



School of Information Technology and
Engineering at the ADA University



School of Engineering and Applied Science
at the George Washington University

IMPLEMENTATION OF DATABASE ARCHITECTURE AND INTERFACE DESIGN
FOR THE
INTERACTIVE URBAN NAVIGATION SYSTEM OF ICHERISHEHER

A Thesis

Presented to the Graduate Program of Computer Science and Data Analytics
of the School of Information Technology and Engineering
ADA University

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Computer Science and Data Analytics
ADA University

By
Huru Algayeva

April, 2025

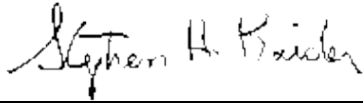
THESIS ACCEPTANCE

This Thesis by: Huru Algayeva

Entitled: *Implementation of Database Architecture and Interface Design for the Interactive Urban Navigation System of Icherisheher*

has been approved as meeting the requirement for the Degree of Master of Science in Computer Science and Data Analytics of the School of Information Technology and Engineering, ADA University.

Approved:



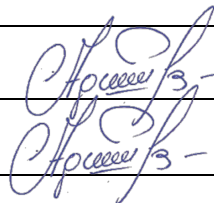
(Adviser)

Abzatdin Adamov

(Program Director)

Abzatdin Adamov

(Dean)



08.08.2025

(Date)

06.05.2025

(Date)

06.05.2025

(Date)

ABSTRACT

The study examines the development of an interactive urban navigation system to be relevant to historical urban areas, such as by the case study of Icherisheher, UNESCO World Heritage site in Baku, Azerbaijan. Indeed, the emphasis in this case, also, would be addressing major inefficiencies in the conventional navigation systems by not accommodating complex layout and cultural significance of historical areas. Here, the transformation from a traditional navigation system into an active exploration activity is achieved using innovative geocaching features, sophisticated database architecture, and an interface design user-centric.

The two basic technological innovations are comprehensive feature modelling system and an image validation system that uses computer vision to verify whether or not specific caches have been discovered once accuracy exceeds a pre-defined threshold. This way, gamification seeks to promote exploration fun but still encourage a larger cultural heritage engagement.

The design implementation is a configuration of database frameworks via Docker containerization technology, thus enabling the integration of a PostgreSQL database with PostGIS extensions for spatial data management and a Neo4J graph database for representing relationships into a viable dual database model that works well for handling complex geospatial queries and relational data.

The user interface places intuitive navigation as a primary objective, in addition to specifying multiple routes-as new, old, cultural, tourist-optimized, generous and scenic, real-time location tracking, and an engaging gamification system. Increases more user involvement in the geocaching interface dynamically, through inclusion of location triggers, image upload mechanisms, and progress tracking during geocaching game.

This research establishes the contours of urban informatics and human-computer interaction by demonstrating the idea that gamification elements indeed enhance the experience of cultural tourism. Deployment through Docker significantly increases the reliability and scalability of the system and enacts a data integrity policy through comprehensive validation and performance testing protocols. This framework thus offers a paradigm for similar implementations in historical urban areas across the globe in marrying technology innovation with cultural preservation for more interactive and informative navigation experiences.

Table of Contents

1	Introduction.....	1
1.1	Problem Statement.....	1
1.2	Definition of Terms.....	1
1.3	Significance of the Study.....	2
1.4	Limitations of the Study.....	2
2	Literature Review.....	3
2.1	Geospatial Data Modeling.....	3
2.1.1	Challenges in Data-Driven Geospatial Modeling.....	3
2.1.2	Spatial Data Representation and GIS Applications.....	3
2.2	Computer Vision in Geospatial Systems.....	3
2.2.1	Urban Analytics and Computer Vision.....	3
2.2.2	Image Validation Techniques.....	3
2.2.3	Geospatial Feature Detections.....	4
2.3	Neo4J and Spatial Graph Databases.....	4
2.3.1	Graph-Based Routing and Neo4J Performance.....	4
2.3.2	Learning Neo4J for Spatial Applications.....	4
2.4	Human-Computer Interaction (HCI) and User Experience (UX) Design.....	4
2.4.1	HCI Principles in Navigation Systems.....	4
2.4.2	Gamification and Interactive Design.....	5
2.5	React and User Interface Development.....	5
2.5.1	React for Web-Based Navigation Systems.....	5
2.5.2	UI/UX Design Strategies.....	5
2.5.3	GIS-Specific UI Design.....	5
3	Methodology and System Architecture.....	5
3.1	System Architecture.....	6
3.1.1	Overall Architecture Design.....	6
3.1.2	Database Architecture.....	7
3.1.3	Integration with Data Processing System.....	8
3.2	Data Structure and Models.....	8
3.2.1	Relational Data Models.....	9
3.2.2	Graph Data Model.....	10
3.3	Implementation Methodology.....	11
3.3.1	Docker Containerization.....	11
3.3.2	Database Implementation.....	12
3.3.3	Geocaching System Implementation.....	13

3.3.4	Image Validation System	14
3.3.5	Frontend Implementation	15
3.4	Testing Methodology	16
3.4.1	System Integration Testing Methodology	16
3.4.2	User Testing Methodology	17
4	Results and Analysis	18
4.1	System Implementation Results	18
4.1.1	Dual-Database Architecture Performance	18
4.1.2	Route Generation System	19
4.1.3	User Interface Implementation	21
4.1.4	Geocaching and Image Validation System	22
4.2	User Testing Results	24
4.2.1	User Demographics and Testing Process	24
4.2.2	User Interface Evaluation	25
4.2.3	Route Preference Findings	25
4.2.4	Geocaching Feature Engagement	25
4.3	Technical Performance Results	26
4.3.1	Database Performance	26
4.3.2	Route Calculation Performance	27
4.3.3	Image Validation System Performance	27
4.3.4	Docker Containerization Benefits	28
4.4	Limitations and Challenges	28
4.4.1	Data Quality and Coverage	28
4.4.2	Image Validation Challenges	28
4.4.3	User Testing Limitations	29
5	Conclusion and Future Work	29
5.1	Research Contributions	29
5.2	Limitations and Future Research Directions	30

LIST OF FIGURES

No	Figure Caption	Page
3.1	High-level system architecture of the interactive urban navigation system	7
3.2	Dual-database architecture with containerization	9
3.3	Integration between data processing and navigation systems	9
3.4	Relational data models class diagram	10
3.5	Basic Road-Intersection relationship in Neo4J graph model	11
3.6	Docker containerization architecture	12
3.7	Geocaching system implementation workflow	15
3.8	Image validation system processing flow	16
3.9	Frontend architecture and component structure	17
4.1	Neo4J graph visualization showing road network relationships	21
4.2	Quickest route visualization between two points in Icherisheher	22
4.3	Historical route visualization highlighting cultural landmarks	22
4.4	Geocaching route showing integration of cache locations	22
4.5	Main navigation interface showing category filtering and route options	23
4.6	Geocaching Panel	24
4.7a	Detailed Geocache interface	25
4.7b	Geocache validation upload interface	25
4.8	PostgreSQL/PostGIS spatial database visualization	28

LIST OF TABLES

No	Figure Caption	Page
4.1	Database Performance Metrics	20
4.2	Route Calculation Performance	29

1 INTRODUCTION

1.1 Problem Statement

Urban navigation within historically significant and culturally rich environments presents a special set of challenges that current map and navigation applications have failed to adequately address. Existing systems such as Google Maps and others do reasonably well in sending their users from point A to point B with basic routing information directly along the shortest distance. These systems have failed to integrate histories, culture, and interactivity to truly serve the exploration of more developed and culturally rich heritage sites. Therefore, the conservation of their own worth in history and culture should give sites such as Icherisheher—a historic medieval quarter of Baku, Azerbaijan—an enriched experience, whereas very few digital tools now exist to assist that development.

It narrates the research work in conceptual designing and developing an Interactive Urban Navigation System incorporates innovative geocaching elements to transform navigation into an exciting and interactive educational journey. The system is designed for the specific set of challenges such as real-time management of spatial data and an image validation system for verifying user-discovered caches. It attempts to create a structured, immersive, and interactive way for users to travel, explore, and appreciate the historical grandeur of Icherisheher using the PostgreSQL database for feature data stores and metadata management and Neo4J graph-based database for efficiently handling relationships between points of interest in such research work.

1.2 Definition of Terms

This section defines frequently used terms in this thesis.

1. **Interactive Urban Navigation System:** An advanced digital platform that would help users explore historically significant sites by using interactive features such as geocaching, real-time adaptation, and user engagement activities that will result in improving user experience on the whole.
2. **Geocaching:** An activity that is location-based and consists of hidden objects, historical locations, and clues using GPS coordinates and puzzles in revealing both historical monuments and cultural landmarks for a user to hunt through.
3. **Image Validation System:** An advanced computer vision-powered mechanism that rigorously examines and verifies user-submitted images of geocache discoveries, ensuring accuracy and authenticity based on predefined thresholds and accuracy metrics.
4. **PostgreSQL:** A powerful open-source technology for managing the relational database that would store all of the geospatial data in what can be termed a structured form, keeping it as points of interest, metadata, and historical sites.
5. **Neo4J:** A high-speed, graph database modeling of the relationships between landmarks, geocaches, and user interaction systems connected dynamically.
6. **React Framework:** It is one of the most extensively used JavaScript front-end framework that allows developers to cater to the development of scalable web applications managing live GPS tracking while also orchestrating geocaching interactions between users and the system.
7. **Gamification:** A strategic design methodology whereby game elements and mechanics—points, challenges, badges, rewards, etc.—are incorporated into a non-game context, is actually implemented in this navigation system to transform the classic view of urban exploration into an engaging interactive experience that motivates users, enhances knowledge retention regarding historical sites, and encourages deeper cultural involvement through achievement-oriented discovery.

1.3 Significance of the Study

The significance of this study arises from its contributions to computer science as far as evolving graph-based data modeling, feature extraction, computer vision, and real-time spatial data processing are concerned. By integrating several advanced computing techniques, it enables better urban navigation and provides a technologically sophisticated method for the exploration of historical sites. This study's major contributions are:

1. **Graph-Based Data Modeling:** Using Neo4J for dynamically modeling the relations of historical sites, geocaches, and user actions facilitates rapid queries for effective route optimization and personalized exploration paths.
2. **Computer Vision:** The image verification system makes use of the computer vision techniques for validating geocache finds in order to recognize historical artifacts and buildings as well as hidden objects under the geocaching framework.
3. **Real-Time Spatial Data Processing:** PostgreSQL with PostGIS gives efficient storage and retrieval of spatial data, which makes real-time navigation and geolocation tracking easy for use in the most historically congested environments.
4. **Scalability and Adaptability:** The research results can be applied to other urban environments and heritage sites, thus producing a fairly versatile framework for rich-featured historical navigation applications.

