DESIGN AND IMPLEMENTATION OF A MAP APPLICATION OF THE UNIVERSITY OF BENIN CAMPUS USING GEOGRAPHICAL INFORMATION SYSTEM

BY

OSAYOMWANBOR OMOEIYEKEWEN JAMES

PSC2003833

DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF PHYSICAL SCIENCES,

UNIVERSITY OF BENIN,

BENIN CITY,

EDO STATE, NIGERIA.

February 2025

DESIGN AND IMPLEMENTATION OF A MAP APPLICATION OF THE UNIVERSITY OF BENIN CAMPUS USING GEOGRAPHICAL INFORMATION SYSTEM

BY

OSAYOMWANBOR OMOEIYEKEWEN JAMES

PSC2003833

A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF PHYSICAL SCIENCES, UNIVERSITY OF BENIN, BENIN CITY

IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF A BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER SCIENCE

JUNE 2025

CERTIFICATION

This is to certify that this project work was carried out by **OSAYOMWANBOR OMOEIYEKEWEN JAMES** with Matriculation Number **PSC2003833** under my supervision. It is adequate and satisfactory, both in scope and content, for the award of Bachelor of Science (B.sc) Degree in Computer Science of the University of Benin

MR. P.E.B. IMIEFOH
Project Supervisor

DATE

APPROVAL

This project work is hereby approved in partial fulfilment of the requirements for the award of Bachelor of Science (B.Sc.) Degree in Computer Science from the University of Benin.

Prof. Godspower O. Ekuobase, PhD	**Date**
(Head of Department)	

DEDICATION

This project is dedicated to God Almighty for giving me the strength and wisdom to see it through to completion, and even throughout my stay in the University of Benin (UNIBEN). It is also dedicated to my parents; Mr and Mrs Imadonmwiyi and my siblings most especially Master Osayande Paul Osayomwanbor; for their love, support and guidance throughout my academic journey.

ACKNOWLEDGEMENT

My utmost acknowledgement goes to God Almighty for giving me the strength, wisdom and direction throughout my academic journey. I would like to express my gratitude to my project supervisor Mr. P.E.B IMIEFOH for his consistent guidance towards ensuring the successful completion of this project.

I would also like to specially thank all the lecturers in the Department of Computer Science who I have been opportune to cross paths with, and have impacted me immensely these past few years: Prof. G.O. Ekuobase, Dr. F.O. Oliha, Prof. K.C. Ukaoha, Prof. A.A. Imiavan, Prof. (Mrs.) F. Egbokhare, Prof. (Mrs.) V.V.N. Akwukwuma, Prof. F.I. Amadin, Prof. (Mrs.) S. Konyeha, Prof. (Mrs.) V.I. Osubor, Dr. (Mrs.) Aziken, Dr. F.O. Chete, Dr. (Mrs) R.O. Osaseri, Dr. J.C. Obi, Mr. P. E.B. Imiefoh, Mr. I.E. Obasohan, Mr. S.O.P. Oliomogbe, Mr. K.O. Otokiti, Mr. I.E. obayagbonna, Mrs. R.I. Izevbizua, Mr. E.C. Igodan, Miss L.O.Usiosefe, Mr J. Okhuoya, Prof. F.A.U. Imouokhome, Mrs. J.I. Adun, Dr. E. Nweli and Mr. D.N. Idehen.

I would also like to thank my family and friends for their support, words of encouragement, and consistent guidance throughout this project.

ABSTRACT

This study develops a web-based map application to enhance navigation within the University of Benin campus, helping users locate buildings, offices, classrooms, and other points of interest. Using Next.js, TypeScript, TailwindCSS, and third-party mapping APIs like Google Maps, the application provides an interactive, mobile-friendly experience. Users can search for locations, get directions, and access regularly updated map data. This tool improves accessibility, saves time, and enhances user experience. By leveraging modern web technologies and geospatial data, the application fosters a more navigable and welcoming campus environment.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

1.0 BACKGROUND STUDY

Geographic Information System (GIS) is a system designed to capture, store, manipulate, analyze, manage, and present all types of geographical data (Baros & Stojanovic 2015). The acronym GIS is sometimes used for geographical information science or geospatial information studies to refer to the academic discipline or career of working with geographic information systems. In the simplest terms, GIS is the merging of cartography, statistical analysis, and database technology. A GIS can be thought of as a system—it digitally creates and "manipulates" spatial areas that may be jurisdictional, purpose, or application-oriented(Maddodi, Udayashankara, Raghavan, & Mathews 2013). Geographical Mapping and Information Systems (GMIS), commonly known as Geographic Information Systems (GIS), are multifaceted software and hardware systems that capture, manage, analyze, and visualize geographic data. These systems combine geographic spatial information (location-based) with descriptive attribute data (information-based) to provide a powerful platform for decision-making, planning, and problem-solving. Generally, a GIS is custom designed for an organization. Hence, a GIS developed for an application, jurisdiction, enterprise, or purpose may not be necessarily interoperable or compatible with a GIS that has been developed for some other application, jurisdiction, enterprise, or purpose. What goes beyond a GIS is a spatial data infrastructure, a concept that has no such restrictive boundaries(Quinn, 2016).

There are several definitions of GIS in existence. However, none of such definitions is universally accepted. Here are some of the various definitions of GIS given over the years. GIS is about uncovering meaning and insights from within data. It is rapidly evolving and providing a whole new framework and process for understanding(Esri Product Team 2017). GIS is an integrated system of computer hardware, software, and trained personnel linking topographic, demographic, utility, facility, image, and other resource data that is geographically referenced(Dempsey, 2023). It is difficult to agree on a single definition for GIS for the simple reason that various kinds of GIS exist, each made for different purposes and for different types of decision making. Early GISs were often standalone and relied on analog maps and data input methods. Today, GIS leverages advances in computing, satellite technology, GPS, remote sensing, and digital cartography, making them indispensable tools in a wide range of fields. GIS allowed for the integration of various data layers, such as topography, land use, and infrastructure, creating more comprehensive and detailed maps. In recent years, the rise of digital mapping systems, like Google Maps and Apple Maps, has transformed how we navigate and explore our surroundings. These systems utilize satellite imagery, GPS technology, and real-time data to provide accurate and up-to-date information. Mapping systems have become crucial in various domains. They are used in transportation and logistics for route planning and optimization. They support urban planning and infrastructure development by analyzing spatial data. Emergency services rely on mapping systems to locate incidents and coordinate responses effectively. Furthermore, mapping systems have become more interactive and personalized. Users can customize their maps, toggle between different layers, and access additional information like reviews, photos, and recommendations. In summary, the background of a mapping system encompasses centuries of cartography, technological advancements, and the integration of various data sources. Today, mapping systems are an essential tool for navigation, planning, and decision-making in our interconnected world. In a general sense, the term describes any information system that integrates, stores, edits, analyzes, shares, and displays geographic information for informed decision making. GIS applications are tools that allow users to create interactive queries (user-created searches), analyze spatial information, edit data in maps, and present the results of all these operations(Bipu, 2019).

This system is intended to help individuals, students, and non-students, to gain knowledge about their environment as well as to help in planning and exploration. Time wasted or spent trying to maneuver their way through the school environment could be better spent performing other useful activities.

1.1 PROBLEM DEFINITION

The implementation of an effective map application or system at University of Benin aims to address several key challenges faced by new students and guests when navigating the campus environment. The following problems have been identified as focal points for this project:

1. Limited Campus Familiarity: New students and guests often lack familiarity with the University of Benin campus layout, leading to confusion and difficulty in finding specific locations such as classrooms, administrative offices, and amenities.

2. Inefficient Navigation: The absence of a centralized and user-friendly navigation system hinders efficient movement across the campus. Current navigation methods do not optimize routes or provide real-time guidance, resulting in time wastage and frustration.

3. Lack of Awareness: Visitors are often unaware of important points of interest within the campus, such as emergency exits, medical centers, and Wi-Fi hotspots. This lack of awareness can compromise safety and access to essential services.

4. Incomplete or Outdated Information: Existing campus maps and informational resources may be incomplete or outdated, failing to provide accurate representations of the constantly evolving campus infrastructure. This discrepancy can mislead users and create confusion.

1.2 AIM AND OBJECTIVES

The aim of this study is to develop a map application utilizing a Geographical Information System (GIS) in University of Benin to aid new students and guests in navigating and planning their routes on the campus effectively while enhancing their awareness of the school environment. The name of this application is Uniben Campus Map Application.

The objectives are as follows:

1. Designing of Uniben Campus Map Application.
2. Implementation of Uniben Campus Map Application.
3. Evaluation of Uniben Campus Map Application.

1.3 RESEARCH METHODOLOGY

This project made use of Object-Oriented Analysis and Design Methodology (OOAD), which is the process of analysing and designing a system as a set of components and objects. It made use of the observation approach for analysis of related and relevant documents on the area of interest to establish the need and necessity of the proposed system and after that ended with a conclusion, summary and recommendation.

1.4 SCOPE OF THE STUDY

1. Campus Mapping: Development of a comprehensive digital map of the University of Benin Main campus. This includes detailed information about buildings, classrooms, administrative offices, cafeterias, libraries, parking lots, sports facilities, and other key locations.
2. Navigation and Route Planning: Implementation of a user-friendly interface that allows new students and guests to input their current location and desired destination within the campus. This may include providing interactive route planning functionality, suggesting the most efficient paths based on the user's location and destination.
3. Geospatial Information: Displaying of additional geospatial information, such as points of interest (e.g., lecture halls, labs, hostels and kiosks), emergency exits and Wi-Fi hotspots.
4. User-Friendly Interface: Designing an intuitive and easy-to-use interface accessible via web browsers and mobile devices. This includes features like search functionality, zoom in/out options, and interactive campus overlays for different purposes (e.g., academic buildings, residential areas).
5. Evaluation and Feedback: Implementation of a feedback mechanism within the map application to collect user opinions and suggestions for improvement. Usability tests may be conducted to gather feedback from new students and guests to assess the effectiveness of the system.
6. Training and Support: Providing training sessions or tutorials for users to familiarize themselves with the map application.
7. Integration with Existing Systems: Exploration of the possibilities for integrating the map application with existing university systems, such as the official Babcock University website or mobile applications, to reach a wider audience.

1.5 RESEARCH SIGNIFICANCE

The creation and development of a map application for University of Benin carries considerable importance and offers various solutions such as:

1. Improved Campus Navigation: The GIS implementation will significantly enhance the navigation experience for new students and guests, reducing confusion and minimizing the time taken to locate specific campus destinations. This improved efficiency contributes to a smoother campus integration process.
2. Enhanced Safety and Security: By providing accurate information about emergency exits, first aid stations, and other safety-related points, the GIS system enhances the overall safety of individuals on campus. It ensures that users are aware of essential safety features, thereby promoting a secure environment.
3. Enhanced User Experience: The GIS application, designed with a user-friendly interface, improves the overall experience for users. Easy access to information and intuitive navigation tools fosters positive interactions, making the campus environment more welcoming.
4. Efficient Decision-Making: The GIS system provides valuable data and insights that can aid university administrators in decision-making processes. Analyzing user behavior and preferences can inform strategic decisions related to campus layout, infrastructure development, and service offerings.
5. Support for Campus Events and Planning: The GIS system facilitates event planning by providing insights into foot traffic patterns and popular locations. It helps in organizing events more effectively and ensures that attendees can navigate to event venues without difficulty.
6. Promotion of Technological Integration: Implementing a GIS system showcases the university's commitment to integrating cutting-edge technology into its operations. It establishes

the University of Benin as a technologically advanced institution, enhancing its reputation among students, staff, and visitors.

7. Optimized Resource Utilization: The GIS data can be utilized by the university administration to optimize resource allocation. By analyzing user movement patterns, the university can make informed decisions about facility placements, ensuring that high-traffic areas are efficiently organized.

1.6 DEFINITION OF TERMS

Geographic Information System (GIS) – An organized collection of computer hardware, software, geographic data, and personnel designed to efficiently capture, store, update, manipulate, analyze, and display all forms of geographically referenced information.

Spatial Analysis – The process of modeling, examining, and interpreting model results. Spatial analysis is useful for evaluating suitability and capability, for estimating and predicting, and for interpreting and understanding.

Map - A map is a visual representation or a symbolic depiction highlighting spatial relationships and connections between elements, typically involving geographical or topographical information.

Metadata – Information about a data set. It may include the source of the data; its creation date and format; its projection, scale, resolution, and accuracy; and its reliability with regard to some standard.

Structured Query Language (SQL) – A syntax for defining and manipulating data from a relational database. Developed by IBM in the 1970s, it has become an industry standard for query languages in most relational database management systems.

Global Positioning System (GPS) – A satellite based device that records x,y,z coordinates and other data. Ground locations are calculated by signals from satellites orbiting the Earth. GPS devices can be taken into the field to record data while walking, driving, or flying.

Attribute – A characteristic of a geographic feature, typically stored in tabular format and linked to the feature in a relational database. The attributes of a well-represented point might include an identification number, address, and type.Base Layer – The foundational layer that provides the primary visual representation of the geographical area. It serves as the background over which other layers, such as markers, overlays, or thematic data, are placed. The base layer typically displays geographical features such as landforms, water bodies, roads, and political boundaries. It can be a simple flat-colored map, a detailed satellite image, or a combination of various elements that provide context and orientation to the user.

1.7 ORGANIZATION OF THE PROJECT

This project work is organized into five chapters. Chapter one is titled introduction. It contains the background to the study, statement of the problem, methodology used in the study, scope of the study, aim and objectives of the study, purpose, and significance for the study. The chapter also contains the organization of the project work and the definition of technical terms used in the study.

Chapter two is titled literature review. It contains a review of related work done in this area of study. The review includes theoretical, historical, and empirical review of related works done on the study. It also contains materials from various other authors and sources.

Chapter three is titled system analysis and design. It contains the analysis and design of both the existing system and the proposed systems. It shows the analysis design requirement, methods, techniques, and the programming language adopted by the proposed system.

Chapter four is titled implementation, it contains the system specification, source of data collection, implementation procedure and tools, description of the testing phase, operations of the system, evaluation and discussion of the results obtained in this study.

Chapter five is titled Summary, Conclusion, and Recommendation. It contains the summary of findings obtained in the study. It also contains conclusions, recommendations, and suggestions for future study in the related area as well as some of the challenges faced in the course of the research.

CHAPTER TWO

LITERATURE REVIEW

2.0 INTRODUCTION

Geographic Information Systems (GIS) have revolutionized the way we perceive and interact with spatial data, offering extensive capabilities for mapping and analysis(Kumar, 2023). GIS-based mapping systems represent a critical component in various fields, enabling visualization, analysis, and interpretation of spatial information for a wide array of applications. The literature review aims to explore the evolution, advancements, and applications of GIS-based mapping systems, understanding their significance, innovations, challenges, and implications across diverse domains.

2.1 HISTORICAL BACKGROUND OF THE RESEARCH

The history of GIS all started in 1854 when Cholera hit the city of London, England. British physician John Snow began mapping outbreak locations, roads, property boundaries, and water lines. Snow achieved this through plotting the residence of each casualty on a map of the area, as well as the nearby water sources. Once these points were marked, he was able to identify the water source within the cluster that was responsible for the outbreak. This was one of the earliest successful uses of a geographic methodology in pinpointing the source of an outbreak in epidemiology(Olamide, 2021).

Another relevant experience that combined computers and geography can be found in 1959, when Waldo Tobler defined the principles of a system called MIMO (map in--map out), with the purpose of applying computers to the field of cartography. He defined the basic ideas for creating, encoding, analyzing, and rendering geographical data within a computer system.

In 1960, Roger Tomlinson who is known to be the father of GIS; through his pioneering work to initiate, plan, and develop the Canada Geographic Information System which resulted in the first computerized GIS in the world in 1963. The Canadian government had commissioned Tomlinson to create a manageable inventory of its natural resources. He envisioned using computers to merge natural resource data from all provinces. He created the design for automated computing to store and process large amounts of data, which enabled Canada to begin its national land-use management program. He also gave GIS its name.

