**TABLE OF CONTENT**

1. Introduction

1.1 Project Details: Broad specifications of the work entrusted to you

1.2 Purpose

1.3 Scope

1.4 Objective

1.5 Technology and Literature Review

2. Project Management

2.1 Feasibility Study

2.1.1 Technical feasibility

2.1.2 Time Schedule feasibility

2.1.3 Operational feasibility

2.1.4 Implementation feasibility

2.2 Project Planning

2.2.1 Project development Approach and Justification

2.2.2 Project Plan

2.2.3 Role and Responsibilities

3. System Requirements Study

3.1 Study of Current System

3.2 Problems and Weakness of Current System

3.3 User Characteristics

3.4 Hardware and Software Requirements

3.5 Constraints

4. System Analysis

4.1 Requirements of New System (SRS)

4.1.1 Functional Requirements

4.1.2 Non-functional requirements

5. System Design

5.1 Use Case Diagram

5.2 Class Diagram

5.3 Sequence Diagram

5.4 Activity Diagram

6. Implementation Planning

6.1 Implementation Environment

6.2 Program Specification

7. Testing

7.1 Testing Plan

7.2 Testing Strategy

7.3 Testing Methods

7.4 Test Cases

8. User Manual

9. Limitation and Future Enhancement

10. Conclusion and Discussion

10.1 Conclusions

10.2 Discussion

10.2.1 Self Analysis of Project Viabilities

10.2.2 Problem Encountered and Possible Solutions

10.2.3 Summary of Project work

11. Bibliography

1.Introduction

1.1 Project Details

The broad specification of work for a full-stack development project typically involves various tasks and responsibilities across the front-end and back-end development. Here's an outline of the work that may be entrusted for this project :

1. Project Planning and Requirement Analysis.
2. Develop responsive and user-friendly web pages using HTML, CSS, and JavaScript
3. Implement interactivity and dynamic content using front-end frameworks like React js.
4. Choose and implement an appropriate back-end technology stack (e.g., Node.js, Django, Flask, Ruby on Rails, etc.).
5. Design and implement the database schema based on the application's data model
6. Implement authentication and authorization mechanisms.
7. Conduct unit testing for both front-end and back-end components.
8. Optimize code and infrastructure for performance

1.2 Purpose

The main purpose of making this web application to have benefits of more than one application in one web application. Teachers have to use google classroom for uploading materials, for sending notice they use gmail and for uploading marks attendance use personal college websites thus we bring one application to get reed of using 4 application and use this one website to fulfil their needs efficiently.

1.3 Scope

The scope of website is broad and continually evolving as technology advances and user needs change.

This web application is educational website which encompasses various features which would be beneficial to not only student but faculties as well.

As the workload increases people find a way which can ease their word and would not be time consuming so this will bi helpful in full filling those needs as well.

1.4 Objective

The objective is to make life easier. To disseminate the educational content and resources. To make relationship of teacher and students more stronger and students can also connect with each other.

1.5 Technology and Literature Review

Following technologies will be used for development / management / tracking activities…

1. Visual Studio Code for developing web application.
2. Mongo Database System for providing 24\*7 services.

2.Project Management

* 1. Feasibility Study
     1. Technical Feasibility

Viewing our project from a technical point of view (thinking about various tools and technologies being used in developing the system). We have decided that the following technologies will be more than enough to develop a complete working system (including tech. & tools used for project tracking, monitoring, etc. along with development).

For web app development: VS Code

Frontend: React JS

Backend: Express JS, Node JS

Database: Mongo database

For testing: Postman

We are equipped with basic workflows of each tool and tech. and capable to explore further if required. Each of the above technologies is freely available and some of the skills are yet to be learned but it is manageable. From this, it is clear that our project is technically feasible.

* + 1. Time Schedule Feasibility

We have planned the steps for the completion of our project in the given duration. Firstly, we will perform requirement gathering & analysis by the mid of January 2023. We will prepare the SRS document and the GUI design tentatively by January 2023 ending. The diagrams required for the design as well as the database design will be tentatively completed by February 2023. For coding and unit testing, 4 weeks and for system and integration testing another 2 weeks will be required. Hence tentatively by the end of March 2023. We will be able to complete the project and ready for its demonstration at starting of April 2023. Being a 2 members team, we will be able to complete our project in the estimated time.

