

DESIGN AND IMPLEMENTATION OF THE CAMPUS NAVIGATION
SYSTEM

BY

OGHAMOMWAN NATHAN OSAMUDIAMEN
(PSC/CSC/PSC1908902)

A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER
SCIENCE, FACULTY OF SCIENCE, UNIVERSITY OF BENIN, IN
PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE AWARD
OF THE BACHELORS OF SCIENCE (B.Sc.) DEGREE IN COMPUTER
SCIENCE

APRIL, 2024

DECLARATION

I hereby declare that this research project entitled “Design and Implementation of the Campus Navigation System” was written by me in the Department of Computer Science, under the supervision of Prof (mrs) Osubor. The information derived from the literature has been duly acknowledged in the work and a list of references has provided. No part of research was previously presented for any other degree or diploma at this or any other institution.

.....
OGHAMOMWAN NATHAN OSMUDIAMEN.

DATE.....

CERTIFICATION

This is to certify that this project entitled “Design And Implementation Of The Campus Navigation System” is the outcome of the research carried out by OGHAMOMWAN NATHAN OSAMUDIAMEN in the department of Computer Science, university of benin

.....

Prof. Mrs Osunbor

(Supervisor)

.....

Date

DEDICATION

This research is dedicated to God Almighty, whom through his words and spirit, encouraged me and has been a driving force that kept me moving, strong and determined

ABSTRACT

This research project provides a solution which would help enable the ease of navigating through a new campus environment as every year there are new students in every university. In this study, I highlighted the importance of the campus navigation system using University Of Benin, ugbowo, as a case study. Agile methodology was adopted for this project, and a few data collection method was adopted. This study evolved into a feasible navigation system, which would help ease the confusion that comes with entering a new university campus

ACKNOWLEDGEMENT

First and foremost, I want to thank God for this unending love and grace and providing me with the wisdom to go through the entire process. I truly appreciate my supervisor, PROF MRS OSUBOR, for helping me with my research to reach an admirable, commendable standard through their constructive criticism and adjustments. I specially thank my parents, Mr and Mrs Oghamomwan for their support, advice, and criticism during the course of my project.

TABLE OF CONTENTS

CHAPTER ONE	10
INTRODUCTION	10
1.1 Background of Study	10
1.3 Aim and Objectives	13
iii. Users will be able to navigate using augmented reality	13
1.4 Motivation	13
1.5 Research Questions	13
1.6 Significance of Study	14
1.7 Organization of Study	14
CHAPTER TWO	Error! Bookmark not defined.
LITERATURE REVIEW	Error! Bookmark not defined.
CHAPTER THREE	42
METHODOLOGY	42
CHAPTER FIVE	59
SUMMARY, CONCLUSION AND RECOMMENDATION	59
REFERENCES	62

LIST OF TABLES

Performance evaluation metric of model	29
--	----

LIST OF FIGURES

Figure3.1	27
Figure4.1	33
Figure4.2.....	
.....34	
Figure4.3	37
Figure4.4.....	
.....39	
Figure4.5.....	
.....40	

CHAPTER ONE

INTRODUCTION

1.1 Background of Study

The Campus Navigation System is a mobile or web-based application designed to help students, staff, and visitors navigate their university campus. Campus Navigation Systems typically have interactive maps that provide information about buildings, classrooms, offices, and other facilities on campus. It may also include information about nearby restaurants, shops, and other amenities, as well as directions on how to get from one location to another (Arengi *et al.*, 2018). The goal of the app is to provide easy-to-use tools for navigating the university campus, helping users easily find their way around and make the most of their time on campus. Campus Navigation System using ARToolKit, that is accessible via mobile devices that uses AR to overlay the information as images of the searched location on campus, especially for indoor navigations for lecture halls, tutorial rooms, laboratories, and offices which rarely are covered by the normal maps. (Noradila Nordin, 2021).

Typically, these systems feature interactive maps that are rich in detail, offering information on various campus facilities including buildings, classrooms, offices, and other essential areas. The maps are designed to be intuitive, allowing users to quickly locate their destinations and understand their surroundings. Additionally, these applications may offer real-time updates and information about campus events, available facilities, and changes in the campus layout, further aiding users in their navigation.

Moreover, modern Campus Navigation Systems extend their utility beyond the immediate campus by including information on nearby amenities such as restaurants, cafes, shops, and public transportation options. This comprehensive approach ensures that users have access to all necessary information, not just limited to the academic buildings but also encompassing the broader campus environment.

Research conducted by Lee and Lin (2018) emphasizes the significance of campus navigation systems, stating that "an effective navigation system can greatly improve the overall campus experience for students and visitors, reducing frustration and enhancing accessibility." Furthermore, Li and Li (2019) highlight the importance of user-centric design in campus navigation systems, stating that "the user interface and experience play a crucial role in the success and adoption of such systems."

The implementation of a robust Campus Navigation System has numerous benefits. It enhances the campus experience by providing a reliable tool for finding locations and planning routes efficiently. This not only saves time but also reduces stress and confusion, particularly for new students and visitors who are unfamiliar with the campus layout. Additionally, by integrating information about nearby amenities and real-time updates, these systems can significantly improve the quality of life on campus.

From an administrative perspective, these systems can also provide valuable data on campus traffic patterns, building usage, and areas that may need improvements or additional signage. This data can be instrumental in campus planning and development, ensuring that resources are allocated effectively and that the campus infrastructure evolves to meet the needs of its users.

By integrating advanced technologies and focusing on user experience, Campus Navigation Systems aim to provide comprehensive, intuitive, and accessible solutions for campus navigation, ultimately enhancing the campus experience for all users.

1.2 Statement of Problem

The difficulty of navigating and finding information about a large university campus is the issue that The Campus Navigation System aims to address. It can be hard for students, visitors, and staff to get around the campus and find what they need because there are so many buildings, facilities, and outdoor areas. This can make it take more time and effort to find information or places, which can make people unhappy and make them work less efficiently. By providing a centralized and simple-to-use platform for finding information about the university campus, The Campus Navigation System addresses this issue. Users can quickly and easily find what they need and get to their destination thanks to features like turn-by-turn directions, real-time updates, and search functionality.

1.3 Aim and Objectives

The objectives of this research are to;

- i. Design a mobile based app to show directions from one building to another
- ii. Create a search bar for users to input directions
- iii. Users will be able to navigate using augmented reality

1.4 Motivation

As a former new student, I can relate to the challenges of exploring a new school grounds, Not knowing where the faculty officers' office is, Not knowing where the administrative building is, and as a result of this lack of information, i missed a few classes also due to this, thus motivating me to design an easy-to-use navigation system that offers information about the campus as well as directions to several offices, classrooms and canteens.

1.5 Research Questions

- i. Which features and functions would have the greatest impact on the campus community?
- ii. How can a student/staff/visitor easily find buildings using the application?
- iii. What are the specific navigation needs and challenges faced by students, staff and visitors on campus?
- iv. What are the potential challenges and limitations in implementing the campus navigation system and how can they be addressed to ensure successful deployment and adoption?

1.6 Significance of Study

University campuses are often vast and complex, consisting of numerous buildings, departments, and facilities. The navigation system provides users with an interactive and detailed map, allowing them to navigate the campus efficiently and locate specific destinations without getting lost. It saves time and reduces frustration associated with finding unfamiliar locations. For new students, faculty, and visitors, navigating a university campus can be daunting. The navigation system serves as an orientation tool, helping newcomers become familiar with the campus layout and find their way around. It provides a sense of confidence and independence, facilitating a smooth transition into the university environment.

1.7 Organization of Study

Chapter One: This shows the introduction into the background of the project, The problem statement and the research aim and objectives.

Chapter Two: Mainly focuses on review and exploration of similar works and their limitations, also reviewing applications and published articles relating to the Campus Navigation System, It also explores the relevant concepts and theories for the project.

Chapter Three: This chapter discusses the methodology of building the project, giving an overview insight on research approach, It also explains the data

collection methods used and the analysis techniques applied to develop the project.

Chapter Four: This chapter focuses on the designing, testing and implementation of the user interface and user experience, Augmented Reality AR, developing the navigation and routing algorithms, incorporating places of interest, It provides a detailed documentation on the step by step process of implementing each function and algorithm into the project.

Chapter Five: This is the conclusion of the project and it gives a summary of the outcomes of each results shown during the testing phase and the contributions made to improve user experience, It also talks about its strengths, limitations and recommendations for future enhancements, It also has a list of cited sources and references used in the study.

CHAPTER TWO

LITERATURE REVIEW

This chapter is dedicated to presenting, pinpointing, and discussing the pertinent scholarly works and research that fall within the domain of this study. It thoroughly examines the existing contributions in the field, clarifying what is established and widely acknowledged, what represents the forefront of academic inquiry, and the present dynamics of the research topic. The chapter further highlights the rapid expansion and growing importance of the study area. Research findings indicate that Campus Navigation Systems have markedly enhanced the ability to navigate on-campus environments. These systems facilitate improved direction-finding, orientation, and navigation by creating a collaborative platform where students can exchange information, report any incidents, and easily access various campus services (Beheshti et al., 2023).