1.4 Limitations of the Study

Although the development and functioning of the Interactive Urban Navigation System has many advances, there are certain limitations in the research that need to be recognized and addressed:

1. **Image validation accuracy:** One area that has an impact on the success of computer vision-enabled image validation systems is that of external environmental conditions like lighting, the image resolution, and camera quality in terms of users; such factors may really affect the precision of images in the verification for geocache discoveries.
2. **Challenges in Real-Time Performance:** With a potential for having network connectivity issues, especially in areas with weak reception where GPS continues to be tracked in the real-time validity of the system, it brings some challenge to fluidity in the user experience.
3. **User Testing Sample Size:** The study will be conducted with a limited sample group of test users, which may not fully capture the diverse range of real-world users and their varied interaction styles with the system.
4. **Concerns regarding the data privacy and security:** As the system entails collection and processing of real-time location data, user interactions, and geocaching progress. There has to be a very good security system which will be made as per data management standards to protect sensitive information of users from unauthorized access and misuse.

Even so, this is an exceptional step toward developing methodologies in computer science-an area of research that includes graph-based modeling, feature extraction, and applications in computer vision. This research not only integrates some of the latest computational methods to advance urban navigation and historical exploration but also contributes to the long-term future development of intelligent, AI-driven, and feature-rich navigation systems that can be applied to many different settings in the real world. Such features make this research an avant-garde one of great scope, full of promise for future iterations, gradual improvements, and broader application context in various historical settings.

2 LITERATURE REVIEW

This literature review analyses exhaustively the theory and related work for the evolution of an interactive urban navigation system with geospatial data modeling, computer vision, graph databases, human-computer interaction (HCI), and user-centered design principles. The literature that has been reviewed covers multiple domains, offering insights into the challenges, methods, and developments that exist in each relevant field.

2.1 Geospatial Data Modeling

Geospatial Data Modeling is an important field of study for location-based services, which provides a structured way of storing, managing, and analyzing spatial data. Various techniques have been explored in the literature for operational efficiencies and accuracy of geospatial data.

2.1.1 *Challenges in Data-Driven Geospatial Modeling*

According to Koldasbayeva et al. (2023), the main challenges in geospatial modeling are integration of data, spatial heterogeneity, and high-performance computing for processing large datasets. The study emphasizes the need for sound architectures for managing data that combine relational database management systems and graph-based management systems.

2.1.2 *Spatial Data Representation and GIS Applications*

In the recent book entitled *Spatial Data Modelling: Mapping the World in Databases* authored by only Johnson and Smith (2022), storage and management practices of spatial data in databases are mostly determined in the context of applications for GIS. The IEEE paper titled "A Database Perspective on Geospatial Data Modeling," (Williams & Chen, 2021), also covers extensively the relational database architectures for geospatial applications with specific emphasis on PostGIS, an important tool that allows processing of spatial queries with a database. Case study on campus navigation published by Tregubova et al. (2022) is such a wonderful example showing the positivity of GIS integration into real-time navigation concerning spatial indexing and route optimization.

2.2 Computer Vision in Geospatial Systems

Computer vision, the artificial intelligence study that allows computers to extract meaningful information from digital images and video, is a major contributor in the geospatial arena, particularly with respect to validating location-based user interactions, automating feature detection, and improving urban analytics.

2.2.1 *Urban Analytics and Computer Vision*

Urban visual intelligence (as explored in Zhang et al., 2023) provides for a fresh perspective on mythology by unraveling the way deep-learned models extract significant urban features from street-view images. Another example is *A Review on Deep Computer Vision in Urban Analytics* (Lee & Garcia, 2022), dealing more with uses of deep learning vision technologies for urban planning on the automated object recognition.

2.2.2 *Image Validation Techniques*

Research in the paper *Validation of Geospatial Data Using Oblique Images* (Kumar et al., 2023) addresses the methods of improving the quality of validation for location-based images. A related study, *Image Feature Extraction Techniques* (Patel & Nguyen, 2022), reviews methods used to extract some

of the most significant features from images, which include edge detection, the Scale-Invariant Feature Transform (SIFT), and CNN-based embeddings.

2.2.3 Geospatial Feature Detections

(Rodriguez & Ahmed, 2023) describes a new A Deep Learning Framework for Urban Features for urban feature detection using convolutional neural networks (CNN). In detail, Computer Vision: Models, Learning, and Inference (Prince, 2022) details the conceptual framework regarding feature extraction, object recognition, and spatial analysis.

2.3 Neo4J and Spatial Graph Databases

Graph databases also offer efficiency of showing relationships amongst geographical datasets, especially when applications are about intricate spacial queries and web analysis.

2.3.1 Graph-Based Routing and Neo4J Performance

Works such as Scifo (2023) Hands-On Graph Analytics with Neo4j and Kemper's (2022) Beginning Neo4j concentrate on how Neo4J handles geospatial analytics. On the other hand, Festl et al. (2024), in Performance of Graph Database Management Systems as Route Planning Solutions, assess the effectiveness of graph-based routing solutions on performance and state their advantages over conventional relational databases for dynamic navigations.

2.3.2 Learning Neo4J for Spatial Applications

Learning Neo4j 3.x by Baton and Van Bruggen is the technical basis for using Neo4J in spatial applications. It further elaborates on indexing, querying, and integrating with graph-based spatial data.

2.4 Human-Computer Interaction (HCI) and User Experience (UX) Design

To ensure usability, accessibility, and engagement, navigation systems should rely on an effective interface. HCI and UX design are an important component of navigation systems in historic city settings, where cultural context should work in unison with wayfinding information. While minimizing load, the interface should maximize information transfer so that the user can keep their attention on the surroundings and not just on the device itself. The balance of virtual orientation versus real-world exploration is critical when applying interactive elements such as the geocaching medium that should enhance rather than compete with the experience of exploring historical areas.

2.4.1 HCI Principles in Navigation Systems

Two important theoretical texts about HCI principles with a strong focus on user experience for interactive map applications are The Encyclopedia of Human-Computer Interaction (Soegaard & Dam, 2022) and Human-Computer Interaction: Concepts, Methodologies, Tools, and Applications (Nakamura, 2023).

The Laugwitz, Held, and Schrepp (2008) User Experience Questionnaire framework provides standardized metrics to assess the quality of user experience in interactive systems. UEQ is a 26-item questionnaire that measures six dimensions: attractiveness, perspicuity, efficiency, dependability, stimulation, and novelty. Besides providing a comprehensive user experience assessment by measuring usability dimensions (efficiency, perspicuity, dependability) and user experience dimensions

(originality, stimulation), it also provides a comparison with a benchmarking database of previous evaluations (Schrepp et al., 2022). This provides an excellent basis upon which the assessment of navigation systems may be built.

2.4.2 *Gamification and Interactive Design*

The study called Geocaching: Interactive Communication Instruments around the Game (Ihamäki, 2023) experimentally examines the gamification of navigation systems to improve user experience. Research in urban informatics shows that certain elements, such as geocaching—they are the most effective—for improving user retention and exploration behavior (Anderson & Taylor, 2023).

2.5 React and User Interface Development

React, a widely used JavaScript framework, plays a significant role in building scalable and dynamic user interfaces for navigation applications.

2.5.1 *React for Web-Based Navigation Systems*

The books *Road to React* (Wieruch, 2023) and *Learning React: Modern Patterns for Developing React Apps* (Banks & Porcello, 2022) lay down the basics of building dynamic front-end applications. The component-based architecture of React beautifully lends itself to building real-time interactive maps.

2.5.2 *UI/UX Design Strategies*

The *Essential Guide to User Interface Design* (Galitz, 2023) and *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (Shneiderman et al., 2022) are two major texts having exhaustive guidelines for user interface design, stressing principles like responsiveness, minimalism, and accessibility.

2.5.3 *GIS-Specific UI Design*

The study *User-Centered Graphical User Interface Design for GIS* (Wilson & Henderson, 2023) outlines best practices for developing GIS-based user interfaces, highlighting the need for intuitive map interactions and real-time data visualization.

3 METHODOLOGY AND SYSTEM ARCHITECTURE

Design science research methodology is applied in this research to create an interactive urban navigation system specifically designed for Icherisheher, the historical center of Baku. The major objective includes addressing the challenges in maneuvering through complex historical urban environments, while enhancing the exploration experience through geographical plays and cultural contexts. The research covers actual designs, implementation, testing, and evaluation in a way that emphasizes both technical functionality and user experience.

The research process follows the following key phases:

1. Problem identification and requirements assessment
2. System architecture design and database modeling
3. System implementation and integration

4. System testing and evaluation

We did iterative development with the continued testing and refinement approach to implement it as a system. The approach permitted periods of changes and integrations based on early findings that would ensure that the system met the requirements in the most efficient manner.

3.1 System Architecture

System architecture forms the base of an interactive urban navigation system and delivers the structure of all features and functionalities. The architecture is targeted to fulfill the navigation system's most correct directions and also facilitate in cultural exploration by way of geocaching and contextual content.

A solution where a multitier paradigm with containerization is presented to maintain the scalability, maintainability, and deployment strategies. This technological application of multitier with containerization ensures not only an operational system with complete functionality, but also offers a proper structure for future scaling in such a way that it both applies no restrictions and allows for the ease and maintenance. The entire technology was developed through its components in almost every layer, but with individual purpose-specific instantiation to cater for the navigation system's specific requirements.

3.1.1 Overall Architecture Design

This high-level architecture contains a layered design pattern that separates presentation, application logic, and data management parts. This separation ensures modularity, maintainability, and scalability of the system, while facilitating independent development and testing of individual components.

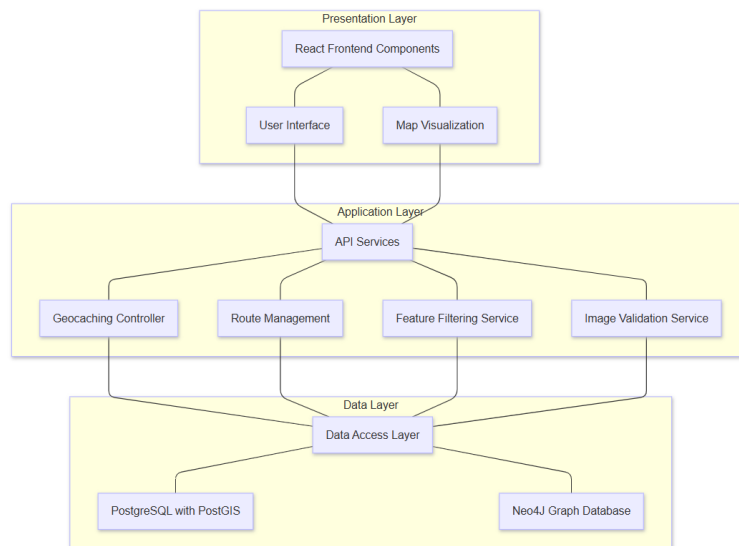


Figure 3.1: High-level system architecture of the interactive urban navigation system

The architecture consists of three primary layers:

1. Presentation Layer

It consists of dealing with the user's responses using a web interface made with React and contains:

- User interface components for navigation, filtering, and geocaching interactions
- The map visualization components use Leaflet for interactive mapping.
- Category filtering interface for different urban features.

- Route selection interface with multiple route options available.