2.1.3. Economic feasibility

This evaluation often includes a cost/benefit analysis of the project, which assists businesses in determining the viability, cost, and advantages of a project before allocating financial resources. It also functions as an impartial project evaluation and enhances project credibility by assisting decision-makers in determining the positive economic advantages that the proposed project would give to the business.

2.1.4. Implementation feasibility

We will be working on developing a full Web application for the first time. So we need to learn the basics of the MERN stack. Also, we need to learn how to connect our project with an online database MongoDB and Learn some other things which can be useful in our project. Since we are well aware of the basics of JavaScript we just need to learn how to implement it according to our needs which will take around 2 or 3 weeks and be completed before starting implementation.

2.2 Project Planning

2.2.1 Project Development Approach and Justification

We would be using the Agile model for project development. Agile methods break tasks into smaller iterations or parts that 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.

Following are the phases in the Agile model are as follows:

1. Requirements gathering

2. Design the requirements

3. Construction/ iteration

4. Testing/ Quality assurance

5. Deployment

6. Feedback

Advantages of Agile model:

* Customer satisfaction by rapid, continuous delivery of useful software.
* People and interactions are emphasized rather than processes and tools. Customers, developers, and testers constantly interact with each other.
* Working software is delivered frequently (weeks rather than months). face-to-face conversation is the best form of communication.
* Close, daily cooperation between business people and developers.
* Continuous attention to technical excellence and good design.
* Regular adaptation to changing circumstances.
* Even late changes in requirements are welcomed.

Disadvantages of Agile model:

* In case of some software deliverables, especially large ones, it is difficult to assess the effort required at the beginning of the software development life cycle.
* There is a lack of emphasis on necessary designing and documentation.
* The project can easily get taken off track if the customer representative is not clear what final outcome that they want.
* Only senior programmers are capable of taking the kind of decisions required during the development process. Hence it has no place for newbie programmers, unless combined with experienced resources

3.System Requirement Study

3.1 Problems and Weakness of Current System

The Following Problem exits in current system

* Students as well as Teachers need to use different apps or websites for different purposes like Whatsapp for sending different message or solving doubt of students.
* Google Classroom for sending materials
* egov for marks and attendance.

3.2 User Characteristics (Type of users who are dealing with the system)

There are 2 types of users

STUDENT:

* Login
* Can use Chatbot
* View Marks and attendance
* View Material in Classroom

TEACHER:

* Login
* Can chat using Chatbot
* Upload Marks and attendance
* Upload Material in Classroom

3.3 Hardware and Software Requirements (minimum requirements to run your system)

There are no such specific hardware requirements other than basic requirements such as a computer with good internet connectivity and a decent browser that supports React & JavaScript.

Software: -

* Operating System: Windows Operating System 2000 and Above and Linux
* Visual Studio Code
* MERN stack
* Mongo database

Mongo database: MongoDB is a Cloud-hosted, NoSQL database that uses a document model. It can be horizontally scaled while letting you store and synchronize data in real time among users.

Visual Studio Code: Visual Studio Code is the Integrated Development Environment (IDE) for Web app development.

3.4 Constraints

3.4.1 Hardware Limitations

There is only one limitation of this web app, the device must have a browser.

3.4.2 Reliability Requirements

The web app does demand much reliability and it is fully assured that the particular information about the users should be secured and flow is maintained and accessed according to the rights.

3.5 Assumptions and Dependencies

1) Users have sufficient privileges to access the internet.

2) Browser on the Device is running smoothly.

3) Database updates are giving expected and accurate results.

4. System Analysis

4.1 Requirements of New System (SRS)

4.1.1 Functional Requirements

1.Login

* Input: User credentials (username and password).
* Process: Validate user credentials against the stored data. Grant access upon successful authentication.
* Output: Successful login message. Error message for unsuccessful login attempts.

2. Attendance

* Input: User identification (e.g., student ID or employee ID). Date and time stamp.
* Process: Record attendance data in a database . Generate attendance reports.
* Output: Confirmation of attendance recorded. Attendance reports for users and administrators.

3. Avatar Selection:

* Input: User's choice of avatar or uploaded image.
* Process: Store user-selected avatars.
* Output: Confirmation message for successful avatar selection. Display selected avatar in the user profile.

