PROJECT SPECIFICATIONS REPORT

GUIDE

(Guided User Itinerary & Destination Explorer)

QUAD-CORE Team

Ebrar Sude Doğan
Erdem Baran
Kayrahan Toprak Tosun
Tuna Kodal

TOBB University of Economics and Technology
Department of Computer Engineering
BIL 495 / YAP 495 Graduation Project

Version: 1.0
Date: January 2026

Contents

1 INTRODUCTION

1.1 Description

GUIDE (Guided User Itinerary & Destination Explorer) is an AI-powered, web-based travel planning system designed to revolutionize how travelers explore cultural, historical, and natural destinations in Turkiye. The system addresses the critical challenge of fragmented travel planning by providing an integrated platform that combines intelligent route generation, personalized POI recommendations, interactive map visualization, and multilingual content delivery in a single, cohesive user experience.

GUIDE eliminates the need for travelers to manage multiple disconnected applications (navigation apps, research websites, and booking platforms) by offering a unified solution that transforms user preferences into actionable, optimized multi-day itineraries. The system's intelligent algorithms analyze user inputs (including trip duration, daily distance constraints, and category preferences) to automatically generate efficient routes connecting relevant points of interest while minimizing travel time and distance.

1.1.1 Key Capabilities

Personalized Route Generation: GUIDE's AI-driven recommendation engine filters and ranks POIs based on user-specified categories (historical sites, museums, natural landmarks, religious sites, etc.), geographic regions, and exclusions. The routing algorithm then constructs multi-day itineraries that balance daily travel distances, visit durations, and user interests. Routes are optimized using the Open Source Routing Machine (OSRM)[1] integrated with OpenStreetMap (OSM)[2] data, ensuring accurate navigation along actual road networks.

Interactive Map Visualization: The system presents generated routes through an intuitive, interactive map interface powered by Leaflet.js[3]. Users can visualize their complete itinerary with day segments, clickable POI markers, and route polylines that clearly illustrate travel paths. The map supports pan, zoom, and marker interactions, enabling users to explore destinations in detail before committing to the itinerary.

Multilingual Content Delivery: GUIDE provides comprehensive audio destination information in three languages (Turkish, English, German) to serve both domestic and international travelers. Each POI includes textual descriptions sourced from authoritative references (UNESCO[4], Wikipedia [5]) and pre-generated text-to-speech (TTS) audio guidance using ElevenLabs API [6]. This multilingual approach enhances accessibility for users with reading difficulties and accommodates travelers with varying language proficiencies.

Dynamic Route Modification: Recognizing that initial AI-selected itineraries may not perfectly align with user preferences, GUIDE enables users to remove unwanted stops from their routes. The system automatically regenerates optimized itineraries based on the modified POI set, maintaining route efficiency while respecting user choices.

Performance-Optimized Architecture: GUIDE is engineered to deliver responsive user experiences with route generation completing within 5 seconds for typical inputs (1-3 day trips). The system employs a four-layer architecture (Presentation, Application, Integration, Data) that supports scalability, maintainability, and efficient resource utilization.

1.1.2 Target Users

GUIDE serves four primary user personas:

- **International Tourists:** Travelers visiting Turkiye who require multilingual support, comprehensive destination information, and efficient routing to maximize their limited time exploring cultural heritage sites.
- **Domestic Travelers:** Turkish residents seeking to explore destinations within their own country, preferring organized itineraries that minimize manual planning effort.
- **Cultural Enthusiasts:** Users with specific interests in particular types of destinations (e.g., museums, UNESCO sites, religious landmarks) who benefit from category-based filtering and educational audio content.

1.2 Constraints

GUIDE's development and operation are subject to several categories of constraints that shape design decisions and implementation strategies. The presentation of the project constraints and their impacts is also provided in the *Project Constraints and Impacts Document*, which is available at this link.

1.2.1 Technical Constraints

Performance Limitations:

- Route generation must complete within 5 seconds on average for trips of 1-3 days to maintain user engagement. This constraint necessitates algorithmic efficiency (polynomial complexity) and local OSRM deployment to minimize network latency.
- Due to the anticipated increase in computational complexity associated with OSRM multi-waypoint routing for longer trips, as well as the growing input size and inference complexity of the AI-based recommendation model as trip duration increases, the system is designed to support trip durations of up to three days.

Geographic Scope:

- Routing and POI data are limited to Turkiye due to project scope and resource constraints. Expanding to additional countries would require OSM data preprocessing for each region and POI dataset creation.

