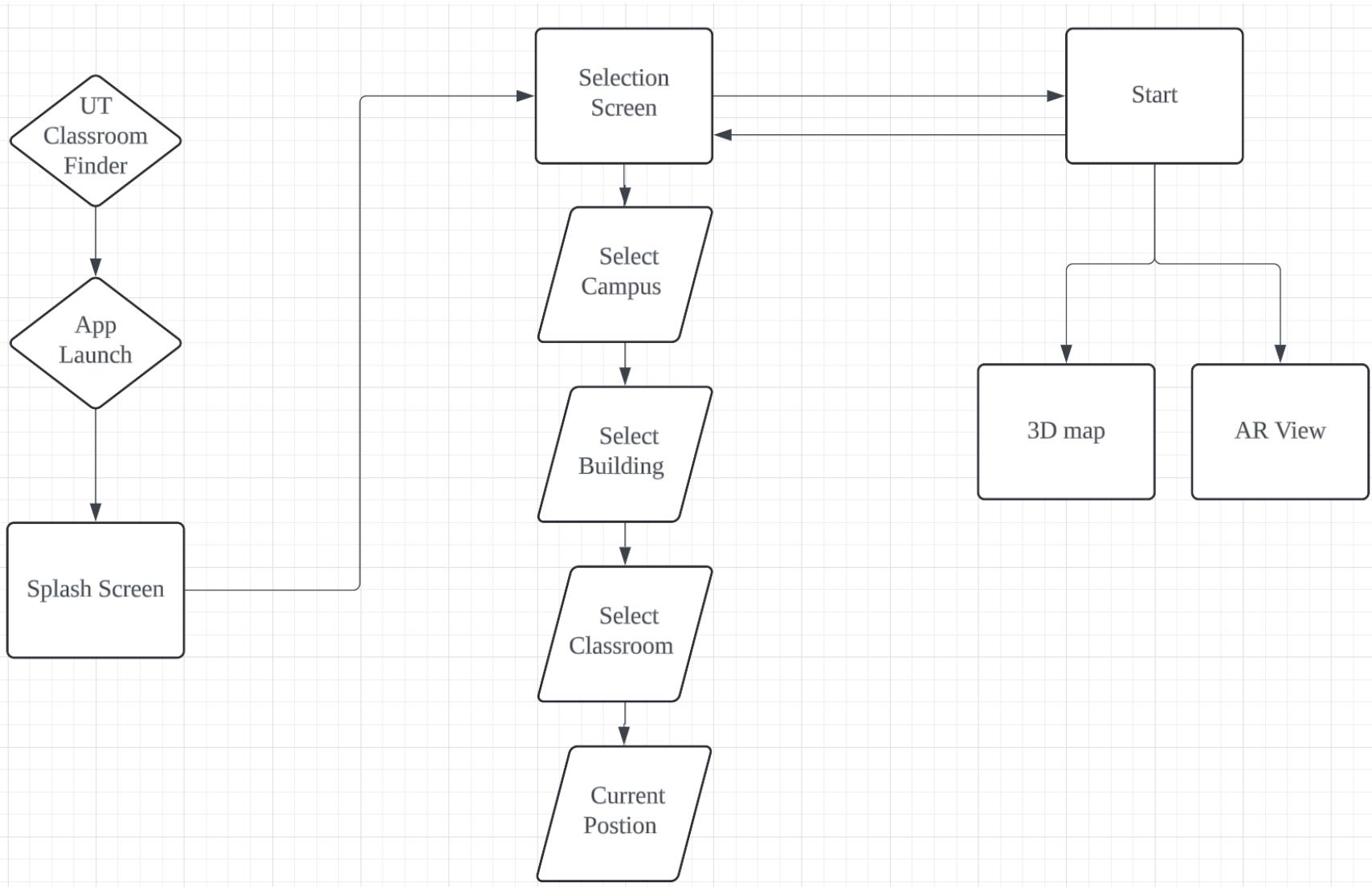


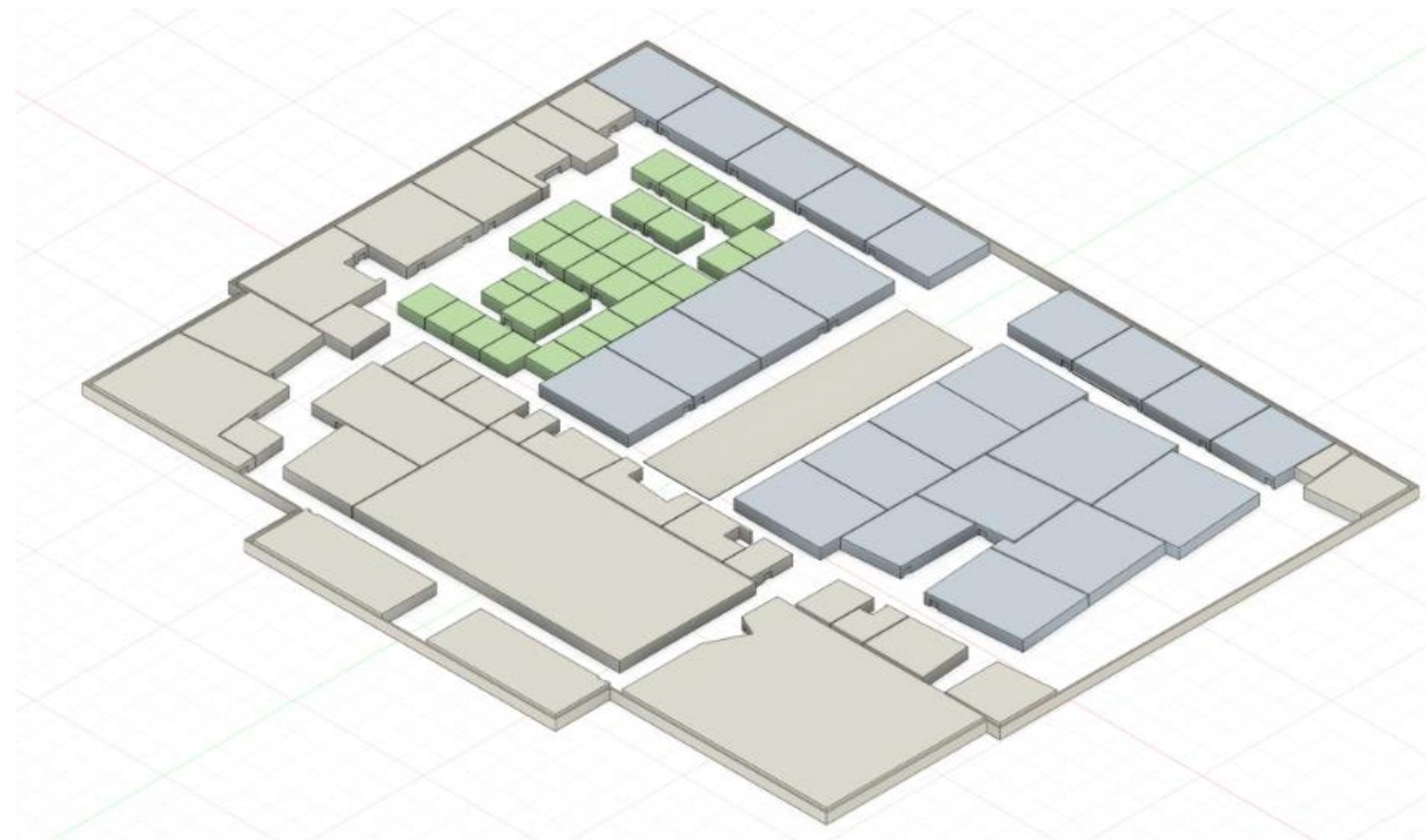
Flowchart



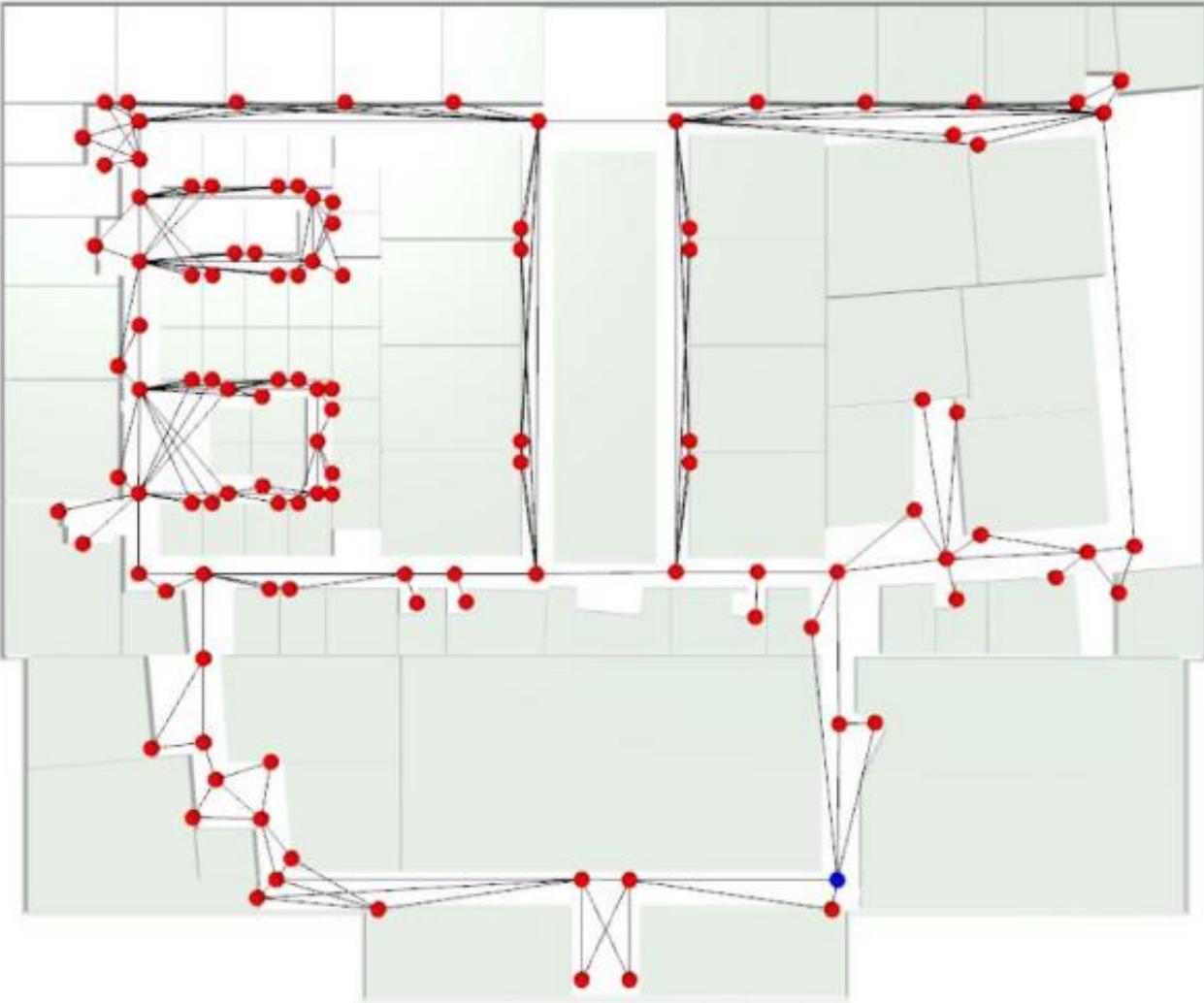
Application Process

1. Launch App
2. Select current position
3. Select destination
4. Generate route
5. AR Navigation
6. 3D Map view

3D Map Modeling

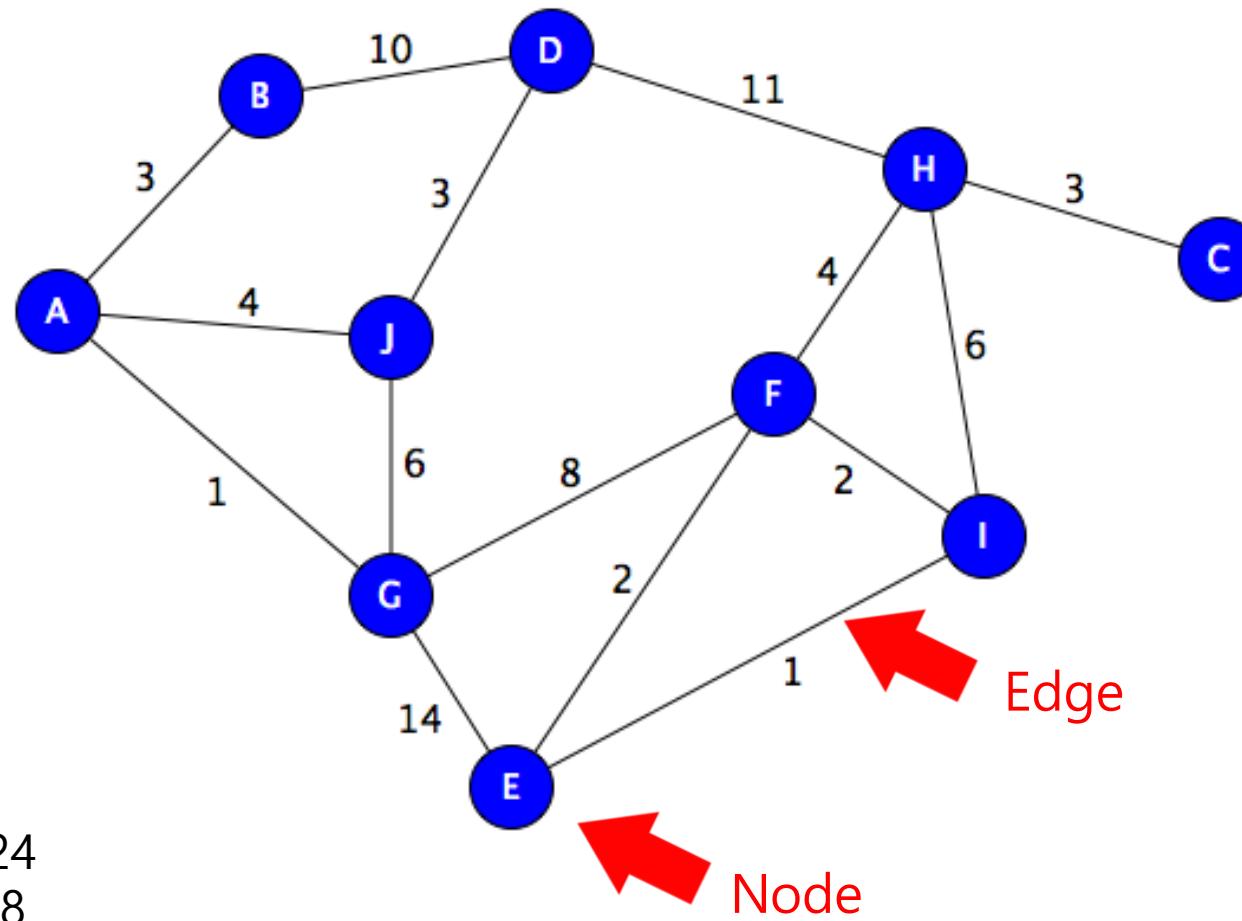


Creating nodes & edges



```
38 let FHnodes: [nodes] = [
