ALUMNI INFORMATION SYSTEM

PROJECT REPORT

18CSC202J/ 18AIC203J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY

(2018 Regulation)

II Year/ III Semester

Academic Year: 2022 -2023

By

CHARVI JAIN (RA2111047010113)

TARUSH KUMAR GOYAL (RA2111047010110)

Under the guidance of

Dr. Poongothai E

Assistant Professor

Department of Computational Intelligence

FACULTY OF ENGINEERING AND TECHNOLOGY

SCHOOL OF COMPUTING

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

Kattankulathur, Kancheepuram

NOVEMBER 2022

BONAFIDE

This is to certify that **18CSC202J - OBJECT ORIENTED DESIGN AND PROGRAMMING LABORATORY** project report titled "**ALUMNI INFORMATION SYSTEM"** is the bonafide work of **CHARVI JAIN (RA2111047010113)** and **TARUSH KUMAR GOYAL (RA2111047010110)** who undertook the task of completing the project within the allotted time.

Signature of the Guide

Dr. Poongothai E

Assistant Professor

Department of CINTEL,

SRM Institute of Science and Technology and Technology

Signature of the II Year Academic Advisor

Professor and Head

Department of CINTEL

SRM Institute of Science and Technology

About the course: -

18CSC202J/ 8AIC203J - Object Oriented Design and Programming are 4 credit courses with **L T P C as 3-0-2-4** (Tutorial modified as Practical from 2018 Curriculum onwards)

Objectives:

The student should be made to:

- Learn the basics of OOP concepts in C++
- Learn the basics of OOP analysis and design skills.
- Be exposed to the UML design diagrams.
- Be familiar with the various testing techniques

Course Learning Rationale (CLR): The purpose of learning this course is to:

1. Utilize class and build domain model for real-time programs
2. Utilize method overloading and operator overloading for real-time application development programs
3. Utilize inline, friend and virtual functions and create application development programs
4. Utilize exceptional handling and collections for real-time object-oriented programming applications
5. Construct UML component diagram and deployment diagram for design of applications
6. Create programs using object-oriented approach and design methodologies for real-time application development

Course Learning Outcomes (CLO): At the end of this course, learners will be able to:

1. Identify the class and build domain model
2. Construct programs using method overloading and operator overloading
3. Create programs using inline, friend and virtual functions, construct programs using

   standard templates

4.Construct programs using exceptional handling and collections

5.Create UML component diagram and deployment diagram

6.Create programs using object-oriented approach and design methodologies

CLAP-1	5= (2(E-lab Completion) + 2(Simple Exercises) (from CodeZinger, and any other coding platform) + 1(HackerRank/Code chef/LeetCode Weekend Challenge)	Elab test
CLAP-2	7.5= (2.0(E-lab Completion) + 2.0 (Simple Exercises) (from CodeZinger, and any other coding platform) + 3.5 (HackerRank/Code chef/LeetCode Weekend Challenge)	Elab test
CLAP-3	7.5= (2.0(E-lab Completion (80 Pgms) + 2.0 (Simple Exercises) (from CodeZinger, and any other coding platform) + 3.5 (HackerRank/Code chef/LeetCode Weekend Challenge)	**2 Mark -** E-lab Completion 80 **Program** Completion from 10 Session (Each session min 8 program) **2 Mark - Code** to UML conversion GCR Exercises **3.5 Mark - Hacker Rank** Coding challenge completion
CLAP-4	5= 3 (Model Practical) + 2(Oral Viva)	• **3 Mark** – Model Test • **2 Mark** – Oral Viva
Total	25	

Table 1: Rubrics for Laboratory Exercises

(Internal Mark Split up: - As per Curriculum)

COURSE ASSESSMENT PLAN FOR OODP LAB

S. No	List of Experiments	Course Learning Outcomes (CLO)	Blooms Level	PI	No of Programs in each session
1.	Implementation of I/O Operations in C++	CLO-1	Understand	2.8.1	10
2.	Implementation of Classes and Objects in C++	CLO-1	Apply	2.6.1	10
3,	To develop a problem statement. 1. From the problem statement, Identify Use Cases and develop the Use Case model. 2. From the problem statement, Identify the conceptual classes and develop a domain model with a UML Class diagram.	CLO-1	Analysis	4.6.1	Mini Project Given
4.	Implementation of Constructor Overloading and Method Overloading in C++	CLO-2	Apply	2.6.1	10
5.	Implementation of Operator Overloading in C++	CLO-2	Apply	2.6.1	10
6.	Using the identified scenarios, find the interaction between objects and represent them using UML Sequence diagrams and Collaboration diagrams	CLO-2	Analysis	4.6.1	Mini Project Given
7.	Implementation of Inheritance concepts in C++	CLO-3	Apply	2.6.1	10
8.	Implementation of Virtual function & interface concepts in C++	CLO-3	Apply	2.6.1	10
9.	Using the identified scenarios in your project, draw relevant state charts and activity diagrams.	CLO-3	Analysis	4.6.1	Mini Project Given
10.	Implementation of Templates in C++	CLO-3	Apply	2.6.1	10
11.	Implementation of Exception of Handling in C++	CLO-4	Apply	2.6.1	10
12.	Identify the User Interface, Domain objects, and Technical Services. Draw the partial layered,	CLO-5	Analysis	4.6.1	Mini Project Given

	logical architecture diagram with UML package diagram notation such as Component Diagram, Deployment Diagram.				
13.	Implementation of STL Containers in C++	CLO-6	Apply	2.6.1	10
14.	Implementation of STL associate containers and algorithms in C++	CLO-6	Apply	2.6.1	10
15.	Implementation of Streams and File Handling in C++	CLO-6	Apply	2.6.1	10

LIST OF EXPERIMNENTS FOR UML DESIGN AND MODELLING:

To develop a mini-project by following the exercises listed below.