2. Application Layer

The application layer contains the core business logic and service components:

- For managing client-specific request-response:
- Geocaching controller, managing challenges and validations.
- Route management services over different navigation paths'(quickest, historical, geocaching).
- Feature-filtering service displaying categorized points of interest.
- Image validation service using computer vision for geocache verification.

3. Data Layer

This layer manages data storage and access through a dual-database approach:

- The data access layer provides a universal interface to both database systems.
- PostgreSQL with PostGIS extension is used for spatial data management.
- Neo4J is a graph database to carry out path finding and route optimizations based on relations.

3.1.2 Database Architecture

An innovative feature of the architecture design is the adoption of a dual-database configuration to realize a highly complex requirement related to historical urban navigation. This provided the ability to exploit the different strengths of relational and graph databases with regard to the various aspects of the data needs of the system.

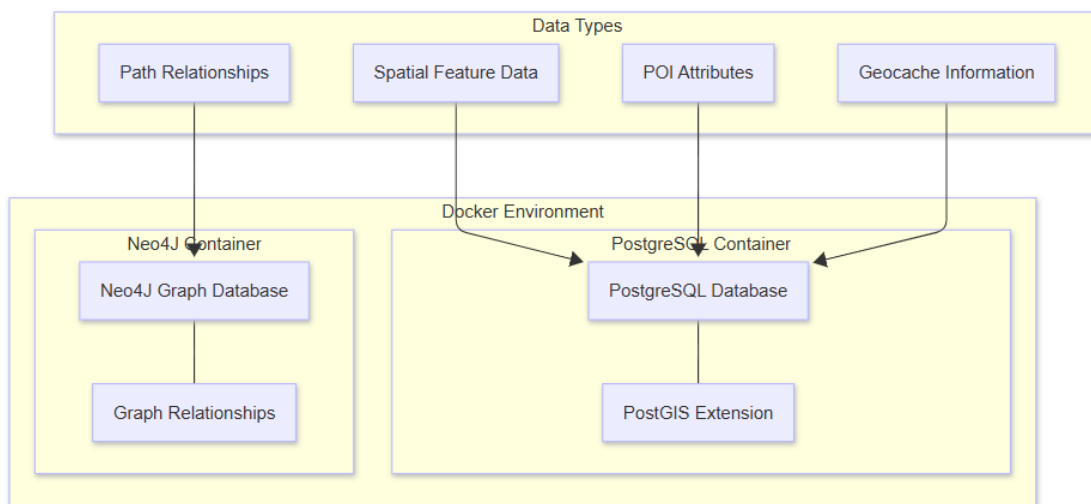


Figure 3.2: Dual-database architecture with containerization

The database architecture consists of:

1. PostgreSQL with PostGIS extension
 - Primarily used for geospatial data management with regard to features and points of interest
 - Implemented by containerization within Docker
 - Includes spatial data like buildings, roads, places of interest, and geocache locations
 - Utilizes PostGIS for spatial functions, indexing, and queries
 - Maintains attributes, metadata, and historical/cultural information
 - Enables feature filtering and category-based retrieval of information

2. Neo4J Graph Database:

- Main purposes are to manage relationships between locations for path computations
- Dual-port implementation (7474 for browser interface, 7687 for bolt protocol)
- Models the road network as a graph with nodes (intersections) and relationships (road segments)
- Stores relationship properties such as distance, travel time, and thematic weights
 - Caters to specialized routing algorithms for different types of routing:
 - Quickest Route: Fastest time
 - Historic Route: Some weightage to sites of historical importance
 - Geocaching Route: Routing will hence contain geocaching activities as well.

The dual-database architecture provides several advantages for the navigation system:

- PostgreSQL/PostGIS is ideal for storing and querying spatial features and their attributes.
- Neo4J's graph structure spatially lays out the naturally interconnected road networks.
- Specialized queries can be directed to the most appropriate database
- Each database could be finely tuned for its particular role in the system.

Docker containerization provides consistency in deployment environments, ensures straightforward dependency management, and improves scalability. Each database runs in its dedicated container with relevant configuration and persistent volumes for data, allowing development and production deployment.

3.1.3 Integration with Data Processing System

My project partner Abbas Aliyev has developed a data processing system that complements both the collection and processing aspects of this navigation system. This becomes the complete pipeline from raw data collection to user-facing navigation features.

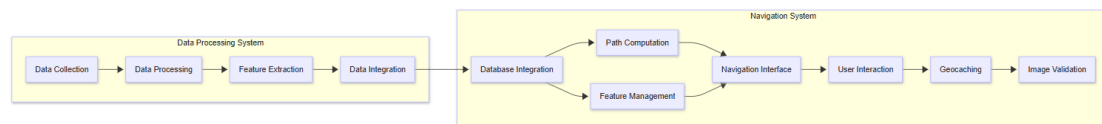


Figure 3.3: Integration between data processing and navigation systems

The integration workflow operates as follows:

1. Database Integration - The navigation system ingests the processed data:
 - Import of spatial features into PostgreSQL/PostGIS
 - Creation of graph structure in Neo4J
 - Validation of data integrity and relationships
2. System Functionality - The navigation system leverages the integrated data:
 - Path computation for multiple route types
 - Feature filtering and visualization
 - Geocaching challenges and image validation

3.2 Data Structure and Models

The data models and data structures of the system represent the underlying complex urban environment of Icherisheher, both for navigation and cultural exploration. Such models would

incorporate the diverse physical features of urban objects with their historical significance, interrelations, and relevance to navigation and geocache functionalities as well.

3.2.1 Relational Data Models

The system implements several key data models to represent urban elements, routes, and geocaching features. These models are implemented across both PostgreSQL and Neo4J databases to leverage their respective strengths.

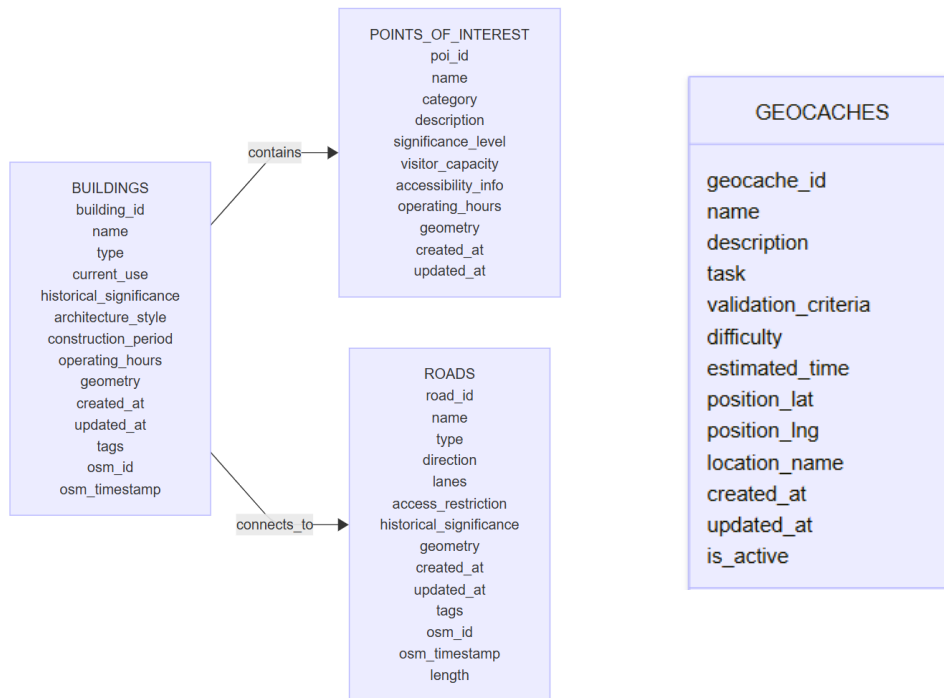


Figure 3.4: Relational data models class diagram

The primary data models include:

1. Building Features:
 - Unique identifiers with comprehensive classification information
 - Functional categorization (residential, commercial, religious, etc.)
 - Historical significance metrics for route weighting
 - Geometric representation using PostGIS data types
 - Architectural style and period documentation
 - Cultural importance indicators
2. Road Network:
 - Detailed street classification and naming
 - Physical attributes such as width and lane configuration
 - Geometric representation as linestrings
 - Historical significance for route weighting
3. Points of Interest:
 - Hierarchical categorization by type and function
 - Significance level metrics for filtering and prioritization

- Detailed descriptions including historical and cultural context
- Temporal information (operating hours, seasonal accessibility)
- Geometric positioning

4. Geocache:

- Difficulty classification (easy, medium, hard) based on estimated completion time
- Detailed instructions and historical context
- Validation criteria and thresholds
- Reference image storage for comparison
- Geometric location information
- Cultural and educational content

These data models provide crucial functionality for core system features like navigation, filtering, and geocaching. The models are implemented in PostgreSQL as tables and in Neo4J as nodes with appropriate relationships set up between related entities.

3.2.2 Graph Data Model

Neo4j utilizes a graph model for establishing topological relationships that will be required during path finding and route optimization. The model emphasizes the connection of locations and their relevant attributes according to routing algorithms.



Figure 3.5: Basic Road-Intersection relationship in Neo4J graph model

The graph data model represents the road network of Icherisheher as a network of nodes and relationships, optimized for pathfinding algorithms. The model consists of:

1. Nodes:

- **Roads:** Representing street segments with properties as shown in Image 2, including:
 - id: Unique identifier (e.g., "r_61")
 - length: Distance metric for path calculations
 - geometry: MULTILINESTRING representation of the road's spatial path
 - label: Road classification label
- **Intersections:** Representing connection points

2. Relationship:

ROAD_TO: Connecting roads to intersections

3. Properties:

ROAD_TO properties: length, road type, direction

This graph structure enables efficient path computation for different routing options:

- Quickest Route: Prioritizes relationships with minimal travel time
- Historical Route: Considers historical significance weights
- Geocaching Route: Incorporates geocache locations into path planning

The model of the graph is one that truly reflects the urban environment and its connections. This makes complex queries possible that may include not only distance but also several other factors. Such a model is especially useful for historical places like Icherisheher, where the quality of environment goes well beyond the physical distance from each other in cultural and historical contexts.

3.3 Implementation Methodology

Here, we are discussing the use of certain technical methodologies, tools, and processes that will aid in the development of the interactive urban navigation system. This section will detail the work done concerning the system implementation from database setup to the final stages of frontend development and feature implementation.

3.3.1 Docker Containerization

The system leverages Docker containerization technology to ensure consistent deployment environments, simplified dependency management, and improved scalability. This approach provides several advantages for both development and production deployment.