2.1 Review of Related Works

In this segment, we delve into an assortment of works that are closely related to Campus Navigation Systems. The scope of this review covers applications that, while catering to various contexts and catering to distinct user needs, bear resemblances to Campus Navigation Systems in terms of functionality, design, and the overall user experience. These comparisons are drawn from similar mapping and navigation tools and provide valuable insights into how these systems are implemented and utilized in different settings (Unger and Chandler, 2023).

Highlighted below are applications and research endeavors that relate directly to the Campus Navigation System framework:

- **City Navigation Systems:** Notable examples include Google Map and Apple Map, which are designed to facilitate navigation across urban landscapes. These platforms are equipped with extensive mapping capabilities, providing users with detailed geographic visuals and comprehensive navigational data. They offer real-time traffic updates and tailored routing options for different modes of transport, such as walking, cycling, and driving. These systems are integral to understanding how navigation tools can be adapted to specific environmental and logistical requirements, thereby offering practical insights for enhancing campus-based navigation solutions (Namoun et al., 2023).

Users have the ability to search for specific locations, identify various points of interest, and obtain information on public transport systems. The distinct advantage of city navigation systems stems from their comprehensive global coverage, their capability to integrate multiple transportation options, and their proficiency in delivering current traffic updates. This extensive functionality makes them invaluable tools for urban dwellers and travelers alike (Djahel et al., 2014).

Comparative Usability Evaluation of Mobile Map Applications

In a detailed study by Patarada Thanachan et al. (2021), the usability of different mobile map applications was rigorously assessed. The primary goal of this research was to evaluate the strengths and weaknesses of these applications focusing on aspects such as user interface, navigation features, and overall user experience. The methodology of this evaluation was thorough, incorporating a variety of techniques including user surveys, interviews, and practical usability tests. These approaches were instrumental in collecting a wealth of data and feedback directly from the users.

The study underscored the critical importance of adhering to user-centered design principles when developing mobile map applications. It highlighted the need for clear and effective visual representations and efficient navigation tools to significantly improve user experience. By focusing on these elements, the study provided key insights that can aid developers and designers in enhancing the functionality and user-friendliness of mobile mapping applications. The findings from this evaluation contribute valuable knowledge to the ongoing enhancement of mobile navigation technology, guiding future developments to better meet the needs of users (Patarada Thanachan et al., 2021).

A Web-Based Campus Navigation System with Mobile Augmented Reality Intervention, authored by Noradila Nordin et al. in 2021, delves into the creation of an innovative solution aimed at enhancing navigation experiences

within university campuses. The study addresses the shortcomings prevalent in conventional navigation systems and underscores the untapped potential lying within web-based platforms and augmented reality (AR) technologies.

The authors meticulously outline the development journey of their navigation system, which involved the creation of a robust map database, seamless integration of AR functionalities, and the crafting of an intuitive user interface. Through rigorous user testing, the efficacy of the system was evaluated, garnering positive feedback from participants who lauded its interactive and immersive AR features.

Central to the discussion are the myriad benefits offered by the system, including heightened navigation efficiency, augmented user engagement, and an enriched campus exploration experience. By seamlessly blending web-based platforms with AR interventions, the research not only addresses the pressing need for improved navigation solutions but also sheds light on the transformative potential of emerging technologies in reshaping our everyday experiences.

In essence, Noradila Nordin et al.'s work constitutes a significant contribution to the realm of campus navigation, offering valuable insights into the symbiotic relationship between web-based platforms and AR technologies in optimizing navigation experiences for diverse user groups.

The limitations inherent in conventional navigation technologies within localized areas often hinder their ability to provide precise and comprehensive

guidance, particularly when it comes to pinpointing specific buildings and accessing pertinent information about their functionalities. In response to this challenge, Qin Yong-Xu et al. (2013) embark on a thorough investigation into the realms of augmented reality (AR) and navigation technologies, aiming to revolutionize the way campus navigation is approached, particularly on mobile platforms.

By leveraging the power of augmented reality, this study pioneers a novel approach to internal campus navigation, transforming the user's mobile device into a sophisticated tool capable of seamlessly interfacing with the physical environment. Through the integration of AR technology, users gain access to precise location data and detailed information about every facet of the campus, empowering them to navigate with unprecedented accuracy and efficiency.

The fusion of navigation systems with augmented reality not only enhances the precision of local applications but also imbues them with a humanized touch, catering to the diverse needs and preferences of users. By overlaying digital information onto the physical environment in real-time, the navigation experience becomes inherently more intuitive and convenient, facilitating seamless interaction between users and their surroundings.

In essence, (Qin Yong-Xu et al.(2021)) pioneering work represents a paradigm shift in the realm of campus navigation, harnessing the transformative potential

of augmented reality to redefine the way individuals navigate and interact with their environment. By bridging the gap between technology and the physical world, this research paves the way for a future where precision, convenience, and human-centric design converge to create truly immersive navigation experiences.

2.2 Summary Of Related Works And Gap Analysis

In summarizing the related works and conducting a gap analysis, it becomes evident that recent advancements in location-based mapping and navigation have undoubtedly propelled the field forward. These studies have contributed significantly to enhancing the accuracy, efficiency, and user experience of Campus Navigation Systems. However, despite these strides, several key areas emerge where further improvement is warranted:

Indoor Navigation: While outdoor navigation has seen substantial progress, indoor navigation remains a challenging frontier. Many existing systems struggle to provide reliable guidance within complex indoor environments such as university buildings or large facilities. Improving indoor navigation capabilities is crucial for ensuring seamless navigation from outdoor to indoor spaces, thus enhancing overall user experience and accessibility.

Accessibility Features: Existing navigation systems often lack comprehensive accessibility features tailored to individuals with diverse needs, including those with disabilities or special requirements. Integrating features such as audio cues, tactile feedback, and customizable route options can significantly enhance the

inclusivity and usability of Campus Navigation Systems, ensuring they cater to the needs of all users.

Real-time Updates and Dynamic Information: Many navigation systems rely on static maps and pre-existing data, which may become outdated or inaccurate over time. Incorporating real-time updates and dynamic information feeds can ensure that users receive the most current and relevant navigation data, including changes in building layouts, construction zones, or event locations, thus enhancing the system's reliability and usefulness.

Multi-modal Navigation: Users often employ multiple modes of transportation, including walking, cycling, driving, and public transit, to navigate campus environments. However, existing navigation systems may lack seamless integration with these diverse transportation modes, leading to disjointed user experiences. Implementing multi-modal navigation capabilities can streamline the navigation process, providing users with optimized routes that accommodate their preferred mode of transportation.

User Interaction and Customization: Personalization and user interaction are essential aspects of any navigation system. Providing users with intuitive interfaces, customizable preferences, and interactive features can enhance user engagement and satisfaction. Additionally, incorporating feedback mechanisms and user-generated content can empower users to contribute to the improvement and refinement of the navigation system over time.

2.2.1 Campus-Specific Features

Academic Building Identification: A fundamental aspect of campus navigation is the ability to accurately identify and locate academic buildings. Navigation systems should provide comprehensive maps that highlight the locations of lecture halls, laboratories, seminar rooms, and other academic facilities. Clear labeling and intuitive icons can help users easily distinguish between different types of buildings.

Departmental Navigation: Given the diverse array of academic departments within universities, navigation systems should offer department-specific navigation options. Users should be able to search for and navigate to individual departments within academic buildings, facilitating efficient movement between classes, meetings, and departmental events.

Class Schedules Integration: Integrating class schedules into the navigation system can greatly enhance its utility for students and faculty. By syncing with the university's academic database, the navigation app can display personalized routes based on users' class schedules, ensuring they arrive at their destinations on time. Additionally, real-time updates and notifications can alert users to any changes or cancellations in their schedules.

Campus Services Directory: Beyond academic facilities, campuses are home to a wide range of support services and amenities. Navigation systems should include a comprehensive directory of campus-specific services, including libraries, research centers, student support offices, dining facilities, and

recreational areas. Users should be able to easily locate and navigate to these essential resources as needed.

Customizable Preferences: Recognizing that different users may have varying navigation preferences and priorities, Campus Navigation Systems should offer customizable settings. Users should be able to tailor their navigation experience based on factors such as preferred walking speed, accessibility requirements, and route preferences (e.g., shortest route vs. scenic route). This customization enhances user satisfaction and ensures that the navigation system caters to the individual needs of each user.

2.2.2 Indoor Mapping and Navigation

In the context of university campuses, where sprawling complexes encompass numerous buildings and facilities, the integration of indoor mapping and navigation functionalities within Campus Navigation Systems assumes paramount importance. Addressing this need necessitates a concerted effort to leverage indoor mapping technologies, thereby facilitating seamless navigation within intricate indoor environments. The following points elucidate the significance and key considerations associated with indoor mapping and navigation within campus settings:

Detailed Floor Plans: Campus Navigation Systems must offer users access to comprehensive and accurate floor plans for individual buildings. These floor plans should delineate the layout of each floor, including the arrangement of rooms, corridors, staircases, elevators, and other relevant features. By providing detailed floor plans, users can navigate with confidence, easily

identifying the locations of classrooms, offices, laboratories, and other indoor destinations (Torres-Sospedra et al., 2015).