1. To develop a problem statement.

2. Identify Use Cases and develop the Use Case model.

3. Identify the conceptual classes and develop a domain model with UML Class diagram.

4. Using the identified scenarios, find the interaction between objects and represent them

using UML Sequence diagrams.

5. Draw relevant state charts and activity diagrams.

6. Identify the User Interface, Domain objects, and technical services. Draw the partial

layered, logical architecture diagram with UML package diagram notation

Suggested Software Tools for UML:

StarUML, Rational Suite, Argo UML (or) equivalent, Eclipse IDE, and Junit

ABSTRACT

The Alumni information system for communication with alumni embodies one of many ways a university can keep track with its graduates. Except for communication between university and its graduates, the information system should allow communication between graduates themselves and their personal presentation in public. The system also should collect actual information about the working experience of graduates, which can improve faculty credits and teaching process. The presented information system includes all these points and focuses on usability and comfortable user interface. The aim of this project is to create a systematic interface for the Alumni of the universities, it is intended to manage the information related to the Alumni of college. The project manages the fresh as well as old graduate students with their respective information in actively participating in making, registering, searching, managing the alumni information for sharing their expertise, network, jobs opportunities and resources.

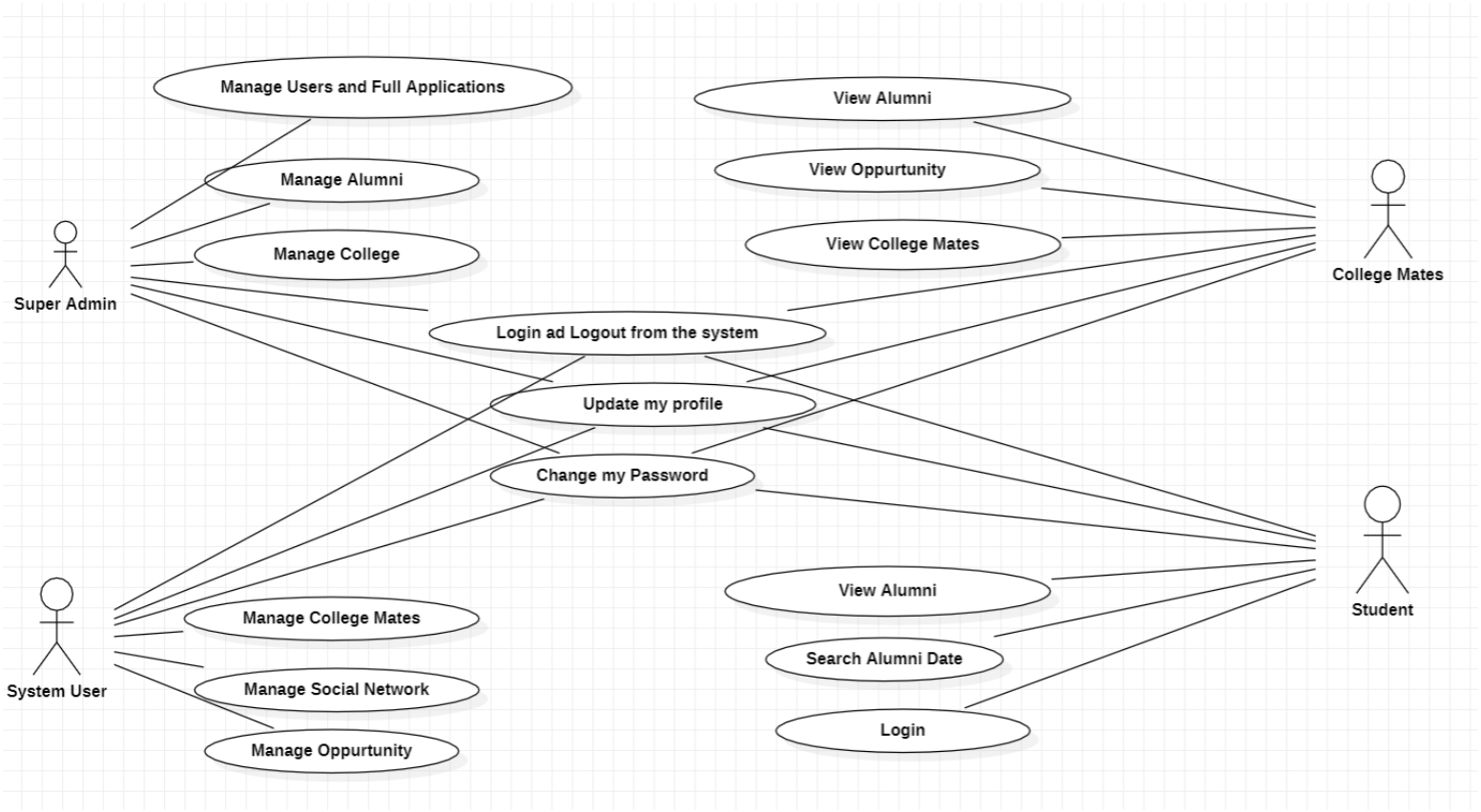
MODULE DESCRIPTION

Alumni Information System is the system responsible for coordinating all aspects of alumni related services such as managing alumni, maintaining their database, viewing opportunities. Managing students, managing college mates. It allows the user to login and access all the features of this system.