39   nodes(name: "Restroom 1020", x: 6.347, y: -0.383, z: 0.2),
40   nodes(name: "Room 1030", x: 4.444, y: 1.549, z: 0.2),
41   nodes(name: "Room 1040", x: 4.42, y: 1.806, z: 0.2),
42   nodes(name: "Room 1050", x: 4.233, y: 1.972, z: 0.2),
43   nodes(name: "Room 1060", x: 3.808, y: 0.485, z: 0.2),
44   nodes(name: "Room 1080", x: 4.722, y: -2.114, z: 0.2),
45   nodes(name: "Room 1090", x: 4.815, y: -3.15, z: 0.2),
46   nodes(name: "Room 1100", x: 4.77, y: -4.238, z: 0.2),
47   nodes(name: "Room 1110", x: 3.885, y: -2.486, z: 0.2),
48   nodes(name: "Room 1120", x: 3.884, y: -3.0, z: 0.2),
49   nodes(name: "Room 1140", x: 2.646, y: -4.584, z: 0.2),
50   nodes(name: "Room 1150", x: 3.585, y: -0.54, z: 0.2),
51   nodes(name: "Room 1180a", x: -0.869, y: -1.365, z: 0.2),
52   nodes(name: "Room 1180b", x: 1.374, y: -1.365, z: 0.2),
53   nodes(name: "Room 1190", x: -0.66, y: -0.72, z: 0.2),
54   nodes(name: "Room 1200", x: 1.121, y: 1.355, z: 0.2),
55   nodes(name: "Room 1210", x: 1.121, y: 1.613, z: 0.2),
56   nodes(name: "Room 1220", x: 1.121, y: 3.977, z: 0.2),
57   nodes(name: "Room 1230", x: 1.121, y: 4.23, z: 0.2),
58   nodes(name: "Room 1240", x: -1.039, y: 4.234, z: 0.2),
59   nodes(name: "Room 1250", x: -1.039, y: 3.976, z: 0.2),
60   nodes(name: "Room 1260", x: -1.039, y: 1.613, z: 0.2),
61   nodes(name: "Room 1270", x: -1.039, y: 1.362, z: 0.2),
62   nodes(name: "Room 1280", x: -1.039, y: -0.772, z: 0.2),
63   nodes(name: "Room 1290", x: -3.634, y: -0.591, z: 0.2),
64   nodes(name: "Room 1295", x: -4.084, y: -0.588, z: 0.2),
65   nodes(name: "Room 1300a", x: -0.588, y: -1.365, z: 0.2),
```