2.2.3 Campus Events and News Integration

In the endeavor to enhance the overall campus experience, the integration of event calendars, campus news updates, and notifications within Campus Navigation Systems emerges as a promising avenue. This multifaceted feature not only facilitates navigation but also fosters a sense of community engagement and connectivity. The following points elucidate the significance and potential benefits of integrating such functionalities:

Comprehensive Event Calendars: By incorporating event calendars into Campus Navigation Systems, users gain access to a centralized platform showcasing a diverse array of campus events, including academic lectures, seminars, workshops, cultural performances, and social gatherings. This feature enables users to stay informed about upcoming events and plan their schedules accordingly, fostering a vibrant campus culture and encouraging participation in various activities.

Timely Campus News Updates: Campus Navigation Systems can serve as a conduit for disseminating timely news updates and announcements relevant to the university community. Whether it be campus-wide initiatives, policy changes, academic deadlines, or emergency alerts, users can receive pertinent information directly through the navigation app, ensuring timely

communication and fostering a sense of transparency and community awareness.

(Graeff, 2014).

2.3 Discussion of Project Specific Concepts and Theories

The concept of Augmented Reality (AR) plays a pivotal role in shaping the user experience of Campus Navigation Systems. AR technology overlays digital information onto the physical environment, providing users with contextually relevant information and enhancing spatial awareness (Azuma, 1997). Integrating AR functionalities into navigation systems offers users immersive and interactive experiences, allowing them to visualize navigation instructions within the real-world environment (Billinghurst et al., 2015).

Moreover, the Theory of Planned Behavior (TPB) provides valuable insights into user behavior and decision-making processes regarding navigation system adoption and usage (Ajzen, 1991). According to TPB, user intentions to use a system are influenced by their attitudes, subjective norms, and perceived behavioral control. By understanding these factors, developers can design navigation systems that align with user preferences and encourage widespread adoption.

2.3.1 Human-Computer Interaction (HCI) Theory

HCI theory focuses on understanding the interaction between humans and computer systems. In the context of the Campus Navigation System, HCI theory provides insights into designing an intuitive and user-friendly interface

(Pitale and Bhumgara, 2019). By applying principles such as user-centered design, usability testing, and user feedback, the app can be tailored to meet the needs and preferences of its target users. HCI theory also emphasizes the importance of considering cognitive factors, such as information processing and decision-making, to ensure that the app provides a seamless and efficient navigation experience (Fleury and Chaniaud, 2024).

2.3.2 Location-Based Services (LBS) Theory

The theory of Location-Based Services (LBS) is centered on the strategic use of geographic location information to deliver personalized and context-aware services to users. This approach is particularly pivotal in applications like the Campus Navigation System, where the effectiveness of the system hinges on its ability to utilize real-time location data accurately (Schiller & Voisard, 2004).

Application of LBS in Campus Navigation System

In the context of a Campus Navigation System, LBS theory underpins the operational framework that uses GPS technology and other location-sensing methods to track the real-time position of users. By leveraging LBS, the application can provide several critical functionalities:

- **Dynamic Positioning:** The system continuously updates the user's location on a virtual map, facilitating easy navigation through the campus.

- **Proximity Services:** It offers information about nearby amenities, such as libraries, lecture halls, cafeterias, and restrooms, enhancing the campus experience.
- **Optimal Routing:** The system suggests the shortest and most convenient routes to the user's desired destinations, which is especially beneficial during peak campus hours or for new students and visitors (Schiller & Voisard, 2004).

Importance of Privacy and Data Security

A core component of LBS theory is the emphasis on privacy and data security. As users disclose their real-time location to access various services, it becomes imperative to:

- **Control User Data:** Ensure that users have substantial control over their location data, allowing them to choose when and how this information is shared.
- **Secure Data Transmission:** Implement robust security protocols to protect the transmission and storage of sensitive location data against unauthorized access.
- **Transparent Data Usage Policies:** Clearly communicate to users how their data is used, stored, and shared within the app ecosystem (Dudhane & Pitambare, 2015).

Challenges and Considerations

Implementing LBS in a Campus Navigation System also involves addressing several challenges:

- **Accuracy of Location Data:** Ensuring the precision of GPS and other location technologies, particularly in densely built areas or indoors.
- **User Interface Design:** Developing an intuitive user interface that effectively displays location-based information without overwhelming the users.
- **Integration with Campus Infrastructure:** Coordinating with campus management to integrate the system seamlessly with existing infrastructure and information systems (Schiller & Voisard, 2004).

2.3.3 Information Visualization and Design

Information visualization and design encompass a set of principles and techniques that are crucial for presenting complex data and information in a visually engaging and comprehensible manner. These principles are especially relevant in applications like the Campus Navigation System, where the clarity of data presentation directly impacts user experience.

Visualization Techniques in Campus Navigation System

In the design of a Campus Navigation System, information visualization theories play a pivotal role in the development of user interfaces that are not only functional but also intuitive:

- **Interactive Maps:** These are essential tools that allow users to visually interact with the campus layout. Features like zooming and panning enable users to explore the campus map in detail (Ware, 2013).
- **Visual Representations of Campus Buildings and Points of Interest:** Using distinctive icons, color coding, and architectural silhouettes helps users quickly identify buildings and key locations (Tufte, 2001).
- **Labels and Signage:** Clear labeling is critical for guiding users effectively around the campus. The use of legible typography and strategic placement of signs ensures that information is easily accessible (Keller & Tergan, 2005).

Importance of Effective Design

Effective design in information visualization not only enhances aesthetic appeal but also improves functionality by:

- **Enhancing Usability:** Well-designed visual elements make the navigation system user-friendly, especially for new students and visitors who are unfamiliar with the campus.
- **Reducing Cognitive Load:** By organizing and presenting information efficiently, the system helps users process campus layouts and navigational data with minimal effort (Tufte, 2001).

- **Facilitating Quick Decision-Making:** Effective visualization aids users in making quick decisions about routes and destinations based on clearly presented data (Ware, 2013).

Challenges and Considerations

Implementing effective information visualization in a Campus Navigation System involves several challenges:

- **Data Integration:** Ensuring that real-time data and static information are seamlessly integrated into the visual design.
- **Accessibility:** Designing visualizations that are accessible to all users, including those with disabilities, which might require the incorporation of features like audio descriptions or high-contrast visuals (Keller & Tergan, 2005).
- **Adaptability:** The system must be flexible enough to update visual elements as the campus evolves, such as changes in building use or construction sites.

2.3.4 Mobile Application Development Frameworks

The development of a mobile application, particularly for projects like the Campus Navigation System, greatly benefits from leveraging established mobile application development frameworks. These frameworks provide a

comprehensive toolkit for developers, enabling the creation of robust, efficient, and scalable mobile applications.

Overview of Popular Mobile Development Frameworks

- **React Native:** Developed by Facebook, React Native allows for the development of mobile apps using JavaScript and React. It offers the benefit of 'learn once, write anywhere,' enabling developers to use the same code for both iOS and Android platforms, which can significantly speed up the development process (Facebook, 2021).
- **Flutter:** Created by Google, Flutter is known for its fast rendering and expressive and flexible designs. It uses the Dart programming language and provides a rich set of pre-designed widgets that help in creating visually attractive and responsive user interfaces (Google, 2021).
- **iOS/Android Native Development:** Native development tools like Swift for iOS and Kotlin or Java for Android provide direct access to device-specific functionalities. This approach is optimal for applications requiring intensive device interaction and high performance, as it allows developers to optimize the app specifically for each platform (Apple, 2021; Android, 2021).

Advantages of Using Development Frameworks

- **Pre-built Components:** Most frameworks come with a library of pre-built components that can be easily integrated into any application. This

includes items like navigation elements, buttons, and form controls, which simplify the development process.

- **Cross-Platform Compatibility:** Frameworks like React Native and Flutter allow developers to create applications that run seamlessly on both iOS and Android devices. This not only reduces development time but also ensures a consistent user experience across platforms.
- **Access to Device Features:** These frameworks provide APIs that allow apps to easily access device features such as GPS, cameras, and accelerometers, which are essential for applications like a Campus Navigation System.

Challenges in Mobile Application Development

While frameworks provide numerous advantages, they also present challenges that need to be addressed:

- **Performance Considerations:** While cross-platform frameworks offer significant development speed, they might not always match the performance of native apps. Critical applications may require native development to achieve optimal performance.
- **Framework Limitations:** Each framework has its limitations and learning curve. For example, while Flutter excels in UI fluidity, it may lack some of the community support and libraries available to React Native.

- **Maintenance and Updates:** Keeping the application updated with the latest versions of these frameworks and ensuring compatibility with new device versions can be challenging.