The main aims of the alumni information system are:

- To maintain alumni database
- To view opportunities
- To manage college mates
- To manage college activities

Use case diagram with explanation

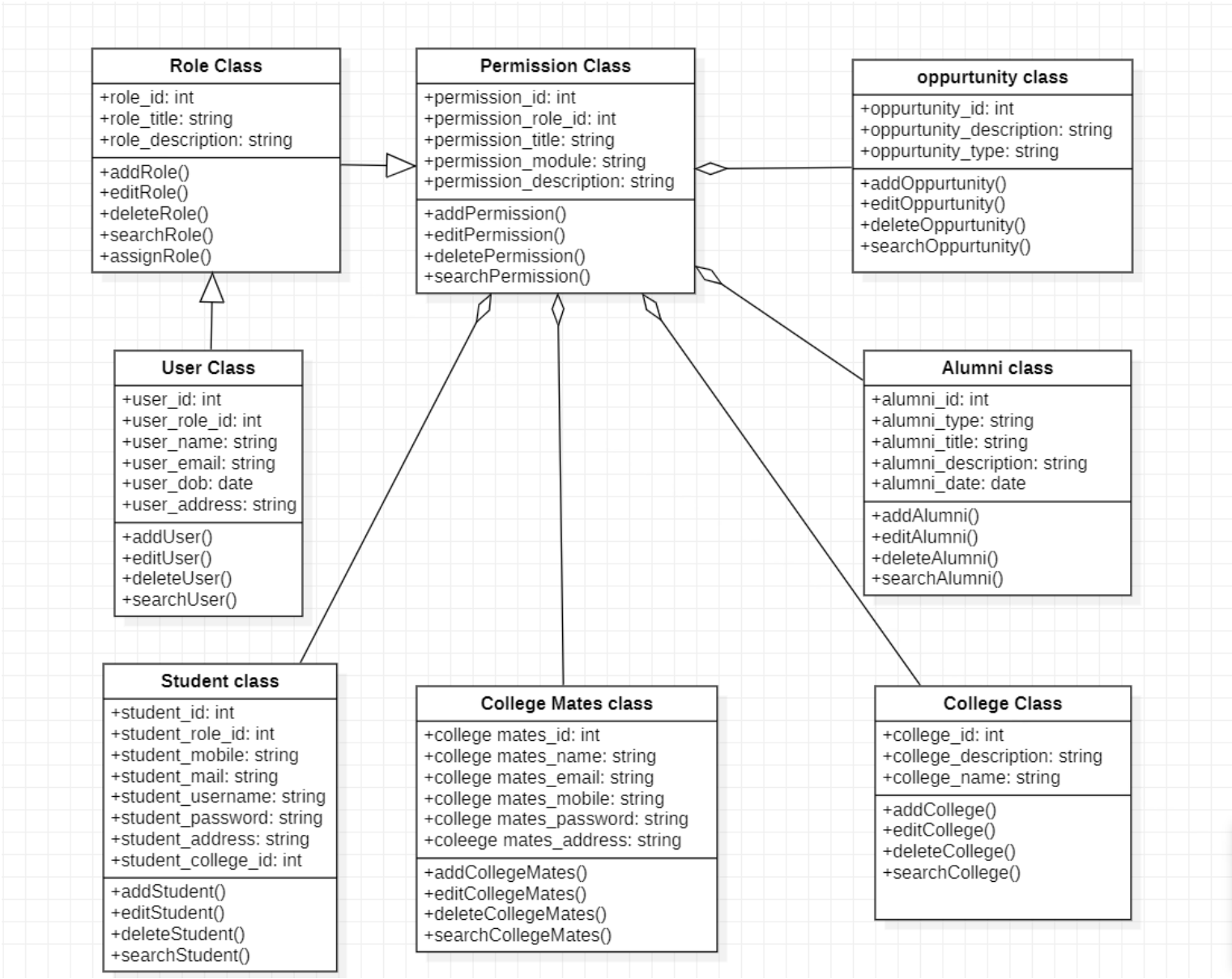


This Use Case Diagram is a graphic depiction of the interactions among the elements of Alumni Information System. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of Alumni Information System. The main actors of Alumni Information System in this Use Case Diagram are: Super Admin, System User, College Mates, Student, who perform the different type of use cases such as Manage Alumni, Manage College, Manage College Mates, Manage Social Network, Manage Opportunity, Manage College News, Manage Student, Manage Users and Full Alumni Information System Operations

The relationships between and among the actors and the use cases of Alumni Information System:

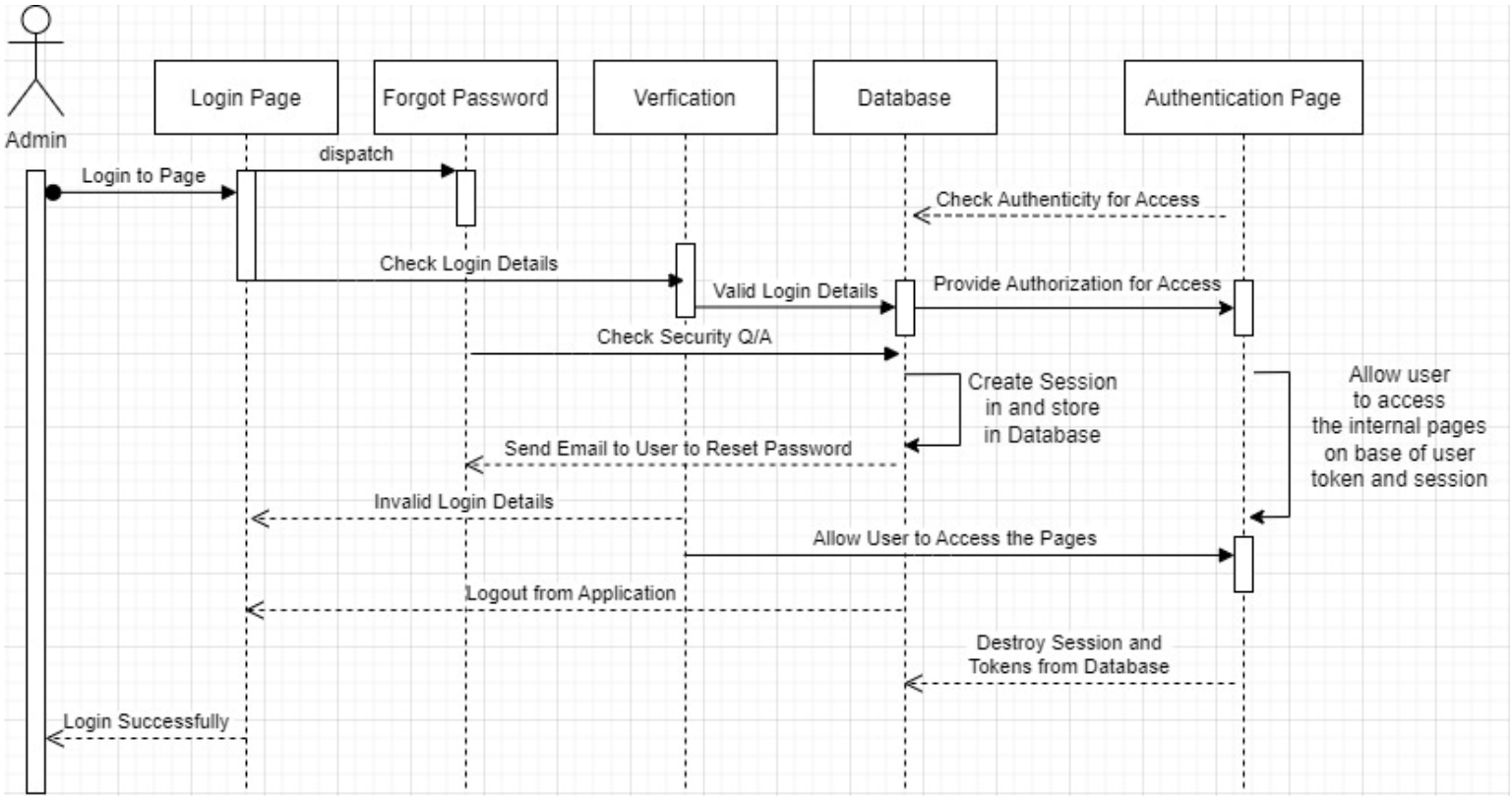
- Super Admin Entity: Use cases of Super Admin are Manage Alumni, Manage College, Manage College Mates, Manage Social Network, Manage Opportunity, Manage College News, Manage Student, Manage Users and Full Alumni Information System Operations
- System User Entity: Use cases of System User are Manage Alumni, Manage College, Manage College Mates, Manage Social Network,
- College Mates Entity: Use cases of College Mates are View Alumni, View opportunity, View College mates, View student
- Student Entity: Use cases of Student are Search Alumni date, View Alumni, View Alumni, Login

Class diagram with explanation



Alumni Information System Class Diagram describes the structure of an Alumni Information System classes, their attributes, operations (or methods). and the relationships among objects. The main classes of the Alumni Information System are Alumni, College, College Mates, Social Network, Opportunity, College News.