By 1965, while at Northwestern University in 1964, Howard Fisher created one of the first computer mapping software programs known as SYMAP. In 1965, he established the Harvard Laboratory for Computer Graphics. While some of the first computer map-making software was created and refined at the Lab, it also became a research center for spatial analysis and visualization. Many of the earlier concepts for GIS and its applications were conceived at the Lab by a talented collection of geographers, planners, computer scientists, and others from many fields.

By 1981 GIS went commercial, as computing became more powerful, Esri (founded in 1969, One of the biggest consulting firms known for computer mapping and spatial analysis to help land-use planners and land-resource managers make informed decisions) improved its software tools. Working on projects that solved real-world problems led the company to innovate and develop robust GIS tools and approaches that could be broadly used. Esri's work gained recognition from the academic community as a new way of doing spatial analysis and planning. Esri developed ARC/INFO—the first commercial GIS product. The technology was released in 1981 and began the evolution of Esri into a software company(Dempsey, 2015).

Up until now, users are starting to adopt GIS technology in different ways. Classrooms, businesses, governments around the world are starting to harness digital mapping and analysis. Companies hesitated to adopt GIS software, Software which was able to handle both vector and raster data.

We are gradually entering an era of open-source GIS software and the importance of spatial analysis for decision-making is becoming recognized. More light is shining on QGIS than ever before. Software companies like Esri provide solutions to practically any spatial problem that exists today(GISGeography. 2015).

2.2 CONCEPTUAL OVERVIEW

A Geographic Information System (GIS) is a powerful tool used for capturing, storing, analyzing, and visualizing spatial or geographic data. Here is a detailed conceptual overview of GIS:

2.2.1 DEFINITION OF GIS

Geographic Information System (GIS): A GIS is a system designed to capture, store, manage, analyze, and present spatial or geographic data. It integrates various types of data to provide a comprehensive understanding of a location or a phenomenon.

2.2.2 KEY COMPONENTS OF GIS

1. Data: GIS revolves around data, which can include spatial data (coordinates, shapes), attribute data (information about the spatial features), and temporal data (information about changes over time).
2. Hardware: GIS systems use hardware such as computers, servers, GPS devices, and data storage systems.
3. Software: GIS software facilitates the processing, analysis, and visualization of geographic data. Examples include ArcGIS, QGIS, and Google Earth.
4. People: GIS professionals, analysts, and users are essential components. They contribute to data collection, analysis, and decision-making.

2.2.3 DATA TYPES IN GIS

1. Spatial Data: Represents the geographic location and shape of features. It can be in the form of points, lines, polygons, or raster images.

2. Attribute Data: Describes the characteristics of spatial features. It is often stored in tables and linked to spatial data through unique identifiers.
3. Temporal Data: Represents changes in geographic features over time, allowing for historical analysis.

2.2.4 GIS FUNCTIONS AND OPERATIONS

1. Data Capture: The process of collecting or inputting spatial data into the GIS system. This can involve digitizing maps, using GPS devices, or importing existing digital data.
2. Data Storage: Organizing and storing spatial and attribute data efficiently, often using databases or file systems.
3. Data Analysis: Performing various analytical operations on spatial data to derive meaningful insights. This can include overlay analysis, proximity analysis, and spatial statistics.
4. Data Visualization: Presenting spatial information in the form of maps, charts, and graphs. Visualization enhances the understanding of patterns and relationships in the data.

2.2.5 APPLICATIONS OF GIS

1. Urban Planning: GIS is used to analyze land use, plan infrastructure, and manage resources in urban areas.
2. Environmental Management: GIS helps in monitoring and managing natural resources, tracking environmental changes, and assessing the impact of human activities.
3. Emergency Management: GIS aids in disaster response and recovery by mapping affected areas, assessing vulnerabilities, and planning evacuation routes.

2.2.6 CHALLENGES AND CONSIDERATIONS

1. Data Quality: Ensuring the accuracy and reliability of spatial data is crucial for meaningful analysis and decision-making.
2. Interoperability: Integrating GIS with other systems and databases to ensure seamless data exchange.

In conclusion, GIS is a versatile and dynamic tool with applications across various domains. It empowers users to make informed decisions by leveraging the spatial dimension of data and facilitating a deeper understanding of our world.

2.3 OVERVIEW OF EXISTING SYSTEM

The realm of map applications using Geographic Information Systems (GIS) has evolved significantly, encompassing a diverse array of systems tailored for various purposes and industries(Lodha, 2021). The existing systems reflect the integration of cutting-edge technologies, improved user experiences, and the growing importance of geospatial data.

Modern GIS map applications leverage advanced technology stacks, incorporating robust GIS software frameworks like ArcGIS, QGIS, and open-source alternatives. Cloud-based GIS solutions, such as those offered by Google Maps, Mapbox, and Esri ArcGIS Online, have gained popularity, allowing for scalable and accessible mapping services.

User interfaces in GIS map applications have evolved to be more intuitive and user-friendly. Interactive maps with features like zoom, pan, and dynamic layer toggling enhance user engagement. Web-based GIS applications often provide seamless and responsive interfaces accessible across various devices, fostering widespread usability.

There are many ways in which Geographical Information Systems have been implemented over the years. Below are a few of them.

1. ArcGIS: ArcGIS, developed by ESRI (Environmental Systems Research Institute), an American multinational geographic information company (GIS) software company, is a comprehensive GIS software suite widely used by professionals and organizations around the world. It offers a broad range of tools for creating, analyzing, and visualizing geospatial data. ArcGIS includes ArcMap for desktop GIS, ArcGIS Pro for advanced 2D and 3D mapping, ArcGIS Online for cloud-based mapping and collaboration, and ArcGIS Enterprise for on-premises deployment. ArcGIS supports various data formats, provides spatial analysis capabilities, and offers extensive geoprocessing tools. It is extensively used in fields such as urban planning, environmental management, natural resource exploration, and disaster response due to its versatility and robust functionalities.

2. QGIS (Quantum GIS): QGIS is a popular open-source GIS software known for its flexibility and user-friendly interface. It allows users to create, edit, visualize, and analyze geospatial data in multiple formats. QGIS supports vector, raster, and database formats, making it compatible with a wide range of data sources. It offers advanced cartographic features, spatial analysis tools, and plugins contributed by the user community. QGIS is widely used in research, education, and various industries due to its open nature and the ability to customize and extend its functionality.

3. Google Maps: Google Maps is a widely used web mapping service developed by Google. It provides detailed maps, satellite imagery, street views, and real-time traffic conditions. Users can explore locations, get directions for driving, walking, cycling, or public transportation, and even view indoor maps of certain facilities. The platform incorporates a powerful search engine,

allowing users to find businesses, addresses, landmarks, and points of interest. Additionally, Google Maps provides information about businesses, including reviews, ratings, and operational hours. Google Maps includes a distinctive Street View feature, allowing users to virtually explore streets at ground level. This immersive experience enhances the understanding of specific locations and aids in visualizing places before visiting. Developers can leverage the Google Maps Platform to integrate mapping and location-based services into their applications. The Google Maps API suite provides a range of tools and services for developers to create customized map experiences.

4. Google Earth Engine: Google Earth Engine is a cloud-based platform that provides petabytes of geospatial data and computational capabilities for analyzing Earth's surface. It offers an extensive archive of satellite imagery from multiple sources, enabling users to conduct large-scale analysis and modeling. Google Earth Engine supports a wide array of remote sensing applications, including land cover mapping, deforestation monitoring, climate change analysis, and disaster response. Its cloud-based infrastructure allows researchers and scientists to process massive datasets efficiently, making it a valuable tool for environmental research and monitoring projects.

5. GRASS GIS: GRASS GIS (Geographic Resources Analysis Support System) is a powerful open-source GIS software suite designed for advanced geospatial analysis and modeling. It provides a comprehensive set of tools for raster, vector, and 3D data processing. GRASS GIS offers functionalities for spatial modeling, image processing, terrain analysis, and geostatistics. It supports various data formats and includes modules for advanced cartography and visualization. GRASS GIS is widely used in scientific research, environmental modeling, and spatial analysis due to its rich set of features and the ability to handle complex geospatial operations.

6. OpenStreetMap (OSM): OpenStreetMap is a collaborative mapping platform where users can create, edit, and share geospatial data freely. It is maintained by a global community of mappers who contribute and update map data regularly. OpenStreetMap offers an extensive dataset of roads, buildings, landmarks, and geographic features, which can be accessed and downloaded for various applications. The data is used in web mapping, navigation systems, mobile apps, and humanitarian efforts. OpenStreetMap data is versatile and can be customized for specific projects, making it a valuable resource for developers, researchers, and organizations requiring free and open geospatial data.

2.4 REVIEW OF RELATED WORK

1. “A COMPLETE FOS APPROACH FOR INDOOR CROWDSOURCED MAPPING: CASE STUDY ON SAPIENZA UNIVERSITY OF ROME FACULTIES” written by (Mascitelli, Ravanelli, Mattoccia, Berardocco, & Mazzoni, 2020) provides an in-depth study of environment mapping processes. Indoor mapping is an essential process in several applications such as the visualization of space and its utilization, security, emergency and resource planning and location-based alerts and, finally, indoor navigation(Mascitelli, Ravanelli, Mattoccia, Berardocco, & Mazzoni, 2020). The key focus of the study was to present a free and open-source (FOS) approach to map indoor and outdoor environments and navigate through them. The research was conducted at Sapienza University of Rome, providing a practical case study. Key terms associated with the study include Crowdsourced Indoor Mapping, Volunteered Geographic Information (VGI), Fiber Optic Sensors (FOS), Geo-crowdsourcing, and involvement of the OpenStreetMap (OSM) Community. The study aimed to develop an interactive tool based on open-source software for indoor and outdoor mapping and navigation,

placing it within the broader context of online cartography. The tool is built on OpenStreetMap, a collaborative project for creating free content maps of the world. The data for these maps are generally collected by volunteers using various tools like GPS units, smartphones, cameras, and even paper. The document further discusses Volunteer Geographic Information (VGI), where individuals contribute geospatial information voluntarily, a model made possible largely due to the Internet. The authors argue that the integration of indoor and outdoor mapping can enhance navigation and have a significant positive impact on safety, security, and emergency interventions management.

2. The paper titled “Android App for Creating a Map of the College to be used with visitor localization” is a paper authored by Swapnil Vaidya and others from the S.N.D. College of Engineering & Research Center, Yeola in the year 2023. The paper addresses the challenge of helping visitors and students navigate to their desired locations easily and timely, particularly in new or unfamiliar environments(Vaidya, Rahane, Gavhane, Shinde, Shaikh, & S. W., 2023).

The authors discussed the widespread use of smartphones among people, including students, and proposed a map application as a solution to this navigation problem. This application would be particularly useful in locating desired places and identifying the shortest paths from the users' current locations. The paper seems to acknowledge the difficulty visitors face in new locations, likening the experience to being lost in a maze without an easy way to reach the destination.

The paper introduces an application for Android mobile devices, which is implemented to allow Android users to add, remove, and review specific locations. Google Maps APIs, Google Direction APIs, PHP, JSON, and MySQL have been integrated and used in this application to obtain solutions.

The paper focuses on the development of a user-friendly navigation application, leveraging mobile technology to enhance the experience of visitors and students in navigating new or complex environments.

3. Publishing and sharing geo-spatial data are becoming important and popular tasks in various applications (Hu & Dai, 2013). In the year 2013 Shunfu Hu and Ting Dai released a journal article on the development of an online map application made using Google Maps API, SQL database and ASP.NET. The article was able to cover in a precise and well-defined manner, the thought process and research methodology that was involved in the development of the application. The case study of this project was the development of an online map service to display tens of thousands of gardens on the Internet for the United States Department of Agriculture (USDA) People's Garden Initiative. The objective of this paper was to demonstrate an online mapping application that requires the access to data sources in the format of a commercial database stored on a secure data server and that offers sophisticated functionalities for the user to manipulate the data(Hu & Dai, 2013).

4. Design Campus Map Using OpenStreetMap Digital Software is an article published in the year 2021 by Mohamad Afi Asyraf Maidin, Faridahanim Ahmad, Nur IzieAdiana Abidin and a slew of other authors. The paper focused on the development of a campus digital map using OpenStreetMap program, a software platform that is available on the Web. It is a community project that can help create the world’s first editable, digital map. The problem facing this study is the lack of information about the location area that gives difficulties for the new student intake and parents to find the locations(Maidin, 2021).. This study aimed to develop the digital mapping of College Campus Pagoh in OpenStreetMap. The website was developed using Joomla 3.8.13

with PHP version 5.6.25. The methodology of the study was begun with identifying all the locations and the name of the building in the College Campus Pagoh. Then, the complete map of College Campus Pagoh in OpenStreetMap was embedded into the Smart Campus and finally improvises the interface website of Smart Campus. Digital Campus based on OpenStreetMap can help students, admin, staff, lecturers, and visitors find location by using the digital platform. The map of the College Campus Pagoh on Smart Campus can be accessed by users to find the location. Thus, a digital map on Campus is very useful for giving effective information about the location and is beneficial to all parties.

5. Shunfu Hu in the year 2012, published an article titled "Online Map Service Using Google Maps API and Other Javascript Libraries: An Open Source Method". The objective of the article was to demonstrate an online mapping application that focuses not only on the functionality to display points of interest with customized icons and the information associated with them, but also on the sophisticated functionalities for marker clustering, searching, filtering, and tabbed interface that offer the user the capability to manipulate the data, which is lacking in most documented web mapping services(Hu, 2012).

6. People are travelling to many different places, and they are unaware of the directions and the route to be followed. Without knowing the directions, this may cause you to get the wrong way. To avoid this situation, many mobile applications are developed to help people to get the right directions(Mohsin & Aldabbagh, 2014). Omar Aldabbagh and Khalid J Mosen published an article in the year 2014, named "Design and Implementation an Online Location Based Services Using Google Maps for Android Mobile". This paper introduces an application for android mobile, which is implemented to provide the android mobile user to add, remove and review specific locations on the online map. The proposed application also presents the basics

navigation operations like showing directions with the optimal path between source and destination and calculating the distance and expected driving time. Google Maps APIs, Google Direction APIs, PHP, JSON, and MySQL have been integrated and used in this application to obtain solutions. This application is a representation for client/server architecture, the android mobile serves as a client and the combination of PHP and MySQL database serve as a server.

7. Location is one of the most important context parameters in mobile computing. Especially relevant is user location(Granlund, Johansson, Andersson & Brännström, 2013). In the year 2013, Daniel Granlund, Dan Johansson, Karl Andersson, and Robert Brännström from the department of computer science, Electrical and Space Engineering Luleå University of Technology released a paper titled "A Case Study of Application Development for Mobile and Location-Based Services". In this paper they presented a case study of a location-dependent service, aimed for tourists. They provided implementation details and compared the different solutions in terms of location support, compliance with required features, and cross-platform functionality. From their experiments it showed that web-based approaches may lead to significant benefits over using native technologies; both versions of the application displayed comparable location support and compliance with required features, while the web version superseded its native counterpart in cross-platform functionality.

8. Research on the Application of University Map Design Based on GIS Technology (2021) by Xiu-qin Liu, Siyu Jia, Yang Cao. This paper discusses the use of GIS technology in campus map design, emphasizing the aesthetic and practical aspects of map applications in university settings. The paper talks about the application development of a campus map and how it is a manifestation of the continuation of campus culture(Liu, Jia, & Cao, 2021). It also states that with the development and expansion of the campus, the map is not only practical but also

aesthetic, highlighting the cultural heritage of the university and showing the soft power of the university culture. In the emerging cultural and creative industries under the background of economic globalization, the image of colleges and universities has great room for creativity.