2.3.5 User Experience (UX)

User Experience (UX) design theory is pivotal in creating applications that are not only functional but also enjoyable and intuitive to use. In the context of the Campus Navigation System, UX design is essential for ensuring that the application is effective in helping users navigate the campus environment smoothly and efficiently.

Key Components of UX Design in the Campus Navigation System

- **User Research:** This foundational step involves understanding the specific needs, behaviors, and expectations of the app's target users, which include students, faculty, and campus visitors. Techniques such as surveys, interviews, and usability testing are critical for gathering insights that drive the design process (Norman & Nielsen, 2021).
- **Information Architecture:** Organizing and structuring content within the app in a logical and accessible way is crucial. This includes defining how the information is grouped, how users navigate from one page to another, and ensuring that important features are easy to find and use (Rosenfeld & Morville, 2017).

- **App Responsiveness:** Ensuring the app is responsive across various devices and platforms enhances user satisfaction. This means the app should perform well on different screen sizes, orientations, and operating systems without losing functionality or aesthetic appeal.
- **Continuous Feedback and Iteration:** UX design is an ongoing process that involves continuously gathering user feedback through beta testing and real-world usage. This feedback should be systematically analyzed and used to make iterative improvements to the app, ensuring that it evolves in line with user needs and technological advancements.

The Role of Empathy in UX Design

Empathy is a core component of UX design, as it involves understanding and addressing the actual needs and emotions of users. In the Campus Navigation System:

- **Understanding User Goals:** By empathizing with users, designers can understand what students and staff truly need from a navigation system, whether it's quick access to class schedules, finding the shortest route to a department, or locating amenities.
- **Tailoring Features to User Needs:** The system should offer customizable features such as adjustable route suggestions for accessibility needs, preferred paths based on user's location history, and

possibly, integration with other campus services like event schedules or emergency alerts.

Challenges in UX Design

Implementing effective UX design in a mobile application like the Campus Navigation System also involves addressing several challenges:

- **Diverse User Base:** Designing an app that meets the varied needs of a diverse campus population can be complex. Different users may require different functionalities, from international students needing language support to physically disabled individuals requiring accessible routes.
- **Balancing Aesthetics and Functionality:** While a visually appealing design is important for user engagement, it must not come at the expense of functionality. The challenge lies in creating a balanced interface that is both attractive and practical.

2.4 Definition Of Terms

2.4.1 Campus Navigation System

A Campus Navigation System is an advanced software application specifically crafted to assist users in efficiently navigating through a campus environment.

This technology serves as an essential tool for students, faculty, staff, and visitors, helping them locate various campus locations and resources with ease and precision.

Core Functions of a Campus Navigation System

- **Real-Time Directions:** The system offers dynamic, turn-by-turn navigation guidance that updates in real-time. This feature is crucial for efficiently managing one's way around often sprawling and complex campus layouts.
- **Interactive Maps:** It provides detailed, scalable maps that users can interact with to explore different parts of the campus. These maps often include features such as zoom and rotate to help users view the campus from various perspectives.
- **Building and Facility Descriptions:** Each building and facility within the campus is cataloged in the system with comprehensive details. This can include information such as operating hours, available services, departmental offices, and even room numbers.
- **Landmarks and Points of Interest:** The system highlights significant landmarks and points of interest around the campus, which might include art installations, historical markers, gardens, and common areas. This not only aids in navigation but also enriches the campus experience.

2.4.2 Routing Algorithm

A routing algorithm is a sophisticated computational technique designed to ascertain the most effective path or route between two or more points within a network or graph. This algorithm is crucial for systems that require dynamic

pathfinding capabilities, such as navigation systems in urban environments, logistics, and, notably, campus navigation systems.

Application in Campus Navigation System

In the context of a Campus Navigation System, routing algorithms play a pivotal role by calculating the most efficient and convenient paths for users to navigate from one campus location to another. This involves considering various factors:

- **Path Length:** The algorithm typically prioritizes the shortest possible route to minimize travel time or distance.
- **Traffic Conditions:** In a dynamic campus environment, the algorithm might also consider pedestrian traffic conditions, adjusting routes during peak times to avoid overcrowded areas.
- **Accessibility:** Routing algorithms can be tailored to accommodate accessibility needs by suggesting routes that are wheelchair accessible or avoiding paths with steep grades or obstacles for those with mobility impairments.

2.4.3 Augmented Reality (AR)

Augmented Reality (AR) is an advanced technology that integrates digital information with the user's environment in real time. Unlike virtual reality, which creates a wholly artificial environment, AR uses the existing

environment and overlays new information on top of it. This technology can greatly enhance the way users interact with their surroundings, providing a more immersive and interactive experience.

Integration into Campus Navigation Systems

In the context of a Campus Navigation System, AR can transform the navigation experience by overlaying visual aids directly onto the real-world view through a user's mobile device. This application of AR technology can significantly enhance the utility and user-friendliness of navigation aids:

- **Visual Cues and Directions:** AR can project arrows, paths, or specific directions onto the live camera feed, guiding users along their walking route in an intuitive manner.
- **Labels and Information:** Information about buildings and landmarks can be displayed as floating texts or icons when users point their camera at these locations, providing context-relevant information without the need for physical signage.
- **Interactive Features:** AR can enable interactive features such as clickable virtual signs or information panels that appear when a user points their device at a building or feature, offering detailed insights or historical data.

2.4.4 User Interface (UI)

The User Interface (UI) encompasses the visual and interactive elements of a software application that users engage with directly. It plays a critical role in shaping the user experience by making software accessible, intuitive, and efficient. In the context of a Campus Navigation System, the UI is essential for facilitating easy and effective navigation through a complex campus environment.

Key Components of UI in a Campus Navigation System

- **Maps:** The centerpiece of the navigation UI, maps provide a visual representation of the campus. They are typically interactive, allowing users to zoom, pan, and click on specific features to get more information.
- **Search Bars:** These allow users to quickly find specific locations or points of interest within the campus by entering text queries.
- **Navigation Controls:** Elements such as directional arrows, zoom sliders, and "locate me" buttons help users orient themselves and adjust the map to fit their needs.
- **Icons and Buttons:** Clear and intuitive icons and buttons facilitate easy access to the system's features, like setting destinations, choosing routes, and accessing building information.

2.4.5 User Experience (UX)

User Experience (UX) is a broad term that refers to the holistic experience and satisfaction a user derives from using a software application or system. It encompasses all aspects of the end-user's interaction with the company, its services, and its products. In the context of a Campus Navigation System, UX is paramount in ensuring that the system is not only functional but also intuitive and enjoyable to use.

UX Components in a Campus Navigation System

- **Ease of Use:** The system should be simple to navigate, allowing users to achieve their goals with minimal effort. This involves intuitive navigation paths, clear labeling, and straightforward interfaces.
- **Efficiency:** Users should be able to perform tasks quickly, such as searching for a specific location or getting directions. The system should respond swiftly to user inputs and provide accurate, timely information.
- **Aesthetics:** The visual design of the system should be pleasing and coherent, which enhances user engagement and satisfaction. Aesthetically pleasing design can make the application feel more welcoming and can increase usage rates.
- **Accessibility:** The system must be accessible to all users, including those with disabilities. This means incorporating features like text-to-speech for visually impaired users or ensuring the app is navigable via keyboard for those who cannot use a touchscreen.

CHAPTER THREE

METHODOLOGY

3.1 Methodology

When implementing a new system, selecting the right methodology is crucial for aligning with the project's specific needs, goals, and the resources available. Agile methodology emerges as a superior choice for projects that prioritize significant user input and aim for high usability. This approach centers around an interactive development process where user feedback is integral, enabling continuous and iterative improvements. Agile's core principles support a highly flexible and responsive development environment, which is perfect for projects with dynamic requirements that may evolve over time.

In practice, Agile encourages frequent collaboration and revisions, which help to ensure the developed system is closely tailored to user needs and preferences. It does this by breaking down the project into smaller, manageable units of work, known as sprints, which allow teams to focus on delivering specific features or enhancements, evaluate their success, and then adjust plans quickly and efficiently based on real feedback. This method not only increases the

adaptability of the development team but also fosters a sense of ownership and closer alignment with the user's expectations.