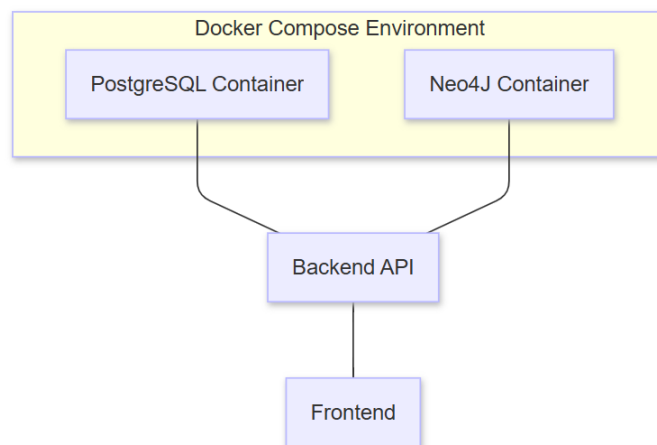


Figure 3.6: Docker containerization architecture

The containerization implementation includes:

1. **PostgreSQL Container:**
 - Base image: PostgreSQL 13
 - PostGIS extension installation
 - Configured port mapping
 - Persistent volume for data storage
 - Environment variables for initial database setup
2. **Neo4J Container:**
 - Base image: Neo4J 4.4 Enterprise Edition
 - Dual-port configuration (7474 for browser, 7687 for bolt)
 - Persistent volume for graph data

- Memory configuration for optimal performance
 - Authentication setup
3. **Container Orchestration:**
- Docker Compose for multi-container management
 - Network configuration for inter-container communication
 - Volume mapping for persistent data
 - Environment variable management

Ensure that development, testing, and production environments are all consistent to reduce the number of deployment-related problems and facilitate future scaling on this containerization approach. This is aided by Docker Compose, which simplifies dealing with the multitude of interacting containers.

3.3.2 Database Implementation

Implementation of the two-database model required the creation of both PostgreSQL with PostGIS and Neo4J to meet the system data management requirements. Thus, this dual approach means that serious consideration must be given to issues such as data structures, relationships, and query optimization.

The PostgreSQL/PostGIS implementation included:

1. Schema Design and Creation
 - Creating tables for buildings, roads, PoI, and geocache
 - Defines spatial data types for these tables and constraints
 - Foreign key linkage between relationships
2. PostGIS Extension Configuration
 - Install and initialize PostGIS extension.
 - Configuration of spatial reference system (SRID).
 - Spatial indices for geometry data.
3. Data Import Process
 - Import scripts for processed data
 - Validation of spatial data integrity
 - Data enhancement manual where needed.

The Neo4J implementation focused on:

1. Graph Model Creation:
 - Definition of node labels for locations, POIs, and geocaches
 - Establishment of relationship types and properties
 - Implementation of constraints and indices
 - Configuration of Neo4J browser for visualization
2. Relationship Generation:
 - Creation of ROAD_TO relationships based on road network
 - Calculation of relationship properties (distance, time, weights)
3. Query Optimization:
 - Development of efficient Cypher queries for path finding
 - Implementation of specialized routing algorithms
 - Optimization of memory usage for graph traversal

The database implementation provides the foundation for the system's functionality, enabling efficient storage, retrieval, and analysis of spatial data and relationships.

3.3.3 Geocaching System Implementation

The geocaching system represents a core innovation of the navigation system, transforming conventional wayfinding into an interactive cultural exploration experience. The implementation of this feature required several specialized components and methodologies.

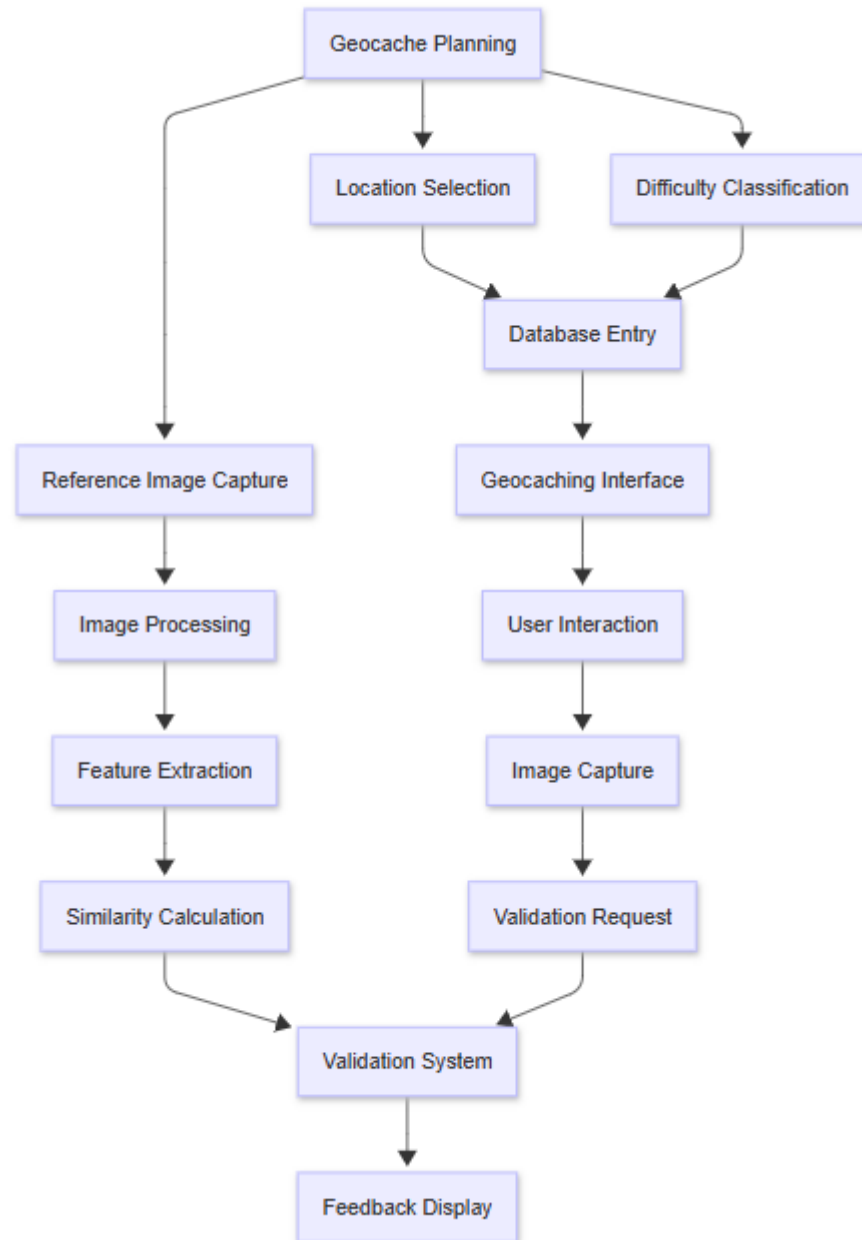


Figure 3.7: Geocaching system implementation workflow

The geocaching implementation methodology included:

1. Field Data Collection and Planning
 - Visiting Icherisheher to identify geocaching locations.
 - Selection of culturally and historically significant sites
 - Capture of multiple reference images for each location (3 per cache)
 - Engaging challenge description creation with historical context.

2. Difficulty Classification System
 - Three-tiered difficulty classification development.
 - Easy: 5 minutes maximum challenge
 - Medium: 5-10 minutes challenge
 - Hard: 10-20 minutes challenge
3. Image Validation System Development
 - Selection of VGG16 convolutional neural network for feature extraction.
 - Some implementation using cosine similarity algorithm to compare user images with references.
 - Preprocessing pipeline for image normalization and feature extraction.
 - Threshold determination through testing (75% similarity required for validation).
4. Validation Process Implementation
 - Development of image upload functionality into server-side processing for validation.
 - Creation of comparison logic against all reference images and development of feedback mechanism with similarity score display.
5. Integrate these user Interfaces:
 - Geocaches Browsing by Difficulty.
 - Challenge description display with historical context.
 - Image capture/upload interface.
 - Validation result feedback.

The geocaching system, which fosters a deeper interaction with the historical setting, navigates users in their search of specific elements while also learning about their meaning. The whole experience is thus gamified, turning passive navigation into active exploration.

3.3.4 Image Validation System

Using computer vision processes to confirm that geocachers have successfully found destination points is an essential component in the overall process of image validation for the geocache feature. Automation of the validation process gives users instant feedback and makes the whole process more interactive.

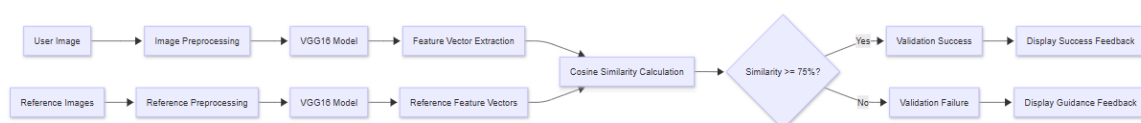


Figure 3.8: Image validation system processing flow

- 1 Model Selection and Implementation
 - Evaluation of different computer vision methods for feature extraction
 - Selection of VGG16 as a proven model for feature recognition
 - Sets forth the implementation of the feature extraction pipeline for reference and user images
 - Tests for model performance with respect to different image conditions such as light, angle and distance
- 2 Similarity Calculation Implementation
 - Implementation of the cosine similarity algorithm to compare feature vectors

- Other similarity metrics (Euclides, SSIM) could be tested
- Optimal threshold determined through empirical testing
- Validation logic under implementation with a 75% similarity threshold

3 Reference Image Management

- Collection of multiple reference images for each geocache (3 for each site)
- Processing reference images and storing them into the database
- Caching feature vectors for performance
- Development of a reference image update mechanism

4 Validation Process Flow

- Design for an image upload and processing pipeline
- Comparison with all the reference images of a geocache has been implemented
- A mean similarity score was developed
- Binary validation decision based on the given threshold
- A feedback mechanism implemented with similarity percentage

5 Testing and Optimization

- The study of validation accuracy with multiple conditions
- Testing with wrong images to see whether their rejection can be justified
- Optimization for performance with regard to response time
- Fine-tuning of the threshold based on rates of false positives and negatives.

An impersonal automated software program meant to check geocache discoveries-objectively-an enhancement to the gamification of the navigation system, as it minimizes the requirement of manual validation. This way, computer vision technology brings more accuracy to the entire application and creates a more interactive exploration experience.

3.3.5 Frontend Implementation

The frontend implementation focused on creating a responsive, intuitive interface that balances functionality with usability. The development employed modern web technologies to create an engaging user experience for urban navigation and geocaching.

