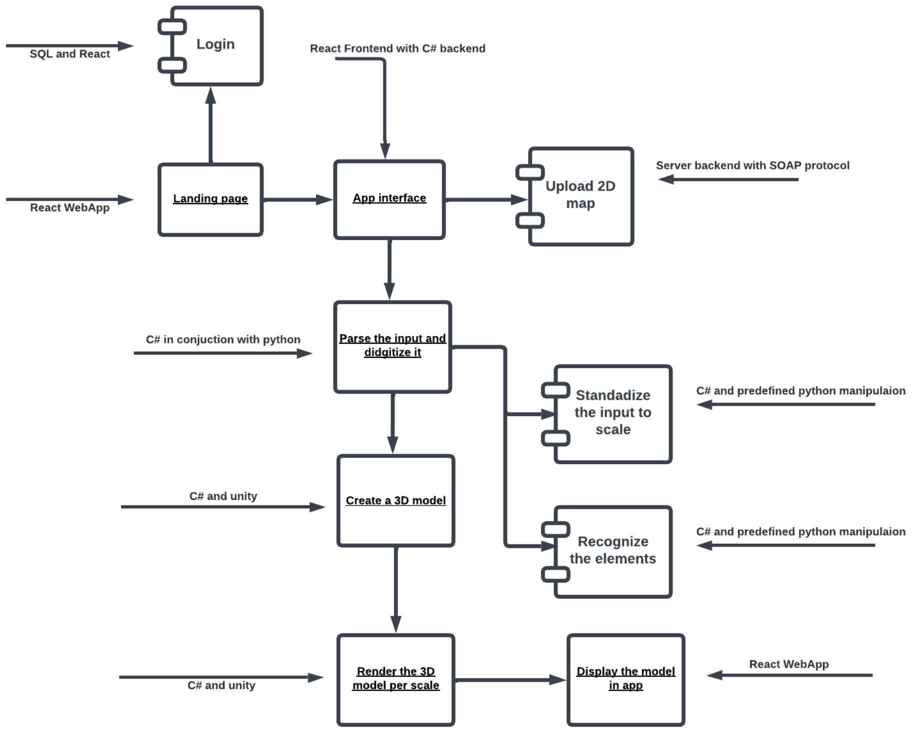
Team Number: 3

Team Members:
Ehtisham Mushtaq
Asrar Farooq
Huy Truong
Mai Duong
Nikhil Singla

Project Name: Mapalyze

Project Synopsis: Our app takes multiple floor plans and generates a 3D layout of a building appropriately marking stairs hallways and rooms, with navigation built on top.

Architecture:
Creating an app that takes multiple floor plans and generates a 3D layout of a building with navigation features is a complex task that requires expertise in several domains, including computer graphics, 3D modeling, and navigation algorithms. Here's a simplified outline of how you might approach building such an app:


SQL and React
Login
React Frontend with C# backend
React WebApp
Landing page
App interface
Upload 2D map
Server backend with SOAP protocol
C# in conjuction with python
Parse the input and didgitize it
Standadize the input to scale
C# and predefined python manipulaion
C# and unity
Create a 3D model
Recognize the elements
C# and predefined python manipulaion
C# and unity
Render the 3D model per scale
Display the model in app
React WebApp

Img1: Initial Architecture

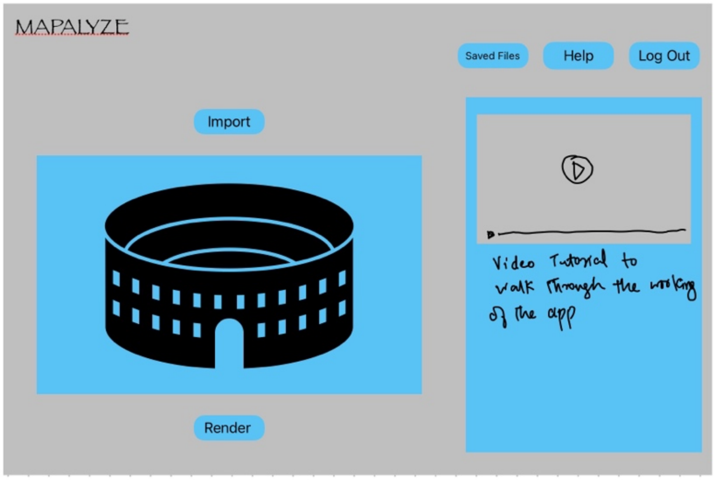
1. Data Input:
   - Develop a mechanism for users to input floor plans. This could involve allowing users to upload floor plan images or draw floor plans within the app. This is going to be a React frontend which has a CSharp backend for Unity support. The React front end will be capable of parsing and saving the input file which would then be sent to the next step in the process for recognition and manipulation.