3.2 Location of Study

This study is conducted in University Of Benin, Ugbowo, Edo State, Nigeria

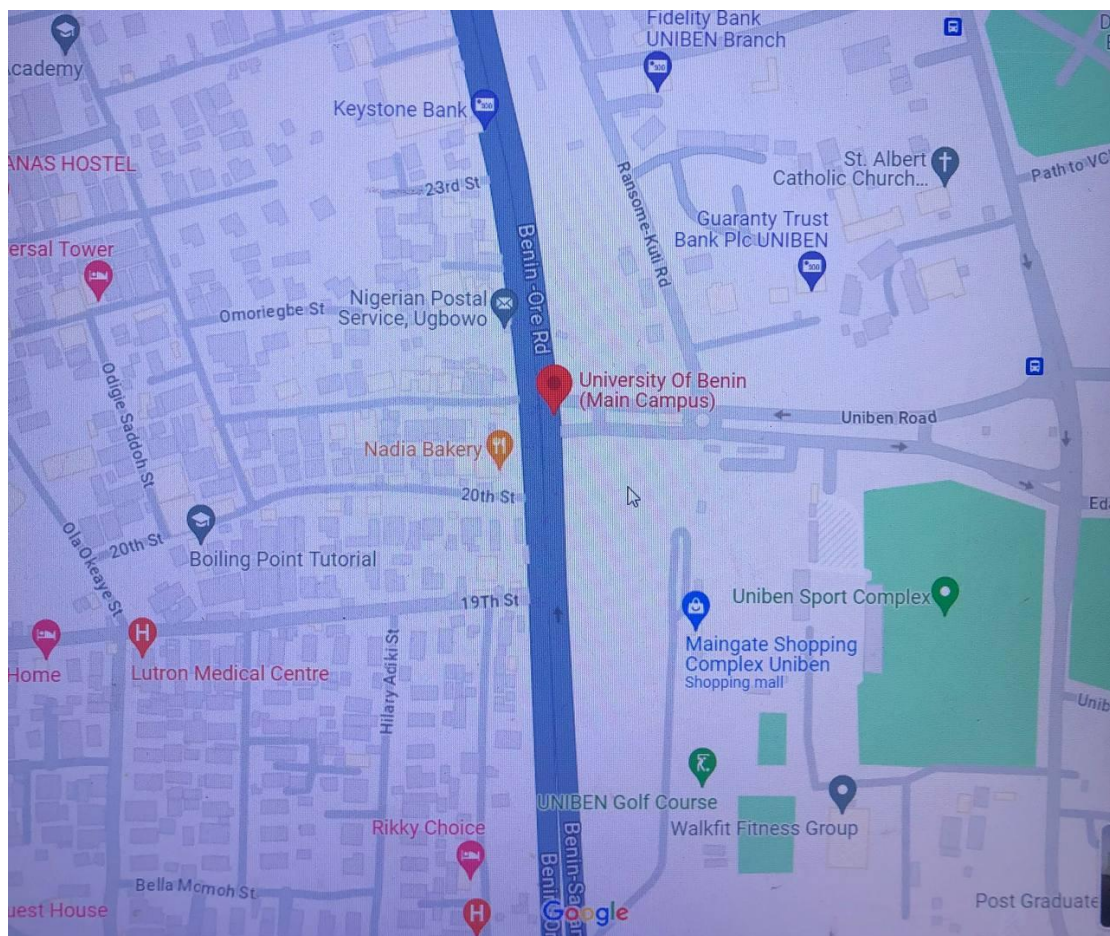


Fig 3.1 Image showing where the University Of Benin is located.

3.3 Method of data collection

The choice of data collection method for the campus navigation system would depend on the specific research objectives and the type of data needed. The research procedures are: Surveys, Documentation and Observation.

Observing users interacting with the app in real-time can provide valuable insights into their behavior, preferences, and challenges. By observing users navigate through the app, researchers can identify usability issues, areas of confusion, and potential improvements. Collecting and analyzing app usage data can provide valuable quantitative insights. This can include popular features, and user paths within the app. Analytics tools and tracking technologies can help capture this data.

3.4 Analysis of existing systems

Every system is composed of some essential components, which are known as modules or subsystems. In certain scenarios, to identify how a system works may require the knowledge of the various modules or subsystems. For this to be achieved, every module is evaluated to see if there is an vulnerability present

3.4.1 Google Maps

Google Maps is a widely used navigation system that offers detailed maps, turn-by-turn directions, and real-time traffic information. While it covers many university campuses, its focus is on general mapping rather than specific campus features. It may lack detailed information about buildings, amenities, and campus-specific points of interest.

3.4.2 Augmented Reality(AR) Apps

Augmented Reality(AR)-based apps use the device's camera and sensors to overlay digital information onto the real-world view, providing a more immersive navigation experience. Users can see building names, directions, and additional information directly on their screens. AR offer a unique and engaging way to explore campuses but may require specific hardware capabilities.

3.5 Problems of existing systems

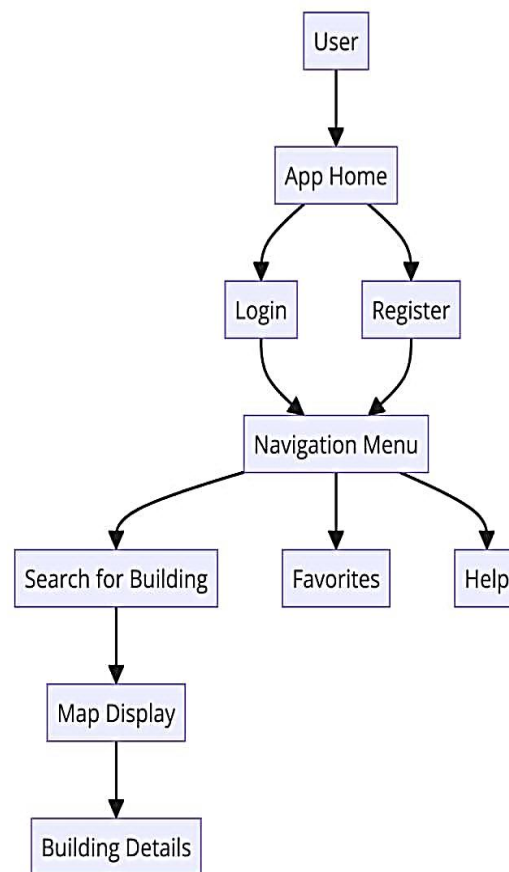
From the analysis carried out through documentation and observation of some of the systems, here are some problems I noted;

- i. University campuses are dynamic environments that undergo changes in building names, locations. And cause of this the existing systems may struggle to keep up with these updates, resulting in outdated or incomplete information that can confuse users and cause inaccurate navigation
- ii. Many existing systems suffer from poor user interface design, confusing navigation or complex features, making it difficult for users to effectively utilize the app.

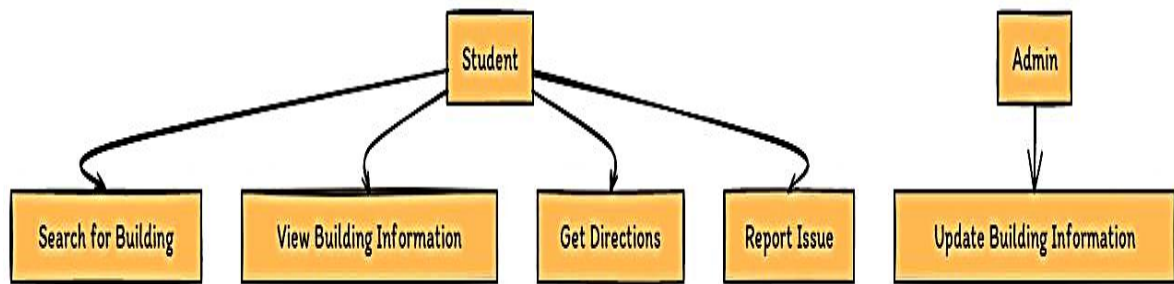
3.6 Analysis of the Proposed system

The proposed campus navigation provides a user-friendly and comprehensive solution for navigating and exploring the campus. It includes turn-by-turn directions, real-time updates, search functionality, make it a valuable tool for students, visitors, and staff to efficiently navigate and access campus information.

3.6.1 Flow Chart Diagram



3.6.2 Use Case Diagram



3.7 Advantages of the proposed system

Below are the advantages of the proposed system:

- i. It has enhanced navigation that can help users easily navigate the campus with distance measuring and comprehensive mapping.
- ii. It provides quick access to information, eliminating the need to search for buildings, departments, or amenities manually.

3.8 Performance Evaluation Metric of Model

Purpose	Survey Question	Evaluation Process	Evaluation Method
To check how	How is the	Using the app to	Test

user friendly the system is.	learning curve	calculate the distance between starting point and ending point	
To know how fast and efficient it is in scanning buildings	How easy is the app able to scan buildings	Scanning the faculty of engineering building	Test
To check if the location tracking is accurate	How reliable the GPS real-time locator is	Using the app from its starting point and calculating the distance between start to finish	Test

CHAPTER FOUR

IMPLEMENTATION

4.1 Introduction

The study implementation and findings are discussed in this chapter. It displays the systems's diagram and choice of programming language that was used to create the system. This assessment makes certain that the system adheres to the original ideas and achieves the desired objectives. Additionally, it provides examples of the system implementation and the programming environment.

The proposed system is a mobile based application with the following functionalities:

- i. Calculating longitude and latitude
- ii. Accurate distance measurement between the user and their destinations.

