Software Engineering

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

Surname, name, matricola ____

Personal Health Monitoring

A personal health monitoring system allows to collect data about one's physical state to get a better view on one's health.

The system has three parts. An app running on a smartphone (IOs or Android), a heart rate monitor (as a chest belt with a sensor, connected via Bluetooth to the smartphone and the app), and a backend web site.

The app collects automatically data like the heart rate (from the chest belt), position and accelerations (from the GPS and accelerometer on the smartphone). From the accelerometer the app estimates the foot steps of the person, and in general her movements and physical activity.
Further, the user of the app can enter on the app (or on the web site) the food eaten and other medical information. The food eaten can be entered in two ways. Inserting the weight of individual ingredients eaten (ex 50g sugar, 100g pasta, 100g broccoli) or inserting the dish eaten (ex pasta carbonara one dish, spritz two glasses). From the food eaten the application computes the specific intake in terms of basic elements (carbohydrates, fat, protein, minerals, vitamins) and energy (calories).
Other medical information inserted is the sex, age, weight of the person, the height, and the belly circumference. From the medical information the application computes the BMI (body mass index).
All data (physical activity, heart rate, food eaten, weight etc) can be shown graphically on time lines.

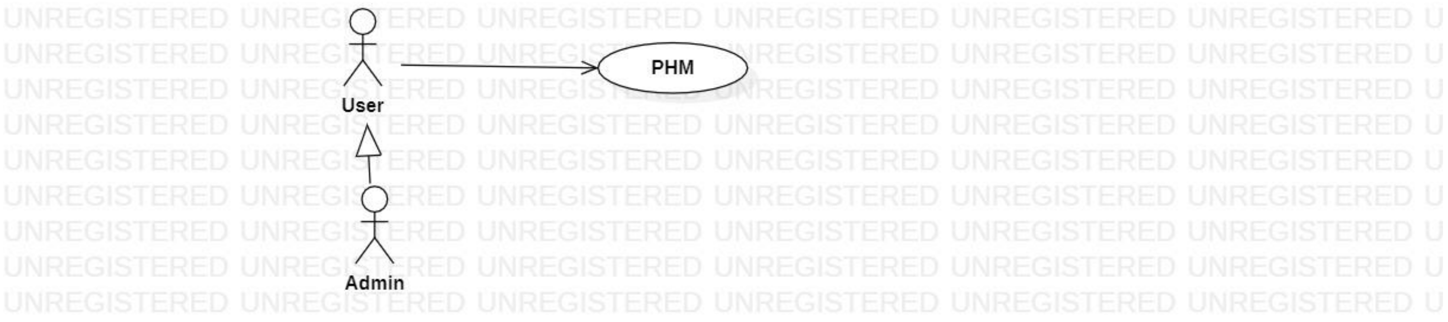
By aggregating all the available information the app estimates and shows on a timeline the level of fitness of the person, and proposes actions to do (like eat less, move more..).
All functionalities are available both on the app and on the web site. While the app is clearly personal, the web site is accessible by registered users only.

In the following you should analyze and model the Personal health monitoring system.