Real-Time Data Limitations:

- The system does not incorporate real-time traffic data, road condition information, or dynamic route adjustments. Route calculations use static OpenStreetMap data without consideration for current traffic congestion, temporary road closures due to construction or maintenance, seasonal road restrictions, or accident-related detours.
- Actual travel times may deviate from system estimates during peak hours or in areas with active construction. Users should validate routes with real-time navigation tools on travel day.

- Integrating real-time traffic APIs would introduce high cost and external dependencies exceeding project scope.

POI Operational Status:

- The system does not track dynamic POI operational information, including opening hours, temporary closures for renovation/maintenance, seasonal schedules, special event impacts, or admission requirement changes.
- Recommended POIs may be temporarily inaccessible when users visit. The system provides historical/cultural information, but cannot guarantee current operational status.
- Maintaining real-time operational data for hundreds of POIs requires continuous manual verification, or commercial API subscriptions are infeasible for an academic project.
- Users should verify POI accessibility through official websites or direct contact before visiting.

1.2.2 Hardware and Infrastructure Constraints

Development Environment:

- OSRM deployment requires a minimum of 2-4 GB RAM for processing Turkiye's OSM dataset. Development team members' personal computers meet this requirement, enabling local testing without additional infrastructure costs.
- OpenStreetMap tile rendering depends on external CDN (OpenStreetMap tile servers)[7], introducing network dependency for map visualization. Offline tile caching is not implemented due to storage and licensing complexities.

Deployment Constraints:

- The system requires sufficient memory capacity to store and serve static POI-related content, including textual descriptions, audio guidance files, and metadata for visual assets.
- For an estimated scale of 3,000–4,000 Points of Interest, the total storage footprint is expected to range between 1.5–2.5 GB, primarily driven by pre-generated multilingual audio files.
- POI data and associated media are served statically.

1.2.3 Functional Scope Constraints

Language Support:

- Multilingual content limited to Turkish, English, and German due to TTS API cost constraints and manual content creation effort. Additional languages would require proportional increases in TTS generation and translation resources. The graphical user interface remains English-only.

POI Content:

- Destination information is limited to publicly available sources (UNESCO[4], Wikipedia[5], Flickr/Pixabay[8], [9] for images) to avoid copyright and licensing issues. Professional content partnerships are out of the scope of an academic project.

Data Currency and Update Frequency:

- POI information (descriptions, images, audio) is curated once during the development phase (2026) and not updated automatically thereafter. The system does not provide continuous content updates reflecting site changes (museum renovations, new exhibits), automatic image refreshes, or real-time synchronization with authoritative sources. development period.
- Automated content update pipelines (web scraping, change detection, validation workflows) exceed project scope.
- Future enhancement: Periodic manual reviews (quarterly/biannually) could refresh high-traffic POI content. Community contribution features could enable user-submitted updates in future versions.

POI Dataset Completeness:

- Database focuses on publicly accessible, well-documented cultural and historical sites. Several categories are excluded or underrepresented: private museums/galleries requiring individual permissions, small-scale local attractions without a comprehensive online documentation, recently opened destinations not yet catalogued, and remote natural landmarks lacking detailed metadata.

User Interaction:

- No user authentication or persistent account management; all interactions are session-based. This simplifies architecture but prevents features like saved itineraries or personalization based on user history.
- Users shall be able to manually modify or override AI-generated route suggestions within the session, ensuring user autonomy and compliance with GDPR[10] regarding automated decision-making.

1.2.4 Academic and Institutional Constraints

Timeline:

- Project must be completed within academic calendar constraints: September 2025 - April 2026. Final demonstration and delivery required by April 2026.
- Sprint-based development constrained by university examination periods and academic breaks.

Documentation Standards:

- All project documentation must comply with BIL 495/YAP 495 course guidelines.

Open-Source Preference:

- Academic context favors open-source technologies over proprietary solutions. ElevenLabs TTS[6] service is used only where open-source alternatives are inadequate.

1.2.5 Process and Development Constraints

Team Capacity:

- The development team consists of 4 Computer Engineering students with 15 hours/week availability during the academic semester. Total team capacity: 60 person-hours/week.
- Skills constrained to team expertise: Python[11], JavaScript[12], web development, software engineering principles. Learning curves anticipated for specialized areas (geospatial processing, OSRM deployment).

