Software Engineering

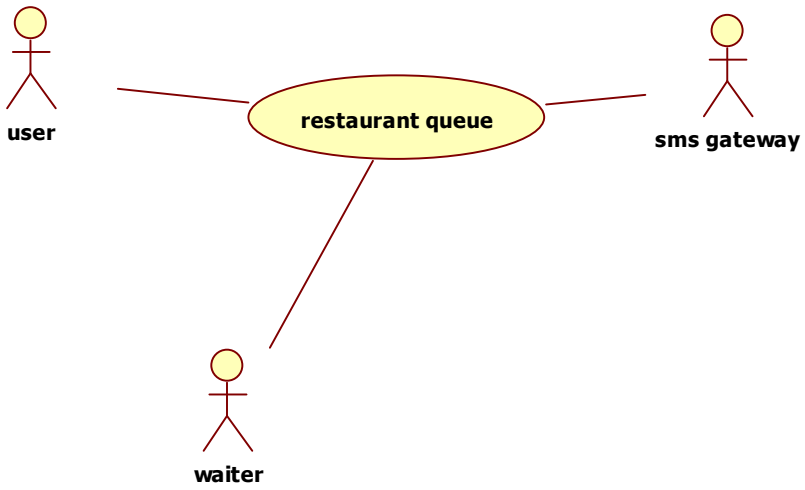
Books or notes are **not** allowed.
Write only on these sheets. **Concise** and **readable** answers please.

Surname, name, matricola _____

Restaurant queue management
At a busy restaurant in a mall, a system is used to manage the waiting queue.
When a new customer arrives, the waiter, given the number of people in the party, checks on the system the expected waiting time to get a table; then the customer provides her name and mobile phone number, the system then sends a confirmation SMS to the customer's phone. At this point the customer has a table reserved.
Then the customer is free to walk away and e.g. visit the shops in the mall. Every five minutes the system will send an SMS with the estimated residual waiting time.
When the table for the customer gets available, a waiter sets the table as ready and the system sends an SMS to the customer asking her to get to the restaurant as soon as possible to get the reserved table.
As the customer gets back to the restaurant she is seated at the reserved table. If the customer does not show up within a fixed time, then the next customer in the waiting queue is recalled.

In the following you should analyze and model the queue management application.

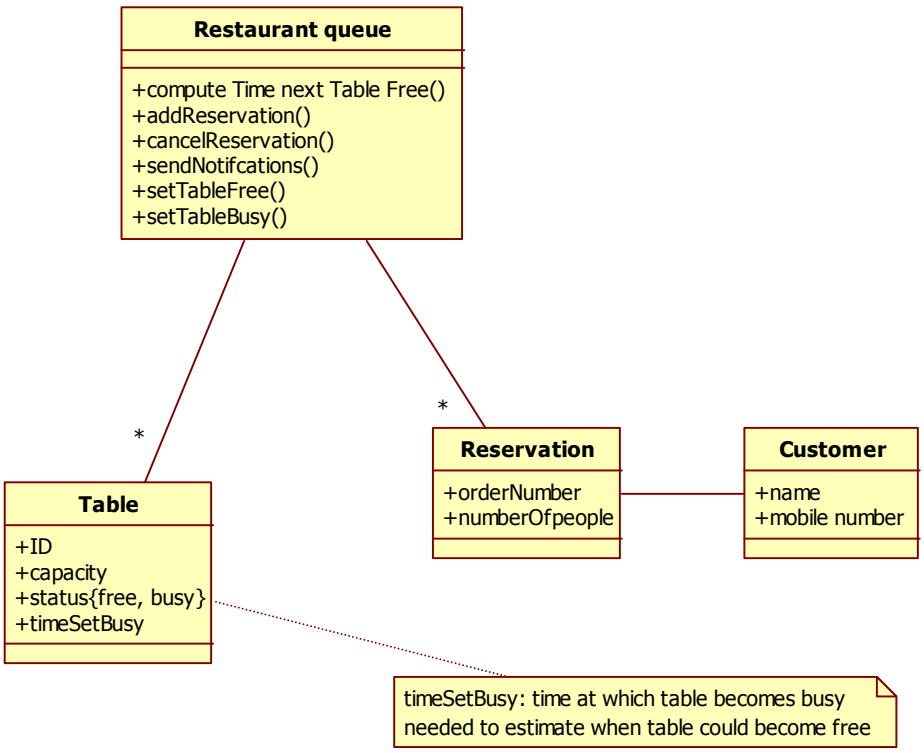
1 (15 points) – a. Define the context diagram (including relevant interfaces)