9. A Web-Based Campus Navigation System with Mobile Augmented Reality Intervention. This paper was published in the year 2021. This research focused on developing an AR-UUM Campus Navigation System accessible via mobile devices for guiding users within the UUM campus. It reviewed existing AR-based navigation systems and applications, highlighting different approaches like augmented arrows, interactive AR character agents, and virtual terrain modeling for improved navigation. The article identified a gap in indoor and outdoor navigation solutions on campuses, particularly the integration of real-world and virtual information to aid navigation. It emphasized the potential of AR in enhancing user experience by making navigation more interactive and user-friendly, addressing limitations in current navigation systems(Nordin, Markom, Suhaimi, & Ishak, 2021).

10. QR Code Generator Application To Access Information On The Campus Building from the International Journal of Computer Sciences and Mathematics Engineering Vol. 2 No. 2 (2023). The article was received on November 19, 2023, revised on November 22, 2023, and accepted on November 27, 2023. The research methodology involved developing an Android application that generates QR codes for each building on the Universitas Pembangunan Panca Budi campus. These QR codes, when scanned by a smartphone camera, provide immediate access to detailed information about the building, such as its location, list of amenities, and emergency contacts. The application was developed using Java programming on the Android Studio platform, and the information about the buildings is stored in the Firebase Realtime Database. The study included both the development of the application and its field testing within the Universitas Pembangunan

Panca Budi to ensure functionality and effectiveness. The research addresses a significant gap in providing accessible information about campus buildings to visitors, especially new students or first-time visitors. Prior to this application, obtaining information about campus buildings required seeking assistance from the administration or related parties, which was time-consuming. The study leveraged QR code technology, which is underutilized in the context of university campus buildings, to offer a convenient, efficient, and innovative solution for accessing building information.

11. Application of GIS in Campus Navigation presented at the 8th International Conference on Education Management Information and Management Society (EMIM 2018). The methodology involved in this study includes the development and design of a campus navigation system based on Geographic Information System (GIS) technology. The system comprises four main function modules: map query, campus navigation, campus environment preview, and campus profile. The development utilized SuperMap GIS 6.0 for spatial data processing and integration, component technology for modular system development, and a combination of SuperMap with Visual Basic 6.0 for the software development environment. This approach facilitated the embedding of map processing functions into the application, allowing it to operate independently of the SuperMap software platform. The system aimed to provide convenient services to school freshmen and campus visitors by offering easy access to campus information, including the optimization of route selection within the campus. The research addresses a significant gap in the application of GIS technology to campus navigation systems, particularly in China, where such applications are still rare and not mature enough. The study highlights the challenges in navigating increasingly large and complex university campuses, especially for new arrivals and visitors unfamiliar with the campus layout.

2.4.1 SUMMARY OF REVIEWED LITERATURES

From the above research works which have been discussed, we can see that the primary goal is the creation of an application that would allow users to navigate around a certain area. These applications leverage technologies like GIS, mobile mapping, real-time data integration, and cloud-based mapping solutions.

2.5 STRENGTH AND WEAKNESSES OF THE EXISTING SYSTEM

Here are some of the strengths of existing mapping applications currently in use today:

1. Comprehensive Mapping Data: Many map applications, especially those built on platforms like Google Maps, offer extensive and up-to-date mapping data, including street maps, satellite imagery, and real-time traffic information.
2. User-Friendly Interfaces: Most popular map applications provide intuitive and user-friendly interfaces, making it easy for users to navigate, search for locations, and access various map layers and features.
3. Mobile Accessibility: Map applications are often available on mobile devices, enabling users to access maps and navigation tools on the go. Mobile mapping contributes to real-time location-based services and enhanced user convenience.
4. Real-Time Data Integration: Many map applications integrate real-time data, such as live traffic updates, weather conditions, and crowd-sourced information. This enhances the accuracy and relevance of spatial information for users.

Now we discuss some of the weaknesses of existing map applications currently being used:

1. Limited Offline Functionality: While offline maps are available, the functionality is often limited compared to online mode. Users may experience restrictions in real-time traffic updates, dynamic data, and other features when using maps offline.

2. Data Privacy Concerns: Map applications often collect and utilize location data, raising concerns about user privacy. Users may be apprehensive about the storage and use of their location information, leading to privacy-related challenges.

3. Complexity in GIS Functionality: While some map applications offer GIS capabilities, these features can be complex for the average user. Understanding and utilizing GIS functionalities may require a certain level of expertise, limiting accessibility.

4. Limited Customization for University Context: Google Maps is a general-purpose mapping tool, and its customization options for specific university settings may be limited. Tailoring the application to the unique needs of University of Benin, such as custom overlays for academic buildings or specific landmarks, might be challenging.

2.6 GENERAL COMMENTS

Map apps are distinguished by their easy interfaces, providing platforms that revolutionize the way people access and traverse geographical data. These programs, which save travel time and increase overall convenience, make effective navigation possible by utilizing GPS technology and real-time traffic data. Users may explore areas all around the world thanks to their global accessibility, which makes them incredibly useful for scholars, vacationers, and anybody looking for information about other regions. The user experience is made more dynamic and pertinent by the incorporation of real-time updates, such as current weather and traffic data. Moreover, there are a variety of uses for maps in outdoor sports, emergency response, urban planning, and personal navigation. Map apps become all-inclusive solutions when they seamlessly integrate with other location-based services, such as ridesharing and restaurant reservations. Users like the user-generated content, including reviews and ratings, which improves the quality of the information offered. Still, there are issues, most notably with relation to security and privacy issues with location data collecting. Long-term battery usage on mobile devices—particularly when using GPS navigation—remains a major concern for users. Notwithstanding these difficulties, continuous innovation—such as the incorporation of AI and AR technologies—guarantees that map apps will always be vital and dynamic resources that enhance daily living and academic pursuits.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.0 INTRODUCTION

This project is driven by the imperative to modernize university campus navigation through the system analysis and design of a web-based map application. Traditional paper maps are proving inadequate, prompting the need for a dynamic, user-friendly solution. Current mapping systems lack user-friendliness and real-time updates, hindering effective navigation on campus. This project seeks to address these limitations by developing an intuitive web-based map application. Key objectives include creating a user-friendly interface, ensuring real-time updates, implementing accessibility features, integrating with existing university systems, and designing a scalable solution for future expansions. The project concentrates on the main ugbowo campus, with potential future phases considering other campuses or additional functionalities based on user feedback. Successfully implementing the web-based map application will significantly enhance campus navigation, fostering efficiency and community engagement among students, faculty, and visitors. The chapter is organized to guide through the system analysis and design process, covering methodology, design considerations, implementation, testing, and evaluation, with the conclusion summarizing findings and proposing future enhancements.

3.1 RESEARCH DESIGN

Research design refers to the structured plan or blueprint that outlines the systematic and logical approach to conducting research. It serves as a framework for the entire research process, guiding researchers in making decisions about the methods and procedures they will use to collect, analyze, and interpret data.

The research design for the development of a web-based map application for University of Benin main campus involves a multifaceted approach. Initially, the research will define the primary objectives, focusing on user-friendly design, real-time updates, accessibility features, and seamless integration with existing university systems. The methodology will include a systematic system analysis, involving interviews and surveys with stakeholders to understand navigation needs and an evaluation of the campus infrastructure. The technology stack will be carefully chosen to meet the application's requirements, with particular emphasis on user interface design, real-time data updates, and accessibility features. The implementation phase will transform design specifications into a working prototype, with iterative testing and refinement based on user feedback. Rigorous testing will be conducted to ensure functionality, usability, and accessibility, followed by an evaluation phase that gathers user feedback through surveys and analytics. The research design will conclude by summarizing key findings, discussing the success of the application in meeting objectives, and outlining potential future enhancements.

3.2 SOFTWARE DEVELOPMENT PROCESS MODEL

The proposed software development process that will be used for this project is the Agile process model but more specifically the Scrum Agile Framework. Scrum is a management framework that teams use to self-organize and work towards a common goal. It describes a set of meetings, tools, and roles for efficient project delivery[23]. Scrum is a framework for Agile software development that uses iterative and incremental methods[24]. Applying Scrum to the development of University of Benin Campus Map Application can bring numerous benefits, aligning well with the dynamic and evolving nature of such projects.

Benefits of Scrum Agile Framework

Scrum offers several benefits that contribute to its popularity as an agile project management framework, particularly in the context of software development. Here are some key advantages of using Scrum:

1. Adaptability to Change: Scrum is highly flexible and can easily accommodate changes in requirements, priorities, and technology. This adaptability is crucial in dynamic environments where customer needs and market conditions can evolve rapidly.
2. Customer Satisfaction: Frequent delivery of potentially shippable increments ensures that customers receive tangible value at the end of each sprint. Continuous feedback loops help in refining the product, leading to higher customer satisfaction.

3. Faster Time-to-Market: Scrum's iterative and incremental approach allows for the delivery of small, valuable increments at the end of each sprint. This results in a faster time-to-market compared to traditional, linear project management approaches.
4. Risk Management: Scrum's short, time-boxed sprints provide a structured approach to risk management. By delivering increments regularly, potential issues and risks are identified and addressed early in the development process.

Risks of Scrum Agile Framework

Here are some of the risks associated with Agile methodology:

1. Lack of Experience: Teams or organizations new to Scrum may face challenges due to a lack of experience and understanding of the framework. This can lead to misinterpretation of roles, ceremonies, and overall Scrum principles.
2. Overemphasis on Speed: While Scrum promotes iterative and fast development, there is a risk of overemphasizing speed at the expense of quality. Teams may prioritize delivering increments quickly but neglect essential quality assurance practices.
3. Incomplete Implementation: Implementing only certain aspects of Scrum without fully embracing its principles and practices can lead to suboptimal results. Incomplete implementation may result in a lack of the expected benefits.

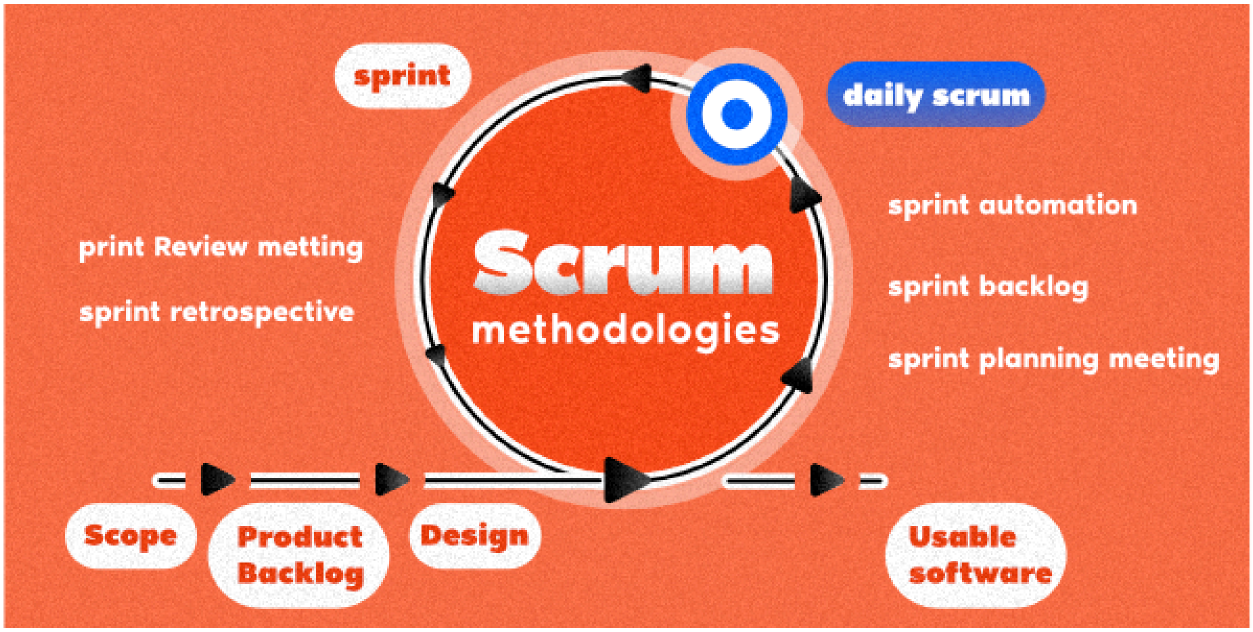


Figure 3. 1 Diagram of Scrum Agile Framework

3.3 REQUIREMENTS ELICITATION

Requirements are specifications or criteria that a product, system, or solution must meet to satisfy the needs and expectations of stakeholders. Requirements are divided into user and system requirements; system requirements are further divided into functional and non-functional requirements. User requirements describes the needs and expectations of end-users in terms of the system's functionality and usability. System requirements describe the overall characteristics and features of the system. Functional requirements describe the functions or features that a system or product must perform. Non-functional requirements specify criteria that are not directly related to the system's functions but are crucial for its overall performance and user satisfaction.

3.3.1 USER REQUIREMENTS

Users

Here are some of the user requirements of the proposed system:

1. Users should be able to search for specific buildings, departments, or points of interest.
2. Users should have access to detailed information about each building, including its purpose and operating hours.
3. Users should have the ability to provide feedback on map accuracy or report issues.
4. Users should be able to zoom in and out of the map smoothly and adjust their location view.

3.3.2 FUNCTIONAL REQUIREMENTS

The functional requirements of the system are:

1. The system should provide an interactive map of the university campus.
2. The system should allow users to search for specific locations on the campus.
3. The system should be able to display information about university buildings.
4. The system should provide detailed directions and turn-by-turn navigation for driving, walking, cycling, or public transportation.
5. The system should provide options for adding markers and labels to specific locations on the map.
6. The system should be able to measure distances and calculate routes between different points on the map.

3.3.3 NON-FUNCTIONAL REQUIREMENTS

Here are the non-functional requirements of the system:

1. The system should respond to the user's input within 2 seconds.

2. The mean time between failures (MTBF) should be at least 1000 hours.
3. The system should be available for atleast 20 hours a day, with scheduled maintenance communicated in advance.
4. The system should maintain acceptable response times with 1000 concurrent users accessing the map.
5. The system should be compatible with the latest versions of major web browsers.
6. The system should implement redundancy and failover mechanisms to ensure continuous service in case of server or component failures.

3.4 SYSTEM REQUIREMENTS

3.4.1 HARDWARE REQUIREMENTS

The hardware requirements for the university map application are:

1. The server hosting our map application should have a minimum RAM of 2gb and at least at least 4 CPUs.
2. The user device should have an Intel or AMD processor with 64-bit support; Recommended*:* 2.8 GHz or faster processor (Mobile /Desktop).
3. The user device should have at least 2gb of hard disk space.
4. The user requires a high-speed and reliable internet connection since our application relies on external mapping APIs or services.
5. The client devices should have up-to-date web browsers.

3.4.2 SOFTWARE REQUIREMENTS

The system software requirements for the Babcock University School Map Application are:

1. The application will be made using HTML, CSS, and JavaScript/TypeScript.
2. The system will make use of a NextJs Framework and Nestjs Framework.
3. The system will integrate with mapping APIs such as Google Maps API, Mapbox API, or Leaflet.

3.5 SYSTEM ARCHITECTURE

System architecture refers to the conceptual structure and high-level design of a complex system. It encompasses the arrangement and interconnection of its components, modules, and services, as well as the relationships and interactions between these elements.

