Chapter 1

Algorithm Design

This section is intended to better explain some functions and algorithms that can be ambiguous in the previous presentation of our system.

Algoritmi speciali usati nel nostro sistema, uno per uno descritti con pseudocodice o flow chart

algoritmi che potremmo mettere:

- quello per selezionare il messaggio e gestirlo (IO Manager)

  1)procedure selectMessage()

  2)

- taxi allocator (selezione delle ride non gestite e gestione)
- queues manager (check positions e ordinamento delle code)
- modify priority –¿ recall della funzione di ordinamento del queue manager
- credentials checker (check email, password)

Algorithm 1 Descrizione Algoritmo

```
1:  procedure EXAMPLE(par1, par2)                ▷ example algorithm
2:      par1 ← 1
3:      if i ≥ maxval then
4:          i ← 0
5:      else
6:          if i + k ≤ maxval then
7:              i ← i + k
8:          end if
9:      end if
10: end procedure
```

1.1 IO Manager Algorithms

Here we describe some functions and algorithms relatives to the IO Manager component.

1.1.1 Select Message

The select message procedure have to select the right message handler for an incoming/outgoing message. It's described by the following pseudocode:

Algorithm 2 Select Message Procedure

1: **procedure** SELECTMESSAGE($rawMessage$) ▷ rawMessage is an xml (or other markup language like json) message
2: $messageHandler \leftarrow messagesTypeMap.getMessageHandler($
3: $rawMessage.type)$
4: $messageHandler.handleMessage(rawMessage)$
5: **end procedure**

Note on handleMessage(): every single message class that is present on the IO Manager component, can handle incoming messages or outgoing messages and the implementation of handleMessage() for each of this sub-components is different depending on the message that has to be handled.

1.2 Requests and Reservations Manager

Here we describe some functions and algorithms relatives to the Requests and Reservations Manager component.

1.2.1 Check Rides To Allocate

The checkRidesToAllocate procedure is invoked by a scheduler every