4. Chatbox:

* Input: Text messages from users.
* Process: Real-time message delivery. Store chat history.
* Output: Display incoming messages. Notification for new messages.

5. Material Uploading Section:

* Input: Educational materials (documents, presentations, etc.).
* Process: Upload and store materials. Categorize materials based on user-provided metadata.
* Output: Confirmation of successful material upload. Material repository with search and categorization features.

6. Grades Section

* Input: Student ID, course details, and grades.
* Process: Calculate and store grades. Generate grade reports.
* Output: Confirmation of successful grade input. Grade reports for users and administrators.

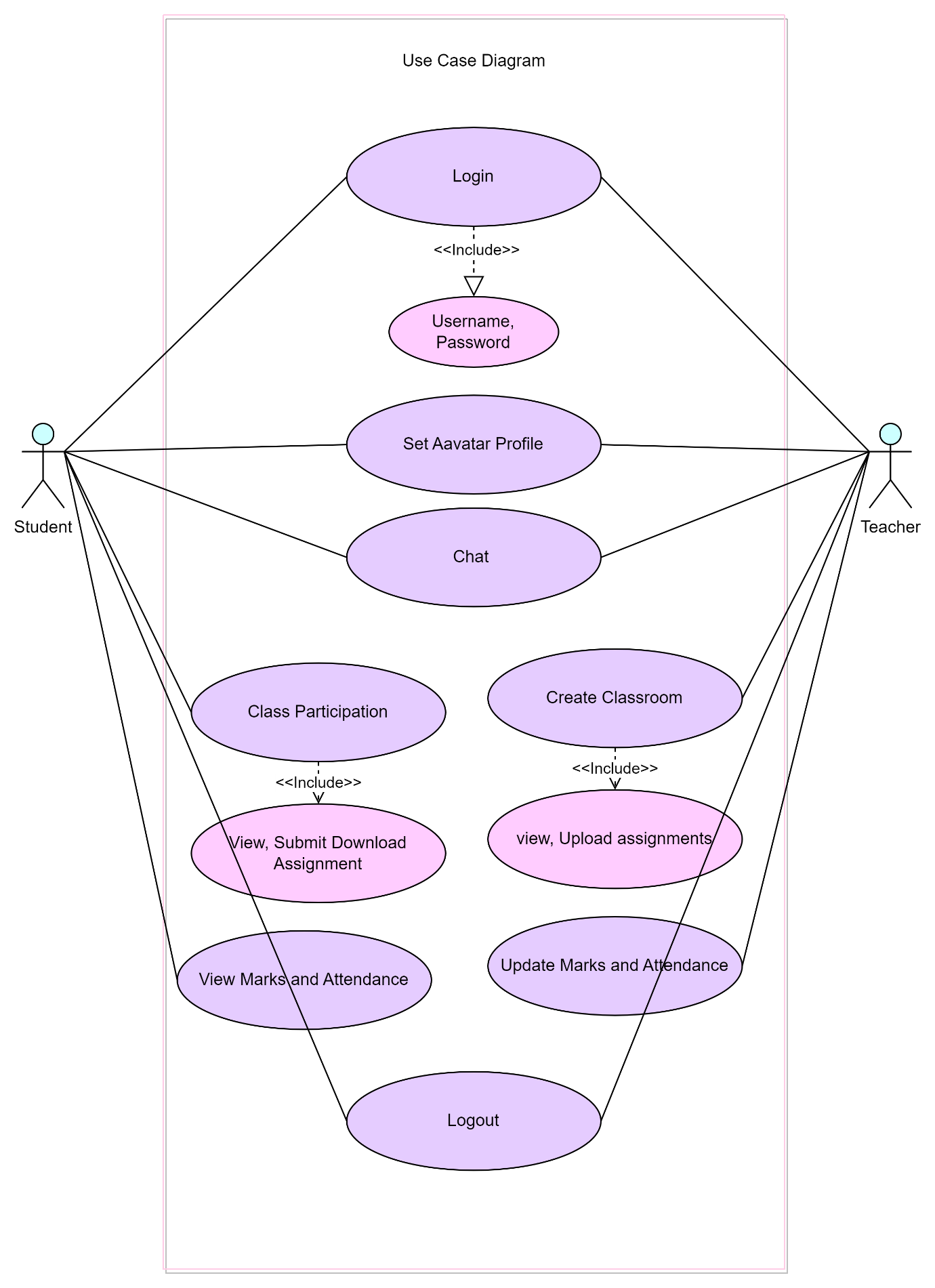
7.Search Functionality

* Input: Search queries from users.
* Process: Search through materials, user profiles, or other relevant data.
* Output: Display search results.
  + 1. Non-Functional Requirements

