**3.2.1.3 Functional requirement 1.3**

**ID**: FR3

**TITLE**: User registration - Mobile application

**DESC**: Given that a user has downloaded the mobile application, then the user should be able to register

through the mobile application. The user must provide user-name, password and e-mail address. The user can choose to provide a regularly used phone number.

**RAT**: In order for a user to register on the mobile application.

**3.2.1.4 Functional requirement 1.4**

**ID**: FR4

**TITLE**: User log-in - Mobile application

**DESC**: Given that a user has registered, then the user should be able to log in to the mobile application.

The log-in information will be stored on the phone and in the future the user should be logged in

automatically.

**RAT**: In order for a user to register on the mobile application.

**DEP**: FR1, FR3

**3.2.1.5 Functional requirement 1.5**

**ID**: FR5

**TITLE**: Retrieve password

**DESC**: Given that a user has registered, then the user should be able to retrieve his/her password by email.

**RAT**: In order for a user to retrieve his/her password.

**DEP**: FR1

**3.2.1.6 Functional requirement 1.6**

**ID**: FR6

**TITLE**: Mobile application - Search

**DESC**: Given that a user is logged in to the mobile application, then the first page that is shown should be the search page. The user should be able to search for a restaurant, according to several search options. The search options are Price, Destination, Restaurant type and Specific dish. There should also be a free text search option. A user should be able to select multiple search options in one search.

**RAT**: In order for a user to search for a restaurant.

**DEP**: FR4

**3.2.1.7 Functional requirement 1.7**

**ID**: FR7

**TITLE**: Mobile application - Search result in a map view

**DESC**: Search results can be viewed on a map. On the map, the relevant and closest restaurants according to the user’s position are shown. A specific pin will represent a specific restaurant location. On each pin there should be an information link. There should be maximally 100 results displayed. The map view should have a default zoom. The map view should include a button that, when selected, should display different filtering options in a filtering menu.

**RAT**: The way results are displayed in a map.

**DEP**: FR6

**3.2.1.8 Functional requirement 1.8**

**ID**: FR8

**TITLE**: Mobile application - Search result in a list view

**DESC**: Search results can be viewed in a list. Each element in the list represents a specific restaurant. Each element should include the restaurant name, telephone number, type of food, distance according to the user’s position, average price, a short two-line description, a link to the restaurant’s web-page and an information link. There should be maximally 100 results displayed. If the result contains more restaurants than what can be displayed on the screen at one time, the user should be able to scroll through them. When searching by price the restaurants should be sorted according to the following order:

1. average price

2. distance

3. restaurant type

4. specific dish

When searching by a search option, other than price, the restaurants should be sorted according to

the following order:

1. distance

2. average price

3. restaurant type

4. specific dish

The list view should include a header with different selectable sorting options. The list view should include a button that, when selected, should display different filtering options in a filtering menu.

**RAT**: The way results should be displayed

**RAT**: In order for a user to switch between result views.

**DEP**: FR7, FR8

**3.2.1.11 Functional requirement 1.11**

**ID**: FR11

**TITLE**: Mobile application - Selecting the information link

**DESC**: A user should be able to select the information link, which is included on all result items. The link will direct the user to an information page, which includes a picture of the restaurant, the restaurant name, address, phone number, e-mail address, type of food, average price, restaurant description and a menu with name, description and price of the different dishes.

**RAT**: In order to show information about the restaurants.

**DEP**: FR7, FR8

**3.2.1.12 Functional requirement 1.12**

**ID**: FR12

**TITLE**: Mobile application - Search by price

**DESC**: A user should be able to input a maximum and a minimum price range. The result is displayed in

a list view by default.

**RAT**: In order for a user to search by price.

**DEP**: FR8

**3.2.1.13 Functional requirement 1.13**

**ID**: FR13

**TITLE**: Mobile application - Search by destination

**DESC**: A user should be able to input a maximum and a minimum distance, according to his/her position.

By default the minimum distance is set to 0 km and the maximum to 10 km. The user should be able to

input a higher or lower maximum distance and a higher minimum distance than set by default. The result

is displayed in a map view by default.

**RAT**: In order for a user to search by destination.

**DEP**: FR7

**3.2.1.14 Functional requirement 1.14**

**ID**: FR14

**TITLE**: Accepted input for price and destination search

**DESC**: Integers should be accepted as input when a user searches by price or destination. If the system receives an invalid input the user should be informed and prompted to insert an accepted input.

**RAT**: In order for a user to search with valid input.

**DEP**: FR12, FR13

**3.2.1.15 Functional requirement 1.15**

**ID**: FR15

**TITLE**: Mobile application - Search by restaurant type

**DESC**: A user should be able to select a restaurant type in a given list as input. The result is displayed in a

map view by default.

**RAT**: In order for a user to search by restaurant type.

**DEP**: FR7

**3.2.1.16 Functional requirement 1.16**

**ID**: FR16

**TITLE**: Mobile application - Search by specific dish

**DESC**: A user should be able to select a specific dish in a given list as input. The result is displayed in a

map view by default.

**RAT**: In order for a user to search by specific dish.

**DEP**: FR7

**3.2.1.17 Functional requirement 1.17**

**ID**: FR17

**TITLE**: Mobile application - Free-text search

**DESC**: A user should be able to conduct a search by providing either restaurant name, restaurant

description, restaurant address, restaurant type or restaurant menu in the free-text search field. The result

is displayed in a map view by default.