2. Floor Plan Parsing:
   - Implement image processing or vectorization algorithms to extract relevant information from the floor plans. This includes identifying rooms, hallways, stairs, and dimensions.
   - Convert the 2D floor plan data into a structured digital format that can be used for 3D modeling. This step basically digitizes the plain 2D input and converts it into a readable format for the next step.

3. 3D Modeling:
   - Use the parsed floor plan data to create a 3D representation of the building. This may involve modeling walls, floors, ceilings, and structural elements. This is done by creating standard classes for each kind of possible floor element (walls, doors etc.) and then assigning various portions of the floor to the most correct class.
   - Ensure that the scale and dimensions are accurately reflected in the 3D model. Also, we have to make sure that all the legends from the floor plans are translated properly into the 3D layout.

4. Navigation System:
   - Develop a navigation system that operates within the 3D model.
   - This system should allow users to:
     + Set starting and ending points for navigation.
     + Calculate and display optimal routes within the building.
     + Consider factors like stairs, elevators, and hallways in route planning.
     + Provide real-time location tracking for users within the building, possibly using indoor positioning technologies like Wi-Fi triangulation or Bluetooth beacons.

5. User Interface:
   - Design an intuitive user interface for interacting with the 3D model and navigation features. This could include touch controls for zooming, panning, and selecting destinations.
   - Make sure that UI will help making the app more user friendly, allowing users to interact with and utilize the app more effectively.

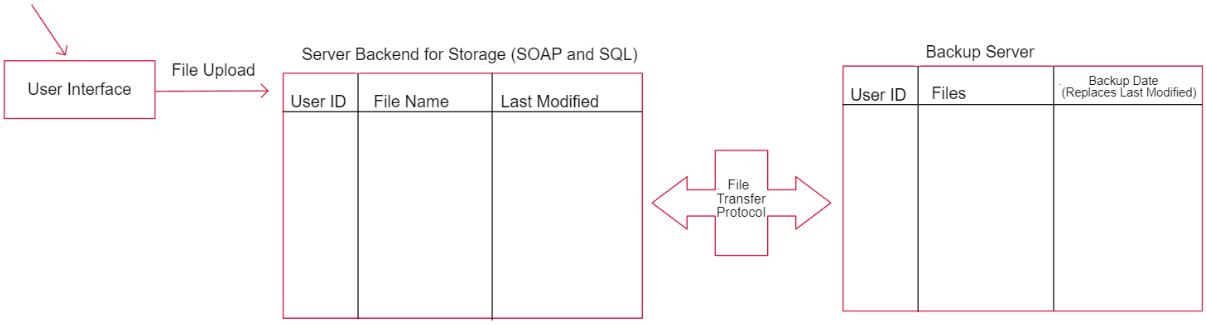


Img2: Tentative User interface

- Ensure that UI can speed up user processes, making it simpler and quicker to input floor plans, set up settings, and move around a 3D layout.

6. Rendering and Visualization:
   - For this step, we need to use 3D rendering techniques to display the 3D model on the user's device. The model will most likely use the Unity Engine to render our picture, and we need to input the instructions using code. We will also work on performance optimizations to ensure smooth navigation even on less powerful hardware by reducing redundant portions of the diagram.
   - We need to be able to create simplified, but large 3D models based on the given input (floor plans), and thus we will do that by digitizing the floor plan into tokens of instructions perhaps that can be followed like an algorithm to render the final output.