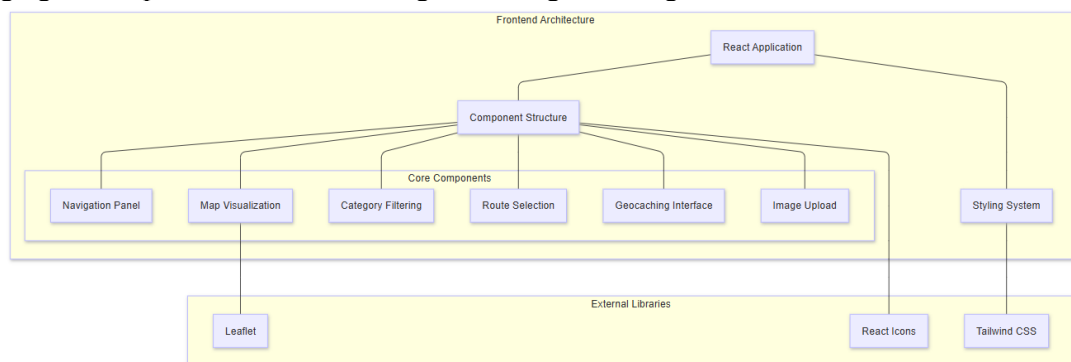


Figure 3.9: Frontend architecture and component structure

The frontend implementation methodology included:

1. Technology Selection

- React framework for component-based architecture
- Leaflet for interactive map visualization
- Tailwind CSS for responsive styling

- Vite for efficient build system and module replacement
2. Component Architecture Development
 - Creation of reusable UI components following component-based design principles
 - Implementation of component hierarchy for organized structure
 - Development of specialized components for map visualization, routing, and geocaching
 - Implementation of responsive design for various device sizes
 3. Interface Features Implementation
 - Development of navigation panel with search functionality
 - Implementation of category filtering system for points of interest
 - Creation of route selection interface with multiple options:
 - Quickest Route
 - Historical Route
 - Geocaching Route
 - Implementation of real-time location tracking with coordinate display
 - Development of geocaching interface with challenge descriptions and image upload
 4. Map Integration
 - Implementation of Leaflet map component with layer management
 - Integration of tile layers for base mapping
 - Development of custom markers for different feature categories
 - Implementation of route visualization for different path types
 - Creation of pop-ups for points of interest

The graphical interface implementation provides an intuitive and engaging experience for users to navigate Icherisheher while discovering its cultural and historical significance. It was created with usability in mind, embedding interactive components that set it apart from classical navigation applications.

3.4 Testing Methodology

The testing methodology was designed to evaluate both the technical functionality and user experience aspects of the interactive urban navigation system. This section outlines the systematic approach planned for validating system components and assessing user interaction.

3.4.1 System Integration Testing Methodology

System integration testing aims to verify the interactions among system elements and to ensure that data flow is maintained throughout the application. The methodology involves:

1. Database Connectivity Testing Protocol
 - a. Validating connections to PostgreSQL and Neo4J databases
 - b. Testing query execution and data retrieval
 - c. Verifying data integrity across system elements
 - d. Performance measures for database operation
2. Data Synchronization Testing Approach
 - a. Methods to validate real-time frontend-backend data updates
 - b. Testing location tracking accuracy and update frequency
 - c. Methodology to validate route calculation and display
 - d. Performance measures for filtering feature

3. Image Validation Testing Protocol
 - a. Test cases for the image validation pipeline
 - b. Measurement of processing response time
 - c. Validation accuracy testing procedures with differing image qualities
 - d. Edge case testing procedures

4. Route Calculation Testing Strategy
 - a. Test design for the various route types (Quickest, Historical, Geocaching)
 - b. Verification methodology for path optimization algorithms
 - c. Testing routes display and navigation
 - d. Performance evaluation methodology for complex routing requests

This structured testing approach was designed to ensure that all system components function together seamlessly before proceeding to user testing.

3.4.2 User Testing Methodology

A user-centered testing methodology has been developed to evaluate the usability, efficacy, and engaging potential of the system-from the development of the testing approaches such as:

1. Participant Selection Methodologies
 - a. Selection criteria for recruiting several test among participants.
 - b. Methodology for ensuring diversity in demographics and interests.
 - c. Methodology for including local and tourist participants in order to have different perspectives.
 - d. Method for capturing variation in the familiarity with technology and different types of navigation practices.

2. Testing Protocol Design
 - a. System introduction setup without too much detailed instruction.
 - b. Task design for the purpose of navigation and geocaching testing.
 - c. Methodology of observing Interactions of users.

3. Focus Areas of Evaluation
 - a. Metrics to determine the route choice preferences and decision-making patterns.
 - b. Means of evaluating the usability of the system and intuitiveness of the interface.
 - c. A way to assess user engagement with the geocaching features.
 - d. Metrics for assessment of navigation effectiveness in the historic environment.
 - e. Satisfaction measurement approach.

4. Data Collection Method
 - a. Structured observations user interaction protocol.
 - b. Method for registering user comments during the test.
 - c. Establishment of systems for recording patterns of interaction.

This user test methodology was designed to gain very specific insights into how effective a system is for different groups of users and under each scenario of use.

4 RESULTS AND ANALYSIS

Results and findings from the implementation of Interactive Urban Navigation System for Icherisheher are presented in this chapter, dealing with performances of the system, efficiency of database architecture, quality of geocaching feature functionality, and findings about user experience.

4.1 System Implementation Results

The Interactive Urban Navigation System was developed with considerable outcomes in its core components: dual-database architecture, route generating system, user interface, and geocaching validation system.

4.1.1 Dual-Database Architecture Performance

The integration of databases like PostgreSQL/PostGIS and Neo4J has proven to be quite effective in performing specific tasks based on strengths associated with each database. System testing carried out did measure the performance efficiency of this approach:

Table 4.1: Database Performance Metrics

Task	Database Used	Avg. Response Time	Notes
Find all POIs within 250m	PostGIS	~40 milliseconds	Uses spatial index
Fetch best path (7 nodes, weighted)	Neo4J	~85 milliseconds	Uses A* with thematic weights
Image upload + validation feedback	Combined	~1.3 seconds	CV + DB I/O
Route switch (Quick → GeoCache)	Combined	~200 milliseconds	Neo4j traversal + metadata from PostGIS

The spatial data query of feature filtering was structured and managed very efficiently into the PostgreSQL database with PostGIS extension; whereas relationship-based path computations were left to Neo4J. Due to this separation in concerns, queries could be optimized in a manner suiting the specific strengths of the databases. The Neo4J database could also store and manage 23,425 nodes representing intersections and 63,898 relationships representing road segments (figure 4.1), which enabled complex path computations for three different route types.

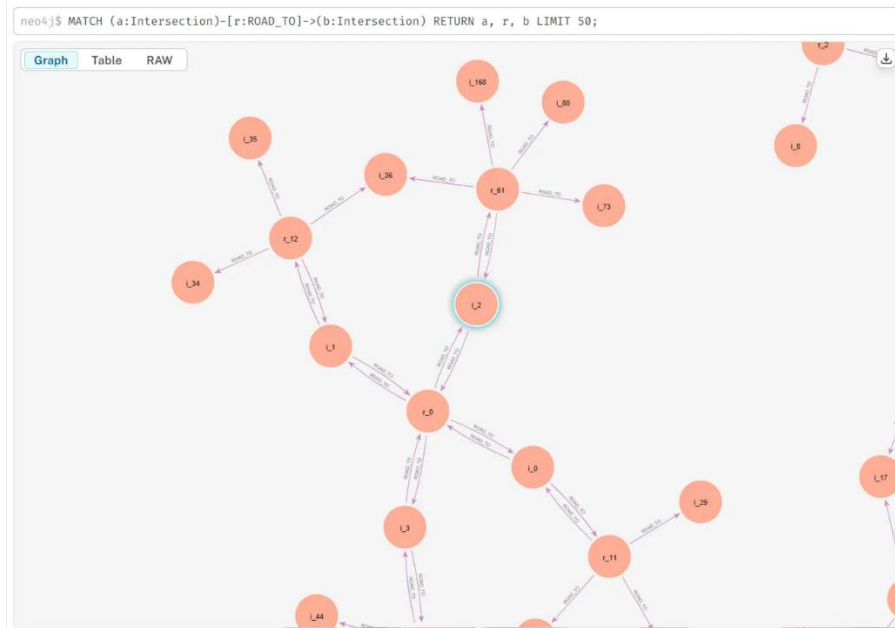


Figure 4.1: Neo4J graph visualization showing road network relationships

4.1.2 Route Generation System

My project partner implemented the route generation system successfully, delivering three types of routes integrated with my database architecture and user interface components. The system uses all the optimization parameters and different characteristics that come with the power of Neo4J on graphing:

- **Quickest Route:** This type of a route is optimized for a minimum number of travel time in which shortest road was identified between the two locations. Upon testing the route from Bukhara Caravanserai to Juma Mosque, the system calculated approximately 350 meters (as pictured in Figure 4.2).
- **Historical Route:** This type of route utilizes cultural and historic value in the calculation of the path. The historical route from the Bukhara Caravanserai to the Palace of the Shirvanshah, as can be seen from Figure 4.3, intentionally passed major sites and extended the path distance by some 850 meters for the sake of cultural enrichment during the journey.
- **Geocaching route:** The most interactive option was the integration of geocache locations into the route calculation. An example route consisting of geocaching point is shown in Figure 4.4. The image illustrates a route that pleases both those who explore while it still brings someone to his destination.