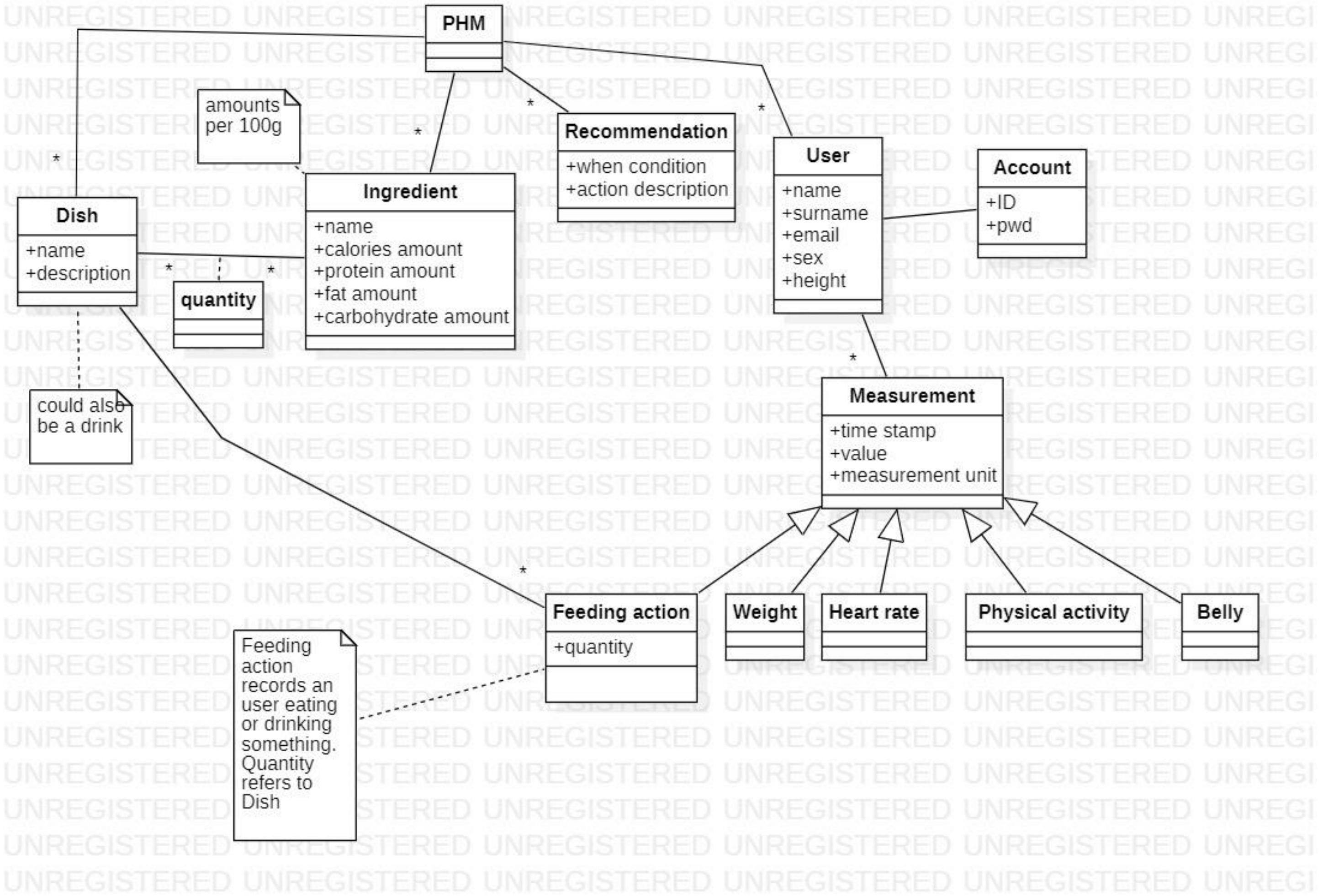
1 – a. Define the **context diagram** (including relevant interfaces)

Actor	Physical interface	Logical interface
User	Smartphone or PC	GUI
Admin	PC	GUI

The heart could be considered an actor, but I consider it as part of User. Accelerometer and GPS in the smartphone could also be considered actors, but I consider them as commodities from the operating system.

