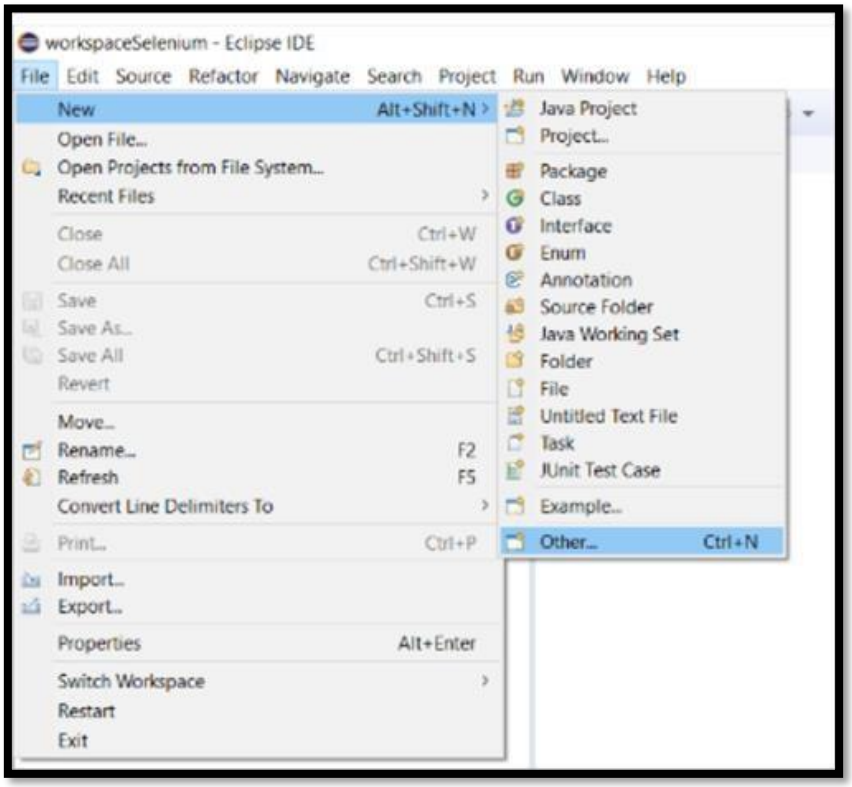
1) Download the setup from the www.selenium.org
2) Check the version of the chrome.
   Open Chrome Browser -> Help -> About Google Chrome
3) Open Chromedriver.exe downloads where you will see the latest ChromeDriver for the latest google chrome version. We will download version – 75 of chromedriver.exe
4) Download the chromedriver.exe file for the respective OS and copy that .exe file into your local. Copied the downloaded .exe
5) The path of the chromedriver (C:\webdriver\chromedriver.exe) will be used in our program.

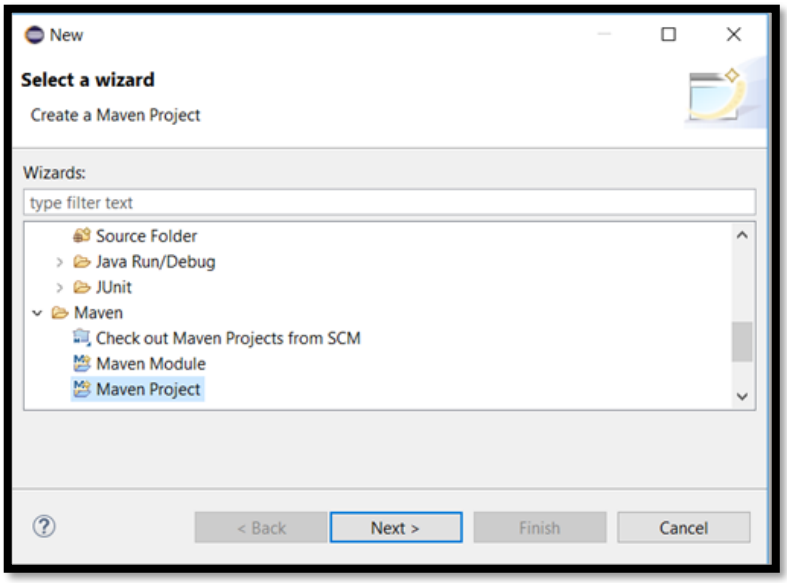
Selenium Setup With ChromeDriver
Now that we are done with setting up of ChromeDriver, we will launch the Eclipse software for executing our Selenium codes.

