Final task ISS-2020 Bologna

[LabISS-lectures site](#)

Tea-room COVID-19

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 *serviceare* 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.
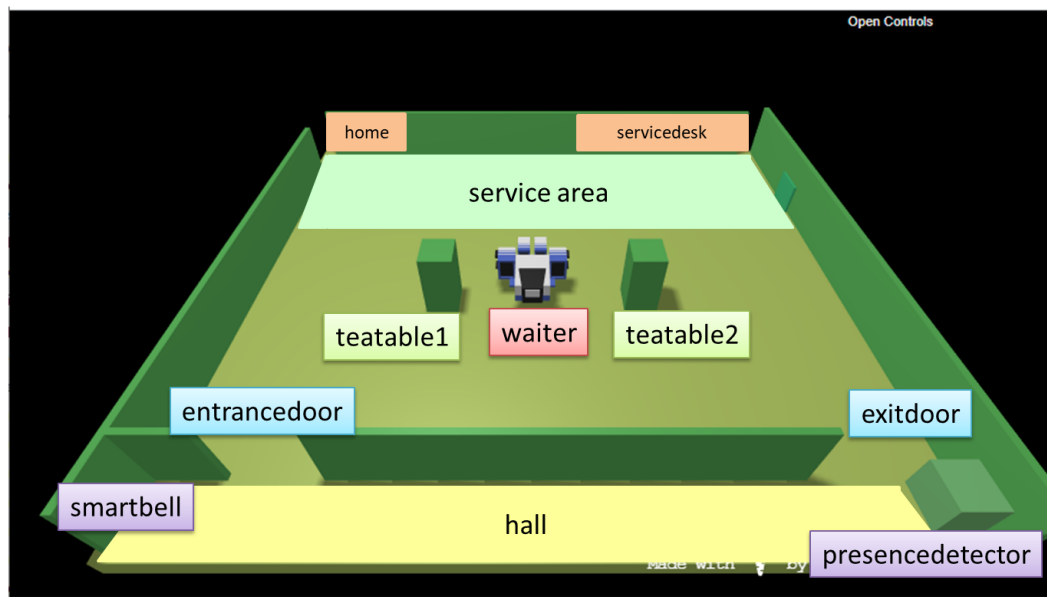
User stories

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 an unique *clientidentifier* .

- I my body temperature is ok, but my request cannot be immediately satisfied (since the room is full), I will be *informed* by the *waiter* aabout 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 web-server associated to the application.

Requirements

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.

Optional: one client in the hall

The *waiter* must open the *exitdoor* only when the hall is free, i.e. it must not open that door if the hall is already engaged by a client waiting to enter at the *entrancedoor* .

Non functional requirements

1. The ideal work team is composed of 3 persons. Teams of 1 or 2 persons (**NOT** 4 or more) are also allowed.
2. The team must present a workplan as the result of the requirement/problem analysis, including some significant **TestPlan**.
3. The team must present the sequence of SPRINT performed, with appropriate motivations.
4. The team must present (in synthetic, schematic way) the specific activity of each team-component.

By AN Unibo-DISI