4.2 System Design and Implementation

This research proposes designing and implementing a campus navigation system that uses Augmented Reality(AR) and GPS location services, My goal is to create an app that can accurately help students and staff navigate their university campus.

```
C:\Users\Admin\Desktop> unityproj > My project (2) > Assets > ARLocation > Scripts > Components > PlaceAtLocation.cs
47 // <summary>
48 // Apply to a GameObject to place it at a specified geographic location.
49 // </summary>
50 [AddComponentMenu("AR+GPS/Place At Location")]
51 [HelpURL("https://http://docs.unity-ar-gps-location.com/guide/#placeatlocation")]
52 [DisallowMultipleComponent]
53 public class PlaceAtLocation : MonoBehaviour
54 {
55     [Serializable]
56     public class ObjectUpdatedEvent : UnityEvent<GameObject, Location, int>
57     {
58     }
59
60     [Serializable]
61     public class PlaceAtOptions
62     {
63         [Tooltip(
64             "The smoothing factor for movement due to GPS location adjustments; if set to zero it is disabled."),
65             Range(0, 1)]
66         public float MovementSmoothing = 0.05f;
67
68         [Tooltip(
69             "The maximum number of times this object will be affected by GPS location updates. Zero means no limits are imposed.")]
70         public int MaxNumberOfLocationUpdates = 4;
71
72         [Tooltip("If true, use a moving average filter.")]
73         public bool UseMovingAverage;
74
75         [Tooltip(
76             "If true, the object will be hidden until the object is placed at the geolocation. If will enable/disable the MeshRenderer or Skl
77             when available, and enable/disable all child game objects.")]
78         public bool HideObjectUntilItIsPlaced = true;
79     }
80 }
```

Fig 4.1 Some of the functions used in Navigation System

4.2.1 Unity Game Development Engine

Unity is an adaptable game creation engine that has skyrocketed in popularity. In order to build dynamic and captivating games across a range of platforms, including PC, consoles, mobile devices, and even augmented reality (AR) and virtual reality (VR) headsets, it offers creators a complete collection of tools and capabilities.

Unity's cross-platform adaptability is one of its main advantages. Developers may save time and work by writing their game code only once and then deploying it across many platforms using Unity. The engine is compatible with many different operating systems and platforms, including Windows, macOS, Linux, iOS, Android, Xbox, and PlayStation. This adaptability enables creators to reach a larger audience and realize the full potential of their game.

Both experienced developers and newbies may utilize Unity because to its user-friendly UI. It has a visual editor that enables users to drag-and-drop game components, construct levels, and create game sceneries. Additionally, C#, which is frequently used by developers to script game behavior and logic, is one of the many programming languages that Unity supports. Developers may incorporate complicated gameplay dynamics, AI systems, manage physics simulations, and much more thanks to the engine's scripting features.

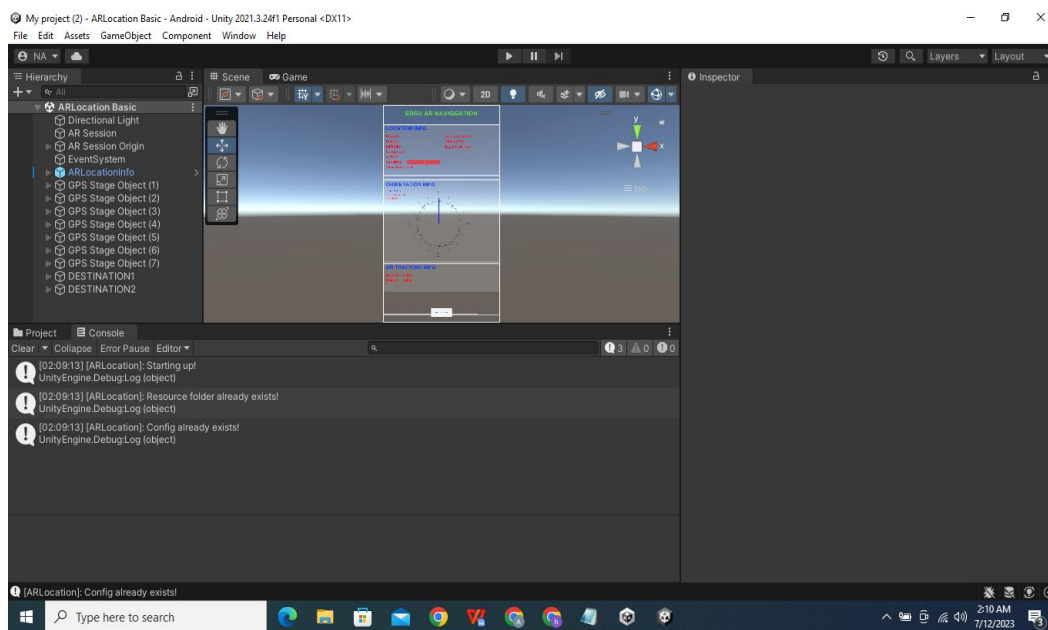


Fig 4.2 Unity Game Dev Engine Interface

4.2.2 AR Plugins

These plugins provide strong integration options and functions that expand the potential of Unity projects. They provide developers the means to optimize backend operations, use cloud services, sell apps with adverts, and build

engaging AR experiences. By using these plugins, you may speed up development, increase the functionality of your app, and enhance user experience.

AR Foundation is a framework provided by Unity that enables developers to create augmented reality (AR) applications that work across multiple platforms, including iOS and Android. It acts as a unified API (Application Programming Interface) that abstracts the underlying AR technologies, such as ARKit for iOS and ARCore for Android, allowing developers to write AR code that is platform-agnostic.

4.2.3 ARCore XR

ARCore XR is a Unity package that enables developers to integrate Google's ARCore technology into their Unity projects. ARCore XR Plugin provides the necessary tools and APIs to build augmented reality (AR) experiences specifically for Android devices. ARCore is Google's platform for building AR experiences on Android devices. It uses advanced computer vision and motion tracking algorithms to enable virtual objects to interact with the real world. ARCore recognizes surfaces, tracks the device's position and orientation, and allows developers to place virtual content in a user's environment.. ARCore XR Plugin seamlessly integrates ARCore with Unity, leveraging Unity's game development capabilities and workflows. It provides a bridge between ARCore and Unity, allowing developers to easily create AR experiences within the Unity Editor. ARCore XR Plugin supports various core AR features provided

by ARCore, including motion tracking, environmental understanding, and light estimation. Motion tracking allows the device to understand its position and orientation in 3D space. Environmental understanding enables the detection of surfaces, such as floors and tables, for placing virtual objects accurately. Light estimation helps virtual objects blend more realistically with the surrounding environment.

4.2.4 AR+GPS Logic

To build immersive experiences in actual locations, AR+GPS blends Augmented Reality (AR) technology with Global Positioning System (GPS) capabilities. The idea behind AR+GPS is to overlay digital information on the user's physical environment by fusing real-time GPS data with virtual material. GPS is a satellite-based navigation system that provides precise location information on Earth. GPS receivers in mobile devices or wearable devices can determine the user's latitude, longitude, and altitude by receiving signals from multiple GPS satellites. To enable AR+GPS, a map of the real world is created or accessed. This involves collecting and processing geographic data to define the physical environment accurately. Maps can include information such as terrain, landmarks, buildings, roads, and points of interest. Virtual content, such as 3D models, animations, text, or images, is created or prepared to be overlaid onto the real-world environment. This content is typically aligned with the geographic data and tied to specific GPS coordinates. As the user moves and

their GPS location updates, the AR+GPS system continuously adjusts the position and rendering of the virtual content to maintain accurate alignment with the user's perspective and the real-world environment.

4.3 Choice of Programming Language

The programming language of choice chosen to implement this research is C#, C# is a general purpose, object-oriented programming language designed to be a versatile language that enables developers to build a wide range of applications, It combines elements from various other languages like C, C++ and Java whilst having it's own features which includes its strong type system, which ensures that variables are assigned specific types and provides compile-time checking for type safety. C# also supports automatic trash collection which simplifies memory management by automatically reclaiming unused memory. Finally C# also has a rich set of libraries and frameworks, which further extends its capabilities. They provide pre-built functionalities for various tasks like working with databases, and creating graphical user interfaces.