Implementation of Dijkstra's Algorithm

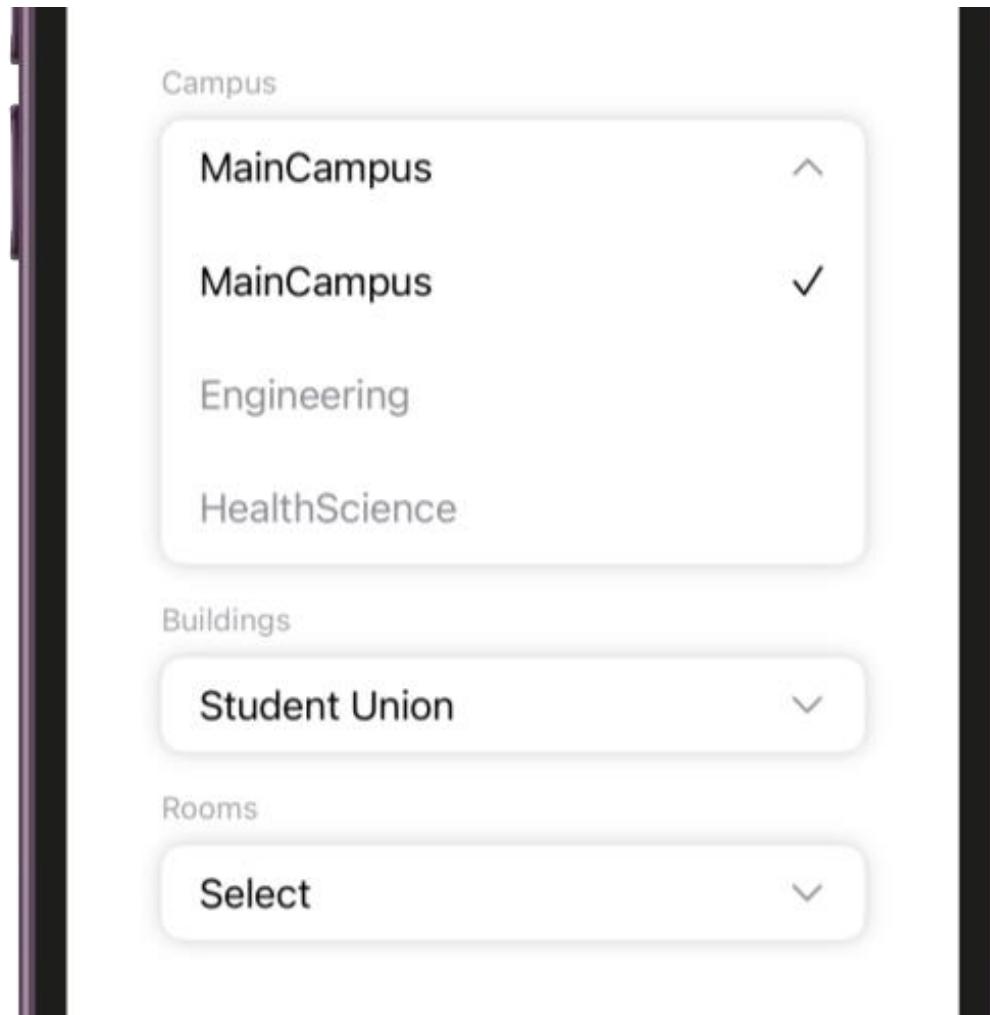


$A \rightarrow B \rightarrow D \rightarrow H = 24$

$A \rightarrow J \rightarrow D \rightarrow H = 18$

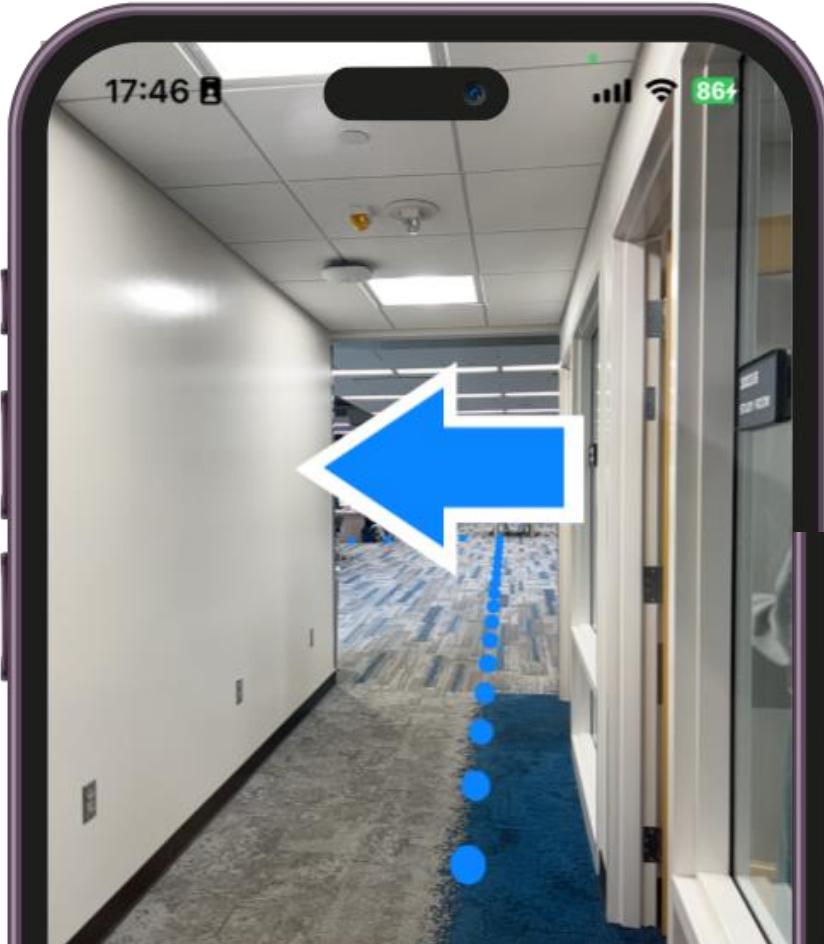
$A \rightarrow G \rightarrow F \rightarrow H = 13$ <- Shortest route!