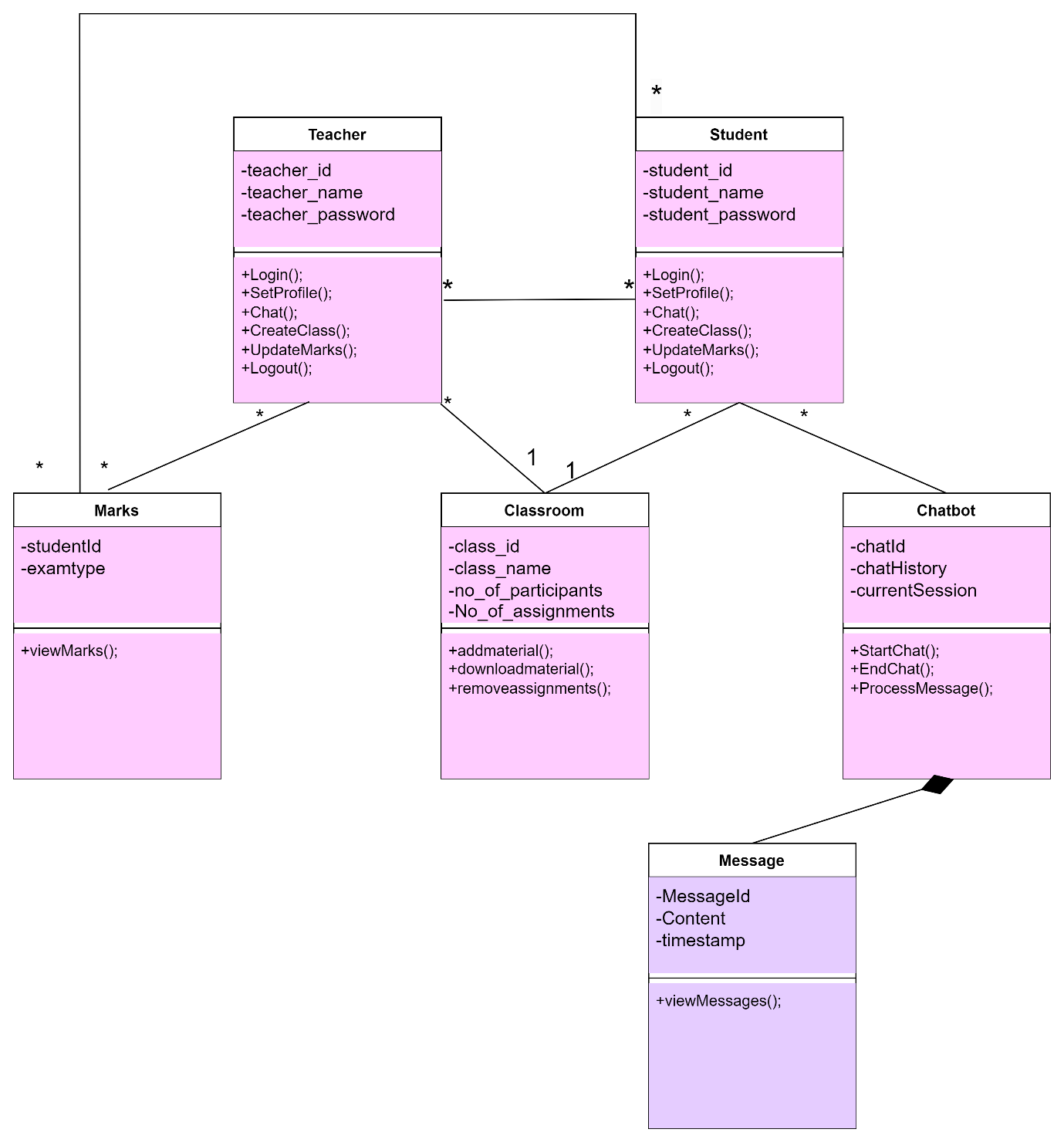
1. Performance: The system should respond to user requests within 2 seconds.
2. Security: User passwords should be securely hashed and stored. SSL/TLS should be implemented for secure data transmission.
3. Scalability: The system should support up to 10,000 simultaneous users.
4. Usability: The user interface should be intuitive, requiring minimal training for users.
5. Reliability: The system should have a 99.9% uptime.