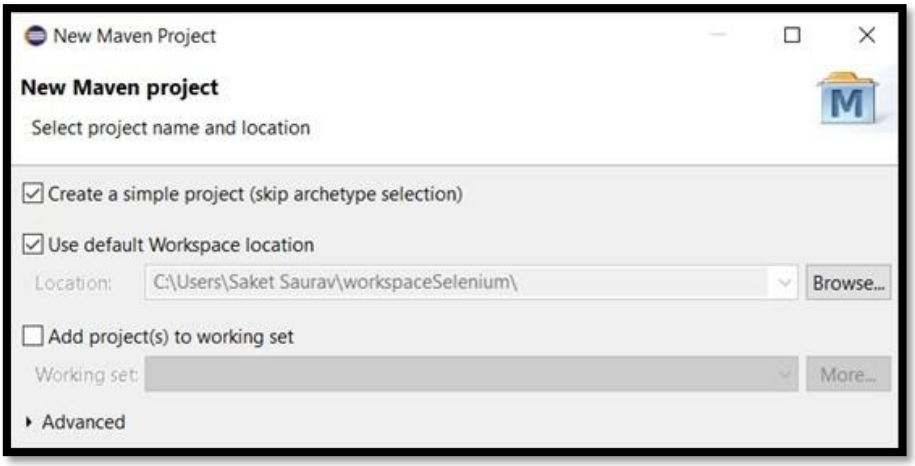
Below are the steps to follow to create and execute our Selenium codes on Eclipse.

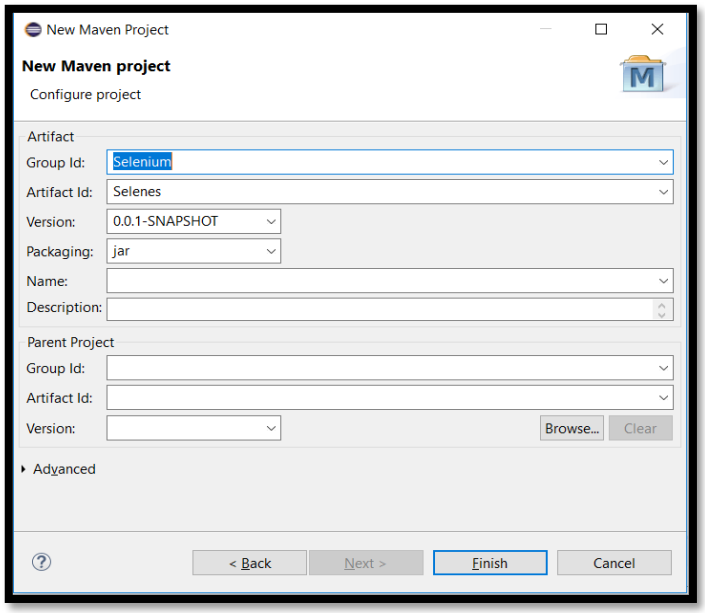
Create A New Maven Project
This step will let you create an empty Maven project in which you can execute your Selenium codes.

All you need to do is to click on File -> New -> Others -> Maven Project.