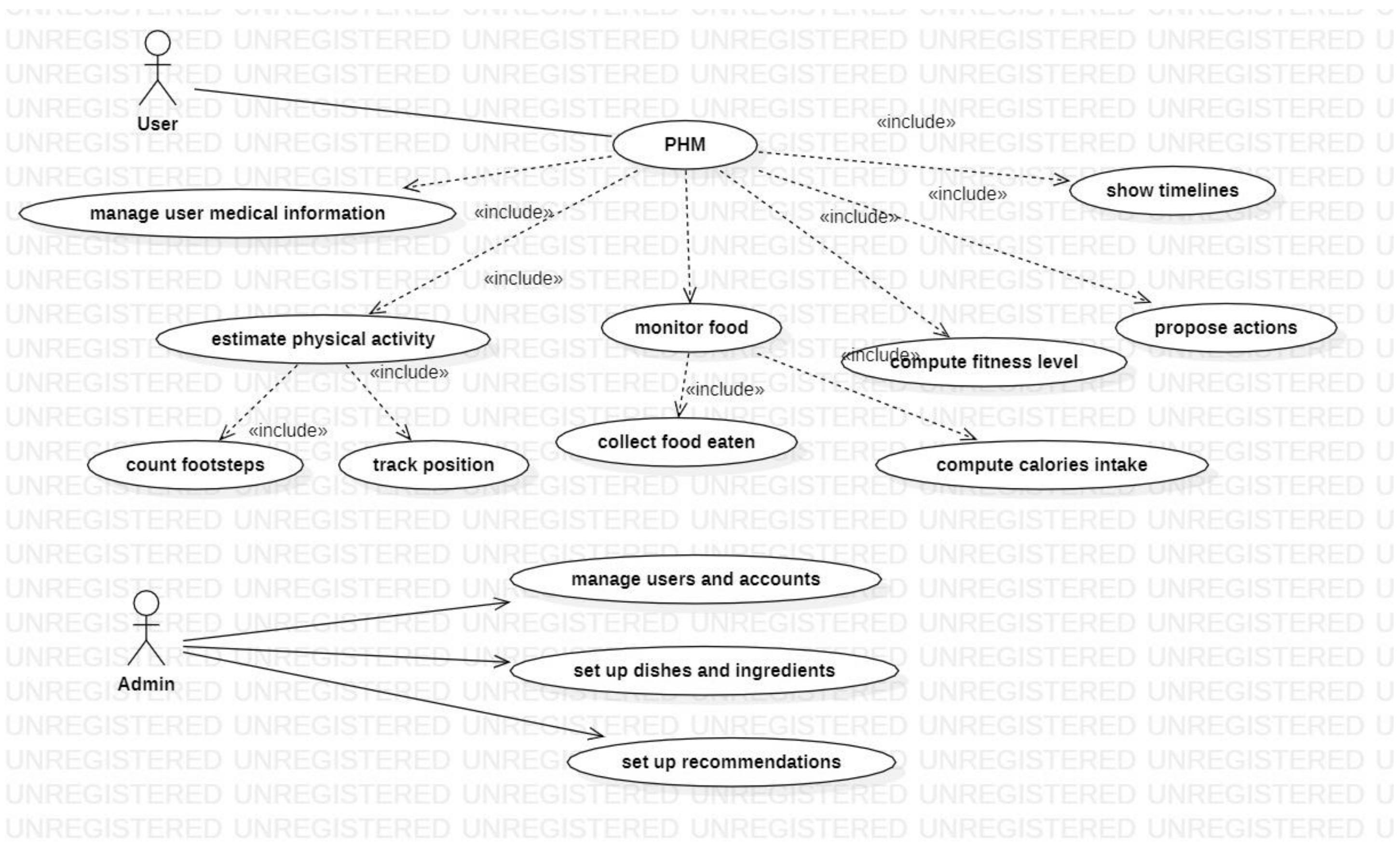


1-b Define the **glossary** (key concepts and their relationships) (UML class diagram)

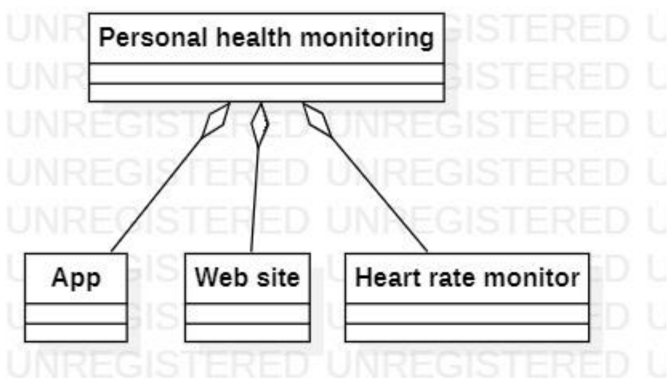


The timestamp of all measures is essential to do timelines and all kind of statistical analyses. Weight is not a property of User because it changes over time (and , again, needs a timestamp). About food, recognize the descriptor part (Dish and Ingredient) that is inserted by the admin (aka Product type), and the user specific part (Feeding action) inserted by the user (aka Product). Dish and ingredient must be connected (ex a dish of carbonara is made of 100g wheat, 50g cheese etc). Recommendation is needed to store knowledge (in term of 'if condition then suggestion') about recommended actions.