Users input the current position
& destination room #'s

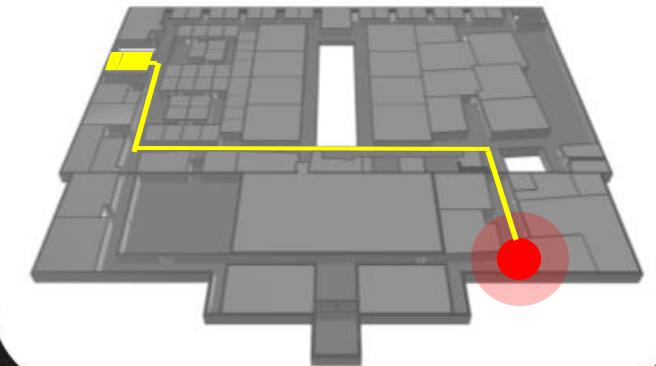


AR dots & an arrow

```
503  
504 func addArrowNodeToScene() {  
505     if self.arrowNode != nil { return }  
506     guard let arrowNode = loadArrowModel() else { return }  
507     arrowNode.name = "ArrowNode"  
508     arView.scene.rootNode.addChildNode(arrowNode)  
509     self.arrowNode = arrowNode  
510 }  
511  
512 func updateArrowNodePosition() {  
513     guard let arrowNode = self.arrowNode,  
514         let currentNode = self.currentNode,  
515         let pointOfView = arView.pointOfView else { return }  
516  
517     let cameraTransform = pointOfViewsimdWorldTransform  
518  
519     let forward = simd_float3(-cameraTransform.columns.2.x,  
520                             -cameraTransform.columns.2.y,  
521                             -cameraTransform.columns.2.z)  
522  
523     let cameraPosition = simd_float3(cameraTransform.columns.3.x,  
524                                         cameraTransform.columns.3.y,  
525                                         cameraTransform.columns.3.z)  
526  
527     let arrowPosition = cameraPosition + forward * 0.2 + simd_float3(0, -0.1, 0)  
528     arrowNodesimdPosition = arrowPosition  
529  
530     let lookAtConstraint = SCNLookAtConstraint(target: currentNode)  
531     lookAtConstraint.isGimbalLockEnabled = true  
532     lookAtConstraint.worldUp = SCNVector3(0, 1, 0)  
533  
534     lookAtConstraint.influenceFactor = 1.0 // only move y axis  
535     arrowNode.constraints = [lookAtConstraint]  
536  
537     let currentEulerAngles = arrowNode.eulerAngles  
538     arrowNode.eulerAngles = SCNVector3(0, currentEulerAngles.y, 0)  
539 }
```

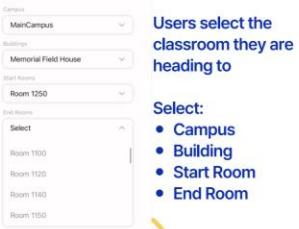


3D map



Expo Brochure

App Snippet



Users select the classroom they are heading to

- Select:
- Campus
 - Building
 - Start Room
 - End Room

The AR overlay displays a directional arrow in real time.