**RAT**: In order for a user to search through the free-text search.

**DEP**: FR7

**3.2.1.18 Functional requirement 1.18**

**ID**: FR18

**TITLE**: Mobile application - No match found

**DESC**: If no match is found the user should be informed but kept on the search page in order to get the

possibility to conduct a new search right away.

**RAT**: In order for user to conduct a new search if no match is found.

**DEP**: FR5

**3.2.1.19 Functional requirement 1.19**

**ID**: FR19

**TITLE**: Mobile application - Sorting results

**DESC**: When viewing the results in a list, a user should be able to sort the results according to price, distance, restaurant type, specific dish or restaurant name. When sorting by restaurant name, specific dish or restaurant type the results should be ordered alphabetically. When sorting by price the results should be ordered from cheapest to most expensive. When sorting by distance the results should be ordered from closets to furthest distance according to the user’s position. When the sort button for a specific search option is clicked, then the order should be reversed and ordered in a descending matter. If the sort button is clicked again the order of the results should be reversed.

RAT: In order for a user to sort results in a list.

DEP: FR8**3.2.2 User Class 2 - Restaurant Owner**

**3.2.2.1 Functional requirement 2.1**

ID: FR22

Feature: Create an account In order to create an account A restaurant owner Should register on the web-portal Scenario: Required information for registration Given the restaurant owner wants to create an account And the restaurant owner does not have an account When the restaurant owner registers on the web-portal by providing user-name

And password

And address

And e-mail address

And phone number

Then the restaurant owner should be able to apply for verification

Scenario: Full information for registration

Given the restaurant owner wants to create an account

And the restaurant owner does not have an account

When the restaurant owner registers on the web-portal by providing user name

And password

And address

And e-mail address

And phone number

And mobile number

Then the restaurant owner should be able to apply for verification

Scenario: Confirmed registration

Given the restaurant owner has applied for verification And has not received a confirmation e-mail after registration When the restaurant owner receives a confirmation e-mail Then the restaurant owner should be able to log in

**3.2.2.2 Functional requirement 2.2**

ID: FR23

Feature: Restaurant owner log-in

In order to use the system

A restaurant owner

Should be logged in to the web-portal

Scenario: Successful log-in

Given the restaurant owner wants to log in

When the restaurant owner logs in with his/her account

Then the restaurant owner should be logged in as a restaurant owner

Scenario: Retrieve password

Given the restaurant owner wants to log in

And has lost the password

When the restaurant owner enters his/her email address in the “Retrieve password” form

And submits the form

Then the restaurant owner should receive an email containing the password

**3.2.2.3 Functional requirement 2.3**

ID: FR24

Feature: Manage information

In order to manage information

A restaurant owner

Should be logged in to the web-portal

Scenario: Show fields for managing information

Given the restaurant owner is logged in

When the restaurant owner wants to manage information

Then the restaurant owner should be able to manage information in a form

Scenario: Filling in mandatory fields

Given the restaurant owner wants to fill in the mandatory fields of the form

When the restaurant owner provides average price

And address

And e-mail address

And phone number

And restaurant name

Then the restaurant owner has filled the mandatory fields of the form

Scenario: Filling in optional fields

Given the restaurant owner of a restaurant wants to fill in optional fields in the form

When the restaurant owner provides restaurant description

And menu

And type of restaurant

And picture of restaurant

And mobile phone

Then the restaurant owner has filled in optional fields in the form

Scenario: Filling in menu field

Given the restaurant owner wants to fill in the menu field in the form

When the restaurant owner provides dish name

And dish description

And dish price

Then the restaurant owner has filled in the menu field in the form

Scenario: Adding information with mandatory fields

Given the restaurant owner has filled in the mandatory fields of the form

When the restaurant owner submits the form

Then the information about the restaurant should be added

Scenario: Adding information with mandatory and optional fields

Given the restaurant owner has filled in the mandatory fields of the form

And filled in one or more optional fields of the form

When the restaurant owner submits the form

Then the information about the restaurant should be added

Scenario: Deleting information

Given the restaurant owner is logged in

And information exists

When the restaurant owner deletes information

Then the information should be deleted

Scenario: Editing information

Given the restaurant owner is logged in

And information exists

When the restaurant owner edits information

Then the information should be edited

**3.2.2.4 Functional requirement 2.4**

ID: FR25

Feature: Restaurant owner - Selecting preferred language on the web-portal

In order to understand the web-portal

A restaurant owner

Should be able to select a preferred language for the web-portal

Scenario: Select English as preferred language

Given the restaurant owner wants to select a preferred language

When the restaurant owner selects English as a new language

Then the web-portal will show all text in English

Scenario: Select Swedish as preferred language

Given the restaurant owner wants to select a preferred language

When the restaurant owner selects Swedish as a new language

Then the web-portal will show all text in Swedish

Scenario: Select French as preferred language

Given the restaurant owner wants to select a preferred language

When the restaurant owner selects French as a new language

Then the web-portal will show all text in French

Scenario: Select Spanish as preferred language

Given the restaurant owner wants to select a preferred language

When the restaurant owner selects Spanish as a new language

Then the web-portal will show all text in Spanish