myTaxiService
Design Document

Jacopo Strada, Luca Riva

December 9, 2015

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Table of Contents

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Component View

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Deployment View

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns
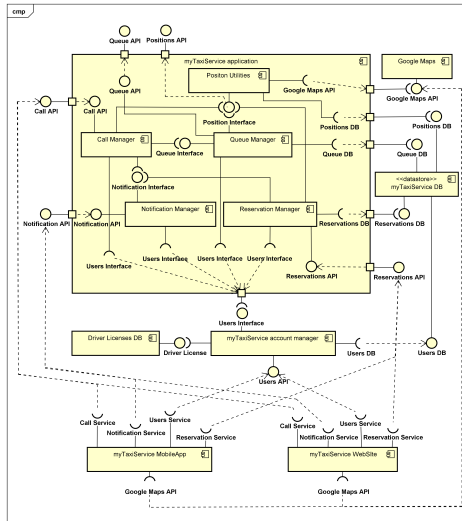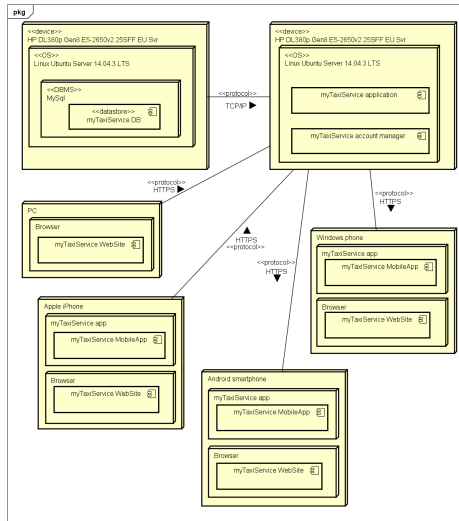
Component Interfaces I

Data Base: (The following methods are not available as APIs but are only private)

createNewClient(email, name, surname, dateOfBirth, phone, password)

createNewDriver(email, name, surname, license, phone, password)

checkPassword(userId, password)

findUserByEMail(email)

getDriverState(email)

updateDriverState(email, state)

getZones()

saveRide(origin, destination, date, time)

setRideAccepted(rideId, driver)

eraseRide(rideId)

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Component Interfaces II

Account Manager:

login(mail, password)
registerClient(email, firstName, lastName, telephoneNumber, password)
registerDriver(email, firstName, lastName, telephoneNumber, license, password)
To receive information about an user: *clientInfo(email)* or *drievrInfo(email)*
To change a driver state when they are taking a call: *changeDriverState(email, state)*

Call Manager:

To call a taxi: *forwardCall(email, GPSPosition)*
To show the details of a call (should be used by a driver): *showCallDetails(call)*

Queue Manager:

To retrieve the drivers' queue of a certain zone: *driversQueue(zone)*

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Component Interfaces III

Reservation Manager:

newReservation(email, origin, destination, time)
to show the details of a reservation:
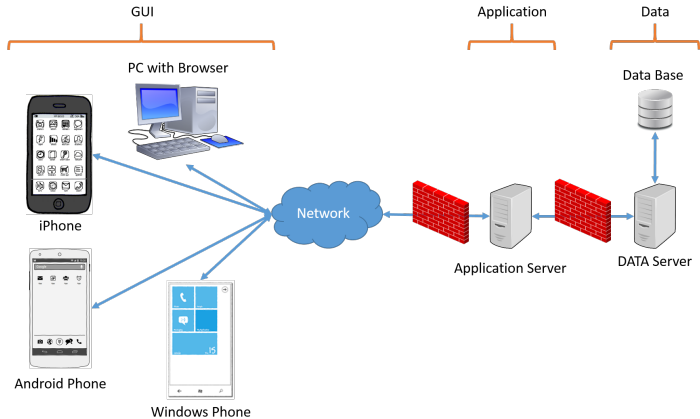*showReservationDetails(reservation)*

Notification Manager:

sendNotification(list<email >, message)

Position Utilities:

to find the zone associated to a given coordinate:
findZone(GPSCoordinate)

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Selected Architectural Styles and Patterns I

Architectural Design
Algorithm Design
User Interfaces Design

Component View
Deployment View
Component Interfaces
Selected Architectural Styles and Patterns

Selected Architectural Styles and Patterns II

Table of Contents

Algorithm Design
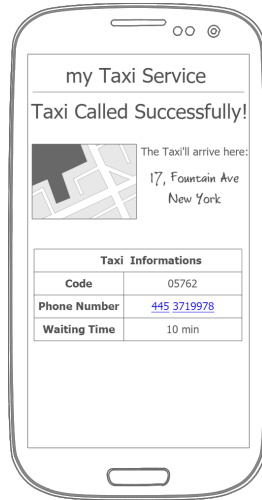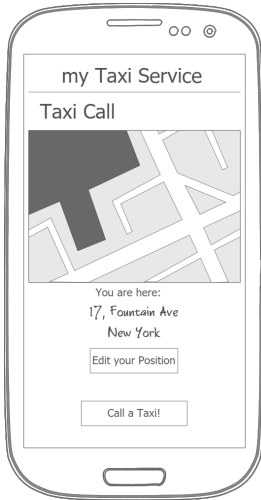
Require: the queue of a zone
Ensure: the return value is the driver who accepted the ride or NULL if nobody accepted
  for all drivers in queue (only once) **do**
      send a notification to the driver
      wait for 30 seconds (or less if a response is received)
      if the driver has accepted **then**
         remove driver from the queue
         set driver state as *unavailable*
         send a notification to the client
         return current driver
      else if the driver has rejected **or** the driver hasn't responded **then**
         move the driver from the begin to the end of the queue
      end if
  end for
  return NULL

Table of Contents

Clients' User Interfaces - Taxi Call

my Taxi Service

Taxi Call

You are here:

17, Fountain Ave
New York

Edit your Position

Call a Taxi!

my Taxi Service

Taxi Called Successfully!

The Taxi'll arrive here:

17, Fountain Ave
New York

Taxi Informations	
Code	05762
Phone Number	445 3719978
Waiting Time	10 min

Taxi Drivers' User Interfaces - New Requests