1-c Draw the Use Case Diagram system. For each Use Case give self-explainable long names, or a short textual description

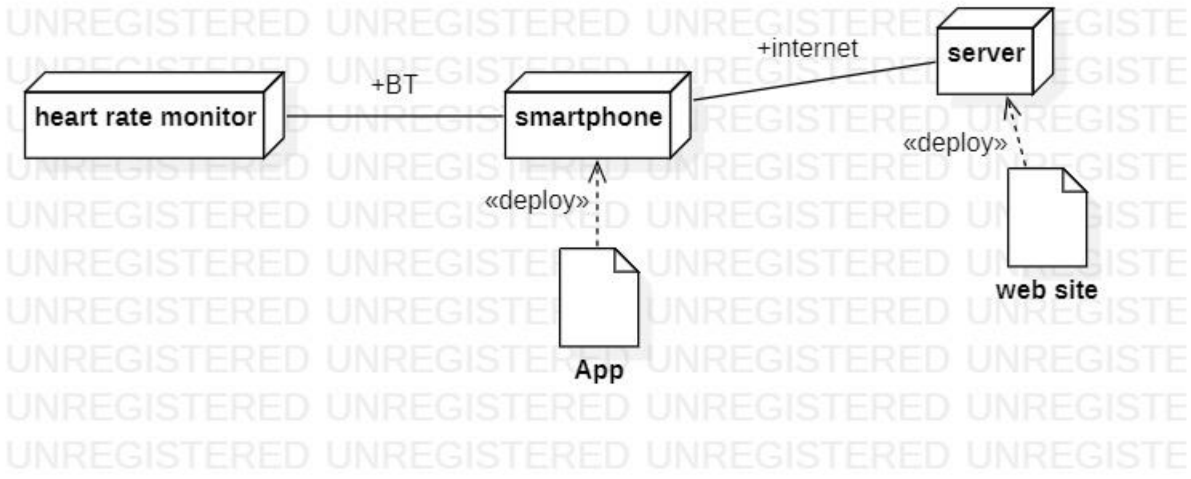


1-d Draw the system design



GPS, accelerometer, smartphone are not part of the system (they can appear on the deployment diagram if needed)

1-e Draw the deployment diagram



1-e Non Functional requirement

2-Black box

Define black box tests for the following function, using equivalence classes and boundary conditions.

Int BMI(int sex, int height, int weight)

The function receives the sex of a person (1 man, 2 female), the height in centimeters (170 = 1meter 70 centimers), the weight in kilograms (80 = 80 kilograms). It returns

0 if the input is not acceptable
1 if the BMI is normal
2 if the BMI is below threshold
3 if the BMI is above threshold

The BMI is = weight / (height * height) and the normal ranges are 18-25 for men, 17-24 for women.

sex	height	weight	BMI	Valid / invalid	Test case
[minint, 0]	-	-	-	I	T(-10, 20,20) → 0
[3 maxint]	-	-	-	I	T(11, 20,20) → 0
-	[minint, 0]	-	-	I	T(1, -10, 20) → 0
-	[300, maxint]	-	-	I	T(1, 400, 20) → 0
-	-	[minint 0]	-	I	T(1, 100, -20) → 0
-	-	[500, maxint]	-	I	T(1, 100, 2000) → 0
1	[1, 299]	[1,499]	Normal	V	T(1, 180, 75) → 1
			Below	V	T(1, 180, 50) → 2
			above	V	T(1, 180, 300) → 3
2	[1, 299]	[1,499]	Normal	V	T(2, 180, 75) → 1
			Below	V	T(2, 180, 50) → 2
			above	V	T(2, 180, 300) → 3

Boundaries to be added