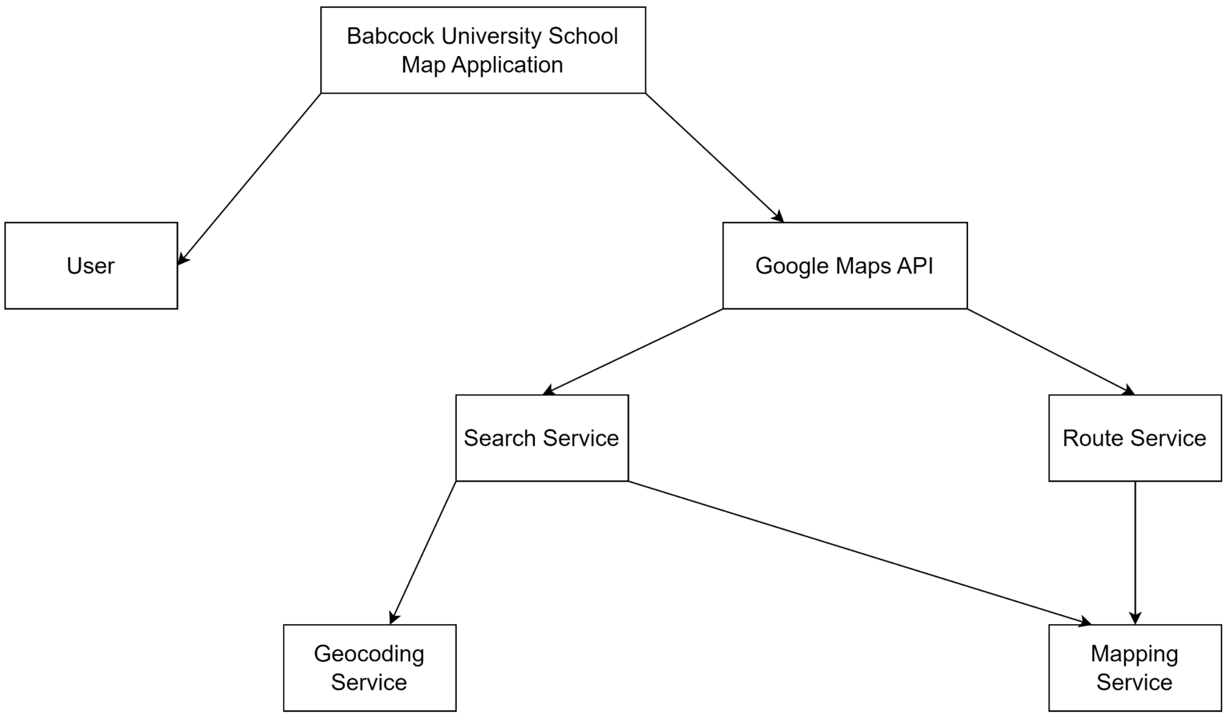


Figure 3. 2 System architecture Diagram of University of Benin Campus Map Application

The above diagram illustrates the system design or architecture for the University of Benin Campus Map Application. Let us now discuss on the individual components and how they are used in the system design:

1. User: User refers to an individual who accesses and interacts with a system or application using a mobile device, such as a smartphone or tablet. This user category encompasses individuals who access the system via a dedicated mobile app or a mobile-responsive web interface.
    a. User Interface (UI): The front-end of the application that users interact with. It includes the map display, search bar, and other user interface elements.
    b. Server-side program: The back-end of the application that handles logic, and connections with the database. This also provides an API for the frontend to access data.
2. Google Maps API Integration:
    a. Map Display: Utilizes the Google Maps API to render the interactive map within the application. The Google Maps API provides map tiles and various mapping functionalities.
    b. Search Service: Integrates with the Backend and postgres database to enable location-based searches. Users can search for specific buildings, departments, or other points of interest on the university campus.
    c. Route Service: Integrates with the Google Maps Directions API to generate and display routes between two or more locations on the map. This feature allows users to find the optimal path from one point to another within the university campus.

3. Communication between Components:
    a. The client-side application communicates with the Server side for user authentication, access control and information fetching.
    b. The Google Maps API is integrated into the client-side application to provide map rendering, search, and routing functionalities.

3.6 SYSTEM DESIGN

When carrying out system design, visual representation plays a pivotal role in conveying complex concepts, relationships, and processes. Diagrams serve as powerful tools to illustrate and communicate design decisions, system architectures, and the dynamic interactions within a software system. An important tool used is the UML (Unified Modelling Language) diagrams, which is widely used in software engineering to visually represent and communicate complex systems and processes. Here, we introduce the several key diagrams employed in system design of this project:

A. Flow Chart: A flow chart is a visual representation of a process or workflow, depicting the sequence of steps, decision points, and the flow of control.
B. Use Case Diagram: A use case diagram provides a high-level view of how users interact with a system, showcasing the system's functionalities from an end-user perspective. It identifies actors, use cases, and the relationships between them.
C. Sequence Diagram: Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration.

3.6.1 FLOW CHART DIAGRAM

Here is a flow chart diagram depicting the main components of the use of the application from user access the app to searching for a specific building or location.

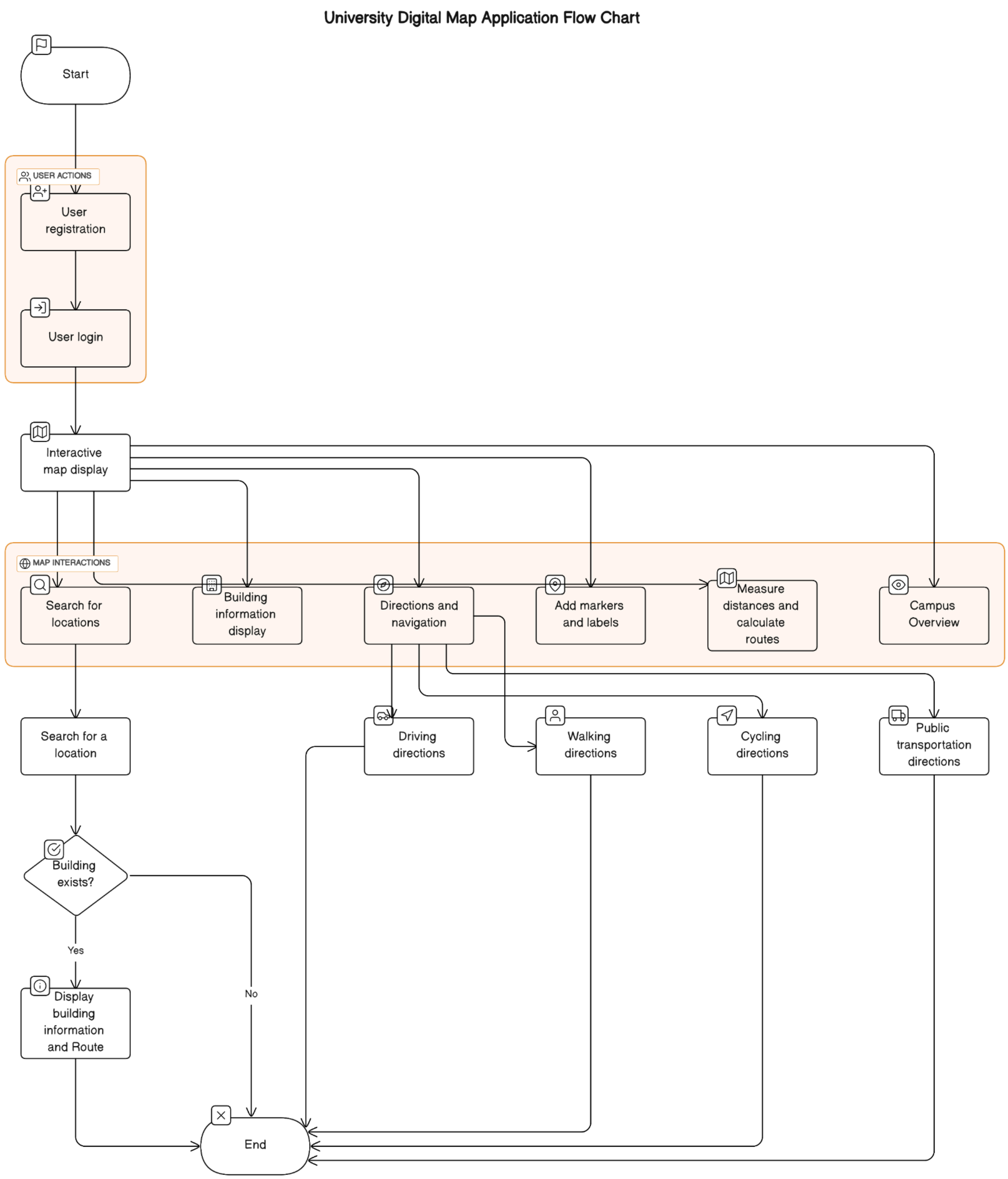


Figure 3. 3 Flow Chart Diagram of University of Benin Map Application

3.6.2 USE CASE DIAGRAM

Below is the use case diagram of the system showing the user, their use cases and how they interact with the system.

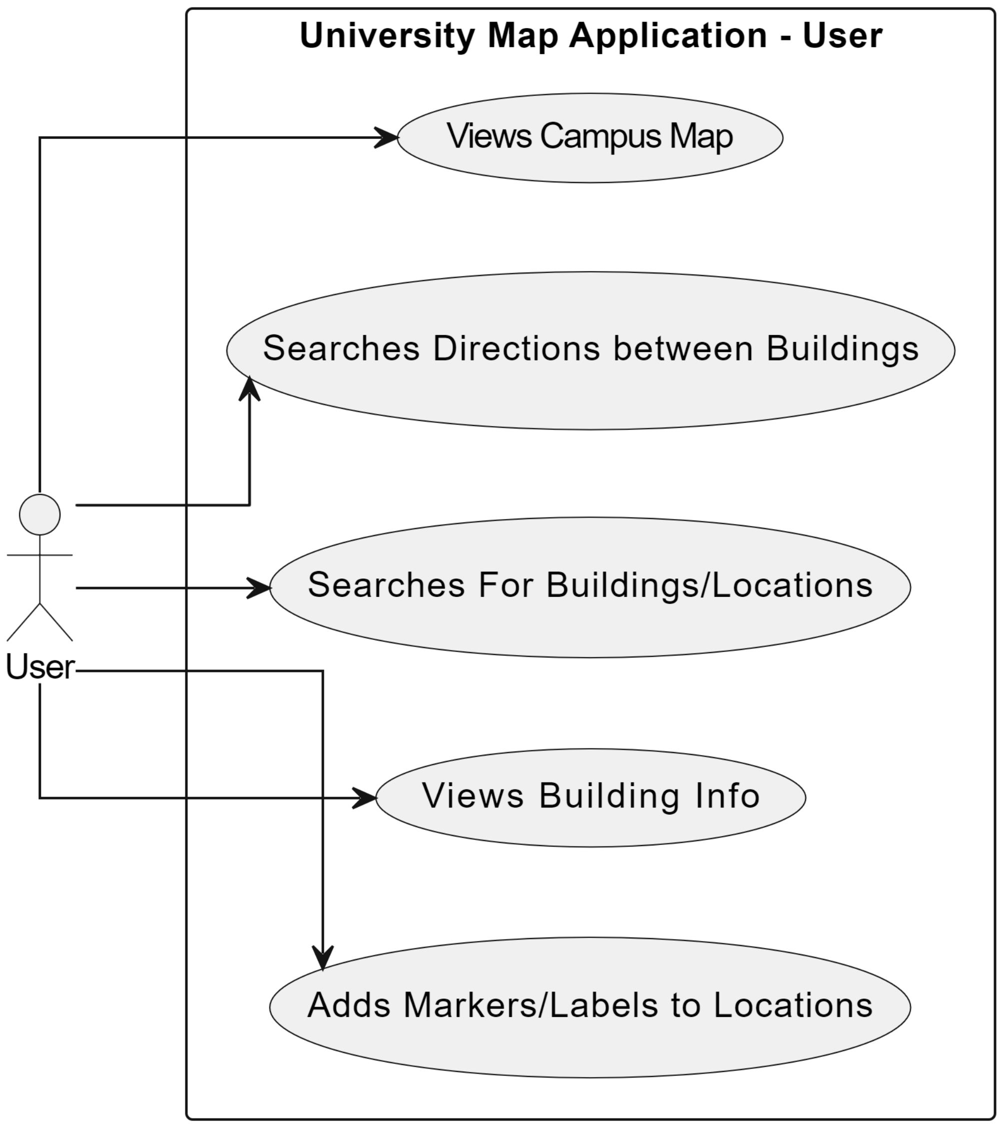


Figure 3. 4 Use Case Diagram of University of Benin Campus Map Application

3.6.3 SEQUENCE DIAGRAM

Here is the sequence diagram for when a user searches for particular location or building:

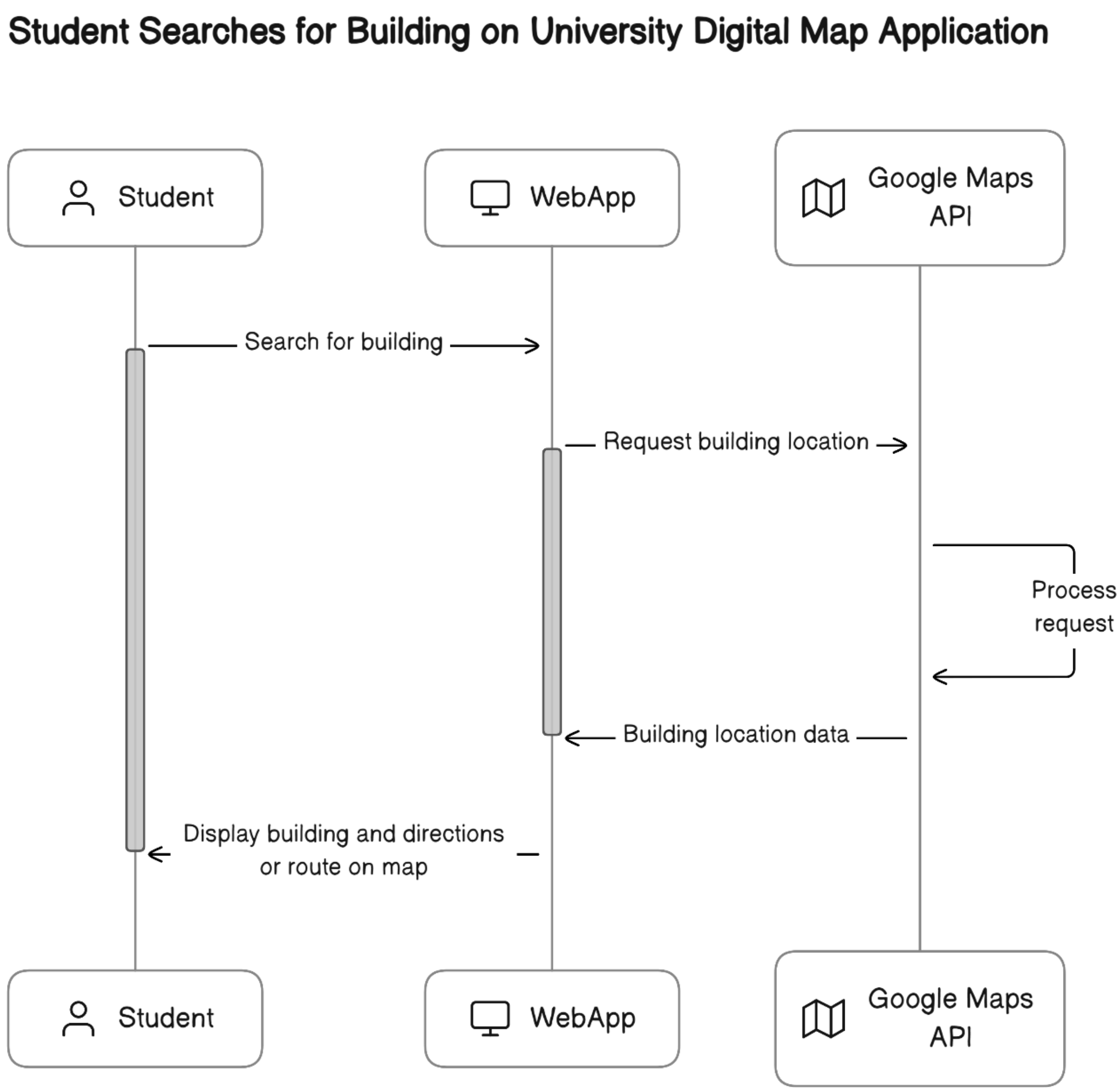


Figure 3. 5 Sequence Diagram of University of Benin Campus Map Application

3.7 DEVELOPMENT TOOLS

Development tools are software applications, libraries, frameworks, or services designed to assist developers in the process of creating, testing, debugging, and deploying software applications. These tools aim to streamline and enhance various aspects of the development lifecycle, making it more efficient, error-free and manageable.

