1.Introduction

1.1 Description of the document

This document is focused on the Data design and on how the interaction between the system and the users should be.

The scope of this document is to provide the design of the application, describing and justifying the reason of our choices, for example the Data Base schema, the User Experience that our application should provide to our users and the main classes that we will make on the Implementation phase.

The Data Base design is essential to build a consistent Data Base that supports in an efficient way all the queries and the manipulations on the data.

1.2 Definitions, acronyms and abbreviations

**User**: is a person who is registered in the database of the application. He has access to all the functions of the program that involves the requiring of a taxi, shared or not. He also has the possibility to save a list of preferred locations, that he can automatically choose when the System require from him an address as starting position or destination.

**Guest**: is a person who is using the application but is not registered in the database. He has access only to the registration functions.

**User Information**: all the information that concern a user; most of them have to be inserted during the registration (Name, Surname, tel. Number, e-mail, password), some of them can be inserted at any time after the registration (such as the personals locations) and others are assigned by the system (for example the number of blank-calls or the feedback).

**Feedback**: the feedback measures the reliability of a user. Is a simple relation between the total number of calls and the number of blank calls that a user have made (so a feedback equals to 1 means that he never missed a call).

**Basic User Information**: The information that a taxi driver visualizes when he receive a call. They are: Name, Surname, Feedback, Telephone Number of the user.

**Blank Call**: we define Blank-call a call for a taxi where the client is not at the starting location when the taxi arrives with a maximum late of X minutes, or a call that the user cancel before X minutes.

**Missed Call**: we define Missed-call a call for a taxi where the client is not at the starting location when the taxi arrives (X+1) or more minutes late.

**Partner**: someone who share a run with a user

**Pick-up place**: the Address where a user asks a taxi to come

**Taxi Driver**: a registered Taxi Driver

1.4 Document Structure

2. Architectual Design

2.1 Overview

We decomposed the system in several sub-systems for visualize better every single part.

The sub-systems in which our system is divided are:

* Call sub-system
* Shared-call cost sub-system
* User sub-system
  + Manage location sub-system
  + Manage account sub-system
  + Manage settings sub-system
* Log in sub-system
* Sign up sub-system

In this document we analyze every sub-system and show how the users will see and interact with our system.

2.2 High level components and their interaction

In this section we provides