Sequence diagram with explanation

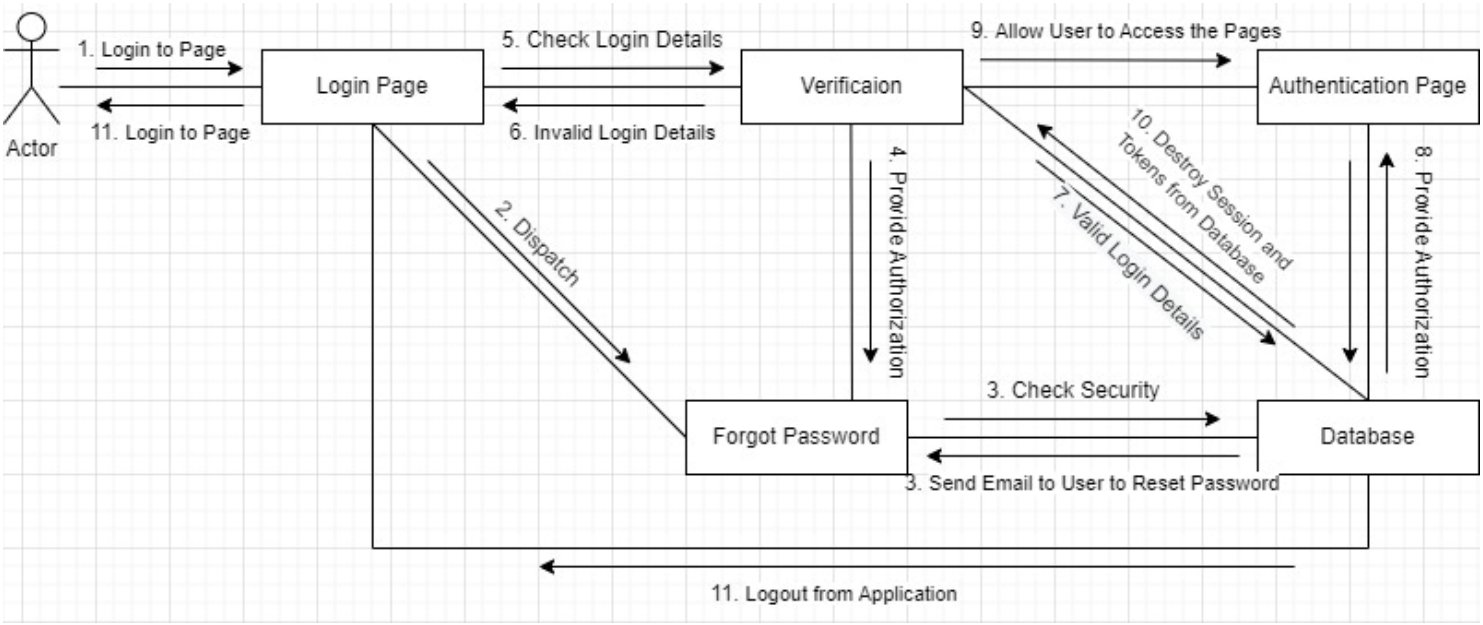


This is the UML sequence diagram of the Alumni Information System which shows the interaction between the objects of Student, Social Network, Alumni, College News, College. Admin will be able to login in their account using their credentials. After login users can manage all the operations on Alumni, Student, Social Network, College, College News. All the pages such as Social Network, College, College News are secure and users can access these pages after login.

The instance of class objects involved in this UML Sequence Diagram of Alumni Information System are as follows:

- Student Object
- Social Network Object
- Alumni Object
- College News Object
- College Object

Communication diagram with explanation

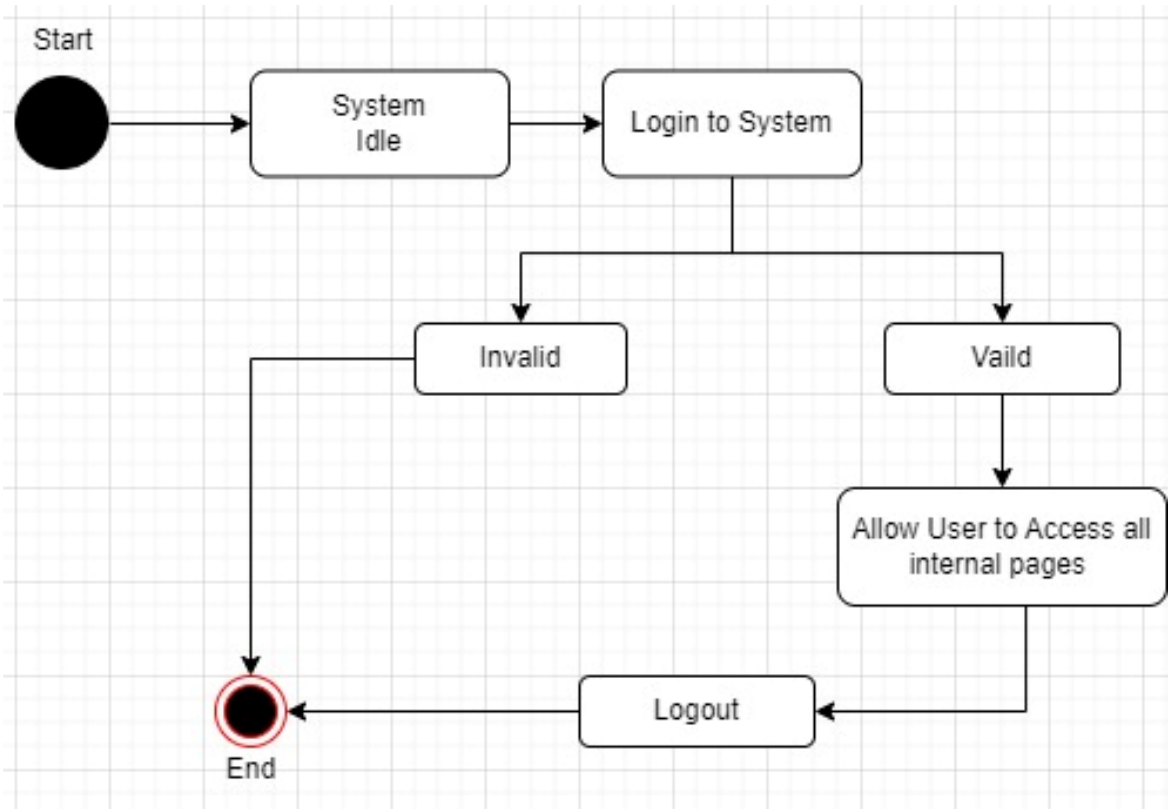


This is the UML communication diagram of the Alumni Information System which shows the communication between the actor and the interface of the login page. Admin will be able to login in their account using their credentials. After login users can manage all the operations and details of login on Alumni, Student and College. All the pages such as Social Network, College, College News are secure and users can access these pages after login.

The steps involved in this UML Communication Diagram of Alumni Information System for a successful session are as follows:

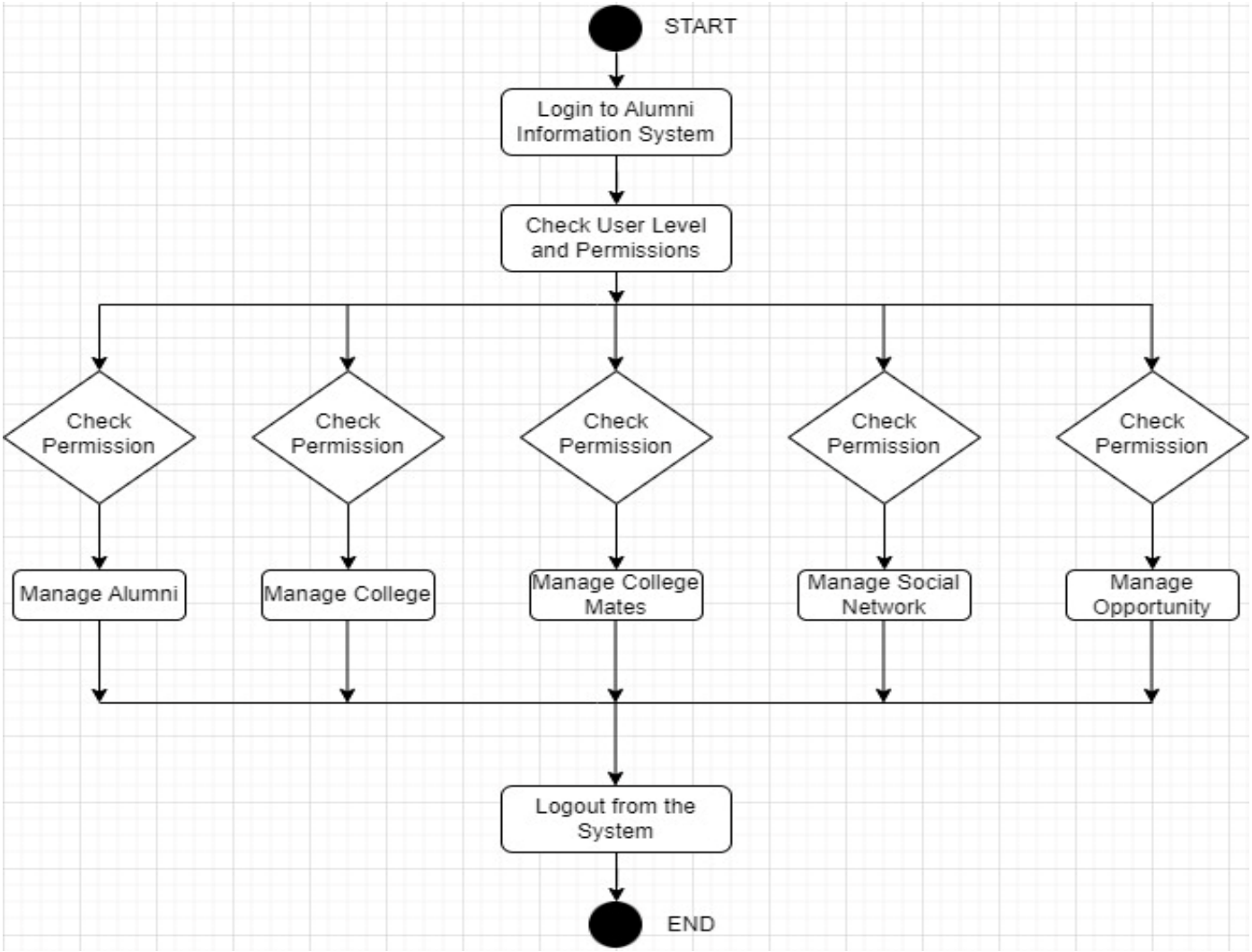
- Login to Page
- Dispatch
- Check Security
- Provide Authentication
- Check Login Details
- Valid Login Details
- Provide Authentication
- Allow user to access the page
- Destroy Session and Tokens from Database
- Logout from Application

State chart diagram with explanation



This is the State Chart diagram of Alumni Information System which shows the flows between the states of the system from Idle, Login, Validation and Logout. The most important purpose of a state chart diagram is to model the lifetime of an object from creation to termination. It is an illustration of the states an object can attain as well as the transitions between those states.

Activity diagram with explanation

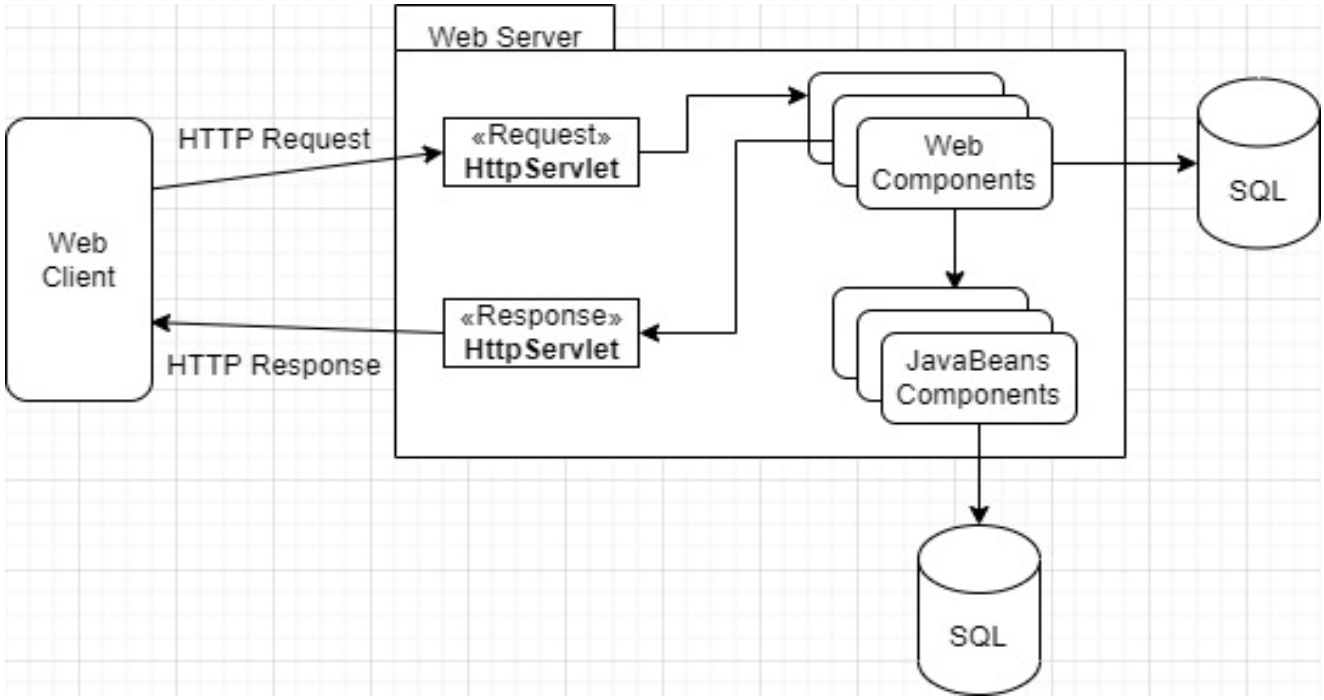


This is the Activity UML diagram of Alumni Information System which shows the flows between the activity of Alumni, Student, College Mates, College News, Social Network.