For the development of the university map application the following tools will be used:

Sure, here are short notes on each of the mentioned technologies and tools:

1. HTML (Hypertext Markup Language): HTML is the standard markup language for creating web pages. It structures content on the web, using tags to define elements such as headings, paragraphs, images, links, and more.
2. CSS (Cascading Style Sheets): CSS is used for styling HTML documents. It controls the layout, appearance, and formatting of web pages, allowing developers to define styles for elements like colors, fonts, spacing, and positioning.
3. JavaScript: JavaScript is a dynamic scripting language used to make web pages interactive. It enables client-side functionality, allowing developers to manipulate the Document Object Model (DOM), handle events, and create dynamic content.
4. Typescript: TypeScript is a superset of JavaScript that adds static typing to the language. It helps catch errors early in the development process and improves code readability and maintainability by providing type annotations for variables, functions, and other elements of your code.

5. React: React is a JavaScript library for building user interfaces, particularly single-page applications where UI updates efficiently with changing data. It follows a component-based architecture, making it modular and reusable.
6. Google Maps API: Google Maps API enables developers to embed Google Maps on web pages. It provides various services, including displaying maps, adding markers, calculating routes, and accessing location-based data.
7. NextJs: Next.js is a React framework for building web applications, focusing on performance and developer experience. It provides features such as server-side rendering, static site generation, and routing out of the box, making it easy to create fast and dynamic web applications.
8. Tailwind CSS: Tailwind CSS is a utility-first CSS framework that allows you to quickly build custom designs without writing custom CSS. It provides a set of utility classes that you can use to style your elements, giving you more control over the design while maintaining a clean and maintainable codebase.
9. Visual Studio Code: Visual Studio Code (VS Code) is a free source-code editor developed by Microsoft. It supports various programming languages and features built-in Git integration, debugging support, syntax highlighting, and extensions for additional functionalities.

CHAPTER FOUR

SYSTEM IMPLEMENTATION

4.0 INTRODUCTION

The development of a web map application requires a structured approach to system implementation, encompassing various technical and operational components. This process involves the integration of geographic information system (GIS) technology, web development frameworks, and user interface design principles to create a seamless and interactive mapping experience. Key steps in the implementation include selecting appropriate mapping APIs, designing a user-friendly interface, ensuring responsive design for various devices, and incorporating features such as searching, filtering and grouping.

4.1 ARCHITECTURAL PATTERN

The proposed system implements a Model-View-Controller (MVC) architectural pattern which is widely used in software development to separate the presentation logic from the application logic. The MVC pattern provides a clear separation of concerns, making it easier to develop and maintain complex applications.

4.1.1 MODEL

This component represents the data and the business logic of the application. In the case of a web map application, the model would manage geographic data, user preferences, and other relevant information. It contains the logic for processing user requests, such as searching for a specific location, calculating routes, or filtering map features based on certain criteria. The model communicates with databases and external APIs to retrieve and update geographic data. It ensures that the data is consistently represented and easily accessible throughout the application.

4.1.2 VIEW

The view component is responsible for presenting the data to the user. It would handle the display of the map, user interface elements, and visual representation of the geographic data. The view for the proposed system is implemented using NextJs. The view is responsible for rendering the map on the user's screen. It uses mapping libraries in this case Google Maps API to display map tiles and vector data. The view also includes UI elements such as buttons, inputs, and checkboxes that allow users to interact with the map, toggle layers, or adjust map settings. It ensures that the map and UI elements are responsive and adapt to different screen sizes and devices, providing a seamless user experience across desktops, tablets, and smartphones.

4.1.3 CONTROLLER

The controller acts as an intermediary between the model and the view. It processes user inputs, such as map zooming, panning, and selection of map layers, and updates the view accordingly. The controller handles user inputs and events, such as clicking on the map, zooming in/out, or selecting a map layer. It processes these inputs and updates the view accordingly. It acts as a bridge between the model and the view, fetching data from the model based on user actions and updating the view to reflect changes in the data. The controller integrates logic for map-related functionalities, such as routing, geocoding, and spatial analysis, ensuring that these features are properly executed and presented in the view.

By employing the MVC pattern, a web map application can achieve a well-organized structure, where data management, user interface, and control logic are neatly separated. This not only enhances maintainability and scalability but also facilitates collaboration among developers working on different aspects of the application. It also facilitates the integration of mapping APIs and services, as the logic for handling these can be encapsulated within the controller and model components.

4.2 GRAPHICAL VIEW OF THE PROJECT

The graphical view of the proposed system i.e a web map application project is a crucial component that provides users with an intuitive and interactive representation of geographic information. It is designed to offer a seamless and visually appealing interface, enabling users to navigate, explore, and interact with the map effortlessly. Through the integration of advanced mapping libraries and technologies, the graphical view supports various functionalities such as zooming, panning, layer toggling, and displaying pop-up information.

4.2.1 WELCOME SCREEN

The welcome screen of the application serves as the initial interface that greets users upon launching the application. It is designed to provide a visually appealing and informative introduction to the application, setting the tone for the user experience. The welcome screen features the application's logo and a brief message or tagline that encapsulates the essence of the application. The purpose of the welcome screen is to establish a positive first impression. It acts as a bridge between the initial launch and the main interface of the application, ensuring a smooth transition for the user.

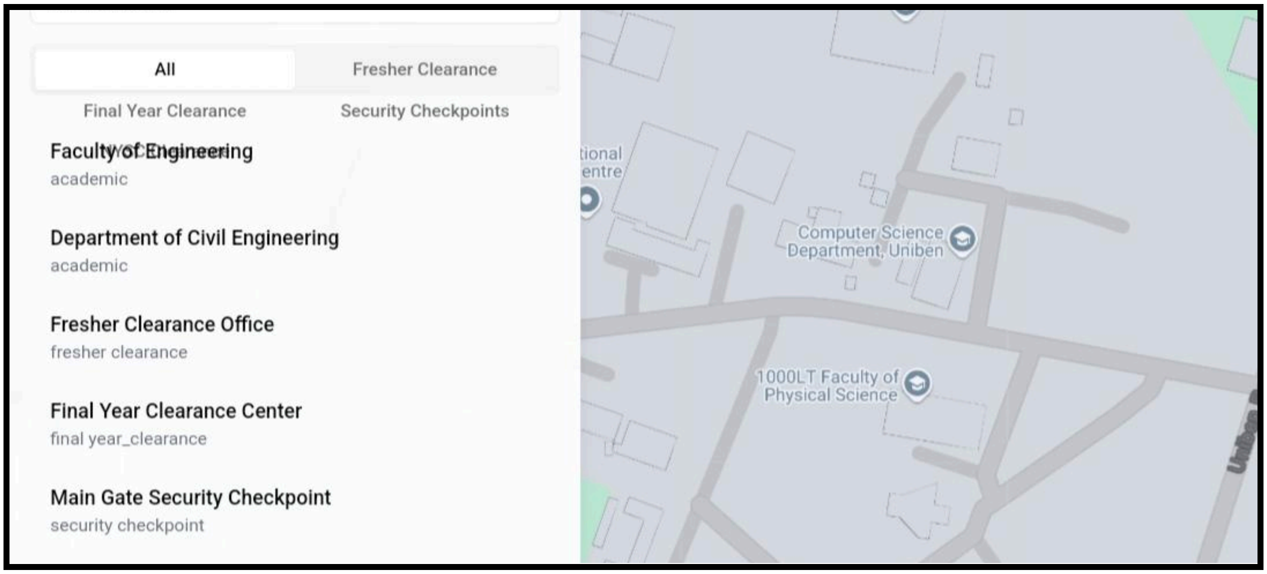


Figure 4. 1 Desktop View of Welcome Screen

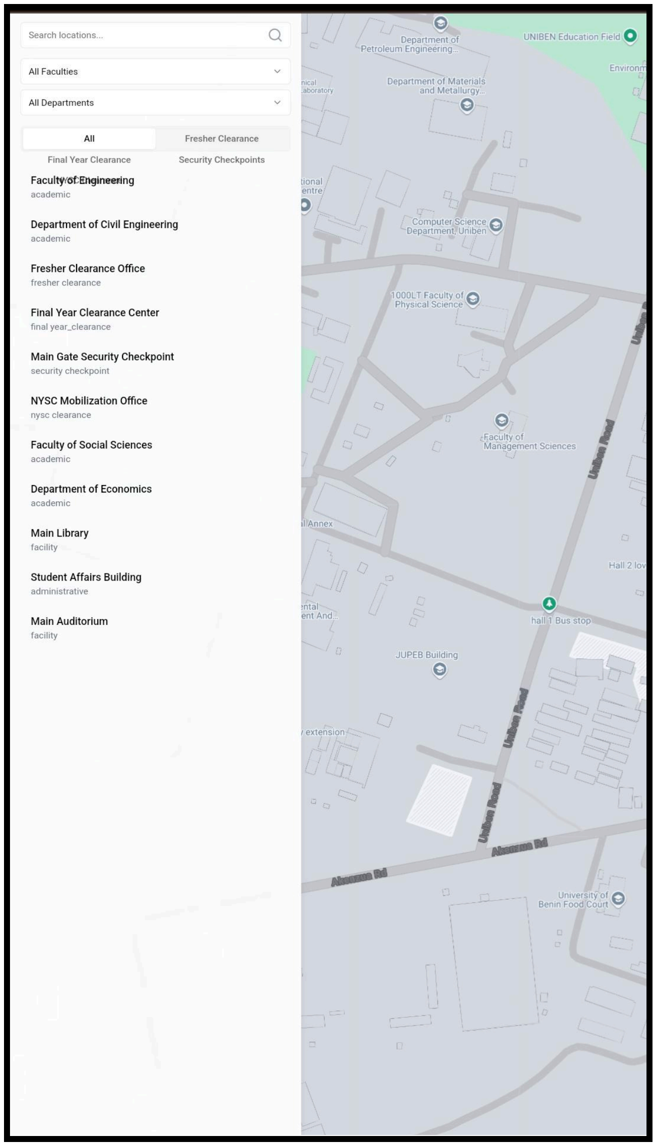


Figure 4. 2 Mobile View of Welcome Screen

4.2.2 MAIN SCREEN

The main screen of the application contains the core functionalities and features. Upon transitioning from the welcome screen, users are presented with a clean and intuitive interface centered around an interactive map. This map serves as the canvas for exploration and analysis, allowing users to pan, zoom, and interact with the given map. Here we have navigation tools, a set of controls for zooming in and out, resetting the map view, and navigating to specific locations. There is also a button that allows users to toggle different map layers on and off, such as satellite imagery or topographic maps. This feature enables users to customize their map view and focus on the information that is most relevant to them.

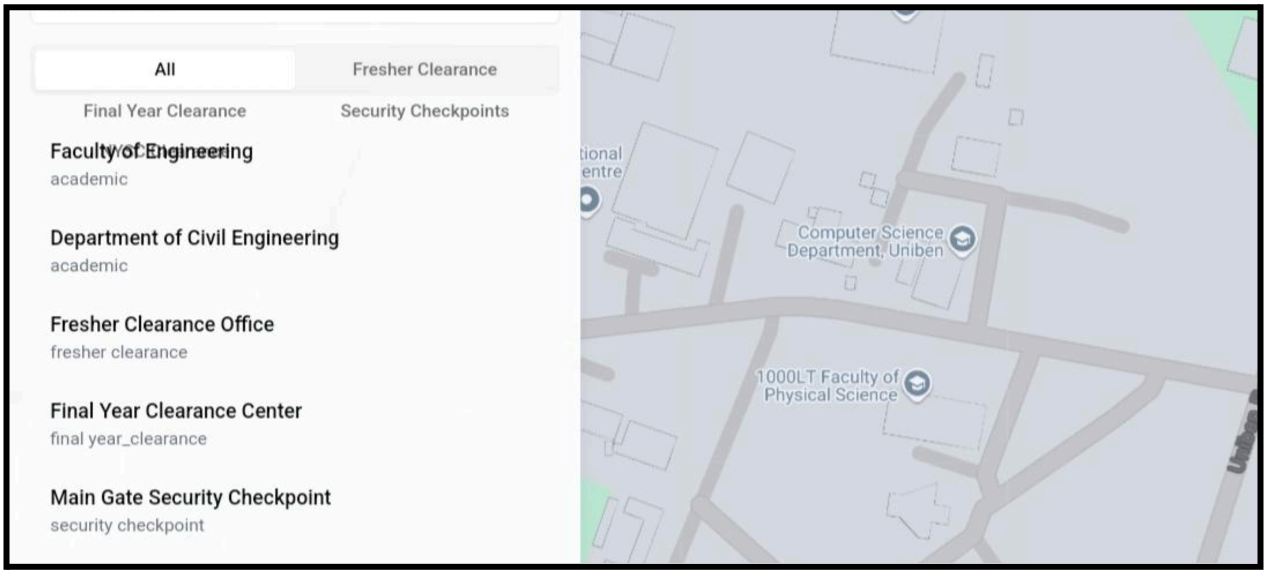


Figure 4. 3 Desktop View of Main Screen

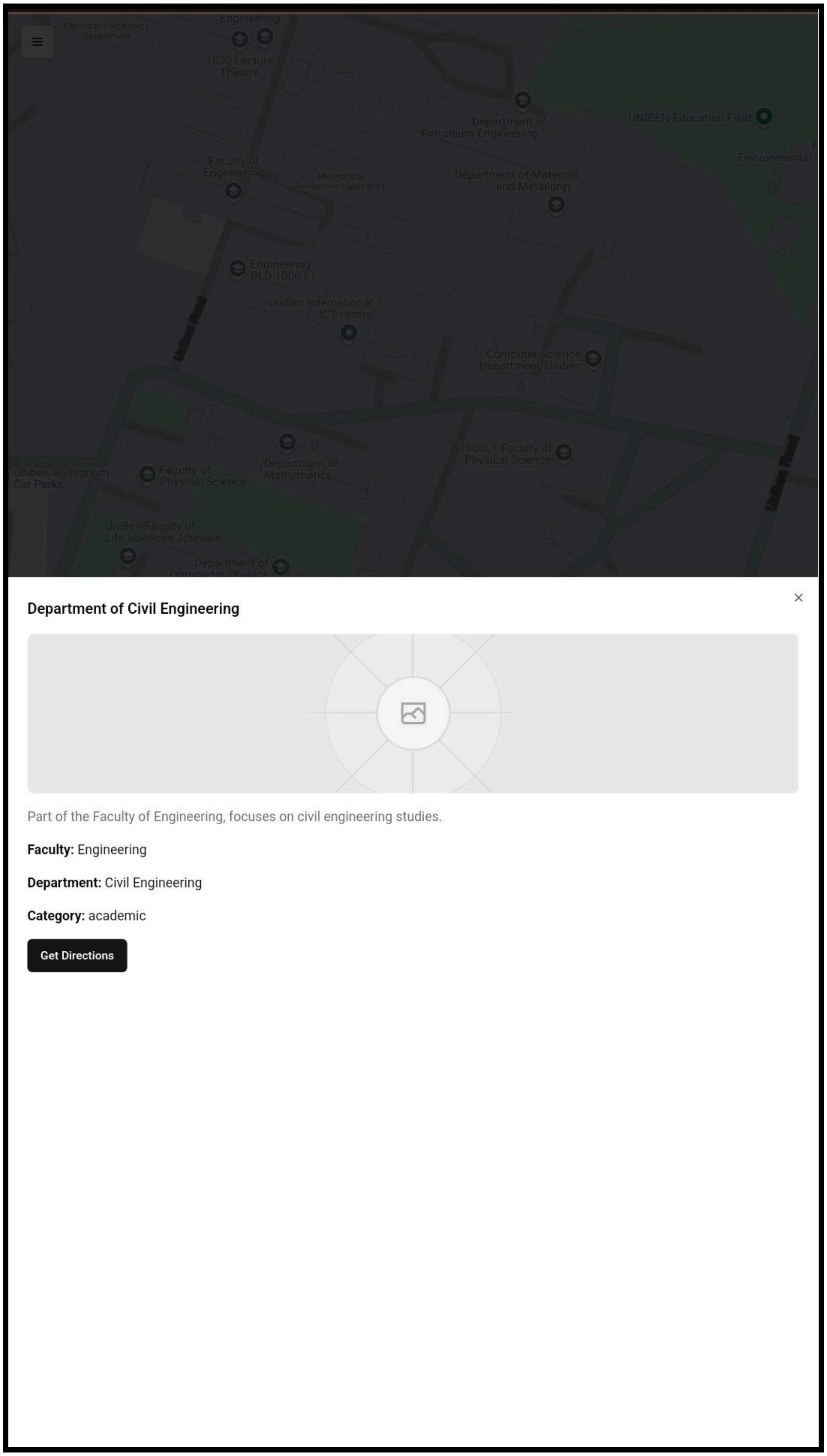

1000 Lecture Theatre
Department of Petroleum Engineering...
UNIBEN Education Field
Faculty of Engineering
Mechanical Production Laboratory
Department of Materials and Metallurgy
Engineering OLD 1000 LT
Uniben International ICT centre
Computer Science Department, Uniben
1000LT Faculty of Physical Science
Uniben Auditorium Car Parks
Faculty of Physical Science
Department of Mathematics
UniBen Faculty of Life Sciences Journals
Department of Civil Engineering
Part of the Faculty of Engineering, focuses on civil engineering studies.
Faculty: Engineering
Department: Civil Engineering
Category: academic
Get Directions

Figure 4. 4 Mobile View of Main Screen

4.2.3 MARKER COMPONENT

The Marker component is a key component of the map application, designed to enhance user interaction by allowing users to add markers to the map. This feature provides a personalized and dynamic mapping experience, enabling users to highlight specific locations, points of interest, or areas of importance. Users can add annotations or labels to their markers, providing additional information or context about the marked location. This can be done through a simple pop-up form that appears when a marker is placed or edited. This component takes in the marker name i.e the name the user wants associated with the marker and the name of the building the user wishes to be marked. It then displays a red marker to mark or point out the building or location on the map.

