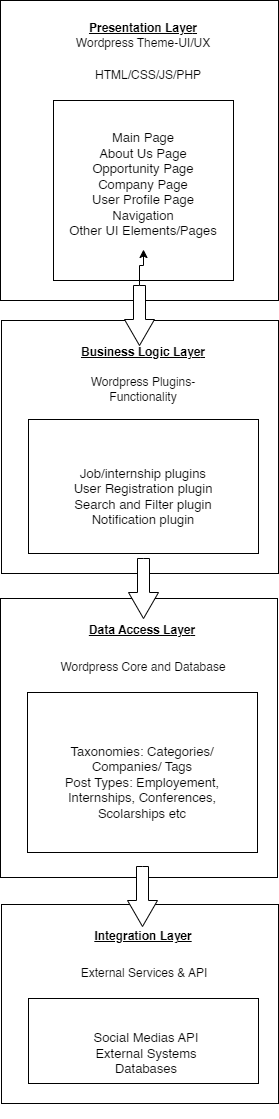
**Phase III: Software Design and Modeling**

Deadline: April 1st, 2024, 23:59

**Software Design and Modeling**

Group Name: Studental

Amanda Peza BINF III-B

**System Architecture:** 

*Explain how different parts of the system work together. Think of it as describing the big picture of your application - what it does and how it does it.*

Studental, is designed as a web-based platform that connects Albanian students with various opportunities such as internships, scholarships, employment, training, and more. Layered Architecture:

* + Description: Organizes components into horizontal layers, promoting separation of concerns and modularity.
  + Applicability: Layered architecture can be suitable for "Studental," because different (horizontal) layers handle specific functionalities. WordPress already provides some level of layered architecture (through plugins).

**Explanation of this system architecture:**

* Presentation Layer (WordPress Theme):
  + - User Interface (UI) design and layout.
    - Rendering content (posts, pages, custom post types).
    - Handling user interactions (forms, buttons, navigation).
* Business Logic Layer (WordPress Plugins):
  + - Business rules and logic (Job/Internship Submission Plugins, User Registration/Login Plugin, Search and Filter Plugin, Notifications Plugin ).
    - Data processing and manipulation.
    - Integration with external services or APIs.
* Data Access Layer (WordPress Core and Database):
  + - Data storage and retrieval ( storing and managing content)
    - Database management and optimization.
    - Interaction with external data sources.
* Integration Layer:
  + - Facilitate communication and data exchange between layers.
    - Manage third-party integrations and services ( e.g.social media platforms)

**Component Diagram:**

*Draw a picture showing the different parts (components) of your application and how they interact with each other. For example, if your application has a login feature, a component diagram would show how the login component talks to other parts of the system.*

Our component diagram consists of the following components:

● Users: The Student User and Admin User, are able to interact with the system.

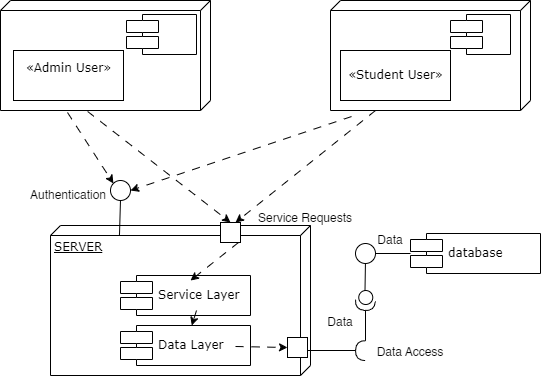
● Processes: The two processes that the users interact with are the Service Requests and Authentication. The actions that the users initiate are handled by these processes.

● Server: It is a representation of the system's host server.

● Service Layer: This layer is in charge of carrying out the system's business logic.

● Data Layer: This layer is responsible for storing and retrieving the data .

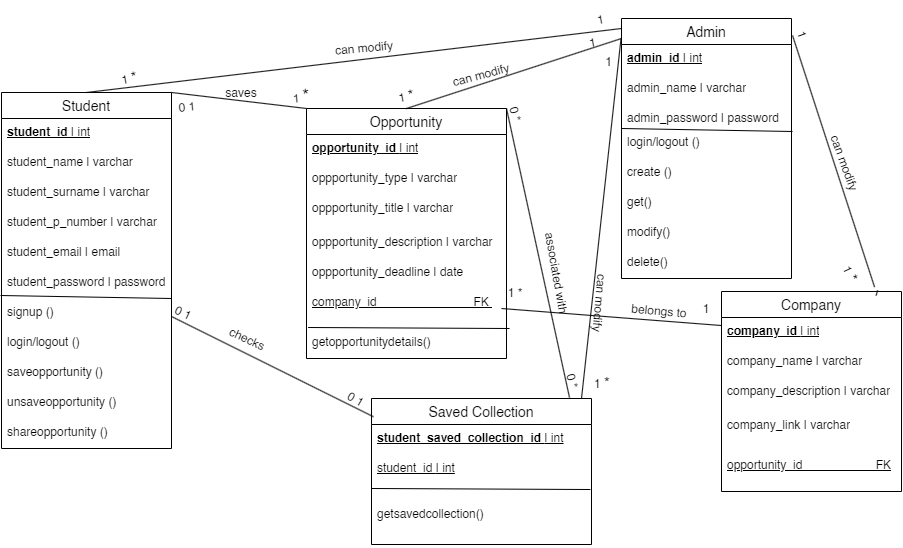
● Data Access and Database: An element named "Data Access" connected to another element labeled "«database»" is located to the right of the Data Layer. They represent the system’s data access mechanism and the actual database where the data is stored.



