Wanderlust System Requirement Document

Content

1. Introduction

This document specifies the requirements of the *Wanderlust* app, providing developers with the necessary information for design and implementation as well as for system testing and approval.

1.1 Document Overview

In addition to this introductory section, the following sections are organized as described below.

1. **Section 2 - System Overview:** Provides an overview of the system, characterizing its scope and describing its users.
2. **Section 3 - Functional Requirements (Use Cases):** Specifies all system use cases, describing the event flows, priorities, actors, inputs, and outputs of each use case to be implemented.
3. **Section 4 - Non-Functional Requirements:** Specifies all non-functional system requirements, divided into usability, reliability, performance, security, distribution, standards, and hardware and software requirements.
4. **Section 5 - References:** Provides references to other documents used to make this document.

1.2 Conventions, terms and abbreviations

Correct interpretation of this document requires knowledge of some specific conventions and terms, which are described below.

1.2.1 Requirements Identification

By convention, requirements are referenced by the name of the subsection where they are described, followed by the requirement identifier according to the following specification:
[name of subsection. requirement identifier]
For example, the functional requirement [Data Recovery. RF016] must be described in a subsection called “Data Recovery” in a block identified by the number [RF016]. The non-functional requirement [Reliability. NF008] must be described in the Non-functional Reliability Requirements section in a block identified by [NF008].
Requirements must be identified with a unique identifier. Numbering starts with the identifier [RF001] or [NF001] and continues to increment as new requirements emerge.

1.2.2 Requirement Priorities

In order to prioritize the requirements, in sections 4 and 5, the designations “essential”, “important” and “desirable” were adopted.

5. **Essential** is the requirement without which the system does not start. Essential requirements are essential requirements that must be implemented without fail.
6. **Important** is the requirement without which the system goes into operation, but unsatisfactorily. Important requirements must be implemented, but if not, the system can be deployed and used nonetheless.

7. **Desirable** is the requirement that does not compromise the basic functionality of the system, ie the system can function satisfactorily without it. Desirable requirements may be left for later versions of the system if there is not enough time to implement them in the version being specified.

2. System Overview

The *Wanderlust* app is a tool developed for the tourism market. Using the app, the user - usually tourists or local travelers - can receive information of the chosen location. The location filter refines the search to a city.
After defining the place of interest, the user will receive suggestions of places to visit in the main menu and also suggestions of typical dishes, bringing details of their main ingredients.
In addition, the user can interact with a chatbot, called Wander, developed for the platform. It is chatbot's function to provide the user with tips on the local code of conduct for their region of interest, thus enabling tourists to get to know the local culture more fully and to respect local habits and spaces.
Chatbot technology will use the data mining process to optimize the performance of this tool.
Wanderlust, with its tips, contributes to a more complete tourist experience in the process of immersion in a new culture, as well as acting as a support for possible unforeseen events.

3. Functional Requirements

3.1 Register

[FR001] Create user profile

Use case description: This use case allows the user to create and store their profile, generating a new user registration in the system.

Priority: ■ Essential □ Important □ Desirable

Entries and preconditions: None.

Exits and Postcondition: A user profile is registered in the system.

[FR002] Delete user profile

Use case description: This use case allows the user to delete their profile from the system user registry. The user can only delete their own registration once it is logged in the system.

Prioridade: ■ Essential □ Important □ Desirable

Entries and preconditions: You receive as input the user profile you want to delete.

Saídas e pós-condição: o usuário consegue excluir seu perfil no cadastro de usuários.

[FR003] Change component

Use Case Description: This use case allows the user to change their user profile data.

Priority: ■ Essential □ Important □ Desirable

Inputs and preconditions: You receive as input the field you want to change and the new data.
Exits and postcondition: A field is changed in the user profile.

3.2 Search

[FR001] Make a search

Use-case description: The user can search for the region they want to receive information about.

Priority: ■ Essential □ Important □ Desirable

Entries and Preconditions: The system must receive as input the name of the country, city or region desired by the user.

Outputs and Postcondition: You view the information proposed by the system and defined in the project scope.

[FR002] Important Supporting Contacts

Use case description: This use case gives the user access to support contacts that can assist the user.

Prioridade: □ Essential ■ Important □ Desirable

Inputs and preconditions: The system collects information from the primary search performed on the main scope of the system.

Outputs and postcondition: User can receive important contact information from support.

[FR003] Place Suggestions

Use case description: This use case suggests places to visit in the region chosen by the user.

Prioridade: ■ Essential □ Important □ Desirable

Inputs and preconditions: The system collects information from the primary search performed on the main scope of the system.

Outputs and postcondition: User can receive the suggestions.

[FR004] Typical dishes suggestions (and ingredients)

Use case description: This use case suggests typical dishes and their main ingredients from the region chosen by the user.

Prioridade: ■ Essential □ Important □ Desirable

Inputs and preconditions: The system collects information from the primary search performed on the main scope of the system.

Outputs and postcondition: User can receive the suggestions.

3.3 Interactivity

[FR001] A ChatBot with AI

Use case description: The system provides the user with an interactivity section with a chatbot, which will guide the user on codes of conduct and socio-cultural demands about the surveyed region.

Priority: ■ Essential □ Important □ Desirable

Inputs and preconditions: From user input or from primary search data, chatbot receives the information.

Outputs and Postcondition: Chatbot responds to inputs received by the system through its learning scope.

Nonfunctional Requirements

[NF001] Usability

The user interface is of vital importance to the success of the system. Mainly for being a system that purports to be a pocket guide.

The system will have a user-friendly primary interface without tiring the most experienced users. Images to illustrate, clean design and interactive screen transitions will help make the project proposal possible.

Prioridade: ■ Essential □ Important □ Desirable

[NF002] Performance

Because it is a mobile application that is characterized as a real-time information system, performance is essential to the system's operation. Response time there is a search, or accuracy of that answer should be prioritized in application development.

Prioridade: □ Essential ■ Important □ Desirable

[NF003] Hardware e Software

The system will be developed for the Android platform, so it will be developed using the Java link as a base through the Android Studio tool. For the infrastructure part, the Python language will be used to be the basis of artificial intelligence for the mining of texts received and sent by BOT. The device, in turn, must have Android as its operating system, as well as internet access and integrated GPS.

Prioridade: □ Essential ■ Important □ Desirable

4. References

1. PRESSMAN, Roger S.; MAXIM, Bruce R. Software Engineering: A PRACTITIONER ' S APPROACH. 8. ed. New York: McGraw-Hill Education, 2015. 993 p. ISBN 978-0-07-802212-8.
2. Artigo Engenharia de Software - Introdução à Engenharia de Requisitos. https://www.devmedia.com.br/artigo-engenharia-de-software-introducao-a-engenharia-de-requisitos/8034