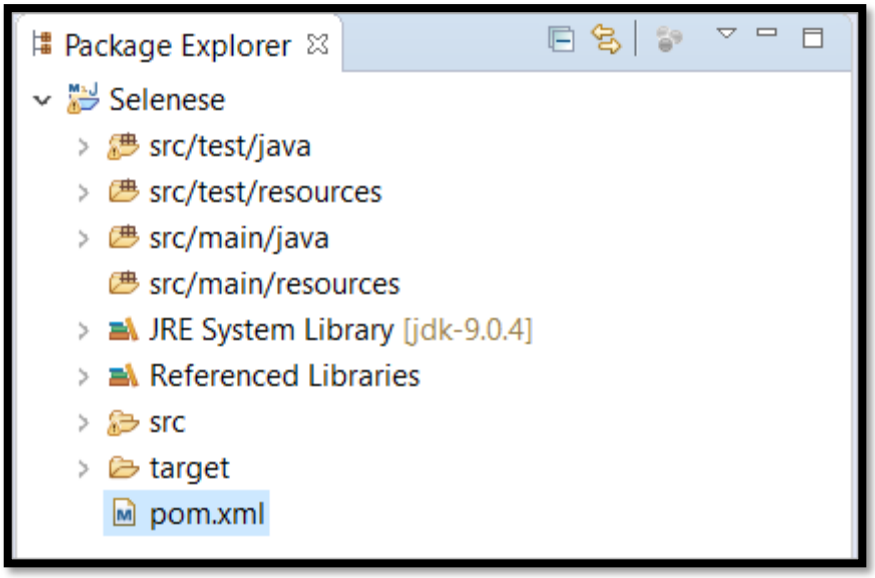

workspaceSelenium - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
New Alt+Shift+N
Open File...
Open Projects from File System...
Recent Files
Close Ctrl+W
Close All Ctrl+Shift+W
Save Ctrl+S
Save As...
Save All Ctrl+Shift+S
Revert
Move...
Rename... F2
Refresh F5
Convert Line Delimiters To
Print... Ctrl+P
Import...
Export...
Properties Alt+Enter
Switch Workspace
Restart
Exit
Java Project
Project...
Package
Class
Interface
Enum
Annotation
Source Folder
Java Working Set
Folder
File
Untitled Text File
Task
JUnit Test Case
Example...
Other... Ctrl+N


New
Select a wizard
Create a Maven Project
Wizards:
type filter text
Source Folder
Java Run/Debug
JUnit
Maven
Check out Maven Projects from SCM
Maven Module
Maven Project
< Back
Next >
Finish
Cancel

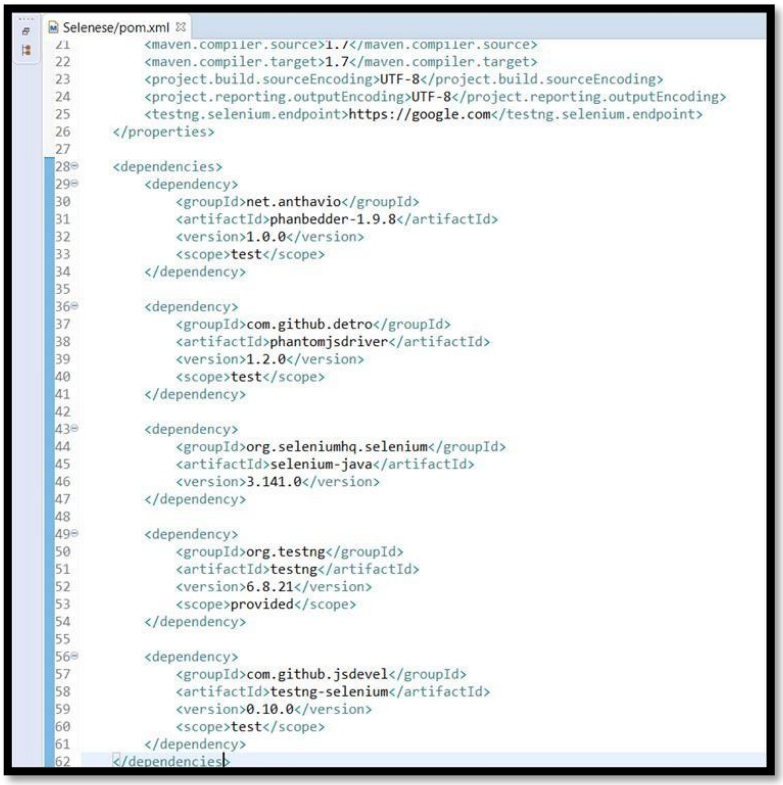




In the above img, we have added the group id and artifact id. The same will be reflected or required in your pom.xml after you have clicked on the finish button.