Tools and Infrastructure:

- Version control limited to GitHub[13] due to team familiarity and zero cost.
- Development environments constrained to team members' personal computers; no access to institutional computing resources beyond standard university IT services.

Software Quality Assurance:

- Automated tests shall be implemented for all critical system modules to ensure reliability and maintain acceptable test coverage.
- All source code shall include clear and consistent inline documentation to support maintainability and team collaboration.

1.3 Professional and Ethical Issues

1.3.1 Professional Responsibility

Software Quality Standards: GUIDE follows a standards-driven quality framework aligned with system and software lifecycle processes, documentation discipline, and verification practices. Project activities and deliverables are organized in accordance with ISO/IEC/IEEE 15288 and ISO/IEC/IEEE 12207 throughout the lifecycle [14], [15].
Quality planning and governance are guided by the Quality Management Plan principles of IEEE 730 and ISO 90003 [16], [17], while software product quality targets are evaluated using the ISO/IEC 25010 quality model [18].
Requirements, design, and testing artifacts are kept consistent and traceable via IEEE-compliant specification and test documentation practices, including IEEE 830 / ISO/IEC/IEEE 29148 and IEEE 829 / ISO/IEC/IEEE 29119 [19], [20], [21], [22].
Code quality is supported through peer reviews, automated testing at multiple levels, and adherence to recognized implementation conventions, ensuring that conformance evidence is available across requirements, architecture, implementation, and test results.

Intellectual Integrity: All external technologies, data sources, and content are properly attributed. No plagiarism or intellectual property violations occur in code, documentation, or content.

Competence and Skill Development: Team members acknowledge knowledge gaps in specialized areas (geospatial processing, OSRM deployment, routing algorithms) and commit to skill development through study, tutorials, and consultation with academic advisors.

1.3.2 Data Privacy and Security

User Data Protection: GUIDE is designed with privacy-by-default principles. No personal data (names, contact information, location history) is collected or stored. User trip preferences (duration, categories, exclusions) are processed transiently within the session context and not retained after completion. Session-based architecture eliminates persistent user tracking.

Secure Communication: All client-server communication occurs over HTTPS/TLS 1.3[23] to protect user inputs during transmission. While trip preferences are not highly sensitive, encryption prevents third-party interception and maintains user trust.

Third-Party Data Handling: GUIDE does not share user data with external services beyond necessary API interactions (OSRM for routing, OSM tile servers for map rendering). No analytics tracking or data monetization occurs.

1.3.3 Algorithmic Fairness and Transparency

Non-Discriminatory Recommendations: POI ranking algorithms are designed to be objective and fair. Rankings are based on some objective features such as category relevance, visit duration, related keywords, etc.

The system does not favor destinations based on commercial partnerships, advertising, or popularity manipulation. All POIs within matching categories receive equal algorithmic consideration.

Transparency of Recommendations: Users understand that route generation is algorithm-driven, not human-created. The system provides clear explanations of route logic (distance minimization, category matching) and enables users to modify recommendations, maintaining user agency rather than imposing algorithmic or AI-based authority.

1.3.4 Cultural Sensitivity and Representation

Respectful Content Curation: Descriptions of cultural heritage sites, religious landmarks, and historical locations are curated with cultural sensitivity. Content sourced from authoritative references (UNESCO, Wikipedia) undergoes manual review to ensure the accuracy of the content.

Multilingual Accuracy: Turkish, English, and German descriptions are reviewed for linguistic accuracy and cultural appropriateness. Turkish proper noun pronunciations in TTS audio are verified to respect cultural authenticity and avoid mispronunciations that could be perceived as disrespectful.

1.3.5 Accessibility and Universal Design

Multilingual Support: Providing content in Turkish, English, and German enhances accessibility for international travelers and Turkish citizens with varying language preferences. This supports inclusive tourism and reduces language barriers as obstacles to cultural exploration.

Audio Guidance for Visual Impairments: Text-to-speech audio serves users with visual impairments or reading difficulties. While GUIDE is not a specialized accessibility application, audio content provides alternative access modalities aligned with universal design principles.
Responsive Interface Design: The web interface is designed to be usable across desktop, tablet, and mobile devices through responsive layouts. This ensures accessibility regardless of users' device preferences or economic constraints.