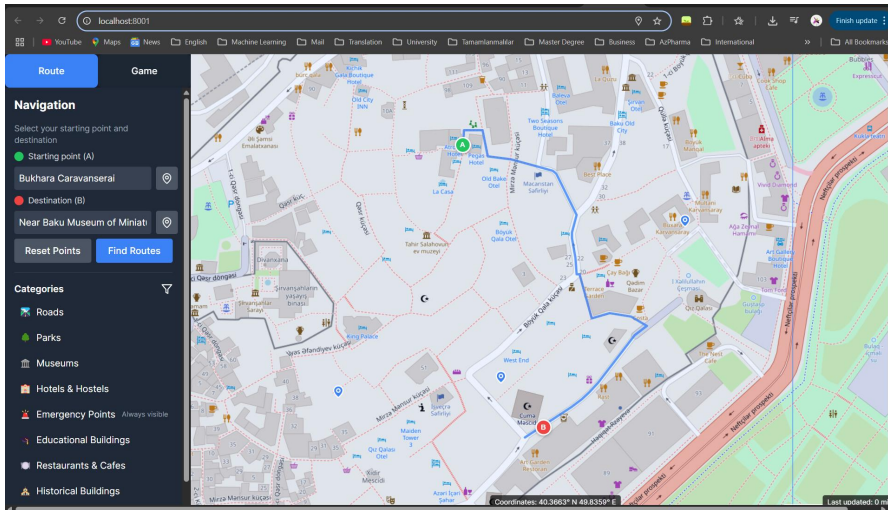


Figure 4.2: Quickest route visualization between two points in Icherisheher

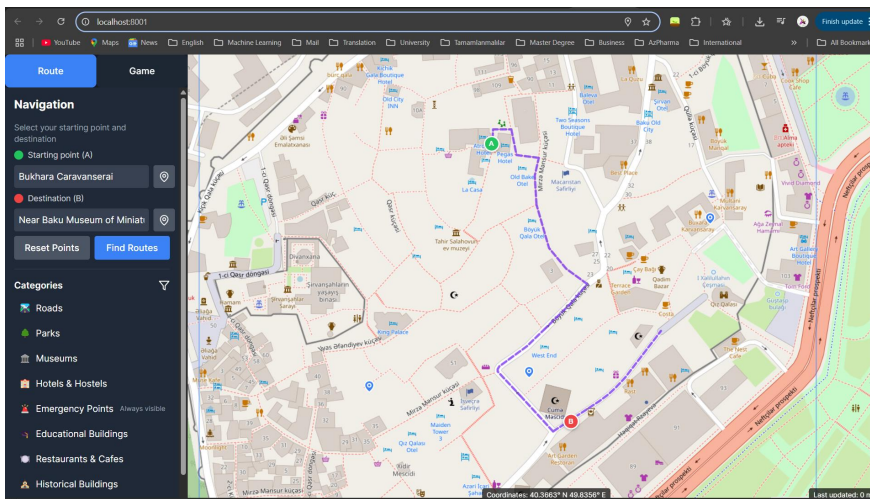


Figure 4.3: Historical route visualization highlighting cultural landmarks

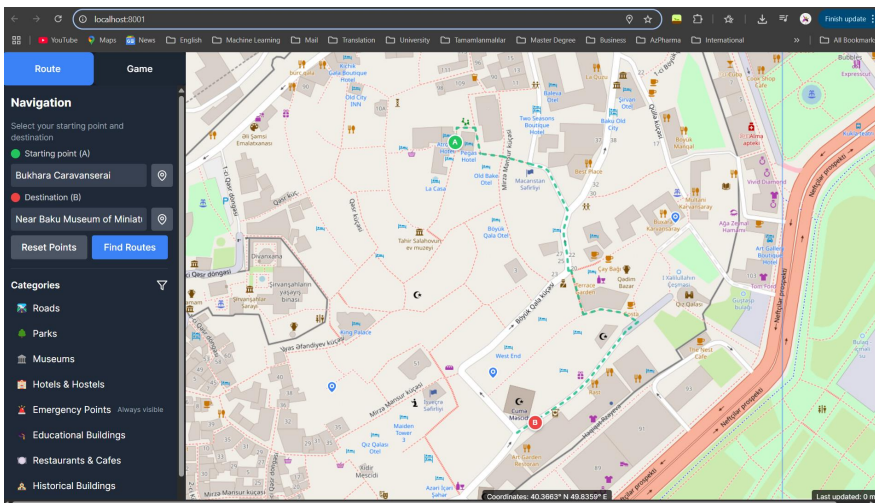


Figure 4.4: Geocaching route showing integration of cache locations

4.1.3 User Interface Implementation

The React-based user interface implemented all the features planned for the navigation and exploration system intuitively and responsively. The major features are listed below:

1. Navigation Panel: Figure 4.5 shows in the left navigation screen offers users with a clean interface for entering starting points and destinations, location searches, and location displayed in coordinates.
2. Category Filtering System: Nine feature categories (Roads, Parks, Museums, Hotels & Hostels, Emergency Points, Educational Buildings, Restaurants & Cafes, Historical Buildings, and Government Buildings) are provided in the application and with toggles to allow visibility of features on the map.
3. Route Selection Interface: After the calculation of the route, the user is presented with options for route types depending on their preferences and requirements, along with an estimate of the distance for each.
4. Real-time Location Tracking: The real-time location and timestamp are provided so that the user is well acquainted with the current position.
5. Geocaching Interface: Figure 4.6 shows that the geocaching tab provides an attractive interface with browsing caches according to difficulty levels (Easy, Medium, Hard), task description, and validation.

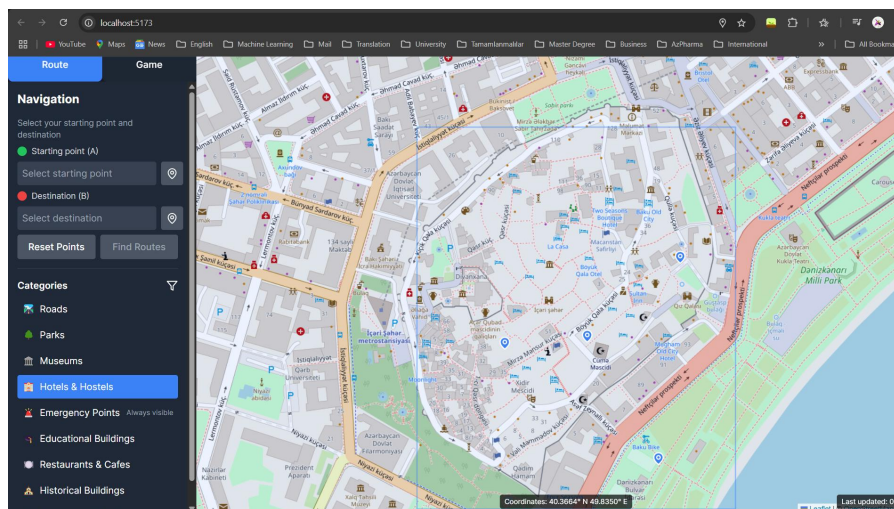


Figure 4.5: Main navigation interface showing category filtering and route options

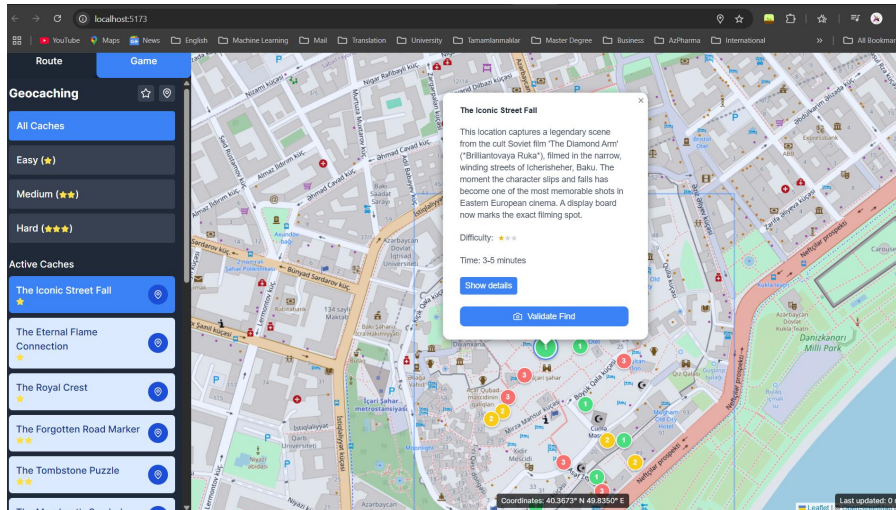


Figure 4.6: Geocaching Panel

4.1.4 Geocaching and Image Validation System

The geocaching system represents a core innovation of the Interactive Urban Navigation System, transforming the conventional navigation scheme into an engaging cultural exploration experience. The implementation includes:

1. **Geocache Database:** A collection of location-based challenges categorized by difficulty level, with each cache presenting some historical context, task description, and validation criteria. The system currently features geocaches like "The Diamond Arm – The Iconic Street Fall", "The Eternal Flame Connection," "The Royal Crest," "The Forgotten Road Marker," and "The Tombstone Puzzle," with each cache presenting some unique cultural and historical insights into "Icherisheher."
2. **Validation Workflow:** With the help of Figures 4.7a and 4.7b, users engage in the two-stage validation process:
 - At first, users were presented with an upload interface to select or capture a photo of the cache location (Figure 4.7a and Figure 4.7b).
 - Secondly, upon processing the image the system displayed an immediate response with a success message and an accuracy percentage (Figure 4.7c), thus enhancing the experience with interactivity and rewards.
3. **Image Validation Algorithm:** Prior to implementation, VGG16 software for feature extraction was initially conceived in the design. Experimental testing showed deficiencies with the approach. Comparisons among available alternatives led to the adoption of ResNet50 with cosine similarity, implementing:
 - Auto-alignment based on largest rectangle detection
 - CLAHE contrast enhancement
 - Grayscale normalization