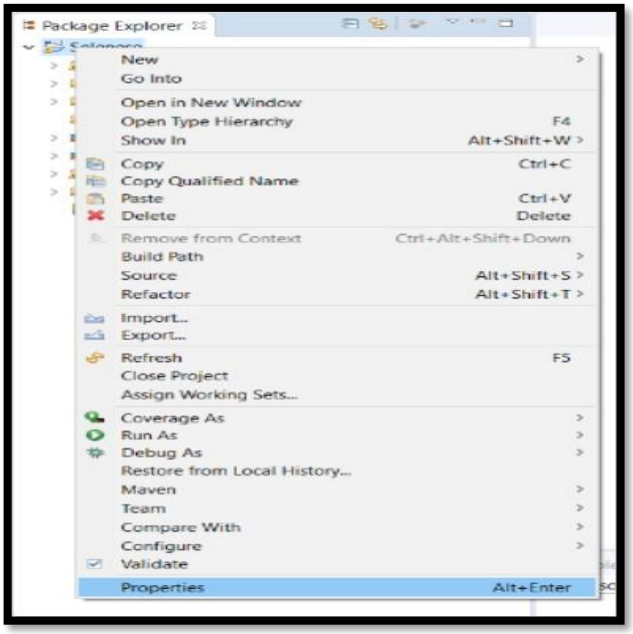
Pom.xml is a file that contains the dependencies. Here we can add as many dependencies as we like. The dependencies could be Selenium, GitHub, TestNG and so on.



Project BuildPath And Importing Jars
The next step is to download the jar files and import them in your project. You can download all the selenium jars from the google or the official maven site

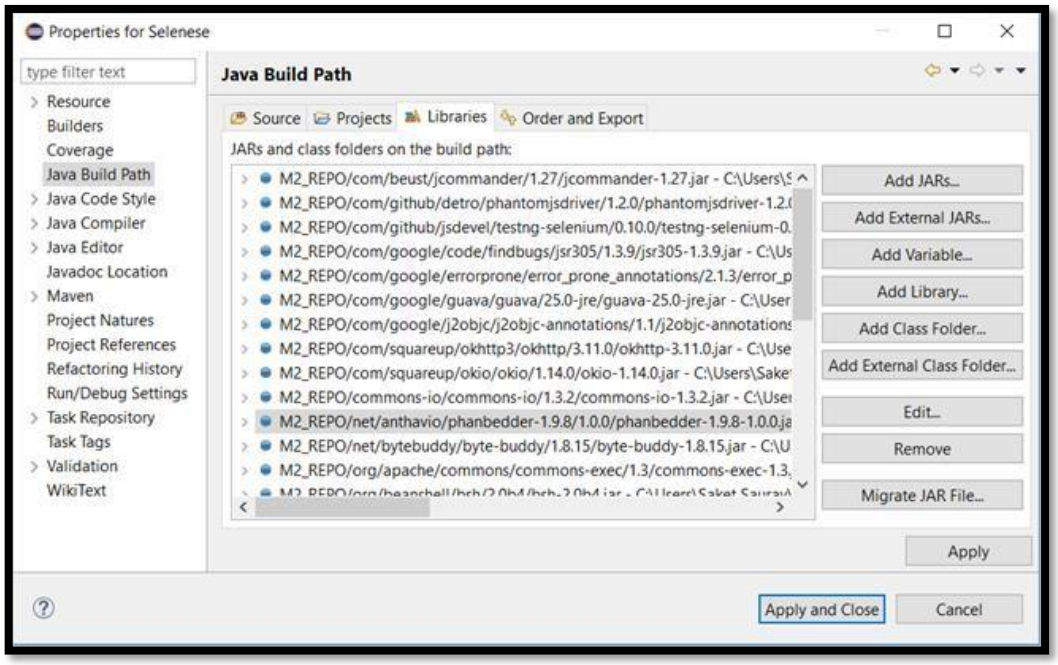
After you have downloaded all the jars, you need to follow the below steps in order.