**Detailed Design**

Class Diagram:

*Think of a class diagram as a family tree for your application. It shows the different types of "things" in your application (called classes) and how they relate to each other. For example, if your application deals with cars, a class diagram would show that a Car class might have attributes like color and model, and methods like drive() and park().*

**

**Sequence Diagrams:**

*Sequence diagrams show the order in which things happen in your application. They're like step-by-step instructions for how different parts of your application interact with each other to accomplish a task. For example, a sequence diagram for ordering food online would show the steps involved, like selecting items, adding them to the cart, and checking out.*

Open the website Studental:

● To log in into the website, the student enters the email and password.

● The website verifies authentication by comparing the student credentials in the database if the credentials are valid.

● The student can access the homepage where he/she can find the opportunities categorized by the company, by its type or can just scroll down the opportunities without any filter.

● In case of unsuccessful authentication the website informs the student to recheck their credentials.

● The list of opportunities is retrieved from the database.

● The student is presented with the retrieved opportunities as mentioned above

Check an opportunity:

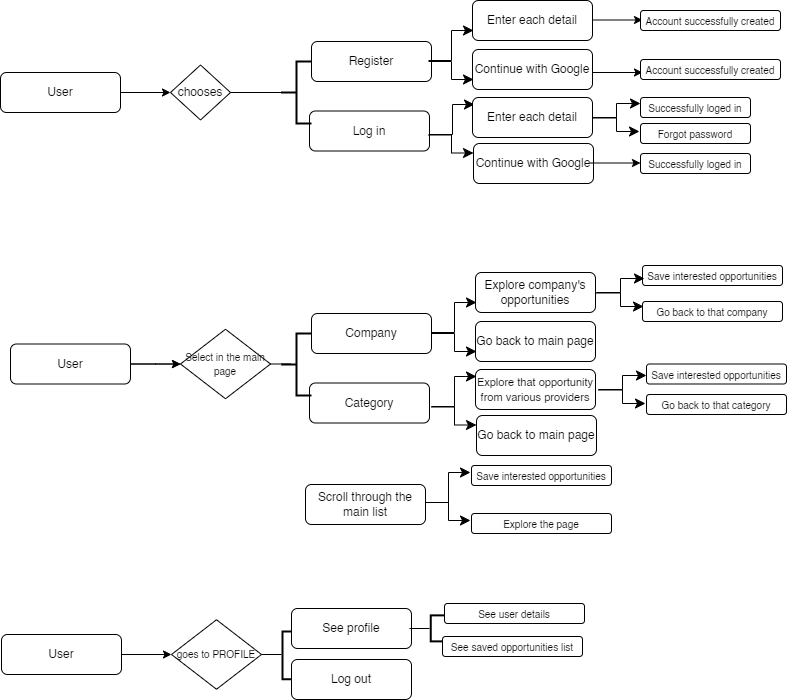
● From the opportunities listed in the website, the student chooses the one he/she is most interested in.

● After clicking on the opportunity, they can see more details about it and also about the company that has posted it.

● They cannot directly apply for a certain position because they have to follow the link/instructions given to make the application at the company offering it.

A *use case diagram* shows the different ways people (or other systems) can use your application. It's like a map of all the different things your application can do. For example, a use case diagram for a music streaming app might show that users can search for songs, create playlists, and listen to music.

*Activity diagrams* show the flow of activities in your application. They're like flowcharts that show the steps involved in completing a task. For example, an activity diagram for booking a flight might show the steps involved, like searching for flights, selecting one, and entering passenger information.



**Database Design:**

Explain how you've organized the data in your application. This includes things like what tables you have in your database, how they're related to each other, and how you've made sure your data is organized efficiently.

1. “student” Table consisting of student data :

○ **student\_id (Primary Key)**

○ student\_name

○ student\_surname

○ student\_p\_number

○ student\_email

○ student\_password

2. “opportunity” Table consisting of opportunities data :

○ **opportunity\_id (Primary Key)**

○ opportunity\_type

○ opportunity\_title

○ opportunity\_description

○ opportunity\_desadline

○ company\_id (Foreign Key)

3. “company” Table consisting of company related data :

○ **company\_id (Primary Key)**

○ company\_name

○ company\_description

○ company\_link

○ opportunity\_id (Foriegn Key)

4. “saved\_collection” Table consisting of saved opportunities by a student:

○ **student\_saved\_collection\_id** ( Primary Key)

○ opportunity\_id (Foriegn Key)

○student\_id (Foriegn Key)

● Student Side

Each student once registered in the website is identified by his unique “student\_id”. The “student\_email” and “student\_password” data are reused every time the student wants to log-in. The “student\_id” for a particular student is created when a student is registered.

● Admin side

The “admin\_username” and “admin\_password” are used by the administrator in order to gain access to the Wordpress website and log in. The admin in our application is able to create, get, modify and delete any particular data from the database that he wants. He is able to access all of the database tables along with changing the given data for each one of them.