4.4 Programming Environment

The programming environment used in our implementation is Unity Game Development Engine as is shown in Fig 4.2. It is a excellent tool for generating the Augmented Reality, it has plugins that can link the AR and GPS, It also has

a text editor that can be used to program apps and other softwares. Then Visual Studio Code offers amazing extensions that enables syntax highlighting

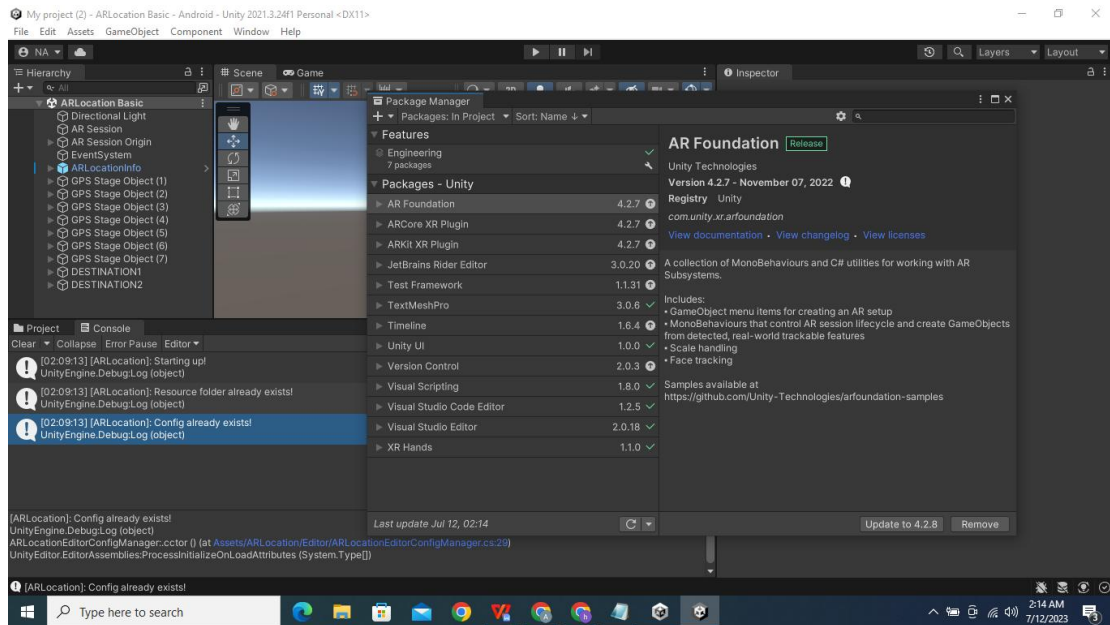


Fig 4.3 Unity Dependency Installation Menu.

4.5 System Implementation

The most time-consuming phase of software development life cycle, The implementation phase is the accomplishment of a technical specification or software module, through computer programming and deployment. At the stage, the physical design specifications were transformed into a working computer code. After which the code passed through the testing stage until the vast majority of the errors and bugs were distinguished and rectified, the system is then installed. Even after this, new highlights and more business rationales would be included due to likely of bugs which clients might experience as they utilize the system. In this study, three design tasks were identified. They were:

- i. The mobile application design
- ii. The testing of the software.

4.5.1 The Mobile Application Design

The app will utilize Google's ARCore SDK and Unity ARCore XR Plugins for augmented reality (AR) functionality and Google Maps SDK for GPS positioning and mapping features. These plug-ins will allow users to overlay digital information onto the real-world view, providing an interactive navigation experience. The AR+GPS plug-ins will be integrated into the app's functionality to accurately position and display information on the user's device.

4.5.2 The Testing

We ran some tests on the system to make sure it satisfies its functional requirements. The tests could be divided into two namely: User view interaction and Technical tests for the backend. We used Unity and Visual Studio Code to run technical tests on the Application Programming Interface (API) created. While for the User View, We used a Samsung S21 to run tests and scan the buildings to get perfect readings.



Fig 4.4 Testing the scanner in Hall 3 Boys Hostel



Fig 4.5 Using the app to determine the distance between current starting point Auditorium and Maingate

4.5.3 System Requirements

The software requirements for implementing the system were:

- i. Unity Game Development Engine
- ii. ARCore - Google for Developers
- iii. Google Earth

4.5.4 Hardware Requirements

- i. Internet Connection
- ii. A network-ready PC with at least, a core i7vpro processor with 2.5 GHz processor speed, 8GB storage.
- iii. A mobile phone.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

In this chapter, we provide an overview of the research to date, the proposed system and its impact on the issue domain, the methods we used to get the results, the conditions we had to operate within, and suggestions for how to advance with the project moving forward.

5.2 Limitations of the study

The following is a list of some restrictions that this investigation came across. Despite the fact that we made a point to utilize each of the devices available to us to come by the result we did, we were by the by restricted by the accompanying conditions.

- i. Technological limitations: The study was limited by the technological capabilities and constraints of the app itself, such as inaccuracy of location tracking, real-time updates, or compatibility with different devices and operating systems.

- ii. Generalizability: The findings of the study was limited to University of Benin. Factors such as campus layout and technological infrastructure may

differ across universities, which could impact the applicability of the study's findings to other institutions.

5.3 Conclusion

In conclusion, the design and implementation of the campus navigation system offer significant benefits for students and staff in navigating and exploring the campus. However, it is crucial to consider the limitations of the study, including generalizability to other universities, and external factors influencing the app's performance.

5.4 Recommendation

This research is still a work in progress. A lot of effort need to be put into this research to fully develop it into a fully functional campus navigation system with AR that can be put into the school system, based on experience and discoveries, we recommend that:

- i. Users should create a GPS APi to accurately locate the user and the entire campus environment
- ii. User should explore the possibility of incorporating indoor mapping capabilities, particularly for large or complex buildings, utilizing technologies such as Wi-Fi signals to provide precise indoor location positioning, ensuring users can easily navigate within buildings and find specific rooms or facilities.

iii. Future works should include turn by turn directions along with the distance measurement to enhance the User Experience.

REFERENCES

- Sandip, Adhikari.(2020, June 23) College Campus Navigation System,
- Chou, T.S., et al.: CaNPAs: A Campus Navigation and Parking Assistant System. In: IEEE International Conference on Systems, Man, and Cybernetics, Taipei, Taiwan, October 8-11, vol. 1, pp. 631–638 (January, 2006).
- Wang, J., Lin, Z.: A General 3D Campus Navigator System. In: 2009 8th IEEE/ACIS International Conference on Computer and Information Science, pp. 1074–1078 (March, 2009).
- Wang, C.S., Hong, C.Y.: A Location and Interest Based Peer to Peer Virtual Navigation System. In: 2nd International Conference on the Applications of Digital Information and Web Technologies 2009, pp. 676–681 (June, 2009).
- Soares, M.G., Malheiro, B.: An Internet DGPS Service for Precise Outdoor Navigation. In: Proceedings of IEEE Conference Emerging Technologies and Factory Automation, vol. 1, pp. 512–518 (November, 2003)
- Liu, W., Liu, Z., Zhang, Y.: Interactive Mobile Campus Based on Position Perception. In: Proceedings of 2011 Fourth International Symposium on Computational Intelligence and Design (ISCID), vol. 1, pp. 147–150 (May, 2011) .
- Gallagher, T.J., Li, B., Andrew, G.D., Rizos, C.: A sector-based campus-wide indoor positioning system. In: 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Zurich, Switzerland, September 15-17, pp. 1–8 (September, 2010).

- Pitale, A., & Bhungara, A. (2019). Human computer interaction strategies—designing the user interface. In *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 752-758). IEEE.
- Fleury, S., & Chaniaud, N. (2024). Multi-user centered design: acceptance, user experience, user research and user testing. *Theoretical Issues in Ergonomics Science*, *25*(2), 209-224.
- Beheshti, M., Naeimi, T., Hudson, T. E., Feng, C., Mongkolwat, P., Riewpaiboon, W., ... & Rizzo, J. R. (2023). A Smart Service System for Spatial Intelligence and Onboard Navigation for Individuals with Visual Impairment (VIS4ION Thailand): Study protocol of a randomized controlled trial of visually impaired students at the Ratchasuda College, Thailand. *Trials*, *24*(1), 169.
- Unger, R., & Chandler, C. (2023). *A Project Guide to UX Design: For user experience designers in the field or in the making*. New Riders.
- Namoun, A., Tufail, A., Mehandjiev, N., Alrehaili, A., Akhlaghinia, J., & Peytchev, E. (2021). An eco-friendly multimodal route guidance system for urban areas using multi-agent technology. *Applied Sciences*, *11*(5), 2057.
- Djahel, S., Doolan, R., Muntean, G. M., & Murphy, J. (2014). A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. *IEEE Communications Surveys & Tutorials*, *17*(1), 125-151.
- Torres-Sospedra, J., Avariento, J., Rambla, D., Montoliu, R., Casteleyn, S., Benedito-Bordonau, M., ... & Huerta, J. (2015). Enhancing integrated

indoor/outdoor mobility in a smart campus. *International Journal of Geographical Information Science*, 29(11), 1955-1968.