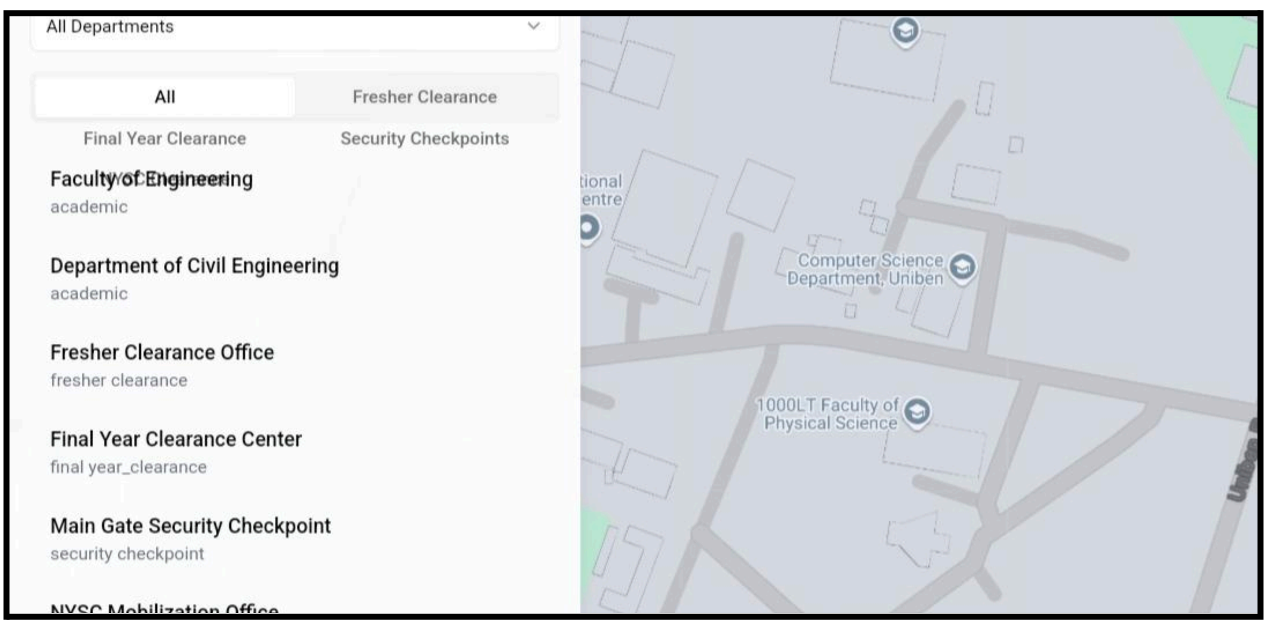


Figure 4. 5 Desktop View of Marker Component Screen

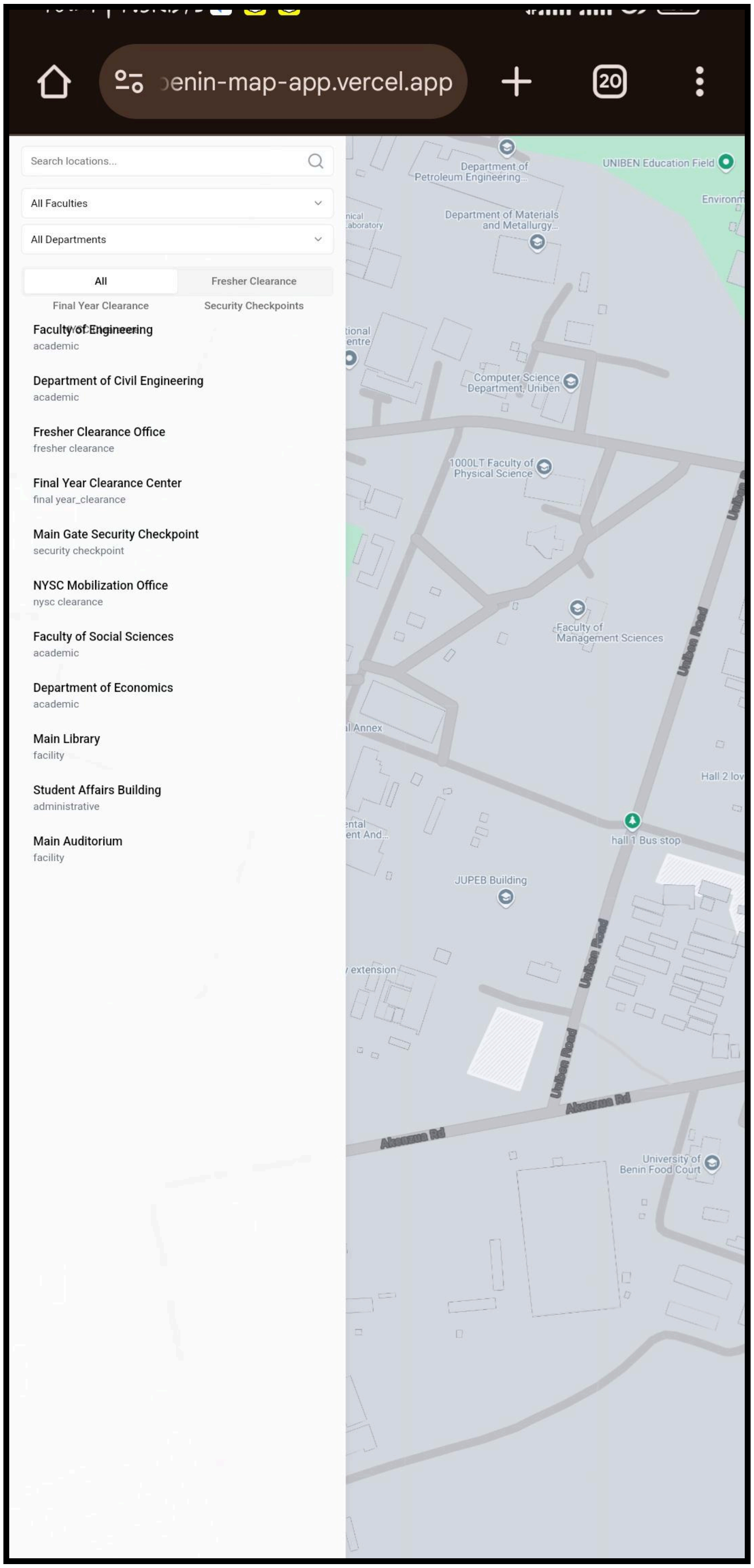
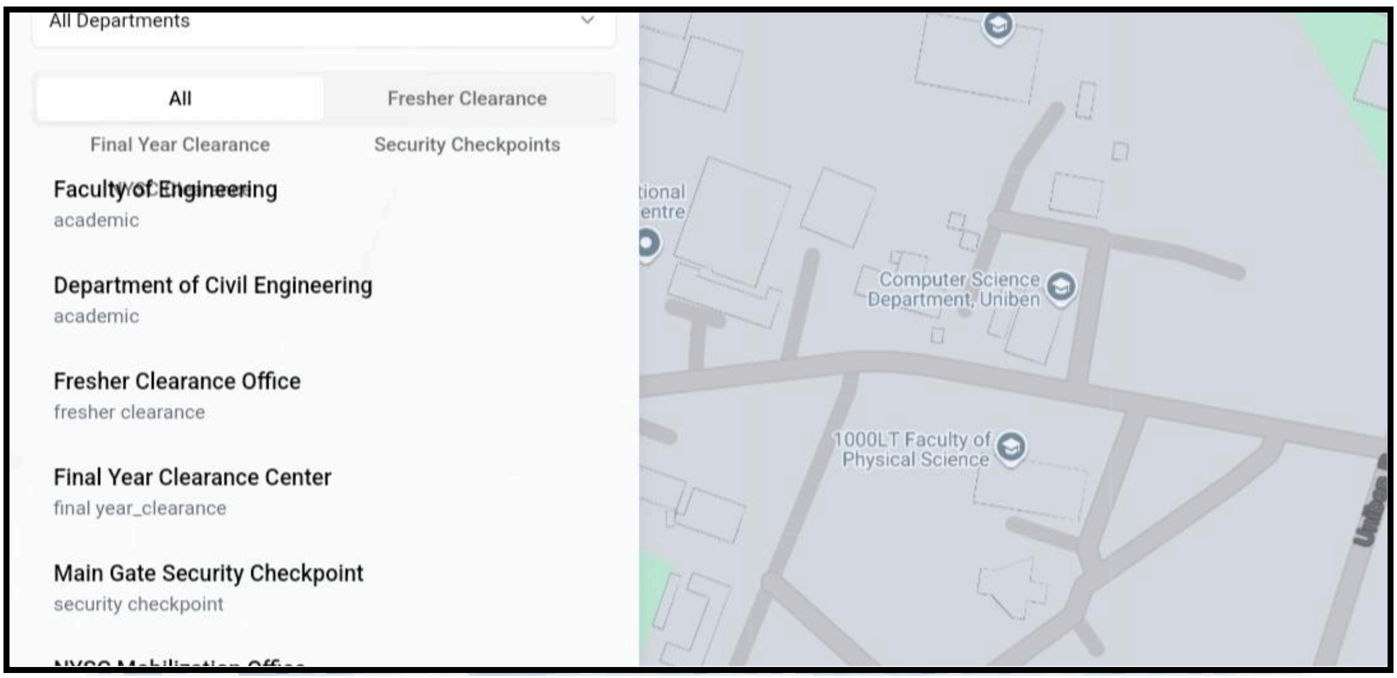


Figure 4. 6 Mobile View of Marker Component Screen

4.2.4 BUILDINGS SEARCH COMPONENT

The Building Information Search Component is a crucial feature of the web map application, designed to provide users with comprehensive details about buildings within the school. The component allows users to enter the names of buildings they wish to have information about and displays necessary information about the building such as the building name, address, an associated phone number etc. The component takes in a building name as input from the user and then returns and displays vital information about the building such as the building name, the address, the building type and a phone number.



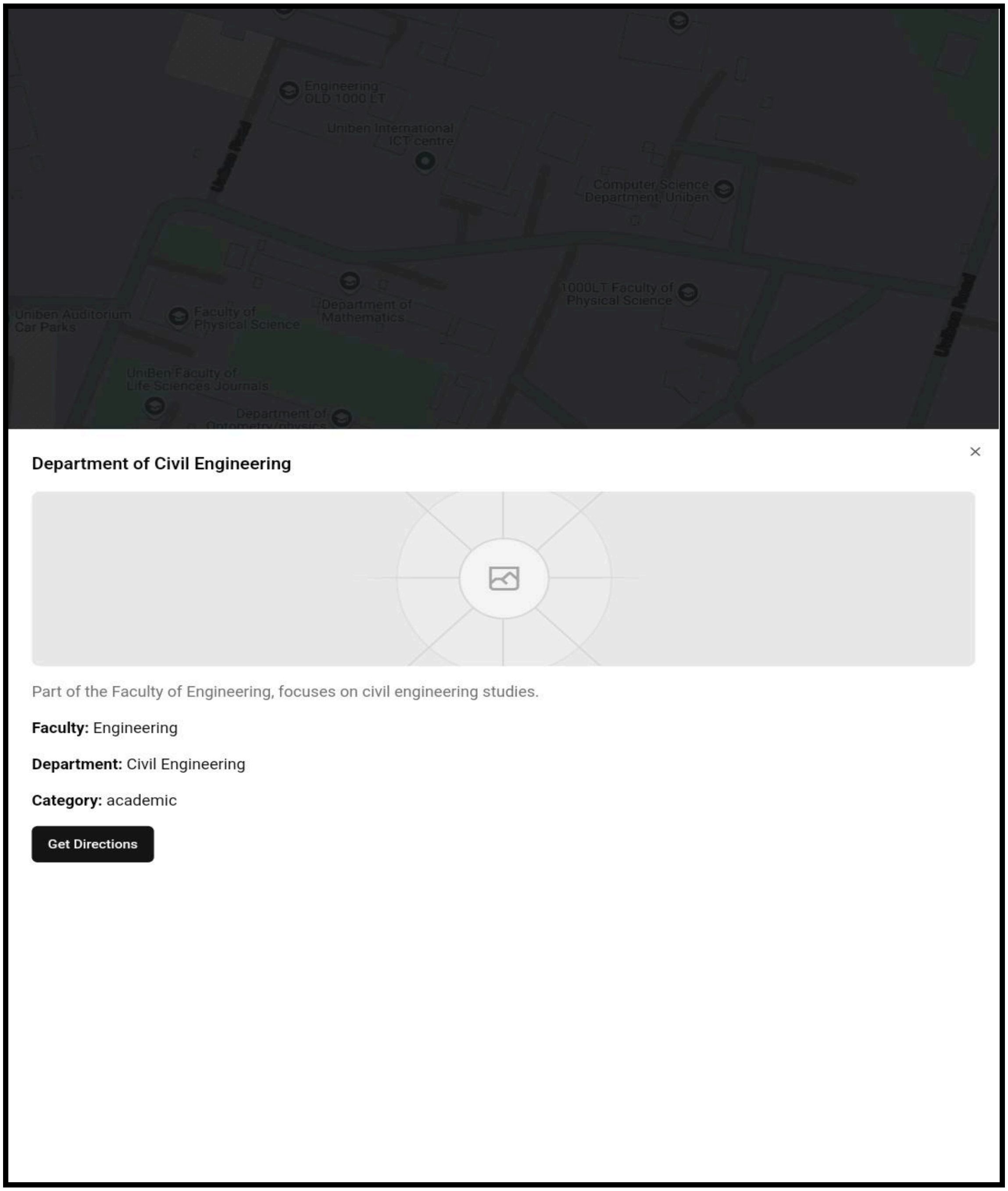
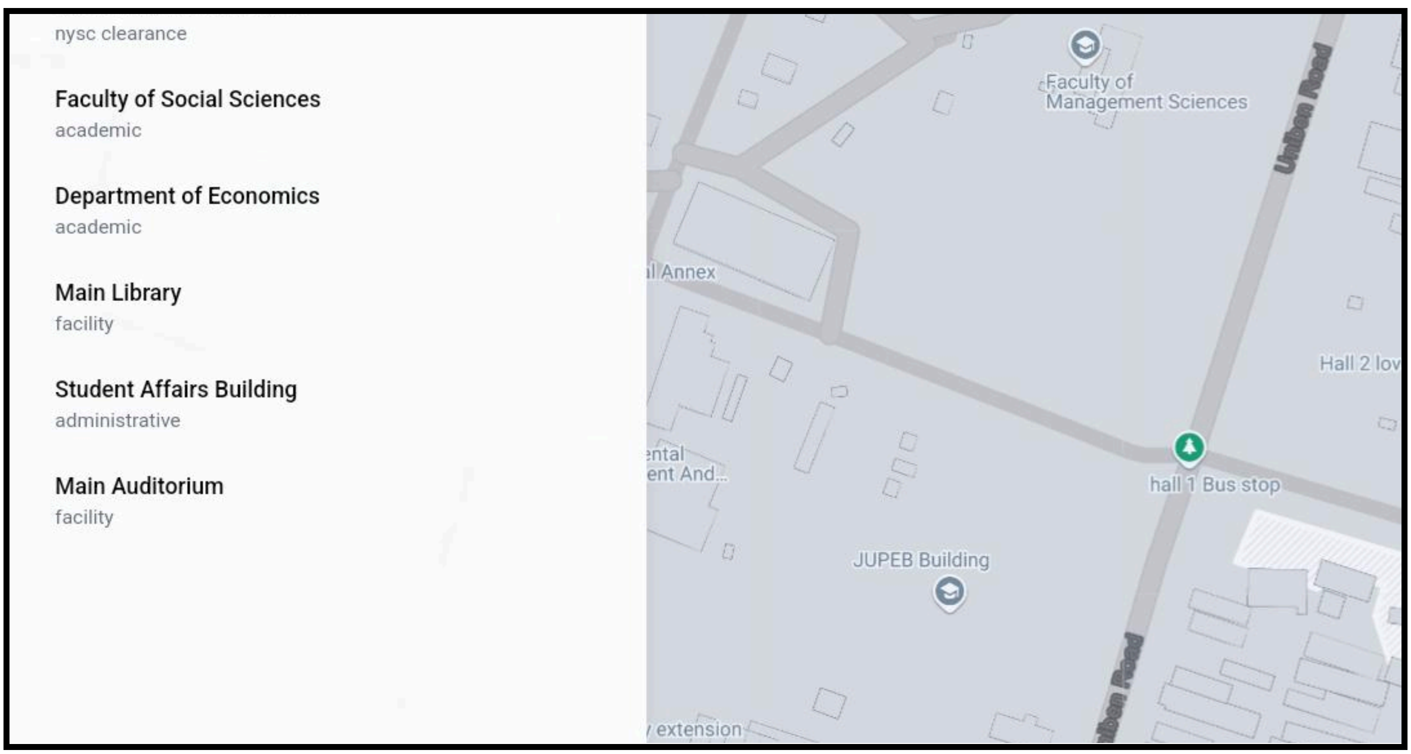