The bottom section provides an overview of the route on a 3D map.

The 3D map indicates the current position with a red dot.



An intuitive AR-powered app that uses 3D mapping and Dijkstra's Algorithm to provide seamless indoor navigation with real-time visual directions.

UT Classroom Finder

About our project

Introducing a smarter way to find your classroom

Problem

Navigating classrooms at the University of Toledo is challenging for students and visitors.

Features and Functionality

The app shows the user's current position, calculates optimal routes, and provides real-time AR directional guidance.

Solution

This solution streamlines indoor navigation, enabling users to reach their destinations quickly and efficiently.

Technologies & Approach



Augmented Reality Navigation

AR navigation uses a smartphone camera to overlay directional arrows in real-time, guiding users to their destination seamlessly. The system adapts dynamically to the user's movement, ensuring accurate and continuous guidance within the physical environment.



Dijkstra's Algorithm Path Finding

Dijkstra's algorithm calculates the shortest route between nodes by iteratively evaluating the smallest cumulative distance. Incorporating Euclidean distance as edge weights ensures spatial accuracy, while the system minimizes unnecessary transitions, such as floor changes, for user convenience.

Tools & Frameworks used



ARKit

- Augmented Reality Development



Swift

- Programming Language for the app



Xcode

- Integrated development environment



Autodesk Fusion 360

- Creating 3D models for floor plans



3D Map Integration

The navigation system incorporates detailed 3D models of buildings, providing a clear and interactive visualization of indoor spaces. Users can view routes superimposed on the 3D map, offering an enhanced spatial understanding of their surroundings.



3D Virtual Coordinate Mapping

Nodes and edges mapped with 3D coordinates ensure precise navigation and robust location tracking within buildings. This system minimizes errors by predefining key locations and pathways, aligning with the physical structure of the environment.

How does it work?

01 Select the initial and destination classrooms

02 Show AR navigation with directional arrows

03 View 3D overview of the entire route

04 Follow the direction & arrive at the destination!

Expo Poster



Group A-06 UT Classroom Finder



Project Problem	Key Features	Target
<p>Students find it challenging to find their classrooms at the beginning of each semester</p>	<p>Augmented Reality Navigation</p> <ul style="list-style-type: none">Real-time guidance with a pointing arrow and AR components <p>Dijkstra's Algorithm for Pathfinding</p> <ul style="list-style-type: none">Efficient route calculation using algorithms for shortest-path navigation <p>3D Map Integration</p> <ul style="list-style-type: none">Enhanced visualization with 3D building models <p>3D Virtual Coordinate Mapping</p> <ul style="list-style-type: none">Location reliability through the pre-mapped node and edge coordinates	<p>University of Toledo faculty and students finding their classrooms</p>
Goal	Tools Used	Team Members
<p>Develop an AR-based indoor navigation app that leverages 3D virtual mapping and advanced algorithms to guide users efficiently to their destinations within campus buildings.</p>	<p>Swift</p> <ul style="list-style-type: none">Programming language for building an iOS app <p>ARKit</p> <ul style="list-style-type: none">Apple's augmented reality(AR) framework <p>Xcode</p> <ul style="list-style-type: none">Integrated development environment to create iOS software <p>AutoDesl Fusion</p> <ul style="list-style-type: none">3D modeling software for creating maps	<p>Sanghak Ryu (CSET)</p> <p>Seungyoun Lee (CSET)</p> <p>Seokhyun Hong (CSET)</p>
Project Solution	App Snippet	Faculty Advisor
<ul style="list-style-type: none">User selects their initial and destination classroom numbers.Dijkstra's Algorithm calculates the most efficient route.The route is shown both in AR and on a 3D map.A dotted line and arrow in the AR view will guide users directly to their destinations.	<p>1. Select destination 2. Show the shortest route 3. Arrive!</p>	<p>Dr. Hong Wang</p>

Gantt Chart