1.3.6 Environmental and Social Impact

Contribution to Tourism Accessibility: GUIDE contributes to the tourism ecosystem by simplifying travel planning and reducing informational barriers for both domestic and international travelers. By providing structured itineraries, multilingual descriptions, and integrated route guidance, the system helps users explore destinations more efficiently, potentially encouraging longer stays and broader engagement with local tourism services.
Cultural Heritage Education: POI descriptions and audio guidance provide users with basic historical and cultural context for visited locations. This helps users better understand the significance of destinations and supports a more informed travel experience.
Open-Source Contribution: GUIDE's reliance on open-source technologies (OSM, OSRM, Leaflet.js, Python, FastAPI) aligns with community-driven development principles. Upon project completion, consideration will be given to releasing GUIDE as an open-source project, contributing to the travel technology ecosystem, and enabling future academic or community enhancements.

2 REQUIREMENTS

This section presents the system's Functional, External Interface, and Performance requirements, which define the core behaviors, interactions, and performance expectations of the GUIDE system. All requirements are organized according to their categories and follow a consistent structure. GUIDE requirements are specified following IEEE 830[19] and ISO/IEC/IEEE 29148[20] standards.

2.1 Functional Requirements

FR-H01 – User Input for Route Generation

Description: The system shall allow the user to provide inputs for route generation.
Related Low-Level Requirements:

- **FR-L01:** The system shall allow the user to specify city/region, trip duration, distance limit, category filters, and places to exclude.
- **FR-L06:** The system shall validate user input for route generation and provide appropriate error messages.

FR-H02 – AI-Based Route Generation

Description: The system shall generate an AI-based route plan according to the user's preferences and time constraints.

Related Low-Level Requirements:

- **FR-L02:** The system shall organize the generated route on a day-by-day basis and optimize it based on user inputs.

FR-H03 – Modification of Suggested Destinations

Description: The system shall allow the user to modify suggested destinations.

Related Low-Level Requirements:

- **FR-L03:** The system shall let the user remove unwanted stops and automatically regenerate an optimized route.

FR-H04 – Destination Information Display

Description: The system shall display descriptive information for each destination.

Related Low-Level Requirements:

- **FR-L04:** The system shall display text descriptions, images, and map locations for each destination.

FR-H05 – Text-to-Speech (TTS)

Description: The system shall support text-to-speech (TTS) functionality for destination descriptions.

Related Low-Level Requirements:

- **FR-L05:** The system shall provide TTS in Turkish, English, and German.

FR-H06 – Viewing the Route on a Map

Description: The system shall allow the user to view the generated route on a map.

Related Low-Level Requirements:

- **FR-L07:** The system shall allow the user to zoom, pan, and interact with the route map.

FR-H07 – Visit Duration Estimation

Description: The system shall estimate the average visiting duration for each stop and use these estimations when planning time-limited trips.

Related Low-Level Requirements:

- None.

FR-H08 – Error Handling and User Feedback

Description: The system shall handle system, network, and data-related errors gracefully and provide meaningful feedback to the user.

Related Low-Level Requirements:

- **FR-L08:** The system shall display user-friendly error messages when route generation fails.
- **FR-L09:** The system shall prevent application crashes due to invalid or incomplete user input.

FR-H09 – Data Source Transparency

Description: The system shall clearly indicate the data sources used for route generation and destination information.

Related Low-Level Requirements:

- **FR-L10:** The system shall inform users that routing data is based on OpenStreetMap.
- **FR-L11:** The system shall indicate that POI descriptions are sourced from publicly available references.

2.2 External Interface Requirements

EIR-H01 – Web-Based User Interface

Description: The system shall operate via a web-based user interface.

Related Low-Level Requirements:

- **EIR-L01:** The system shall support access from both desktop and mobile browsers.
- **EIR-L02:** The system shall provide intuitive navigation with clear menus, buttons, and interactive elements.
- **EIR-L03:** The system shall support access from different screen sizes.

2.3 Performance Requirements

PR-H01 – Route Generation Performance

Description: The system shall generate and display a complete route suggestion within an average of 5 seconds after receiving the user's request.

Related Low-Level Requirements:

- None.

PR-H02 – User Capacity Performance

Description: The system shall support at least 1000 concurrent users.

Related Low-Level Requirements:

- None.

REFERENCES

[1] OSRM Project. "Open Source Routing Machine HTTP API Documentation," Accessed: Jan. 24, 2026. [Online]. Available: http://project-osrm.org.

[2] OpenStreetMap Foundation. "OpenStreetMap," Accessed: Jan. 24, 2026. [Online]. Available: https://www.openstreetmap.org.