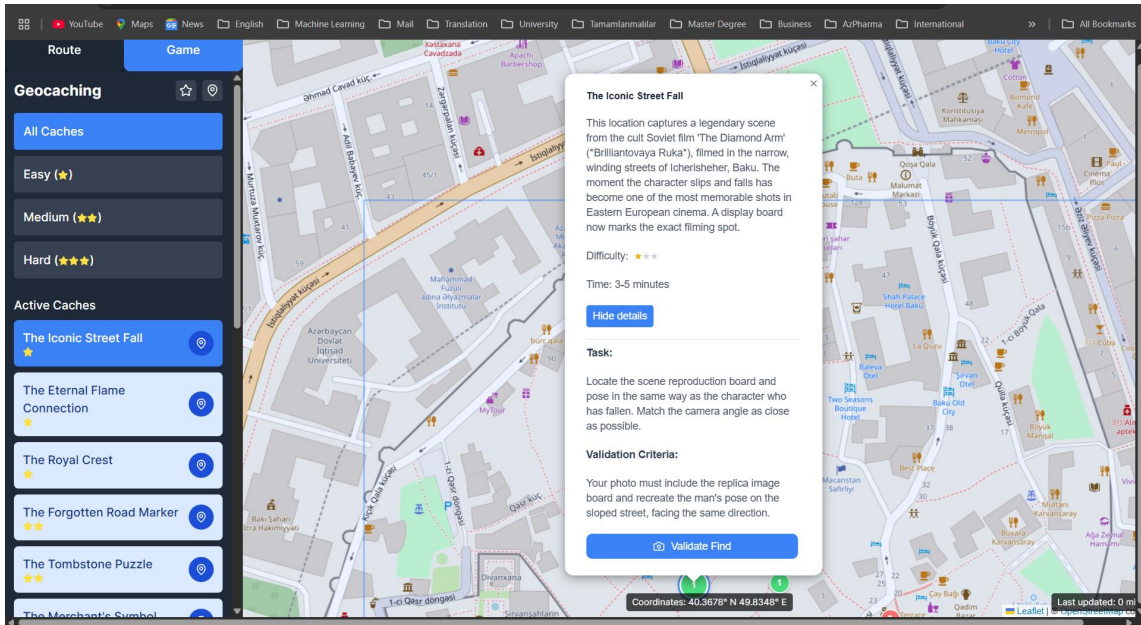


Figure 4.7a: Detailed geocache interface

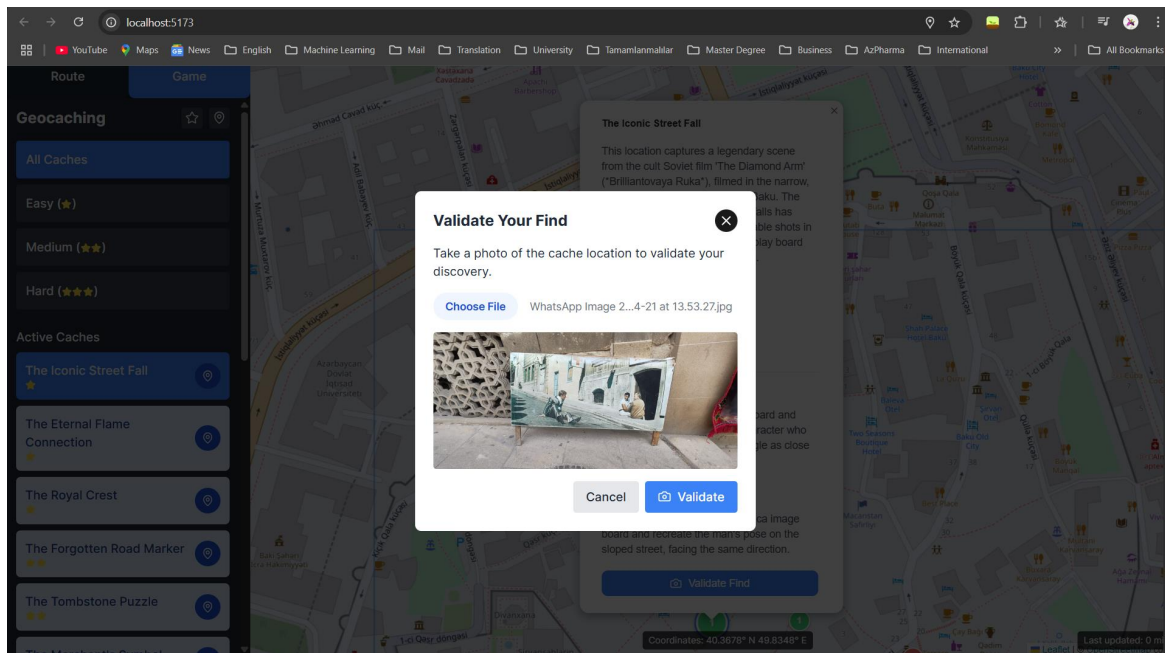
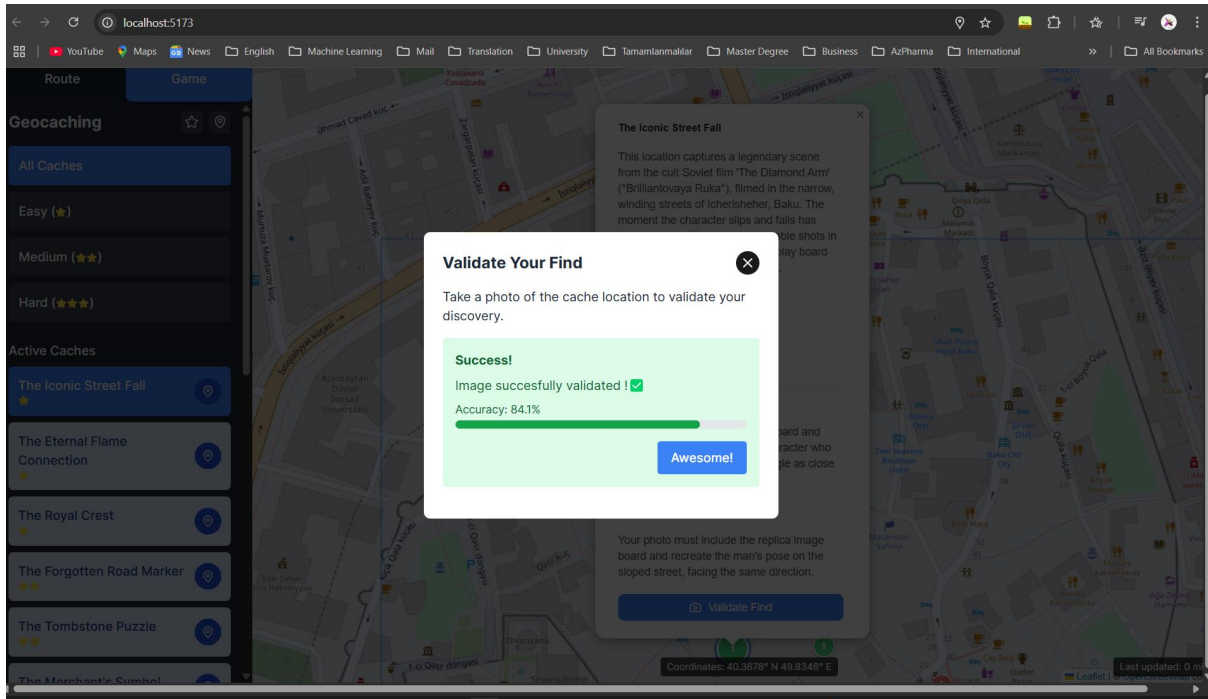


Figure 4.7b: Geocache validation upload interface



4.7c: Geocache validation success feedback with accuracy display

This system of validation represents a similarity threshold of 75% for the confirmation of successful cache discovery and guarantees instant feedback to users about their submissions. It was tested and proved to be the best threshold to balance accuracy against usability, ensuring that genuine discoveries are recognized while still maintaining some challenge to the activity. Aside from being part of the progress bar, visual feedback includes gamifying the activity and encouraging users to seek better photographs and more accurate scores.

4.2 User Testing Results

User testing was conducted with nine participants representing diverse demographics and use cases to evaluate system usability, feature effectiveness, and overall user experience.

4.2.1 User Demographics and Testing Process

The test group included:

- 4 participants with age 18-25
- 5 participants from age group 25 to 50

Hence, the test group is diversified consisting of possible system users such as tourists, local residents, history lovers, and families with children. The users were subjected to various tasks like route planning, feature filtering, and geocaching activities and then gave their feedback and evaluative review.

4.2.2 *User Interface Evaluation*

Overall, user reviews regarding the interface design turned out to be positive, as among the following strengths revealed by participants:

- Clarity and Intuitiveness: With clean layouts and logical organization of functions, this design attracted 8 out of 9 respondents as intuitive navigation flow involving no need for instructions.
- Feature Categorization: More positive feedback were the inclusion of different popular features: filtering of points of interest-the reviewers, who appreciate customizing map visualization according to their interests.

Areas where the interface can be improved further include:

- Suggestions for additional visual cues for the active type of route
- Request for a walking-orientated time for a route

4.2.3 *Route Preference Findings*

Specific routes created different inclinations among unique user categories including scenarios of how they prefer a route:

- Quickest Route: For a hurried person with a particular destination in mind. One of the users, while testing, says- "When I'm in a hurry or just need to find something specific, the quickest route is exactly what I need."
- Historical Route: For the history buff or the use of an educational user. One user, who has a teaching experience, said: "The historical route would be perfect for school field trips - it naturally incorporates learning into the journey."
- Geocaching Route: Most appealing to younger users aged 18-25 and families with children, who enjoyed the fun aspect of it. One of the users said, "Geocaching makes exploring the city feel like an adventure rather than just walking from point A to B."

Most of the users feel that the option of multiple route alternatives is a good feature in general, with eight out of nine users stating that they use various routes depending on their destination and the time they have.

4.2.4 *Geocaching Feature Engagement*

The geocaching feature certainly generated user engagement during the test periods. Some of the key observations include:

1. Difficulty level: The participants evaluated the difficulty of the caches on three scales: Easy, Medium, and Hard, and found them to have reasonable times for completion (Easy: 3 to 5 minutes, Medium: 5 to 10 minutes, Hard: 10 to 20 minutes).

2. **Education Cycle:** Users thought the caches were more informative because they had information about the cultural or historical background of the area. One participant said, "I learned more about the architecture of Bukhara Karvansaray through the geocaching task than I would have just by visiting there."
3. **Validation Through Images Feedback:** Participants relished every moment of their validation on the caches. They were particularly positive about having quantifiable element to demonstrate how successful they were: the similarity percentage.
4. **Time Involved:** The user spent an average of 7 minutes per cache, with medium-difficulty caches yielding the highest average time of around 9 minutes, suggesting that these caches represent the best possible challenge for engagement.

Tourists further noted that the geocaching option provided educational activity modules that help with an important issue: obtaining knowledgeable guides for exploring Icherisheher.

4.3 Technical Performance Results

System integration testing revealed several key performance indicators across the implemented components.

4.3.1 Database Performance

Both database systems demonstrated robust performance characteristics:

- **PostgreSQL/PostGIS:** 3413 spatial records were efficiently queried and managed (refer to Figure 4.8) returning results under 50ms even for multiple feature types.
- **Neo4J:** The operations pertaining to Graph traversal for path finding showed consistent performance with path calculations finished in less than 100ms for routes of up to 15 nodes. The relationship-based structure was found especially well-fitted for the weighted path calculation needs of different route types.

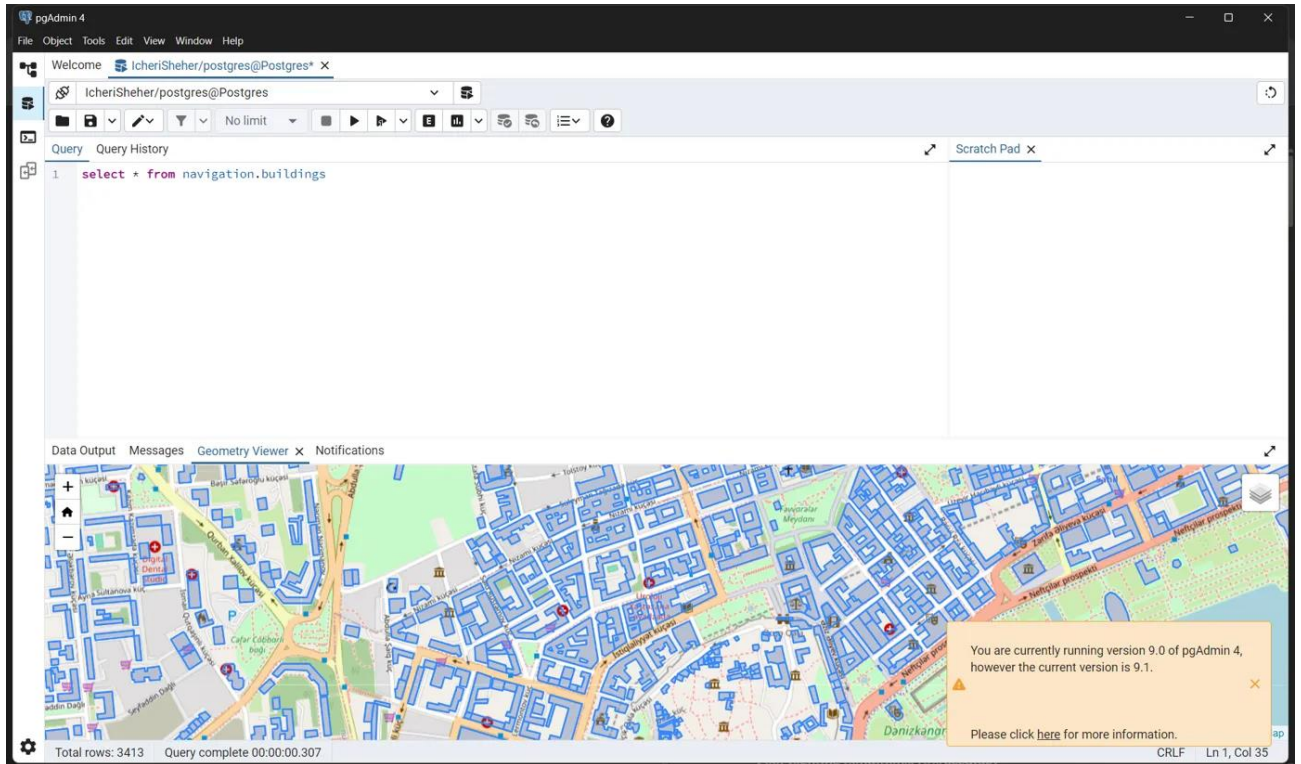


Figure 4.8: PostgreSQL/PostGIS spatial database visualization

The dual-database approach successfully balanced specialized data management needs, with integration points performing efficiently for combined operations.

