Design Document

FIT AR Navigation App (FITARNA)

Vincenzo Barager vbarager2022@my.fit.edu

Dathan Dixon ddixon2022@my.fit.edu

Jacob Hall-Burns jhallburns2021@my.fit.edu

Ethan Wadley ewadley2022@my.fit.edu

Faculty Advisor: Eraldo Ribeiro eribeiro@fit.edu

Client: Florida Tech Library

Version 1.0

Florida Institute of Technology

9/12/2025

TABLE OF CONTENTS

1. Introduction

1.1. Purpose

This software design document describes the architecture and system design of the FIT AR Navigation App. The intended audience for this document would be our clients, Florida Tech Evans Library, our Faculty Advisor Eraldo Ribeiro, and the Team Members to use as a baseline.

1.2. Scope

The FIT AR Navigation App (FITARNA) is designed to help students and visitors with navigation and familiarization of Florida Tech's Evans Library, as well as lighten the workload of staff. The app will accomplish this through its Navigation, Tour, and pop-up modes.
The goals of FITARNA would be to:

- Simplify and Improve the Navigation of FIT Buildings
- Provide engaging and informative AR Tours
- Enhance user experience by making the library and any additional buildings easier to explore and learn about.

1.3. Overview

Section 2 System Overview - General overview of the system functionality and context.
Section 3 System Architecture - High-level architectural design through subsystems and components.
Section 4 Data Design - Provides a Description of how data will be saved and designed in the system.
Section 5 Component Design - Provides Examples of how each of the components may be designed.
Section 6 Human Interface Design - Overview of User Interface and front-end design.
Section 7 Requirements Matrix - Review of functionalities provided by FITARNA.

1.4. Reference Material

FIT AR Navigation App Software Requirements Document
Unity, AR Foundation, ARKit, ARCore, and language documentation.

1.5. Definitions and Acronyms

FITARNA - FIT AR Navigation App
FIT - Florida Institute of Technology
AR - Augmented Reality
SDK - Software Development Kit

POI - Point of Interest

2. SYSTEM OVERVIEW

Currently, locating specific rooms inside buildings can be confusing for anyone unfamiliar with the buildings. The FITARNA app will help users navigate these unfamiliar spaces quickly and effectively, or allow the user to become familiar with the space themselves. The app is built on Unity and integrates AR Foundation, ARKit, and ARCore for its primary functions, and will support both iOS and Android. The system will use device sensors such as the camera and GPS for positioning inside buildings and identification of Points of Interest (POI). The maps that the system uses to navigate will be stored locally for the least interference and fast operation. In Navigation Mode, users select a destination via a search bar or dropdown. The system calculates a path and overlays AR directions in real time. In Tour Mode, the system guides users through predefined library stops of all the library's most useful and interesting spaces alike. In either mode, AR Pop-ups will be displayed to users when they are near any POI to inform them about its use, history, or any other relevant information.

3. SYSTEM ARCHITECTURE

3.1. Architectural Design

3.2. Decomposition Description

User Interface will be responsible for the Main Menu, Navigation UI, Tour UI, Pop-up UI, and Search UI.

The Tour Manager will be responsible for the Tour Controller, Progress Tracker, Quiz Handler, and Voting Handler.

Navigation Manager will be responsible for the Path Calculator and Route Updater

The Pop-up System will be responsible for providing and managing the Pop-up data.

AR System will be responsible for managing and utilizing Unity data, AR location, and navigation.

Stored Data will be responsible for storing all app data locally.

3.3. Design Rationale

We chose a modular architecture to:

- Allow subsystems to update without updating others.
- Simplify testing, debugging, and ease of understanding.
- Support scalability when extending to new buildings.

4. DATA DESIGN

4.1. Data Description

The FITARNA is designed to rely on lightweight storage to aid navigation, tours, and pop-ups. All app data is stored in JSON format locally.

Data Sets:

Floor Data

- Contains floor data as a graph structure.
- Stores POI locations and navigation nodes.

Tour Data

- Contains data for the tour.
- Saves an ordered sequence of POIs
- Saves tour-specific pop-ups and quiz questions

Pop-up Data

- Contains data on the context and history found in the various pop-ups utilized in both Navigation and Tour Mode.

User Data

- Contains data for users.
- Stored locally on device
- Stores session state: Tour progress, user settings, navigation history.