5.UML DIAGRAMS

5.1 Use Case Diagram

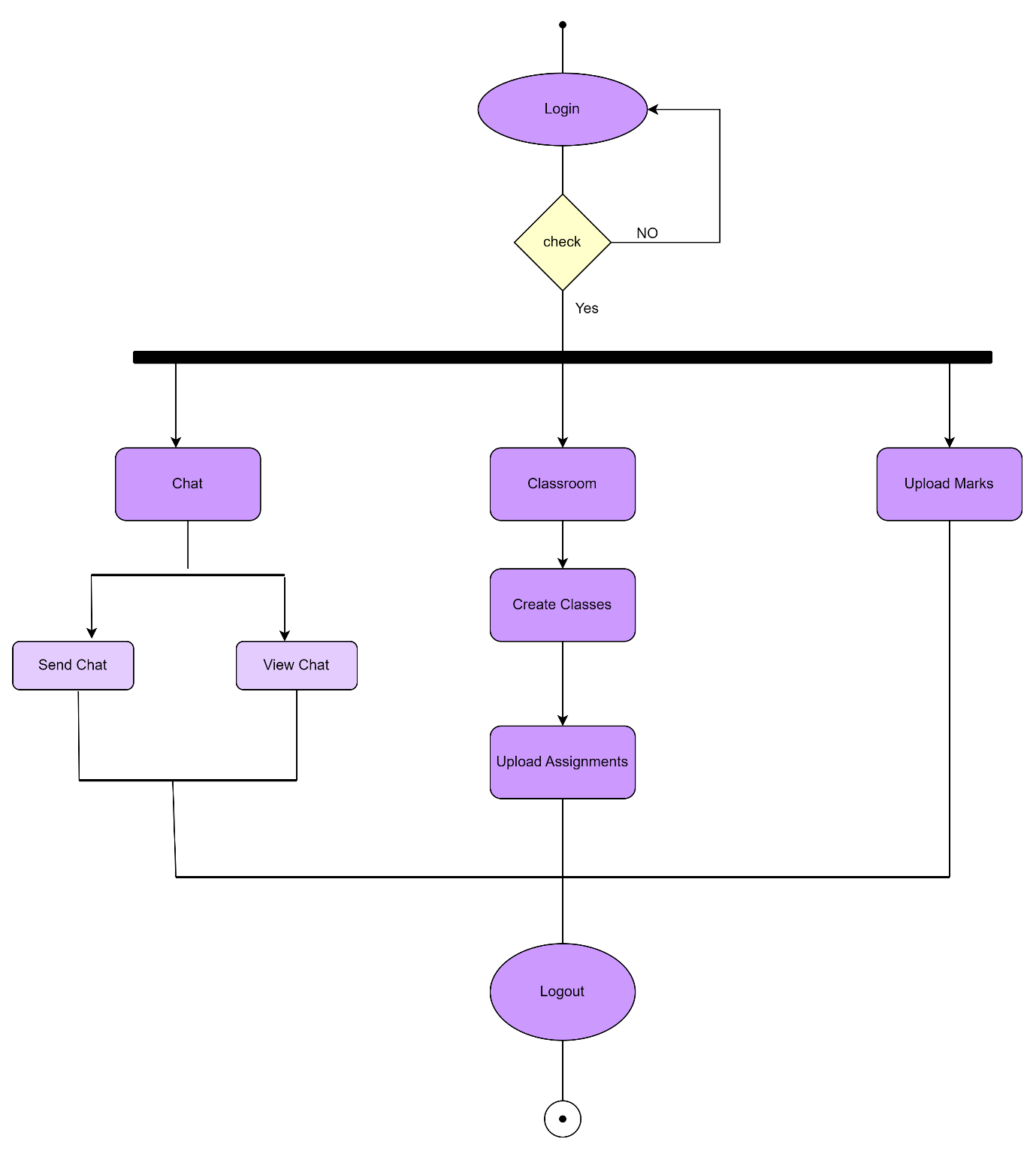


5.2 Class Diagram

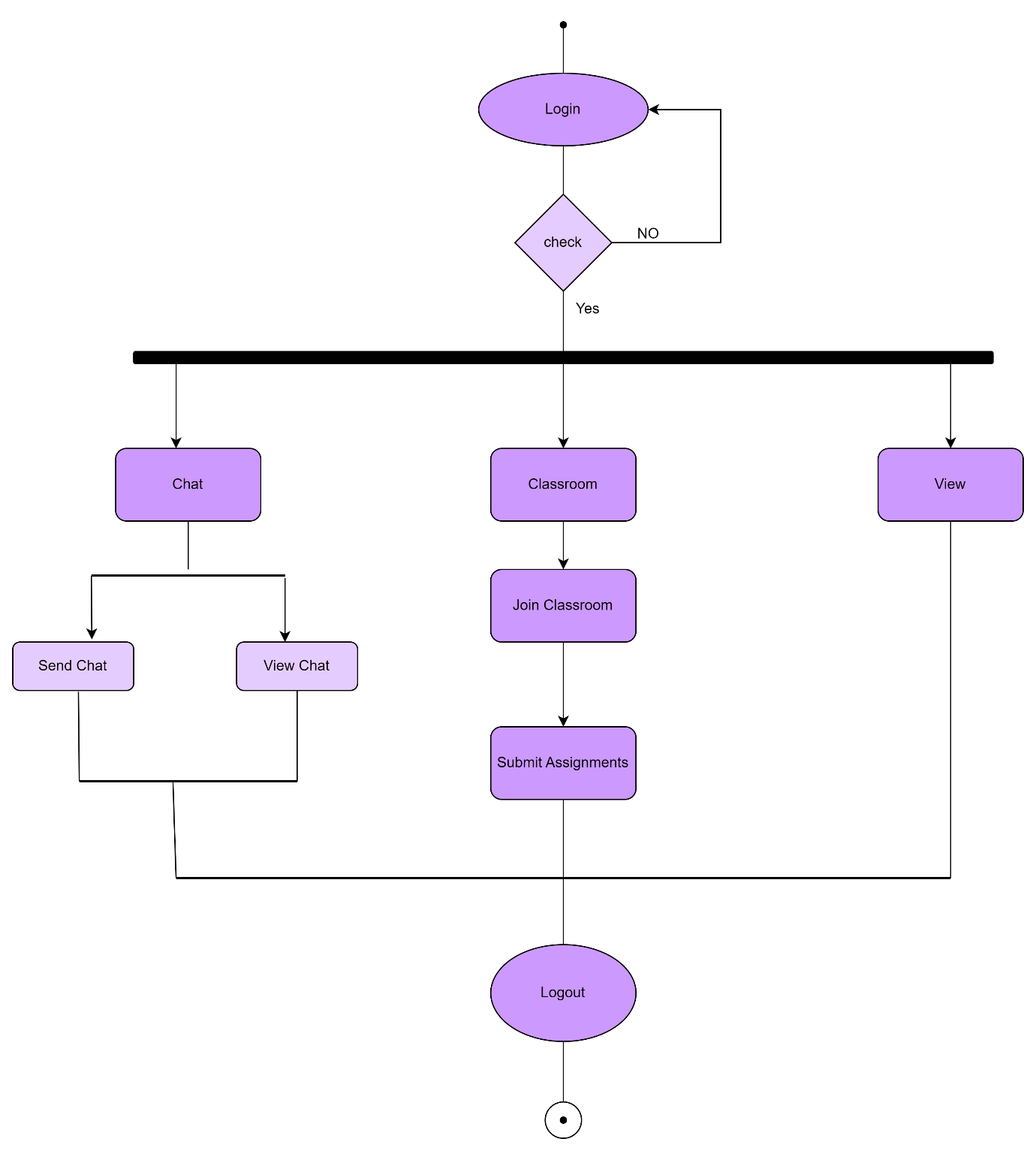


5.3 Activity Diagram

Teacher

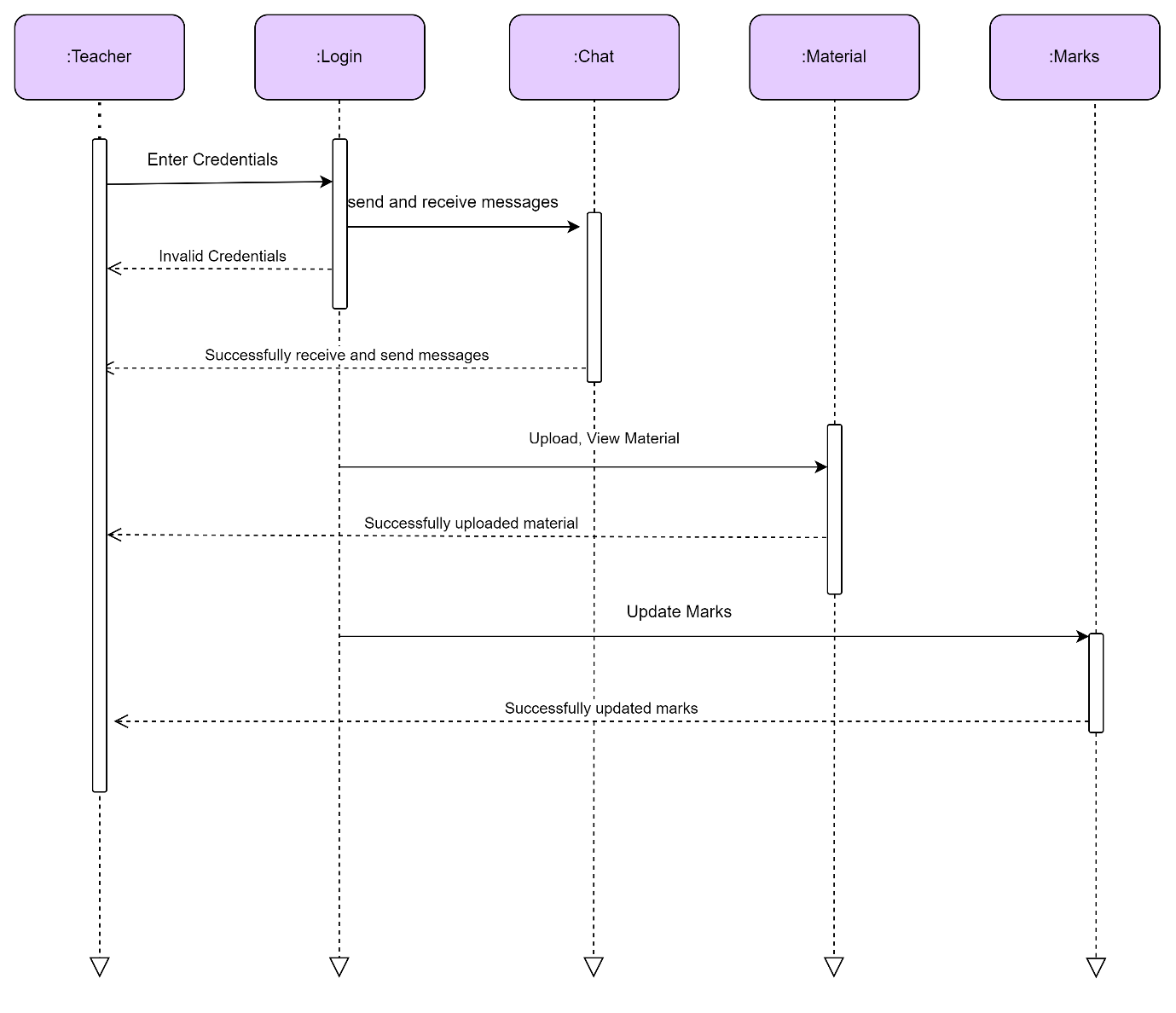


Student

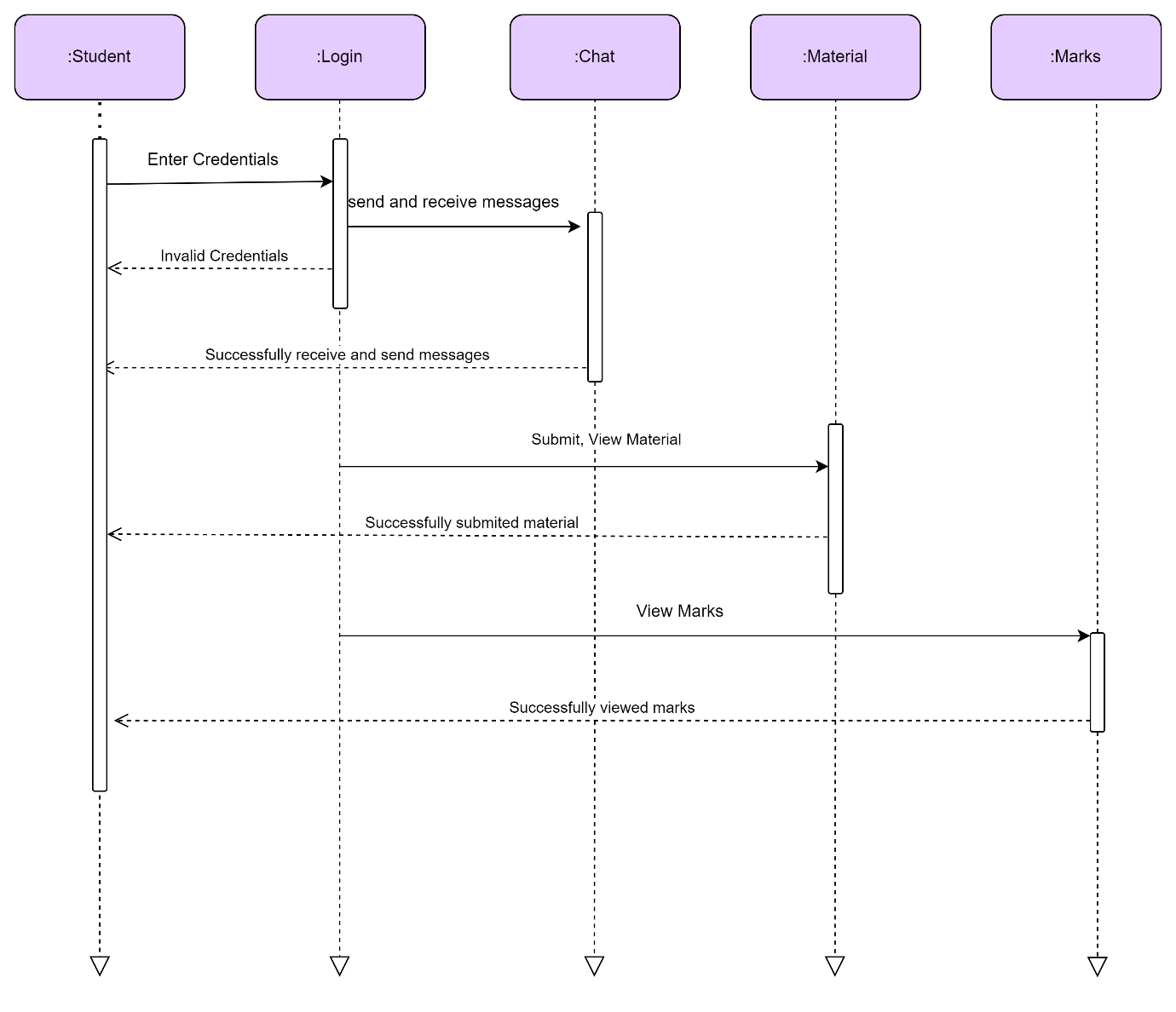


5.4 Sequence Diagram

Teacher



Student



6.IMPLEMENTATION PLANNING

6.1 Implementation Environment (Single vs Multi-User, GUI vs Non-GUI)

For the implementation, we have used: Visual Studio Code Our project is built using Visual Studio Code seeing that it is a web application we saw fit that Visual studio code provided us with all the required basis for the successful implementation of our web app. Also for storing our data we have used Mongo database which enables our web app to run at all times.

6.2 Program/Modules Specification

The following Modules are implemented:

STUDENT:

* Login
* Can use Chatbot
* View Marks and attendance
* View Material in Classroom

TEACHER:

* Login
* Can chat using Chatbot
* Upload Marks and attendance
* Upload Material in Classroom

6.3 Coding Standards

To make the system coding easy, easy to remember, and reduce the chances of errors, some techniques are used at the time of coding of the application which is called coding standard. The coding standard which we adopted during the coding is explained as follows:

* Each nested block should be properly indented and spaced. The code should be properly commented on for understanding easily.
* Comments regarding the statements increase the understandability of the code. 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 for using the function clearly and briefly.