Figure 4. 8 Mobile View of Buildings Search Component Screen

4.2.5 DIRECTIONS COMPONENT

The Directions Tool is a vital component of our web map application, designed to provide users with the ability to determine the optimal route between two specified locations, namely the origin and the destination. This feature is indispensable for navigation and route planning, offering clear and concise directions that guide users from their starting point to their intended endpoint. The directions component takes both the origin or starting point and the destination and end point as input from the user. The user can also select the transportation mode which they would like to use such as walking, driving and cycling. The component then displays the most optimum route on the map and also displays the distance of the journey in kilometers as well as the estimated duration of the journey in minutes.



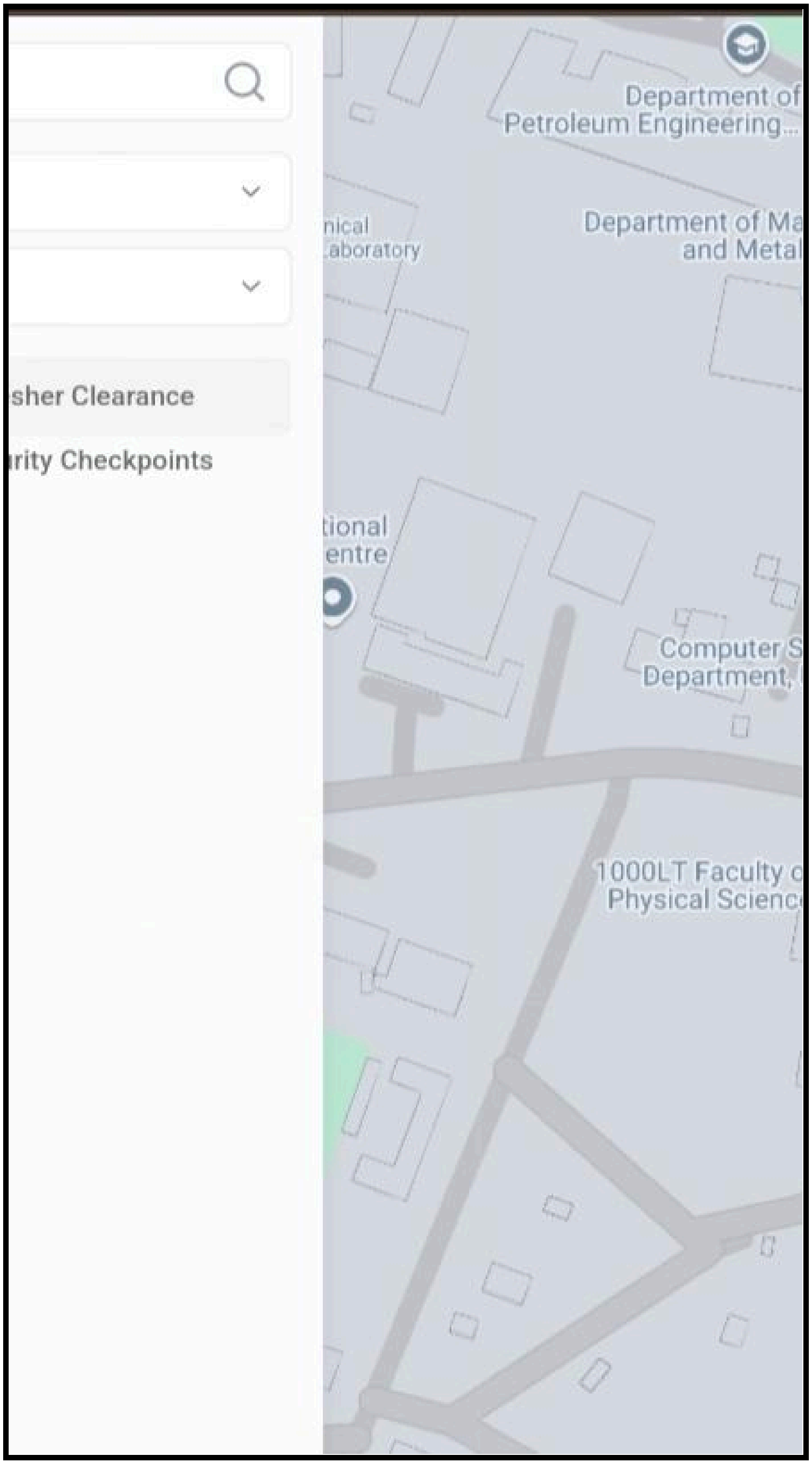


Figure 4. 10 Mobile View of Directions Component Screen

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATION

5.0 SUMMARY

The map application for University of Benin is essential for enhancing campus navigation and communication. It provides students, faculty, and visitors with an interactive tool to easily locate buildings, classrooms and other facilities. This can be particularly useful for new students and visitors who are unfamiliar with the campus layout, helping them find their way around more efficiently. This study focuses on designing and implementing a web-based map application using Geographic Information Systems (GIS) that would allow users to navigate between specific locations or points, search up information regarding a building and add markers to the map to note points of interest and other notable locations required of the user. For the system architecture of the application, selecting suitable technologies and frameworks for the frontend (e.g., mapping libraries, user interface components) is crucial for scalability and performance. Designing an intuitive user interface is also important, incorporating features such as zooming, panning, searching, and adding markers, along with layer toggling and pop-up information. There is also the integration with third-party APIs and services, such as geocoding and routing services, which is often necessary for additional features and interoperability. Testing of the application thoroughly across different devices and browsers helps to ensure functionality and compatibility.

5.1 RECOMMENDATION

The developed application focuses on a user-friendly interface suitable for students, staff, and visitors. Detailed mapping information, including buildings, classrooms and parking areas have also been provided. The application also features robust search functionality allowing users to easily find specific locations by name and additional filters for Faculty and Department. Mobile compatibility has also been considered in order to ensure access on the go.

5.2 CONCLUSION

In conclusion, the development of a map application for University of Benin campus is a crucial endeavor that can significantly enhance the overall campus experience for students, staff, and visitors. By providing a user-friendly interface, detailed mapping information, robust search functionality, and mobile compatibility, the application can simplify navigation and improve communication. Integration with campus systems, campus events and schedules and accessibility features are also key elements that could further enhance the quality of the project. Overall, the map application has the potential to greatly benefit the campus community by providing a comprehensive and efficient tool for navigating and accessing campus resources.

REFERENCES

Baros, T., & Stojanovic, T. (2015). Geographic information system (GIS) in mapping of mine suspected area in the Republic of Serpska. *Global Journal of Science Frontier Research, 15*(H3), 1–4.

Maddodi, B. S., Udayashankara, H. N., Raghavan, B. R., & Mathews, A. E. (n.d.). (2013). An assessment of the spatial relativity of health, literacy and socio-economic growth index using GIS techniques: A case study on Udupi south west coast of India, Karnataka. *Semantic Scholar*. Retrieved from https://api.semanticscholar.org/CorpusID:131629995

Quinn, S. (2016). *GIS essentials*. CreateSpace Independent Publishing Platform.

Esri Product Team. (2017). Understanding the science of where. *Esri Blog*. Retrieved from https://www.esri.com/arcgis-blog/products/product/uncategorized/understanding-the-science-of-where/

Dempsey, C. (2023). What is GIS? *GIS Lounge*. Retrieved from https://www.gislounge.com/what-is-gis/

Bipu, N. (2019). Geographic information system & spatial analysis. *International Journal for Empirical Education and Research*, *2*(2), 70–90.

Kumar, R. (2023). Unveiling the benefits of geographic information systems (GIS). *LinkedIn*. Retrieved from https://www.linkedin.com/pulse/unveiling-benefits-geographic-information-systems-gis-kumar-f rgs

Olamide, O. (2021). Evolution of GIS. *GeoInfotech*. Retrieved from https://geoinfotech.ng/evolution-of-gis/

Dempsey, C. (2015). A brief history of GIS. *GIS Lounge*. Retrieved from https://www.gislounge.com/history-of-gis/

GISGeography. (2015). The remarkable history of GIS. *GISGeography*. Retrieved from https://gisgeography.com/history-of-gis/

Lodha, K. G. (2021). Web GIS development starter guide - Geek culture. *Medium*. Retrieved from https://medium.com/geekculture/web-gis-development-starter-guide-e9ecd32e7fad

Mascitelli, A., Ravanelli, M., Mattoccia, S., Berardocco, C., & Mazzoni, A. (2020). A complete FOS approach for indoor crowdsourced mapping: Case study on Sapienza University of Rome faculties. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIII*-B4-2020, 361–365. https://doi.org/10.5194/isprs-archives-xliii-b4-2020-361-2020

Vaidya, S., Rahane, D., Gavhane, A., Shinde, V., Shaikh, K., & S. W., A. (2023). Android app for creating a map of the college to be used with visitor localization. *International Journal of Advanced Research in Computer and Communication Engineering, 12*(5). https://doi.org/10.17148/ijarcce.2023.125295

Hu, S., & Dai, T. (2013). Online map application development using Google Maps API, SQL database, and ASP.NET. *International Journal of Information and Communication Technology*

Research, 3.

Maidin, M. A. A., et al. (2021). Design campus map using OpenStreetMap digital software. In *Proceedings of the 3rd International Conference on Separation Technology*, Singapore, 113–129.

Hu, S. (2012). Online map service using Google Maps API and other JavaScript libraries: An open-source method. In *Lecture Notes in Geoinformation and Cartography* (pp. 265–278). Springer Berlin Heidelberg.

Mohsin, K., & Aldabbagh, O. (2014). Design and implementation of an online location-based service using Google Maps for Android mobile.

Granlund, D., Johansson, D., Andersson, K., & Brännström, R. (2013). A case study of application development for mobile and location-based services. In *Proceedings of the International Conference on Information Integration and Web-based Applications & Services.* https://doi.org/10.1145/2539150.2539202

Liu, X., Jia, S., & Cao, Y. (2021). Research on the application of university map design based on GIS technology. In *2021 The 3rd World Symposium on Software Engineering.* https://doi.org/10.1145/3488838.3488876

Nordin, N., Markom, M., Suhaimi, F., & Ishak, S. (2021). A web-based campus navigation system with mobile augmented reality intervention. *Journal of Physics: Conference Series, 1997*, 012038. https://doi.org/10.1088/1742-6596/1997/1/012038

Wahyuni, S., Lubis, A., & Purba, A. F. B. (2023). QR code generator application to access information on the campus building. *International Journal of Computer Sciences and*

Mathematics Engineering, 2(2), 216–224. https://doi.org/10.61306/ijecom.v2i2.44

Lin, D., & Li, B. (2018). Application of GIS in campus navigation. In *Proceedings of the 8th International Conference on Education, Management, Information and Management Society (EMIM 2018)*. https://doi.org/10.2991/emim-18.2018.72

Amazon Web Services. (n.d.). What is Scrum? Scrum methodology explained. *AWS Documentation*. Retrieved from https://aws.amazon.com/what-is/scrum/

L. Expert. (2021). Scrum methodology. *MDevelopers Blog*. Retrieved from https://mdevelopers.com/blog/scrum-methodology

Xu, A. (2022). Design Google Maps. *ByteByteGo Blog*. Retrieved from https://blog.bytebytego.com/p/design-google-maps

APPENDICES

Client-side code

PAGE.TSX

```
"use client"
import Link from "next/link"
import Image from "next/image"
import { useEffect, useState } from "react"
import LoadingSpinner from "./components/Loading-Spinner"

const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||
"http://localhost:3000"

export default function Home() {
  const [featuredCategories, setFeaturedCategories] = useState<any[]>([]);
  const [isLoading, setIsLoading] = useState(true)

  useEffect(() => {
    const fetchCategories = async () => {
      try {
        const response = await fetch(`${API_BASE_URL}/products/category`)
        const data = await response.json()
        setFeaturedCategories(data)
      } catch (error) {
```

```
      console.error("Failed to fetch featured categories:", error)
    } finally {
      setIsLoading(false)
    }
  }

  fetchCategories()
}, [])

if (isLoading) {
  return <LoadingSpinner />
}
return (
  <div className="space-y-8">
    <section className="text-center">
      <h1 className="text-3xl md:text-4xl font-bold text-green-600 mb-4">Welcome to Grocery
Shop</h1>
      <p className="text-lg md:text-xl text-gray-600">
        Fresh produce and quality groceries delivered to your doorstep.
      </p>
    </section>

    <section>
```

```
    <h2 className="text-2xl font-semibold mb-4">Featured Categories</h2>
    <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-4 gap-4 md:gap-6">
      {featuredCategories.map((category) => (
        <Link
          key={category.id}
          href={`/products?category=${category.id}`}
          className="bg-white shadow-md rounded-lg overflow-hidden hover:shadow-lg
transition-shadow duration-300"
        >
          <Image
            src={category.image || "/placeholder.svg"}
            alt={category.name}
            width={300}
            height={200}
            className="w-full h-40 md:h-48 object-cover"
          />
          <div className="p-4">
            <h3 className="text-lg font-semibold text-green-600">{category.name}</h3>
          </div>
        </Link>
      ))}
    </div>
  </section>
```

```
      <section className="text-center">
        <Link
          href="/products"
          className="inline-block bg-green-600 text-white px-6 py-3 rounded-md font-semibold
hover:bg-green-700 transition-colors duration-300"
        >
          Shop All Products
        </Link>
      </section>
    </div>
  )
}
```