4.2. Data Dictionary

poi_id (string) → Unique ID for each poi. Saved in the floor data set.

loc (array[float]) → position in AR space for object.

popup_id (string) → Unique ID for each pop-up. Saved in Pop-up data set.

popup_type (string) → “info”, “tour”, “quiz” or “poll”. Saved in Pop-up data set.

popup_text (string) → Text found in a pop-up. Saved in pop-up data set.

5. COMPONENT DESIGN

Example pseudocode for Navigation Manager:

```
function calculateRoute(start, destination):
   mapData = loadFloorLayout()
   route = shortestPath(mapData, start, destination)
   return route

function updateRoute(userPosition, destination):
   if userPosition deviates from the current route:
      recalculate route
   return updatedRoute
```

Example pseudocode for Tour Manager:

```
function startTour():
   loadTourStops()
   currentStop = 1
   displayRouteToStop(currentStop)

function onStopReached():
   showPopUp(currentStop)
   if userInteractsWithPopUp():
      currentStop = currentStop + 1
      displayRouteToStop(currentStop)
```

Example pseudocode for User Interface:

```
function displayMainMenu():
   showButton("Navigation")
   showButton("Tour")
   showButton("Pop-ups")
   showButton("Location")
   showButton("Help")

function onButtonClick(button):
   if button == "Navigation":
      NavigationManager.startNavigationUI()
   if button == "Tour":
```

```
    TourManager.showTourMenu()
  if button == "Pop-ups":
    PopUpSystem.activatePopUps()
  if button == "Location":
    displayLocationDropdown()
  if button == "Help":
    openHelpPage()
```

Examples of pseudocode for the AR System:

```
function startARSession():
  initializeCameraFeed()
  enableTracking()
  loadFeatureMaps()

function updateUserPosition():
  currentPose = getDevicePose()
  updateAnchors(currentPose)

function renderOverlay(type, position, data):
  if type == "arrow":
    drawArrowAt(position, data.direction)
  if type == "popup":
    drawPopupAt(position, data.content)
```

6. HUMAN INTERFACE DESIGN

6.1. Overview of User Interface

From the user's perspective, the FIT AR Navigation App (FITARNA) provides a simple and intuitive interface that makes exploring the Evans Library straightforward and engaging. Upon launching the app, the user is presented with a clean main menu that clearly separates the available features: Navigation, Tour, Pop-ups, Help, and Location settings. This ensures that first-time users can immediately understand the app's core functionality without requiring external instructions.

When using Navigation Mode, the user can easily find destinations through a search bar or dropdown selection. Once a destination is chosen, AR arrows and overlays appear directly on the live camera feed to guide the user to their destination. The interface provides real-time feedback by updating the route dynamically as the user moves through the building. Users also have the option to end navigation at any time using the back button, which returns them to the main menu.

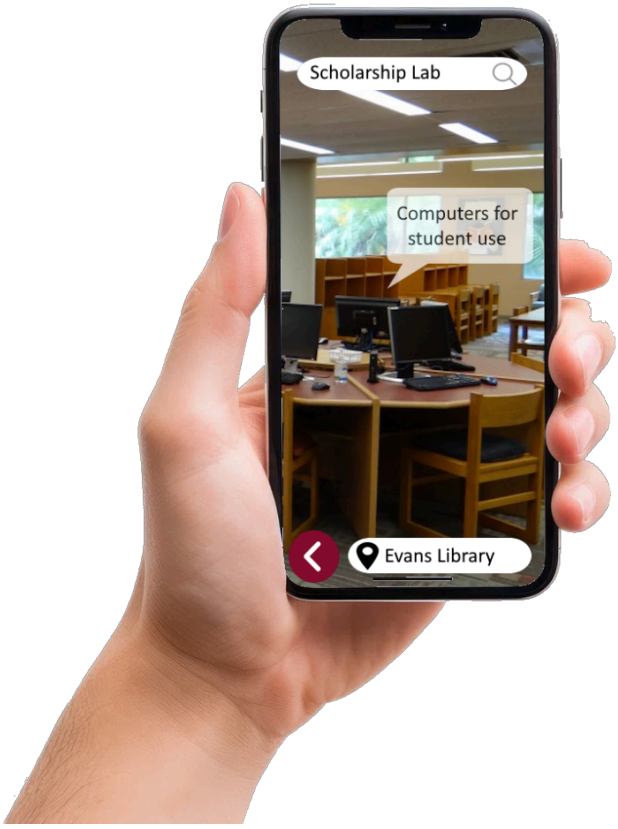
In Tour Mode, the app takes a more guided approach. The user selects a curated tour, after which the system displays a tour guide panel that shows the next destination. As the user reaches each stop, interactive pop-ups appear with contextual information, images, or quiz questions. These provide immediate feedback and encourage engagement, ensuring the tour is both informative and interactive.

