



Requirement Specification

MunichX - Mobile application

CityMob

July 6, 2007

Contents

1	Introduction	1
2	Product Functions	3
2.1	Determined functions	3
2.2	Functions to be defined	4
2.3	Category definition and content specification	4
2.4	Premium functions	4
3	Graphical User Interface	5
4	Server Communication	8
4.1	Communication	8
4.2	API	8

CHAPTER 1

Introduction

This document constitutes the requirement specifications that define the mobile application that CityMob will develop for MunichX.

This document contains the following parts and chapters:

Chapter 2 - Functions

This chapter specifies the main functionality of the mobile application.

Chapter 3 - The Graphical User Interface

This chapter gives a view of the graphical user interface of the mobile application. This may be diverging from the actual implementation.

Chapter 4 - Server Communication

This chapter details the communication between the CityMob Server and the MunichX Server.

CHAPTER 2

Product Functions

We will in this chapter describe the functions that the mobile application should provide. We then specify the categories of content that will be available to the user. Finally, we define the functions that only the Premium Users gets access to.

2.1 Determined functions

These are elements that have been fixed during the development of the prototype, and that CityMob believes are necessary in order to keep the integrity, value and completeness of the mobile application. How these functions should be further defined and integrated into the mobile phone is dependent on the user tests and user prototyping that CityMob will perform. This to ensure adapted functions and usability.

- A main menu
- Categories mapping content of MunichX, each with its own logo
- Favorites
- Last searches
- Subway information
- Map

The integration of subway information must be further defined according to the technical possibilities offered by MVV. This also concerns the map. The main menu gives quick access to the main functionality. Other functions will be accessible through the softkeys and softkey menus.

2.2 Functions to be defined

These are elements that have to be further discussed and defined by both MunichX and CityMob.

- Ranking
- Location Based Positioning - GPS, manual, third-party
- Tell a friend function
- Community functions
- Advertisement

Finally, the softkeys must be defined according to the final functions that will be implemented.

2.3 Category definition and content specification

Here we specify what content should be brought to the mobile application, and how to categorize this content.

2.4 Premium functions

Here we define what functions are related to the premium users.

CHAPTER 3

Graphical User Interface

We will in this chapter specify the graphical user interface - GUI of the MunichX mobile application. The GUI incorporates the functionality defined in chapter 2.

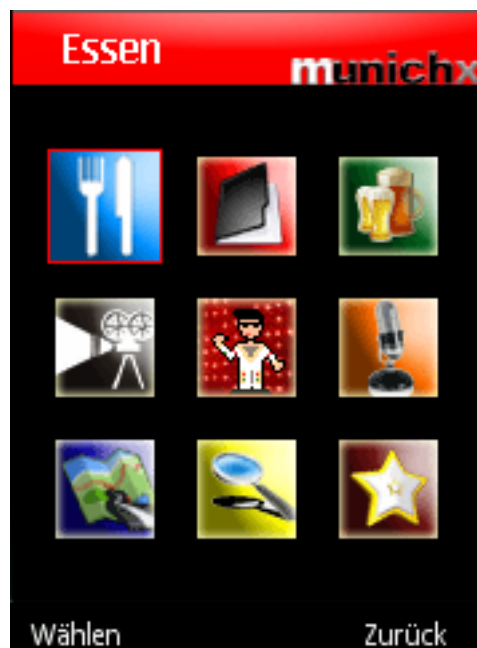


Figure 3.1: The main menu



Figure 3.2: Application screenshot 1 - Leftmost picture shows the searchpage. Middle picture shows pop-up where type of restaurant can be chosen. Rightmost picture shows search results.



Figure 3.3: Application screenshot 2 - Left picture shows the “was” part of the element pop-up. Right picture shows the “warum” part of the element pop-up.

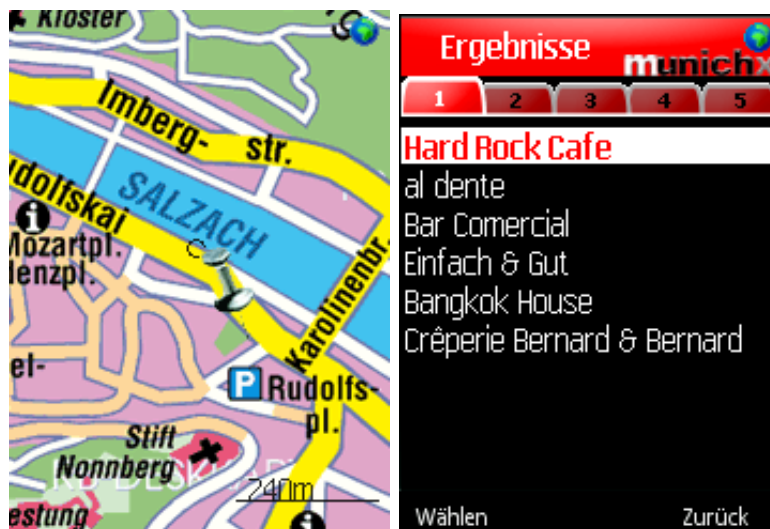


Figure 3.4: Application screenshot 3 - The left picture shows a screen shot of the map. The right picture shows the last search menu.

CHAPTER 4

Server Communication

This chapter deals with how the MunichX server communicates with the CityMob Server - CMF. We first define the communication protocol that should be used. Then we define the Application Programming Interface - API, that specifies what functionality the CMF must be able to access on the MunichX Server.

4.1 Communication

The preferred way of server interaction for CityMob is Web Services, but this can also be done through the use of SQL queries or other.

4.2 API

According to the functionality and content that the mobile application will have access to, we define a series of functions that the CityMob server can access, in order to perform its task in a satisfying and complete manner.