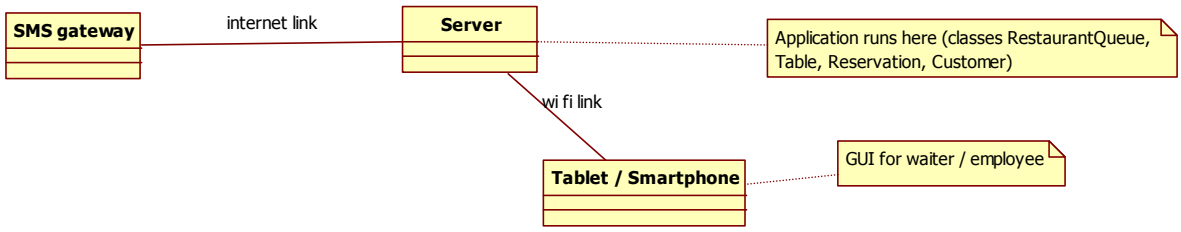
List the requirements in tabular form

ID	Type (Functional Non Functional)	Description
1	F	Set up (number and capacity of tables, waiters)
2	F	Compute time to have table (with no reservation yet)
3	F	Ask to reserve table (create a reservation)
4	F	Assign table, given reservation
5	F	Set table busy
6	F	Set table free
7	F	Communicate time to have table (given reservation)
8	F	Cancel reservation if not used within given time (ex 5 min)
9	NF usability	GUI on smartphone usable by non IT expert with max 5 min training
10	NF privacy	Users cannot access data of customers

Define the key concepts and entities and their relationships (UML class diagram) for the system



Define the system design (key hardware and software components of the system, UML class diagram) for the system. Describe the key design choices.



Define one scenario describing a customer that asks for a table at the restaurant
Precondition: No free table

Postcondition: Added Reservation R for Customer C at time t0. Estimated free table for C at time t1.

Step	Description	Req ID
1	Customer asks for a table	-
2	Waiter checks waiting time and tells to customer	2
3	Customer agrees to wait and asks for reservation	-
4	Waiter creates reservation	3
5	System sends to customer notification of time to wait	7

2 (8 points) -Define black box tests for the following class, using equivalence classes and boundary conditions.

```
Class LightController{
   private boolean onOff ;
   private int intensity;
   public void dimUp();
   public void dimDown();
   public void on();
   public void off();
   public int getIntensity();       // getter for intensity
   public boolean getOnOff();  // getter for onOff

}
```

The class controls a light, that can be on or off. When 'on', its intensity can range from 1 to 100. When 'off', its intensity is 0. At creation the light is off. The on() command sets the light on, at intensity 50. Functions dimOn() and dimOff() respectively add / reduce intensity by 10. When a sequence of dimOff() reduces the intensity to zero, the light is switched off.

Light on, off	Intensity <1, 1-100, >100	Valid	Test case
Off	[minint, 1[	N	T1. L = new LightController(); L.getIntensity() ==0; L.dimOff(); L.dimOff(); L.getIntensity() ==0; L.getOnOff() == off;
	[1, 100]	N	T2 L = new LightController(); L.getIntensity() ==0; L.dimOn(); L.dimOn(); L.getIntensity() ==0; L.getOnOff() == off;
	]100, maxint]	N	T3 L = new LightController(); L.getIntensity() ==0; repeat 20 times L.dimOn() L.getIntensity() ==0; L.getOnOff() == off;
On	[minint, 1[	N	T4 L = new LightController(); L.on(); L.getIntensity() ==50; L.getOnOff() == on; repeat 20 times L.dimOff() L.getIntensity() ==0; L.getOnOff() == off;
	[1, 100]	Y	T5 L = new LightController(); L.on();

			L.getIntensity() ==50; L.getOnOff() == on;
	]100, maxint]	N	T6 L = new LightController(); L.on(); L.getIntensity() ==50; L.getOnOff() == on; repeat 20 times L.dimOn() L.getIntensity() ==100; L.getOnOff() == on;

Boundary conditions: intensity 0 (T4), 1 (test to be added) and 100 (T6)

3 (7 points) – For the following function define the control flow graph, and define test cases to obtain the highest possible node coverage, edge coverage, multiple condition coverage, loop coverage, path coverage.
For the test cases, **write only the input value**.

```
1. int is_sorted(int A[], int n){
2.   int i;
3.   for(i=1; i<n; ++i){
4.         if(A[i] < A[i-1]){
5.         return 0;
6.         }
7.   }
8.   return 1;
9. }
```

Coverage type	Number of test cases needed to obtain 100% coverage	Coverage obtained with test cases defined (%)	Test cases defined
Node	2	100	T1 in {2,1,3} out 0 T2 in {1,2,3} out 1
Edge	2	100	T1, T2
Multiple condition	Not applicable		
Loop	3	100	T1, many times T3 in {1} no times T4 in {1,2} 1 time
Path	infeasible		

4 (1 points) – Describe briefly the ‘repository’ architectural style

5 (1 points) – What are the possible type of defects in a code module?

6 (1 points) – In the context of configuration management, what is a baseline ?

7 (1 points) – In the context of project management, give the definition of ‘deliverable’