- Right-click on your Maven Project and click on Properties.



- Click on Java Build Path -> Libraries -> Add Jars -> Apply and Close.

Adding Jars in Java Build Path



Example: Let's take the example of Facebook. Whenever you try automating www.facebook.com using Chrome, you will see the following alert.

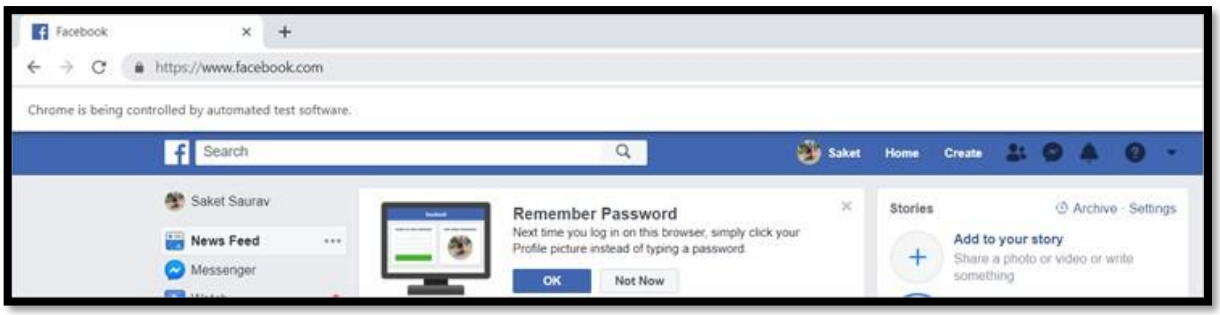
HandlingAlerts.java

```
1  package tests;
2
3  import org.openqa.selenium.By;
4  import org.openqa.selenium.WebDriver;
5  import org.openqa.selenium.WebElement;
6  import org.openqa.selenium.chrome.ChromeDriver;
7
8  public class HandlingAlerts {
9
10     public static void main(String[] args) {
11         // TODO Auto-generated method stub
12         System.setProperty("webdriver.chrome.driver", "C:\\webdriver\\chromedriver.exe");
13         WebDriver driver = new ChromeDriver();
14         driver.get("https://www.facebook.com");
15         WebElement element = driver.findElement(By.xpath("//*[@id = 'email']"));
16         element.sendKeys("Your Email Id or Phone Number");
17
18         WebElement element2 = driver.findElement(By.xpath("//*[@id = 'pass']"));
19         element2.sendKeys("Your Password");
20
21         element2.submit();
22     }
23
24 }
25
```

In the above script, we have passed our ChromeDriver path as an argument in the system.setProperty(). This will let the WebDriver to control Google Chrome.

Upon executing the above script, we will be logged into Facebook using the email id and password. However, an alert will pop up which would further deny any operation that we will do on the website through our script.

Final output:



Practical 11

AIM: Write test cases for Current semester project

Description of Main Menu

- ADMIN: It should contain various options performed by administrator only.
- SCHEME: It should provide different scheme.
- CONTROLS: It should be used to control client terminals.
- EXIT: It will exit from software.
- HELP: It will show a help page.

Graphical user interface:

1. Each control button should show tool tip associated with it.
2. Name of the window.
3. Text type face, size and format used through out the system were checked for uniformity.
4. Grouping information is displayed appropriately.
5. Tab order should be maintained.
6. Alignment of each text box and equal spaces between the controls.
7. Each child form should fit into the MDI form.

Functional Test cases:

1. All menu functions and sub functions were verified for correctness.
2. A validation for all inputs was done.
3. Each menu function was tested, whether it invokes the corresponding functionality properly.
4. Reset button will reset all setting.
5. Clear button will clear all values of control.
6. Check the number of users and authentication among the privileged user should be their.
7. Registration should not allow special characters or digits in the name.
8. Browsing time should not exceed the left time in particular user account.
9. According to the use user left time should be decremented.
10. Status should be cleared as and when user logs in/out.
11. Controls of client terminal like shut down and restarts should be checked.