Features Of the Activity UML Diagram of Alumni Information System:

- Admin User can search Alumni, view description of a selected Alumni, add Alumni, update Alumni, and delete Alumni.
- It shows the activity flow of editing, adding, and updating of Student
- User will be able to search and generate report of College Mates, College News, Social Network
- All objects such as (Alumni, Student, Social Network) are interlinked
- Its shows the full description and flow of Alumni, College News, Social Network, College Mates, Student

Package diagram with explanation

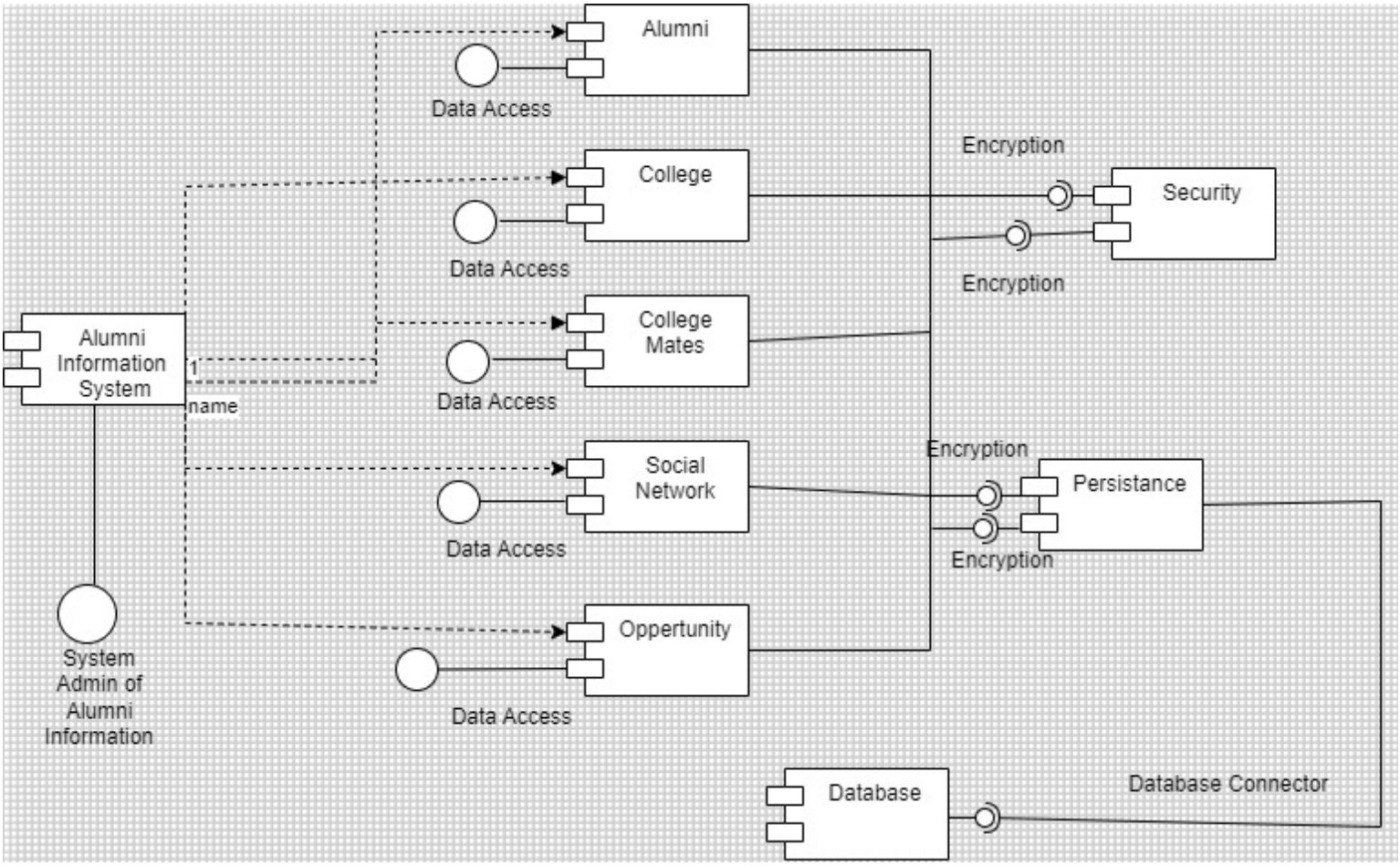


This is the UML Package diagram of the Alumni Information System which shows the organization and arrangement of various model elements in the form of packages.
Each element is nested within the package, which is depicted as a file folder within the diagram, then arranged hierarchically within the diagram. These are most commonly used to provide a visual organization of the layered architecture within any UML classifier.

Features Of The Package UML Diagram Of Alumni Information System:

- It provides a clear view of the hierarchical structure of the various UML elements within a given system.
- These diagrams can simplify complex class diagrams into well-ordered visuals.
- It offers valuable high-level visibility into large-scale projects and systems.
- Package diagrams can be used to visually clarify a wide variety of projects and systems.
- These visuals can be easily updated as systems and projects evolve.