[3] Leaflet.js. "Interactive Maps Library for JavaScript," Accessed: Jan. 24, 2026. [Online]. Available: https://leafletjs.com.

[4] UNESCO. "UNESCO World Heritage Centre – Türkiye." Access Date: 2025-09-16, Accessed: Sep. 16, 2025. [Online]. Available: https://www.unesco.org.tr.

[5] Wikimedia Foundation. "MediaWiki REST API Documentation," Accessed: Jan. 24, 2026. [Online]. Available: https://www.mediawiki.org/wiki/API:REST_API.

[6] ElevenLabs. "Text-to-Speech API Reference," Accessed: Jan. 24, 2026. [Online]. Available: https://elevenlabs.io.

[7] OpenStreetMap Foundation. "Tile Usage Policy." Describes CDN-based tile delivery, usage limits, and external dependency of OpenStreetMap tile servers., Accessed: Jan. 24, 2026. [Online]. Available: https://operations.osmfoundation.org/policies/tiles/.

[8] Flickr. "Creative Commons on Flickr." Describes Creative Commons licensing options available for images hosted on Flickr., Accessed: Jan. 24, 2026. [Online]. Available: https://www.flickr.com/creativecommons/.

[9] Pixabay. "Pixabay License." Defines royalty-free usage terms for images and videos provided by Pixabay., Accessed: Jan. 24, 2026. [Online]. Available: https://pixabay.com/service/license/.

[10] European Union, *Regulation (EU) 2016/679 – General Data Protection Regulation (GDPR)*, Official Journal of the European Union. Access Date: 2026-01-24, 2016.

[11] Python Software Foundation. "Python 3.12 Documentation," Accessed: Jan. 24, 2026. [Online]. Available: https://docs.python.org/3.12/.

[12] Mozilla Developer Network. "JavaScript — MDN Web Docs." Authoritative documentation for the JavaScript programming language., Accessed: Jan. 24, 2026. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript.

[13] GitHub, Inc. "GitHub Documentation." Official documentation for GitHub version control, collaboration, and repository management platform., Accessed: Jan. 24, 2026. [Online]. Available: https://docs.github.com.

[14] ISO/IEC/IEEE, "ISO/IEC/IEEE 15288:2015 – Systems and software engineering – System life cycle processes," International Organization for Standardization, Tech. Rep., 2015, Access Date: 2026-01-24.

[15] ISO/IEC/IEEE, "ISO/IEC/IEEE 12207:2017 – Systems and software engineering – Software life cycle processes," International Organization for Standardization, Tech. Rep., 2017, Access Date: 2026-01-24.

[16] IEEE Computer Society, "IEEE Std 730-2014 – IEEE Standard for Software Quality Assurance Processes," IEEE, Tech. Rep., 2014, Access Date: 2026-01-24.

[17] ISO/IEC, "ISO 90003:2014 – Software engineering – Guidelines for the application of ISO 9001:2015 to computer software," International Organization for Standardization, Tech. Rep., 2014, Access Date: 2026-01-24.

[18] ISO/IEC, "ISO/IEC 25010:2011 – Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models," International Organization for Standardization, Tech. Rep., 2011, Access Date: 2026-01-24.

[19] IEEE Computer Society, "IEEE Std 830-1998 – IEEE Recommended Practice for Software Requirements Specifications," IEEE, Tech. Rep., 1998, Access Date: 2026-01-24.

[20] ISO/IEC/IEEE, "ISO/IEC/IEEE 29148:2018 – Systems and software engineering – Life cycle processes – Requirements engineering," International Organization for Standardization, Tech. Rep., 2018, Access Date: 2026-01-24.

[21] IEEE Computer Society, "IEEE Std 829-2008: Software and System Test Documentation," IEEE, Tech. Rep., 2008, Access Date: 2026-01-24.

[22] ISO/IEC/IEEE, "ISO/IEC/IEEE 29119-1:2013 – Software and systems engineering – Software testing – Part 1: Concepts and definitions," International Organization for Standardization, Tech. Rep., 2013, Software testing standard series (29119). Access Date: 2026-01-24.

[23] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3," IETF, Tech. Rep., 2018, RFC 8446. Access Date: 2026-01-24.

Note: Spell checking and basic grammatical verification were performed using Grammarly: (`https://www.grammarly.com`).

END OF PROJECT SPECIFICATIONS REPORT

GUIDE (Guided User Itinerary & Destination Explorer)
QUAD-CORE Team | TOBB ETÜ
Version 1.0 | January 2026