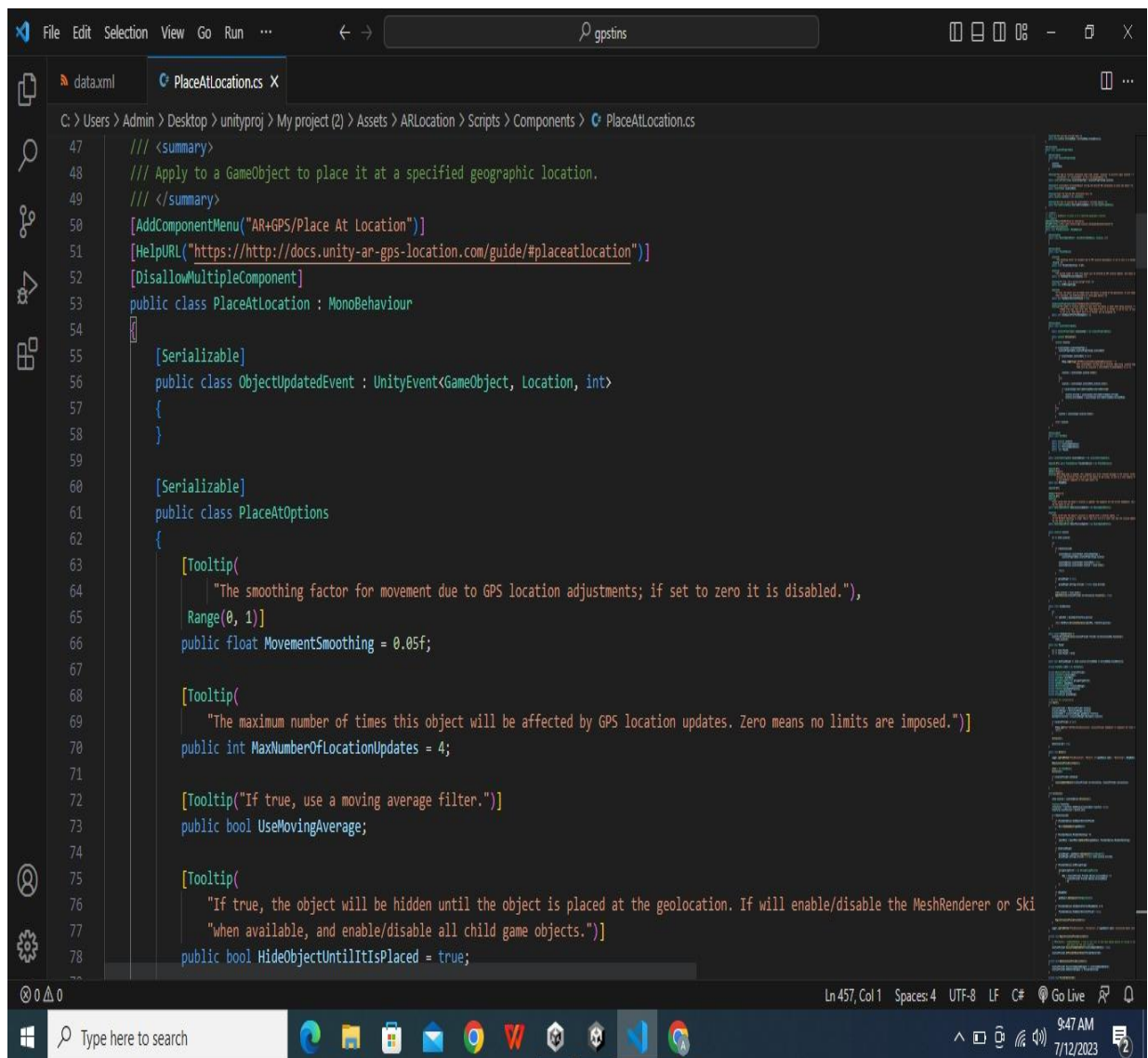
Graeff, E. (2014). Crowdsourcing as reflective political practice: building a location-based tool for civic learning and engagement. *Internet, politics, and policy*.

Dudhane, N. A., & Pitambare, S. T. (2015). Location based and contextual services using bluetooth beacons: New way to enhance customer experience. *Lecture Notes on Information Theory Vol*, 3(1).

Sedig, K., & Parsons, P. (2022). *Design of visualizations for human-information interaction: A pattern-based framework*. Springer Nature.

Arengi, A., Belometti, S., Brignoli, F., Fogli, D., Gentilin, F., & Plebani, N. (2018, November). Unibs4all: A mobile application for accessible wayfinding and navigation in an urban university campus. In *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good* (pp. 124-129).

APPENDIX



The image shows a screenshot of the Visual Studio Code editor. The main window displays the file `PlaceAtLocation.cs` with the following C# code:

```
47  /// <summary>
48  /// Apply to a GameObject to place it at a specified geographic location.
49  /// </summary>
50  [AddComponentMenu("AR+GPS/Place At Location")]
51  [HelpURL("https://http://docs.unity-ar-gps-location.com/guide/#placeatlocation")]
52  [DisallowMultipleComponent]
53  public class PlaceAtLocation : MonoBehaviour
54
55      [Serializable]
56      public class ObjectUpdatedEvent : UnityEvent<GameObject, Location, int>
57      {
58      }
59
60      [Serializable]
61      public class PlaceAtOptions
62      {
63          [Tooltip(
64              "The smoothing factor for movement due to GPS location adjustments; if set to zero it is disabled."),
65              Range(0, 1)]
66          public float MovementSmoothing = 0.05f;
67
68          [Tooltip(
69              "The maximum number of times this object will be affected by GPS location updates. Zero means no limits are imposed.")]
70          public int MaxNumberOfLocationUpdates = 4;
71
72          [Tooltip("If true, use a moving average filter.")]
73          public bool UseMovingAverage;
74
75          [Tooltip(
76              "If true, the object will be hidden until the object is placed at the geolocation. If will enable/disable the MeshRenderer or Ski
77              when available, and enable/disable all child game objects.")]
78          public bool HideObjectUntilItIsPlaced = true;
```

The status bar at the bottom of the editor shows "Ln 457, Col 1 Spaces: 4 UTF-8 LF C#". The Windows taskbar is visible at the bottom of the screen, showing the search bar and several application icons.

```
File Edit Selection View Go Run ... gptins
data.xml PlaceAtLocations.cs X
C:\Users\Admin\Desktop> unityproj > My project (2) > Assets > ARLocation > Scripts > Components > PlaceAtLocation.cs
222 private GroundHeight groundHeight;
223
224 // Use this for initialization
225 void Start()
226 {
227     locationProvider = ARLocationProvider.Instance;
228     arLocationManager = ARLocationManager.Instance;
229     arLocationRoot = arLocationManager.gameObject.transform;
230     mainCameraTransform = arLocationManager.MainCamera.transform;
231
232     if (locationProvider == null)
233     {
234         Debug.LogError("[AR+GPS][PlaceAtLocation]: LocationProvider GameObject or Component not found.");
235         return;
236     }
237
238     Initialize();
239     hasInitialized = true;
240 }
241
242 public void Restart()
243 {
244     Logger.LogFromMethod("PlaceAtLocation", "Restart", $"{gameObject.name} - Restarting!", DebugMode);
245
246     RemoveLocationProviderListeners();
247
248     state = new StateData();
249     Initialize();
250
251     if (locationProvider.IsEnabled)
252     {
253
```

```
File Edit Selection View Go Run ... gptins
data.xml PlaceAtLocations.cs X
C:\Users\Admin\Desktop> unityproj > My project (2) > Assets > ARLocation > Scripts > Components > PlaceAtLocation.cs
65 range(0, 4);
66 public float MovementSmoothing = 0.05f;
67
68 [Tooltip("The maximum number of times this object will be affected by GPS location updates. Zero means no limits are imposed.")]
69 public int MaxNumberOfLocationUpdates = 4;
70
71 [Tooltip("If true, use a moving average filter.")]
72 public bool UseMovingAverage;
73
74 [Tooltip("If true, the object will be hidden until the object is placed at the geolocation. If will enable/disable the MeshRenderer or Skl
75 when available, and enable/disable all child game objects.")]
76 public bool HideObjectUntilItIsPlaced = true;
77
78 [ConditionalPropertyAttribute("HideObjectUntilItIsPlaced")]
79 [Tooltip("The number of location updates to wait until the object is shown after being initially " +
80 "hidden from view. Only works when 'Hide Object Until It Is Placed' is set to true. If this " +
81 "is set to 0, 'Hide Object Until It Is Placed' will be disabled.")]
82 public uint ShowObjectAfterThisManyUpdates = 1;
83
84 }
85
86 [Serializable]
87 public class LocationSettingsData
88 {
89     public LocationPropertyData LocationInput = new LocationPropertyData();
90
91     public Location GetLocation()
92     {
93         Location location;
94
95         if (LocationInput.LocationInputType ==
96             LocationPropertyData.LocationPropertyType.LocationData)
97
```

```
346 }
347
348 Vector3 targetPosition;
349 var location = state.Location;
350 var useSmoothMove = smoothMove != null;
351 var isHeightRelative = location.AltitudeMode == AltitudeMode.DeviceRelative;
352 // If we have reached the max number of location updates, do nothing
353 if ((PlacementOptions.MaxNumberOfLocationUpdates > 0) &&
354     (state.LocationUpdatedCount >= PlacementOptions.MaxNumberOfLocationUpdates) && !forceUpdate)
355 {
356     return;
357 }
358
359 // Calculate the target position where the object will be placed next
360 if (movingAverageFilter != null)
361 {
362     var position = Location.GetGameObjectPositionForLocation(
363         arLocationRoot, mainCameraTransform, deviceLocation, location, isHeightRelative
364     );
365     var accuracy = locationProvider.CurrentLocation.accuracy;
366     movingAverageFilter.AddEntry(new DVector3(position), accuracy);
367     targetPosition = movingAverageFilter.CalculateAveragePosition().toVector3();
368 }
369 else
370 {
371     targetPosition = Location.GetGameObjectPositionForLocation(
372         arLocationRoot, mainCameraTransform, deviceLocation, location, isHeightRelative
373     );
374 }
375
376
377 }
```