Final task ISS-2020 Bologna

Introduction

*Remember our **motto**:*
*there is no **code** without a **project**, no **project** without **problem analysis** and no **problem** without **requirements**.*

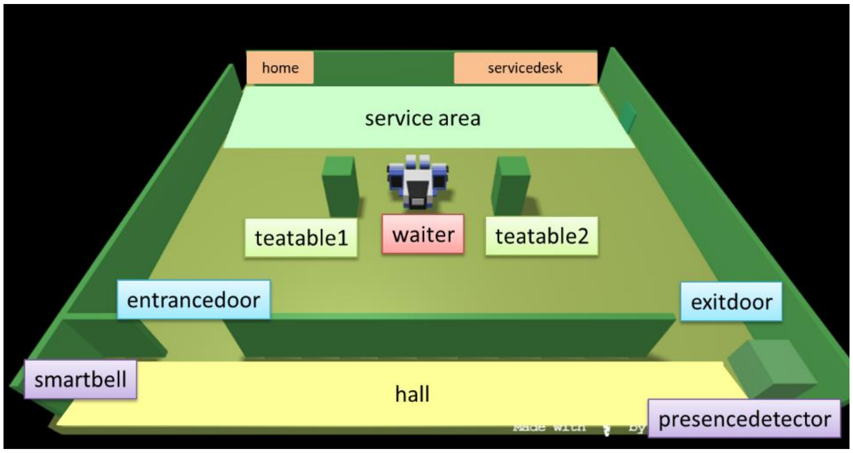
Requirements

The **manager** of a **tearoom** intends to regulate the access to the service by means of a ddr robot (**waiter**).

The **tearoom** is a rectangular room that includes:

- an **entrancedoor** to enter in the room and an **exitdoor** to exit form it;
- a number N (N=2) of **teatable**;
- a **servicearea** including a **servicedesk** at which works a **barman**;
- a **hall** equipped with a **presencedetector**, i.e. a device (e.g. a sonar) that can detect the presence of a person (or some other entity) in it.

The **waiter** can freely move along the borders of the tearoom, since there are no obstacles there.



User stories

As a **client**:

I intend to **notify** my interest in entering in a **safe tearoom**, **sitting** at a free teatable, **ordering** some tea, **consuming** it (within a limited amount of time **maxstaytime**) **paying** the service with my credit card and finally **leaving** the room.

For **safe tearoom**, I intend a tearoom with clean tea-tables posed at a proper distance; the room is populated by human clients whose body temperature is less than 37.5 degrees.

I can submit my notification of interest by hitting the **smartbell** located near the **entrancedoor** that will automatically measure my body temperature and send a request message to the **waiter**, by giving to me a unique **clientidentifier**.

If my body temperature is ok, but my request cannot be immediately satisfied (since the room is full), I will be **informed** by the **waiter** about the maximum waiting time.

As a **manager**:

I intend to be able to see the **current state** of the **tearoom** by using a browser connected to a webserver associated to the application

The **waiter** should perform the following tasks:

- **accept** the request of a client to enter in the tearoom if there is at least one teatable in the state **tableclean**, i.e. the table is free and has been properly cleaned
- **inform** the client about the maximum waiting time if there is no **tableclean**
- **reach** the **entrance** door and **convoy** the accepted client to the selected teatable
- **take** the order of the client and transmit it (using a wifi-device) to the **barman**
- **serve** the client when the barman says that the requested drink is ready
- **collect** the payment from the client when he/she has finished to consume or when the **maxstaytime** is expired
- **convoy** the client to the **exitdoor**
- **clean** the tea-table just freed by the client
- **rest** at my **home** when there is nothing to do.

Since the room could contain N clients at the time, the **waiter** should reduce as much as possible the waiting time of the requests coming from each client.

Per l'**analisi dei requisiti** riferirsi al documento.