THANK-YOU.TSX

```
import Link from "next/link"

export default function ThankYouPage() {
  return (
    <div className="text-center max-w-md mx-auto">
      <h1 className="text-2xl md:text-3xl font-bold mb-6">Thank You for Your Order!</h1>
      <p className="text-lg md:text-xl mb-8">Your order has been successfully placed and is
being processed.</p>
      <Link
```

```
        href="/"
        className="bg-green-600 text-white px-6 py-2 rounded-md font-semibold
hover:bg-green-700 transition-colors duration-300"
      >
        Continue Shopping
      </Link>
    </div>
  )
}
```

CART.TSX

```
"use client"
import Link from "next/link"
import Image from "next/image"
import { useEffect, useState } from "react"
import LoadingSpinner from "../components/Loading-Spinner"
import { useRouter } from "next/navigation"



const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||
"http://localhost:3000"
```

```
export default function CartPage() {
  const [cartItems, setCartItems] = useState<any[]>([]);
  const [isLoading, setIsLoading] = useState(true)
  let id = ''
  const router = useRouter()

  useEffect(() => {
    const fetchCartItems = async () => {
      id = localStorage.getItem("id")|| ""
      if (!id) {
        router.push("/login")
        return
      }
      try {
        const response =  await fetch(`${API_BASE_URL}/orders/cart`, {
          method: "GET",
          headers: {
            "Content-Type": "application/json",
            "Authorization": `Bearer ${id}`
          }
        })
        const data = await response.json()
        setCartItems(data)
```

```
    } catch (error) {
      console.error("Failed to fetch cart items:", error)
    } finally {
      setIsLoading(false)
    }
  }
  fetchCartItems()
}, [])

if (isLoading) {
  return <LoadingSpinner />
}

const total = cartItems.reduce((sum, item) => sum + item.price * item.quantity, 0)

return (
  <div>
    <h1 className="text-2xl md:text-3xl font-bold mb-6">Your Cart</h1>
    {cartItems.length === 0 ? (
      <div className="text-center py-10">
        <p className="text-lg text-gray-500 mb-4">Your cart is empty.</p>
        <Link href="/" className="text-green-600 font-semibold hover:underline">
          Continue Shopping
```

```
          </Link>
        </div>
      ) : (
        <>
          <div className="space-y-4">
            {cartItems.map((item) => (
              <div key={item.id} className="flex flex-col sm:flex-row items-center justify-between
border-b pb-4">
                <div className="flex items-center space-x-4 mb-2 sm:mb-0">
                  <Image
                    src={item.image || "/placeholder.svg"}
                    alt={item.name}
                    width={64}
                    height={64}
                    className="rounded-md"
                  />
                  <div>
                    <h2 className="text-lg font-semibold">{item.name}</h2>
                    <p className="text-gray-600">
                      ₦{item.price.toFixed(2)} x {item.quantity}
                    </p>
                  </div>
                </div>
```

```
              <p className="font-semibold">₦{(item.price * item.quantity).toFixed(2)}</p>
            </div>
          ))}
        </div>
        <div className="mt-6 text-right">
          <p className="text-xl font-bold">Total: ₦{total.toFixed(2)}</p>
          <Link
            href="/checkout"
            className={`inline-block mt-4 px-6 py-2 rounded-md font-semibold transition-colors
duration-300 ${
              cartItems.length === 0
                ? 'bg-gray-400 text-white cursor-not-allowed'
                : 'bg-green-600 text-white hover:bg-green-700'
            }`}
            aria-disabled={cartItems.length === 0}
            tabIndex={cartItems.length === 0 ? -1 : 0}
          >
            {cartItems.length === 0 ? 'Cart is Empty' : 'Proceed to Checkout'}
          </Link>
        </div>
      </>
    )}
  </div>
```

```
  )
}
```

ORDER.TXS

```
"use client"
import Link from "next/link"
import { formatDate } from "@/lib/utils"
import LoadingSpinner from "../components/Loading-Spinner"
import { useEffect, useState } from "react"
import { useRouter } from "next/navigation"

// Mock data for orders
// const orders = [
//   { id: "ORD001", date: "2023-05-15", total: 7530, status: "Delivered" },
//   { id: "ORD002", date: "2023-05-10", total: 4500, status: "Shipped" },
//   { id: "ORD003", date: "2023-05-05", total: 2090, status: "Processing" },
//   { id: "ORD004", date: "2023-04-30", total: 32980, status: "Delivered" },
// ]
const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||
"http://localhost:3000"

export default function OrdersPage() {
  const [orders, setOrders] = useState<any[]>([]);
```

```
const [isLoading, setIsLoading] = useState(true)
let id = ''
const router = useRouter()
useEffect(() => {
 id = localStorage.getItem("id")|| ""
 if (!id) {
  router.push("/login")
  return
 }
 const fetchOrders = async () => {
  try {
   const response = await fetch(`${API_BASE_URL}/orders`, {
    method: "GET",
    headers: {
     "Content-Type": "application/json",
     "Authorization": `Bearer ${id}`
    }
   })
   const data = await response.json()
   setOrders(data)
  } catch (error) {
   console.error("Failed to fetch cart items:", error)
  } finally {
```

```
        setIsLoading(false)
      }
    }
    fetchOrders()
  }, [])

  if (isLoading) {
    return <LoadingSpinner />
  }

  return (
    <div className="container mx-auto px-4 py-8">
      <h1 className="text-3xl font-bold mb-6">Your Orders</h1>
      <div className="bg-white shadow-md rounded-lg overflow-hidden">
        <table className="min-w-full divide-y divide-gray-200">
          <thead className="bg-gray-50">
            <tr>
              <th className="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase
tracking-wider">
                Order ID
              </th>
              <th className="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase
tracking-wider">
```

```
        Date
      </th>
      <th className="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase
tracking-wider">
        Total
      </th>
      <th className="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase
tracking-wider">
        Status
      </th>
      <th className="px-6 py-3 text-left text-xs font-medium text-gray-500 uppercase
tracking-wider">
        Actions
      </th>
    </tr>
  </thead>
  <tbody className="bg-white divide-y divide-gray-200">
    {orders.map((order) => (
      <tr key={order.id}>
        <td className="px-6 py-4 whitespace-nowrap text-sm font-medium
text-gray-900">{order.id}</td>
        <td className="px-6 py-4 whitespace-nowrap text-sm
text-gray-500">{formatDate(order.date)}</td>
```

```
                <td className="px-6 py-4 whitespace-nowrap text-sm
text-gray-500">₦{order.total.toFixed(2)}</td>
                <td className="px-6 py-4 whitespace-nowrap">
                  <span
                    className={`px-2 inline-flex text-xs leading-5 font-semibold rounded-full
${order.status === "Delivered"
                        ? "bg-green-100 text-green-800"
                        : order.status === "Shipped"
                          ? "bg-blue-100 text-blue-800"
                          : "bg-yellow-100 text-yellow-800"
                      }`}
                  >
                    {order.status}
                  </span>
                </td>
                <td className="px-6 py-4 whitespace-nowrap text-sm font-medium">
                  <Link href={`/orders/${order.id}`} className="text-green-600
hover:text-green-900">
                    View Details
                  </Link>
                </td>
              </tr>
            ))}
```

```
        </tbody>
      </table>
    </div>
  </div>
 )
}
```

CHECKOUT.TXS

```
"use client"
import Link from "next/link"
import Image from "next/image"
import { useEffect, useState } from "react"
import LoadingSpinner from "../components/Loading-Spinner"
import { useRouter } from "next/navigation"



const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||
"http://localhost:3000"

export default function CartPage() {
  const [cartItems, setCartItems] = useState<any[]>([]);
  const [isLoading, setIsLoading] = useState(true)
```

```
let id = ''
const router = useRouter()


useEffect(() => {
  const fetchCartItems = async () => {
    id = localStorage.getItem("id")|| ""
    if (!id) {
      router.push("/login")
      return
    }
    try {
      const response =  await fetch(`${API_BASE_URL}/orders/cart`, {
        method: "GET",
        headers: {
          "Content-Type": "application/json",
          "Authorization": `Bearer ${id}`
        }
      })
      const data = await response.json()
      setCartItems(data)
    } catch (error) {
      console.error("Failed to fetch cart items:", error)
    } finally {
```

```
      setIsLoading(false)
    }
  }
  fetchCartItems()
}, [])

if (isLoading) {
  return <LoadingSpinner />
}

const total = cartItems.reduce((sum, item) => sum + item.price * item.quantity, 0)

return (
  <div>
    <h1 className="text-2xl md:text-3xl font-bold mb-6">Your Cart</h1>
    {cartItems.length === 0 ? (
      <div className="text-center py-10">
        <p className="text-lg text-gray-500 mb-4">Your cart is empty.</p>
        <Link href="/" className="text-green-600 font-semibold hover:underline">
          Continue Shopping
        </Link>
      </div>
    ) : (
```

```
<>
  <div className="space-y-4">
    {cartItems.map((item) => (
      <div key={item.id} className="flex flex-col sm:flex-row items-center justify-between
border-b pb-4">
        <div className="flex items-center space-x-4 mb-2 sm:mb-0">
          <Image
            src={item.image || "/placeholder.svg"}
            alt={item.name}
            width={64}
            height={64}
            className="rounded-md"
          />
          <div>
            <h2 className="text-lg font-semibold">{item.name}</h2>
            <p className="text-gray-600">
              ₦{item.price.toFixed(2)} x {item.quantity}
            </p>
          </div>
        </div>
        <p className="font-semibold">₦{(item.price * item.quantity).toFixed(2)}</p>
      </div>
    ))}
```

```
          </div>
          <div className="mt-6 text-right">
            <p className="text-xl font-bold">Total: ₦{total.toFixed(2)}</p>
            <Link
              href="/checkout"
              className={`inline-block mt-4 px-6 py-2 rounded-md font-semibold transition-colors
duration-300 ${
                cartItems.length === 0
                  ? 'bg-gray-400 text-white cursor-not-allowed'
                  : 'bg-green-600 text-white hover:bg-green-700'
              }`}
              aria-disabled={cartItems.length === 0}
              tabIndex={cartItems.length === 0 ? -1 : 0}
            >
              {cartItems.length === 0 ? 'Cart is Empty' : 'Proceed to Checkout'}
            </Link>
          </div>
        </>
      )}
    </div>
  )
}
```

LOGIN.TSX

```
"use client"

import Link from "next/link"

import Image from "next/image"

import { useEffect, useState } from "react"

import LoadingSpinner from "../components/Loading-Spinner"

import { useRouter } from "next/navigation"




const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||

"http://localhost:3000"


export default function CartPage() {

  const [cartItems, setCartItems] = useState<any[]>([]);

  const [isLoading, setIsLoading] = useState(true)

  let id = ''

  const router = useRouter()


  useEffect(() => {

    const fetchCartItems = async () => {

      id = localStorage.getItem("id")|| ""

      if (!id) {
```

```
      router.push("/login")
      return
    }
    try {
      const response =  await fetch(`${API_BASE_URL}/orders/cart`, {
        method: "GET",
        headers: {
          "Content-Type": "application/json",
          "Authorization": `Bearer ${id}`
        }
      })
      const data = await response.json()
      setCartItems(data)
    } catch (error) {
      console.error("Failed to fetch cart items:", error)
    } finally {
      setIsLoading(false)
    }
  }
  fetchCartItems()
}, [])

if (isLoading) {
```

```
  return <LoadingSpinner />
 }

 const total = cartItems.reduce((sum, item) => sum + item.price * item.quantity, 0)

 return (
  <div>
   <h1 className="text-2xl md:text-3xl font-bold mb-6">Your Cart</h1>
   {cartItems.length === 0 ? (
    <div className="text-center py-10">
     <p className="text-lg text-gray-500 mb-4">Your cart is empty.</p>
     <Link href="/" className="text-green-600 font-semibold hover:underline">
      Continue Shopping
     </Link>
    </div>
   ) : (
    <>
     <div className="space-y-4">
      {cartItems.map((item) => (
       <div key={item.id} className="flex flex-col sm:flex-row items-center justify-between
border-b pb-4">
        <div className="flex items-center space-x-4 mb-2 sm:mb-0">
         <Image
```

```
            src={item.image || "/placeholder.svg"}
            alt={item.name}
            width={64}
            height={64}
            className="rounded-md"
          />
          <div>
            <h2 className="text-lg font-semibold">{item.name}</h2>
            <p className="text-gray-600">
              ₦{item.price.toFixed(2)} x {item.quantity}
            </p>
          </div>
        </div>
        <p className="font-semibold">₦{(item.price * item.quantity).toFixed(2)}</p>
      </div>
    ))}
  </div>
  <div className="mt-6 text-right">
    <p className="text-xl font-bold">Total: ₦{total.toFixed(2)}</p>
    <Link
      href="/checkout"
      className={`inline-block mt-4 px-6 py-2 rounded-md font-semibold transition-colors
duration-300 ${
```

```
            cartItems.length === 0
              ? 'bg-gray-400 text-white cursor-not-allowed'
              : 'bg-green-600 text-white hover:bg-green-700'
          }`}
          aria-disabled={cartItems.length === 0}
          tabIndex={cartItems.length === 0 ? -1 : 0}
        >
          {cartItems.length === 0 ? 'Cart is Empty' : 'Proceed to Checkout'}
        </Link>
      </div>
    </>
  )}
 </div>
)
}
```

SIGNUP.TSX

```
"use client"


import { useState } from "react"
import { useRouter } from "next/navigation"
import Link from "next/link"
```

```
const API_BASE_URL = process.env.NEXT_PUBLIC_API_BASE_URL ||
"http://localhost:3000"

export default function SignupPage() {
  const [name, setName] = useState("")
  const [email, setEmail] = useState("")
  const [password, setPassword] = useState("")
  const [error, setError] = useState<string|null>(null);
  const router = useRouter()

  const handleSubmit = async (e: React.FormEvent<HTMLFormElement>) => {
    e.preventDefault()
    try {
      const response = await fetch(`${API_BASE_URL}/user/signup`, {
        method: "POST",
        headers: {
          "Content-Type": "application/json",
        },
        body: JSON.stringify({ email, password, name}),
      })

      const data = await response.json()
```

```
    if (!response.ok) {
      setError("An error occured")
      return
    }

    router.push("/login")
  } catch (error) {
    console.error("Failed to fetch orders:", error)
    setError("Invalid credentials")
    return
  } finally {

  }
  // Simulate signup process
  console.log("Signing up with:", name, email, password)
  router.push("/login")
}

return (
  <div className="max-w-md mx-auto">
    <h1 className="text-2xl md:text-3xl font-bold mb-6">Sign Up</h1>
    <p className="text-red-500 text-center">{error? error :''}</p>
    <form onSubmit={handleSubmit} className="space-y-4">
```

```
<div>
  <label htmlFor="name" className="block text-sm font-medium text-gray-700">
    Name
  </label>
  <input
    type="text"
    id="name"
    value={name}
    onChange={(e) => setName(e.target.value)}
    required
    className="mt-1 block w-full rounded-md border-gray-300 shadow-sm
focus:border-green-500 focus:ring-green-500"
  />
</div>
<div>
  <label htmlFor="email" className="block text-sm font-medium text-gray-700">
    Email
  </label>
  <input
    type="email"
    id="email"
    value={email}
    onChange={(e) => setEmail(e.target.value)}
```

```
            required
            className="mt-1 block w-full rounded-md border-gray-300 shadow-sm
focus:border-green-500 focus:ring-green-500"
          />
        </div>
        <div>
          <label htmlFor="password" className="block text-sm font-medium text-gray-700">
            Password
          </label>
          <
```