March 10th 2011

Requirements Specification Document

Introduction to Human-Computer interfaces

Project: Design of an interactive application

Table of Contents

[I. Objectives of the project 3](#_Toc287449662)

[II. Targeted user-base 3](#_Toc287449663)

[1. General description 3](#_Toc287449664)

[2. Personas 3](#_Toc287449665)

[III. Functional Requirements 3](#_Toc287449666)

[1. User input collection 3](#_Toc287449667)

[2. Analysis 3](#_Toc287449668)

[3. Use Cases 4](#_Toc287449669)

[a. Use Case diagram 4](#_Toc287449670)

[b. Scenarios 5](#_Toc287449671)

[IV. Quality requirements 8](#_Toc287449672)

# Objectives of the project

# Targeted user-base

## General description

## Personas

# Functional Requirements

## User input collection

## Analysis

## Scenarios

### Use Case diagram



Figure : Use cases diagram

One of Emma’s friends just called her and invited her to attend a pillow party (as the parents are out of town). However, her friend’s is a bit far, and Emma doesn’t know that area of the city very well. Since she is already a bit late, she wants to go there as fast and soon as possible. She would like to know the earliest available public transportation itinerary to go from the bus station in front of her house, to her friend’s address. A little over 12 hours later, the party is over, and now Emma would like to go back home. Unfortunately, it turns out that for some reason she has trouble remembering at which station she had to take the bus, and the bus number as well for that matter! Hence she would like to do the opposite of what she did when she came to her friends’. Namely find the itinerary between her friend’s address and the station in front of her house.

Today was a day like any other at the end of January for Lucas. Daylight had long waned and the bone chilling wind was howling furiously under the full moon. He had worked hard all day on the multi-million contract on which the future of the company depended and was hopelessly tired: it was time to think going back home. Were it up to him, he would have done so ages ago; however, he had some very urgent work to finish, which he estimated would take him about an hour. Usually not staying at work so late, he wanted to know how to get from the station next to his office to the station near his house as fast as possible in an hour time.

Pascale has an appointment with her oncologist this afternoon. Usually she goes to the hospital on foot, bus she has started feeling the weight of all her years and she is too tired to walk today. Since she does not own a car, she has the idea to take the bus. Unfortunately, she doesn’t really know the network at all and she would like to know what itinerary she has to take between her address and the hospital’s address. Since her meeting is at 15:00 she would really like to be able to specify she wants to arrive at 14:50.

### Scenarios

|  |  |
| --- | --- |
| **Station to Station Itinerary** | |
| **Issues:** | The user wants to know how to get from a station to another |
| **Requirements:** | * An Internet Access * Prior knowledge of the start and destination stops. |
| **Actors:** | * The user * The SEMITAG servers |
| **Goals:** | Find the optimal (in relation to travel time) itinerary between two stations at a given time |
| **Assumptions:** | The user has a least moderate knowledge of the SEMITAG network |
| **Scenario steps:** | * + The user starts typing the name of the start station   + The system incrementally shows the available stops matching the currently typed characters     - The user fully types the name OR Picks one of the propositions.   + The same process is repeated for the input of the destination station |
| **Exception cases:** | * If one of the source or destination station does not exist.   + The system displays a list of the first few (limited to 7) stations closest to what was typed. * If there are no available itineraries between the stations selected by the user, the system displays a message stating that no itineraries could be found with the selected parameter. |

|  |  |
| --- | --- |
| **Address to Address Itinerary** | |
| **Issues:** | The user wants to know how to get from an address to another address |
| **Requirements:** | An internet access. |
| **Actors:** | The user, The SEMITAG servers |
| **Goals:** | Find the optimal (in relation to travel time) itinerary between two addresses at a given time |
| **Assumptions:** | The user has no knowledge about the SEMITAG network |
| **Scenario steps:** | * + The user starts typing the name of the departure street/city   + The system incrementally shows the available streets/cities matching the currently typed characters   + The user fully types the name OR picks one of the propositions. * The user starts typing the street number * The system incrementally shows the available numbers matching the currently input characters * The user either fully types the number OR selects one of the propositions.   + The same process is repeated for the selection of the destination address. |
| **Exception cases:** | * If one of the street/city combination do not exist:   + The system displays a list of the first few (limited to 7) closest to what was typed. * The closest stop to the selected addresses are used for the itineraries * If no street number is selected, the start of the street is considered. * If there are no available itineraries between the addresses selected by the user, the system displays a message stating that no itineraries could be found with the selected parameter. |

|  |  |
| --- | --- |
| **Station to Address Itinerary** | |
| **Issues:** | The user wants to know how to get from a station to an address |
| **Requirements:** | An internet access.  Partial knowledge of the tag network |
| **Actors:** | The user, The SEMITAG servers |
| **Goals:** | Find the optimal (in relation to travel time) itinerary between two addresses at a given time |
| **Assumptions:** | The user has a partial knowledge about the SEMITAG network |
| **Scenario steps:** | * + The user starts typing the name of the start station   + The system incrementally shows the available stops matching the currently typed characters     - The user fully types the name OR Picks one of the propositions.   + The user starts typing the name of the destination street/city   + The system incrementally shows the available streets/cities matching the currently typed characters   + The user fully types the name OR picks one of the propositions. * The user starts typing the street number * The system incrementally shows the available numbers matching the currently input characters * The user either fully types the number OR selects one of the propositions. |
| **Exception cases:** | * If the departure station does not exist:   + The system displays a list of the first few (limited to 7) stations closest to what was typed. * If the destination street/city combination does not exist:   + The system displays a list of the first few (limited to 7) closest to what was typed. * The closest station to the selected arrival addresses is used for the itineraries. * If no street number is selected, the start of the street is considered. * If there are no available itineraries between the addresses selected by the user, the system displays a message stating that no itineraries could be found with the selected parameter. |

|  |  |
| --- | --- |
| **Address to station Itinerary** | |
| **Issues:** | The user wants to know how to get from an address to a station |
| **Requirements:** | An internet access.  Partial knowledge of the tag network |
| **Actors:** | The user, The SEMITAG servers |
| **Goals:** | Find the optimal (in relation to travel time) itinerary between an address and a station at a given time |
| **Assumptions:** | The user has a partial knowledge about the SEMITAG network |
| **Scenario steps:** | * + The user starts typing the name of the start street/city   + The system incrementally shows the available streets/cities matching the currently typed characters   + The user fully types the name OR picks one of the propositions. * The user starts typing the street number * The system incrementally shows the available numbers matching the currently input characters * The user either fully types the number OR selects one of the propositions.   + The user starts typing the name of the destination station   + The system incrementally shows the available stations matching the currently typed characters     - The user fully types the name OR Picks one of the propositions. |
| **Exception cases:** | * If the destination station does not exist:   + The system displays a list of the first few (limited to 7) stations closest to what was typed. * If the start address does not exist:   + The system displays a list of the first few (limited to 7) closest to what was typed. * The closest station to the selected arrival addresses is used for the itineraries. * If no street number is selected, the start of the street is considered. * If there are no available itineraries between the addresses selected by the user, the system displays a message stating that no itineraries could be found with the selected parameter. |

# Quality requirements

Info perturbations/ meilleures infos

Trop d’étapes