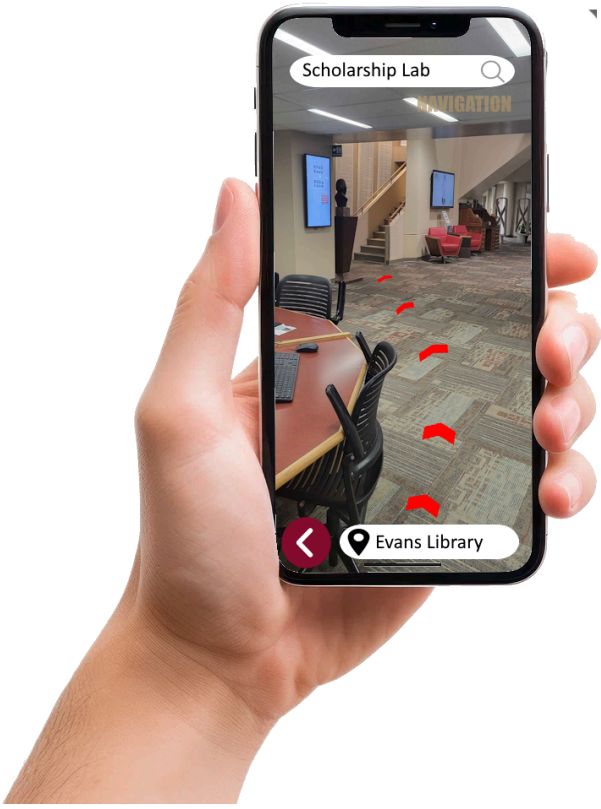
In Pop-up Mode, the system highlights nearby POIs as AR overlays. Users receive visual and textual information about their surroundings, and interactive elements allow them to open links, answer questions, or vote in polls. This mode provides a more exploratory experience compared to the structured navigation and tour modes.

6.2. Screen Images

6.3. Screen Objects and Actions

Main Menu





Help - Clicking this button from the main menu will bring you to a page containing the FITARNA Website, as well as emails for our team.

Location - clicking Location will display a drop-down menu where you can select your location from a set of select locations for use in the app's features.

Navigation - clicking Navigation will bring you to the AR Navigation screen in Navigation mode.

Tour - clicking Tour will bring you to a secondary menu where you can select a tour you would like to go on from a list of curated tours. Selecting a tour within this menu will launch the AR Navigation screen in Tour mode

.
Pop-ups - clicking Pop-ups will bring you to the AR Navigation screen in Pop-up mode.

AR Navigation

Current Location - Displays the selected location. The user can click on this to bring up a drop-down menu containing other locations, and will be able to change the location to any listed.

Back Button - When selected, it will bring the user back to the Main Menu screen.

Search Bar - In Navigation Mode, this object will be empty until clicked, where the user can then type in a keyword using the phone's keyboard to find what they are looking for in the given location.

Tour Guide - In Tour Mode, your next destination will be displayed within this object. Clicking it will show you the destinations of this tour and will allow you to skip ahead or go back to any location in the tour.

Pop-ups, in Pop-up Mode, will be displayed to the user when near and looking at POIs. These Pop-ups will display relevant information, features, context, and history. If the Pop-up contains a link, the user will be able to click on it to open the link in the default browser.

Navigation Arrows - When navigating to a destination, arrows will be displayed informing the user where to go to reach their desired destination.

Camera - The app will use the phone's camera to display the surroundings on the screen and use AR functionality.

7. REQUIREMENTS MATRIX

Main Menu/ Mode Selection - handled by User Interface

Destination Selection - handled by User Interface and Tour Manager

Navigation/AR Overlays - handled by User Interface, Navigation Manager, and AR systems

Track Location Effectively - handled by Navigation Manager and AR Systems

Real Time Route Updates - handled by Navigation Manager

End of Navigation - handled by User Interface and Navigation Manager

Display Pop-ups - handled by User Interface, Pop-up System, and AR Systems

Quiz Integration - handled by Tour Manager

Store and Call Relevant Data - handled by Stored Data