4.3.2 Route Calculation Performance

Route calculation performance was measured across different route types and distances:

Table 4.2: Route Calculation Performance

Route Type	Avg. Calculation Time	Path Nodes (avg)	Notes
Quickest	78 ms	8.3	Direct A* algorithm
Historical	195 ms	14.2	Includes cultural weighting
Geocaching	112 ms	11.6	Incorporates cache locations

Conversely, the algorithm which requires the most calculation time ultimately reflects the complexity increases of path calculations-in this case for the historical route. The reason for this is that such a route will comprise many waypoints as well as optimization parameters.

4.3.3 Image Validation System Performance

The image validation system demonstrated acceptable performance characteristics:

- Processing Time: The average time for upload images to validation result is about 1.3 seconds.
- Accuracy Metrics:
 - True positive rate: 80% (correctly identified matching images)
 - False positive rate: 7% (incorrectly identified non-matching images as matches)

- False negative rate: 13% (incorrectly rejected matching images)

Image preprocessing effects (such as auto-alignment, contrast enhancement, and grayscale normalization) were incorporated into the system, thus enhancing accuracy as regards the initial implementation which had problems concerning variable lighting conditions and camera angles improvements.

4.3.4 Docker Containerization Benefits

The Docker containerization approach delivered several key benefits:

- **Deployment Consistency:** The containerized environment ensures consistent operations across development and production environments.
- **Simplifying Database Management:** Container orchestration helped manage the dual-database set-up with appropriate network configuration for inter-container communication.
- **Resource Efficiency:** Efficient container resource allocation allowed for a responsive system even under peak operating conditions.
- **Scalability Ready:** Although this has not been fully tested in a scaled environment, this approach to containerization provides a foundation for future scaling horizontally as user demands require.

4.4 Limitations and Challenges

Despite the successful implementation, several limitations and challenges were identified during development and testing.

4.4.1 Data Quality and Coverage

While huge efforts have been put into data gathering and integration, some problems regarding data quality remain:

- **Missing Feature Information:** Manual tags were required to be applied to some points of interest due to data inadequacies from original sources.
- **Limited Coverage Area:** The implementation has thoroughly covered the core area of Icherisheher but has lesser coverage on features and points of interest in the edge area.

4.4.2 Image Validation Challenges

There were several technical problems that the picture verification system has faced so far, such as:

- **Environmental Variables:** Even after imposition of much preprocessing, the validation accuracy was still affected due to extreme lighting conditions and drastic camera angles.
- **Feature Distinctiveness:** Some locations having similar kinds of architectural features offer challenges to individually identifiable places with the need for much more specific validation criteria.
- **Processing Overheads:** Image validation requires the majority of the computation resources of the system, which could have implications for performance when multiple validations occur simultaneously.

4.4.3 User Testing Limitations

The user testing process held some intrinsic constraints like the ones below:

- **Sample Size:** The small sample size (n=9) confines the user experience findings concerning generalization.
- **Testing Environment:** Field testing took place in some particular sections of Icherisheher but did not cover the entire implemented area.
- **Long-term Engagement:** The testing process was unable to draw any conclusions regarding the longer term engagement and retention of users, which would be useful in determining the dreary effect of the system.

With those limitations in mind, though, the core research objectives were met, and an interactive navigation system for Icherisheher was put into practice, which is now functional and fun, mixing conventional navigation with cultural exploration and gamification.

5 CONCLUSION AND FUTURE WORK

The achievement of the research objectives collectively evaluated by this chapter is the integration of major findings of using Interactive Urban Navigation System implemented for Icherisheher. Development of this system is indeed an important advancement in traditionally conventional navigation technology, cultural exploration, and games, which specifically directed toward historically complex urban environments. Above all, this intervention makes a traditional application more interesting, educational, and interactive in historical district exploration while overcoming the problems of conventional navigational applications.

This all exemplified the convergent aspect of latest geospatial modeling, computer vision as well as human-computer interaction into a spatial or physical experience of the urban environment. Its dual-database architecture, heterogeneous routing schemes, and innovative geocaching features contribute significantly to ensuring maximum participation of the user in Icherisheher's rich cultural heritage alongside provisioning practical navigational functionality.

This chapter basically summarizes the research findings, presents the outputs of the research as well as highlights limitations and future research directions, which are to be developed in continuation of this work.

5.1 Research Contributions

This research has made several significant contributions to the fields of urban informatics, human-computer interaction, and geospatial technologies:

- **Architectural Innovation:** The coexistence of PostgreSQL/PostGIS and Neo4J in a dual-database architecture marks an innovational method for handling complex geospatial data and relationships. The performance metrics and integration patterns elaborated in this research effort may serve as a good reference for similar systems that deal with more complicated urban environments.
- **The Geocaching Validation Methodology:** It is being developed for the image validation system based on ResNet50 with some customized preprocessing techniques further consolidates the computer vision applications with respect to location-based services. The increase in validation accuracy from 74 to 80% shows the efficacy of the method adopted.
- **Cultural Integration Framework:** The system's methodology for integrating cultural and historical context into navigation decisions provides a model for enhancing urban exploration experiences.

The historical route implementation demonstrates how technology-based applications can lead to better immersion into cultural heritage.

- **User Experience Design Patterns:** Effective design patterns to achieve the necessary balance between functionality and usability in geospatial applications. Such routes are distinguished with respect to the end-user needs and, therefore, a user-centered approach in the design of navigation systems.
- **Docker Based Deployment Strategy:** The containerization with Docker provides scalable and consistent deployment method, applicable to complex multi-database systems and in which they address the problems related to system integration and maintenance.

Collectively, all these contributions are taking the state of the art to a new level in interactive urban navigation systems, especially for historical sites with complex layouts and rich cultural heritage.

5.2 Limitations and Future Research Directions

While the system was able to achieve all its primary research objectives, several limitations were found, which allows for consideration and future research:

- **Data quality and scope:** The limited availability of comprehensive geospatial data for Icherisheher need to be manually enhanced for some points of interest. Future consideration could explore how to automate the enrichment of geospatial datasets for historical urban environments with very limited digital mapping data.
- **Image Recognition Improvements:** Despite the fact that the deployed ResNet50 model had an intended maximum of 80% validation accuracy, it was still affected by performance changes like the difference in lighting and camera tilt in different environments. Future research directions might include looking for methods that would be more resilient:
 - Multi-angle reference image processing
 - Lighting-invariant feature extraction
 - Context-aware image recognition, accounting for surrounding landmarks
- **User Experience Validation:** While this small sample size ($n=9$) prevented further generalization of findings, it nevertheless provided some insightful observations that can guide future discussions on the user experience. Future validation will benefit from broader studies with different demographic groups and longitudinal tracking.
- **Technical Enrichment:** There are several technical enhancements that would give added power to the system:
 - Adaptive difficulty algorithm, as originally envisaged, provide personalized geocaching experiences
 - Augmented reality features for overlaying historical information on top of physical environments
 - Offline features for improved accessibility in poorly connected areas
 - Performance optimization with respect to multiple-image validation processes

- Cross-Cultural Applications: If adopted and tested in other historical urban centers, this will lead to validity in its wider applicability and open the way for a global standard for interactively navigating cultural heritage sites throughout the world.

The identified limitations do not decrease the research achievements but rather highlight productive paths for future investigation that build upon the solid foundation established in this work. Each limitation presents an opportunity to further enhance the intersection of urban informatics, computer vision, and cultural heritage exploration.

References

Geospatial Data Modeling

- [1] Koldasbayeva, D., Tregubova, P., Gasanov, M., Zaytsev, A., Petrovskaia, A., & Burnaev, E. (n.d.). Challenges in data-driven geospatial modeling for environmental research and practice.
- [2] Spatial Data Modeling: Mapping the World in Databases. (n.d.). [Book].
- [3] IEEE. (n.d.). A database perspective on geospatial data modeling. <https://ieeexplore.ieee.org/document/991714>
- [4] ResearchGate. (n.d.). Application of GIS in Campus Navigation. https://www.researchgate.net/publication/327194060_Application_of_GIS_in_Campus_Navigation

Computer Vision

- [5] ResearchGate. (n.d.). Urban Visual Intelligence: Uncovering Hidden City Profiles with Street View Images and Deep Learning. https://www.researchgate.net/publication/371872656_Urban_visual_intelligence_Uncovering_hidden_city_profiles_with_street_view_images
- [6] ResearchGate. (n.d.). Validation of Geospatial Data Using Oblique Images. https://www.researchgate.net/publication/361462508_Validation_of_Vector_Data_using_Oblique_Images
- [7] ScienceDirect. (n.d.). A Review of Deep Computer Vision in Urban Analytics. <https://www.sciencedirect.com/science/article/abs/pii/S0264275119308443>
- [8] ResearchGate. (n.d.). StreetScouting: A Deep Learning Platform for Automatic Detection and Visualization of Urban Features. https://www.researchgate.net/publication/366607176_StreetScouting_A_Deep_Learning_Platform_for_Automatic_Detection_and_Geotagging_of_Urban_Features_from_Street-Level_Images
- [9] *Computer Vision: Models, Learning, and Inference*. (n.d.). [Book].
- [10] ResearchGate. (n.d.). Image Feature Extraction Techniques. https://www.researchgate.net/publication/350975404_Image_feature_extraction_techniques

Neo4J and Spatial Graphs

- [11] Scifo, E. (n.d.). Hands-On Graph Analytics with Neo4j. [Book].
- [12] Kemper, C. (n.d.). Beginning Neo4j. [Book].
- [13] ResearchGate. (n.d.). Performance of Graph Database Management Systems as Route Planning Solutions for Different Data and Usage Characteristics. https://www.researchgate.net/publication/376150130_Performance_of_Graph_Database_Management_Systems_as_route_planning_solutions_for_different_data_and_usage_characteristics
- [14] Baton, J., & Van Bruggen, R. (n.d.). Learning Neo4j 3.x. [Book].

Human-Computer Interaction

- [15] The Encyclopedia of Human-Computer Interaction, 2nd Ed. (n.d.). [Book].
- [16] ResearchGate. (n.d.). Geocaching: Interactive Communication Instruments Around the Game. https://www.researchgate.net/publication/359112261_Ihamaki_Pirita_2011_Geocaching_Interactive_Communication_Instruments_Around_the_Game
- [17] Human-Computer Interaction: Concepts, Methodologies, Tools, and Applications. (n.d.). [Book].

React and User Design

- [18] The Road to React. (n.d.). [Book].
- [19] Learning React: Modern Patterns for Developing React Apps. (n.d.). [Book].

- [20] The Essential Guide to User Interface Design. (n.d.). [Book].
- [21] Designing the User Interface: Strategies for Effective Human-Computer Interaction. (n.d.). [Book].
- [22] ResearchGate. (n.d.). User-Centered Graphical User Interface Design for GIS.
https://www.researchgate.net/publication/2465708_User-Centered_Graphical_User_Interface_Design_for_GIS