Component diagram with explanation

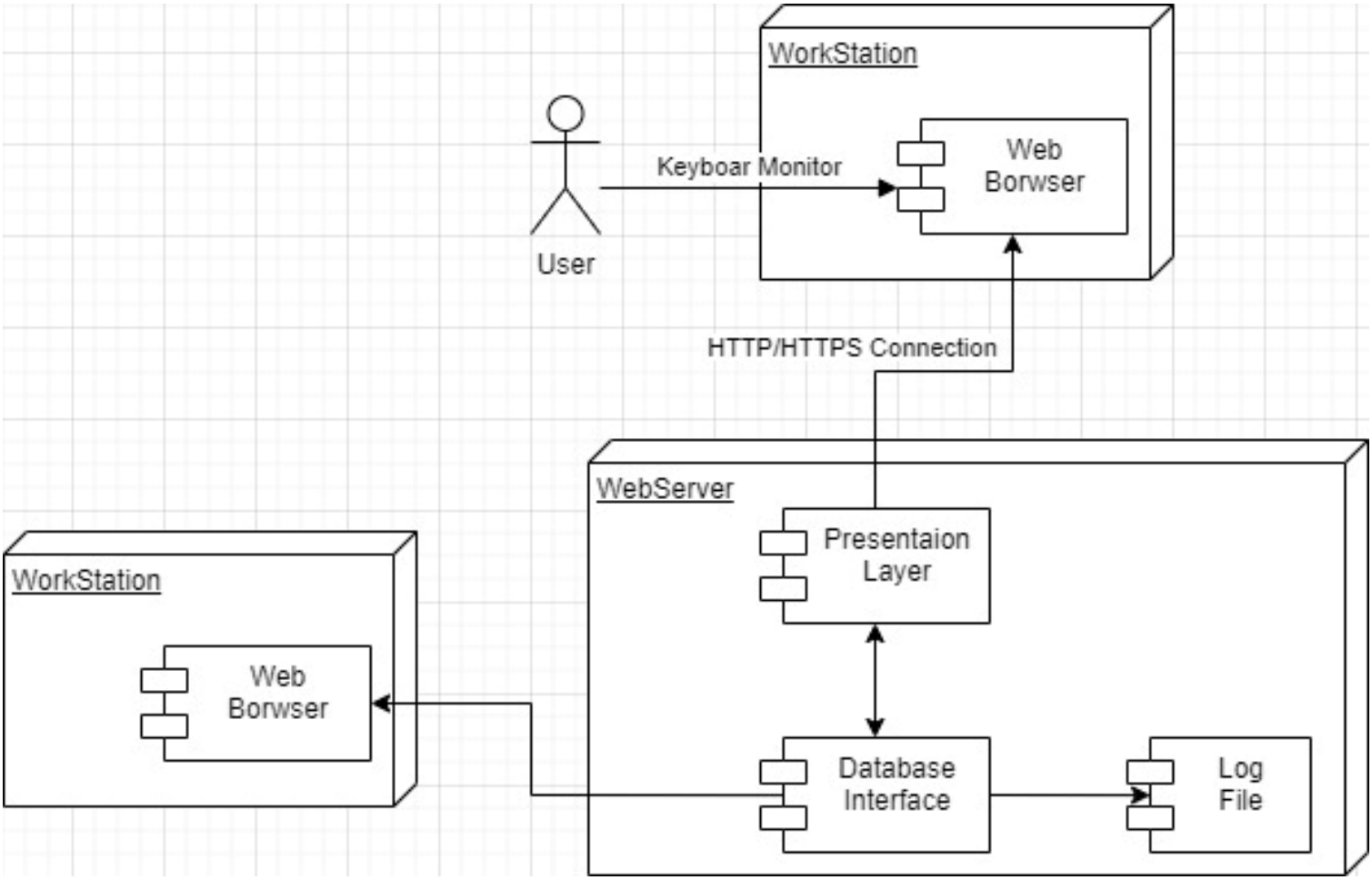


This is a Component diagram of Alumni Information System which shows components, provided, and required interfaces, ports, and relationships between the Social Network, Student, Alumni, College Mates and College. Alumni Information System UML component diagram, describes the organization and wiring of the physical components in a system.

Features of Alumni Information System Component Diagram:

- You can show the components of the Alumni Information System.
- Model the database schema of Alumni Information System.
- Model the executables of an application of Alumni Information System.
- Model the system's source

Deployment diagram with explanation



This is the UML Deployment diagram of Alumni Information System which shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

The components and active objects involved in this UML Deployment Diagram of Alumni Information System are as follows:

- User
- Work Station 1
- Web Server
- Work Station 2

These are connected by using communication association. It shows the path of communication between nodes.

Conclusion

The existing system is a manual one in which users are maintaining documents, paperwork to store the information like Alumni details, college details, student details. It is very difficult to maintain historical data. It is difficult to maintain important information in documents and paperwork. More manual hours are required to generate the required reports. It is a tedious task to manage historical data which needs much space to keep all the previous years' alumni information, documents, and paperwork. The design and implementation of the Alumni Information System will be available for college entity's use through the web interface. A non-registered visitor can look at the list of graduates according to year of graduation or a field of study. They can also look at graduates' profiles. The level of profile details shown to the public is limited. By default, a public visitor can only see the name and surname of a graduate, year of graduation and a field of study. The faculty endeavours to propagate its graduates. Therefore, graduates can also add some information about themselves into the system during the study such as working experience, knowledge. This project aims to create an environment which would connect the current students of the university with their peers, so that they can gain more knowledge, experience and exposure from people who are already working in the industry and are from similar backgrounds.

References

- **https://www.freeprojectz.com/**

- **https://www.inettutor.com/**