7. Data Storage and Management:
   - Implement a backend system for storing and managing floor plan data, 3D models, and user preferences.
   - Make sure that all floor plans, 3D models, and navigational information are preserved over time by being kept in a structured database. Even after exiting the app or moving devices, users may still access their projects, revisions, and data.
   - Develop automatic data backup and recovery mechanisms to protect against data loss due to user errors, hardware failures, or other unforeseen events so users can rest assured that their work is safe.
   - We try to optimize data storage and retrieval mechanisms so we can ensure that the app performs smoothly, even with complex 3D models and large datasets.



Img3: Database setup

8. Testing and Validation:
   - Develop and execute comprehensive unit and integration tests to validate each component and their interaction with the app.
   - Engage a select group of users to test the app in real world conditions to ensure the app meets user expectations and the intended use case scenarios.

9. Security and Privacy:
   - Implement strong encryption algorithms for stored data which will protect user data and sensitive building layouts.

- Ensure that application codebase adheres to secure coding standards by regularly conducting code reviews and using automated tools to detect any potential vulnerabilities.

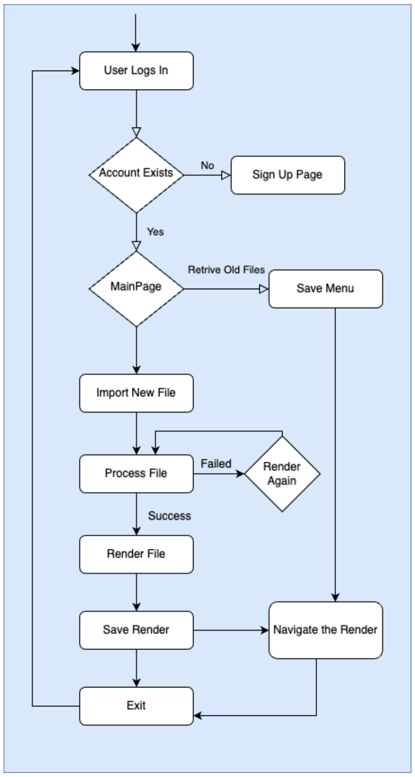
10. Deployment and Maintenance:
    - This portion of the project is related to deploying the created app on the various platforms (through the medium of a webpage), and then having architecture in place to consistently update it and fix any arising bugs. The team uses this tool to implement backend fixes at the press of a button and also to immediately update any frontend code. This can be done by synchronizing the updates to the main server and pushing it to the root branch.

Final Workflow

The web application offers a dynamic and innovative solution for users seeking to transform their floor plans and diagrams into stunning 3D architectural renderings. It simplifies the complex process of creating multi-story buildings by utilizing sophisticated algorithms on the backend to seamlessly stack and connect each floor plan, resulting in a cohesive and visually captivating 3D representation of a building.

Upon entering the application, users are greeted with the option to upload one or more floor plans or diagrams. These blueprints serve as the foundation for the virtual construction process. Users can easily save their uploaded files, allowing for quick access and continued work when they return to the platform in the future.

A key feature of the application is its ability to interpret and incorporate legends found within the floor plans. These legends are a critical component in accurately identifying and placing architectural elements such as doors, stairs, and hallways in the 3D layout. This attention to detail ensures that the final rendering faithfully reflects the original design intent.



Img4: Tentative Workflow

Once the application has successfully rendered the 3D building layout, users are empowered to explore and navigate within the virtual structure. The navigation menu offers a user-friendly interface for selecting a starting and ending location, facilitating effortless movement between rooms and floors. As a forward-looking expansion, the application is poised to extend its functionality to allow seamless navigation between multiple buildings, enhancing its utility for a wide range of architectural and urban planning projects.

In summary, this web application simplifies the transition from 2D floor plans to immersive 3D architectural representations. It leverages cutting-edge algorithms, preserves user input, and enables precise navigation within and potentially between buildings, offering a comprehensive and dynamic tool for architects, designers, and anyone interested in visualizing architectural spaces.