

Final Year Project 2020  
**COVID-19 contact tracing**

**Subject:** Software Requirement Analysis

**Subject code:** SCS254

**Project ID:** PM-446

**Under the supervision of:** Dr. Hanan Moussa

Dr. Iman Helal

TA. Samar Taha

ID	Name	Group
20186031	Nada Mohamed	3
20186007	Ayat Hany	
20186008	Sarah Khaled	
20186043	Mark Rofaeel	

(1) Describe the main functionalities of the application:

**1. Functional requirements:**

- 1.1 User shall **sign in** and **sign up** to the application.
- 1.2 **Get** a quick **snapshot** of the official status within country.
- 1.3 User's **location** shall be **traced** 24/7 by GPS to the application's database.
- 1.4 User shall **report** when having coronavirus symptoms.
- 1.5 User shall **answer** questions when reporting symptoms before requesting home test.
- 1.6 User shall **request** home test if questions' answers were related to many symptoms.
- 1.7 Close user will be **notified** immediately when contacted with an infected individual to self-isolate.
- 1.8 Close user will **have** an **automated test request** when contacted with an infected individual.
- 1.9 Nearby user will be **notified** and advised to social distancing and self-quarantine.
- 1.10 User shall **scan** QR code(that is displayed in places where there is no signal to use GPS) to know the location.

**2. Non-functional requirements:**

- 2.1 Privacy.
- 2.2 Space.
- 2.3 Security.
- 2.4 Safety.
- 2.5 Availability.

**3. Business requirements:**

The process of identifying persons ("contacts") who may have been in contact with an infected individual.

**4. Business rule:**

Data should be limited to COVID-19 purposes.

**5. External interface:**

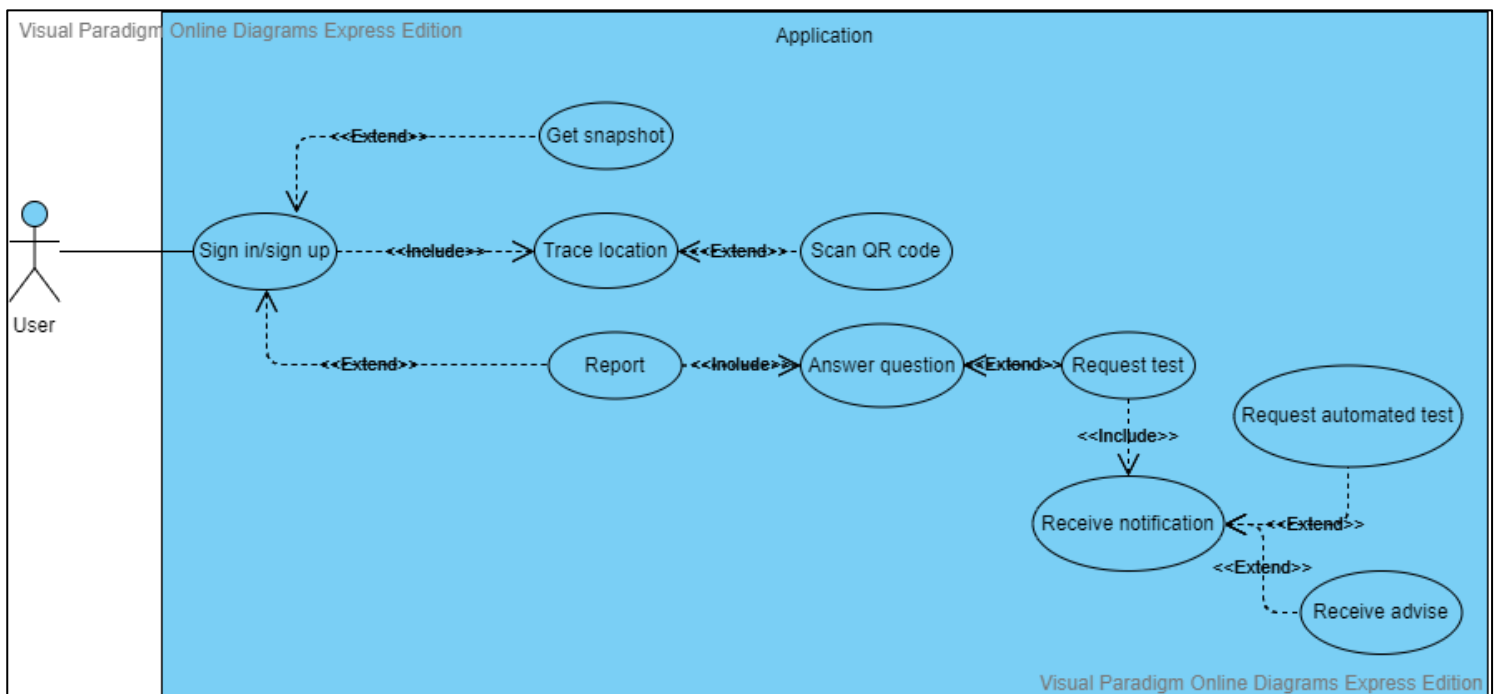
User's hardware(mobile phones) that support the application.

**6. System requirements:**

System should provide high availability features as user's location shall be traced 24/7.

(2.1) Provide the main use cases for the functional requirements of the application:

- 2.1.1 Sign in/sign up.
- 2.1.2 Get snapshot.
- 2.1.3 Trace location.
- 2.1.4 Report.
- 2.1.5 Answer question.
- 2.1.6 Request test.
- 2.1.7 Receive notification.
- 2.1.8 Request automated test.
- 2.1.9 Receive advise.
- 2.1.10 Scan QR code.



Use Case ID:	2.1.1	
Use Case Name:	Sign in/sign up	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection in his device.	
Post-conditions:	The user has now his own account on the application, that will save his data to the application's database.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows two different options either "sign in" or "sign up".
	3- User may select "sign in" or may select "sign up".	
		4- Application lets the user fill in his e-mail and password and then verify this data.
	5- User has his own account now and can use the different functionalities of the application.	
		6- Application will save the user's data to the application's database and will start tracing the user's locations.
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User may select "sign in" or may select "sign up".	
		2- Application lets user fill in his e-mail and password and then find that this data is invalid, so it rejects the user's request.
Includes:	Trace location	
Notes and Issues:	E-mail address should be valid, and password should contain at least 8 characters.	

Use Case ID:	2.1.2	
Use Case Name:	Get snapshot	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection in his device.	
Post-conditions:	The user has now his own account on the application, and he will get a quick snapshot of the official status within country.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows the option of "Get snapshot".
	3- User selects "Get snapshot".	
		4- Application lets the user view a snapshot of total cases, number of new cases, total deaths, number of new death cases and total recovered.
Notes and Issues:	The snapshot is updated every 24 hours.	

Use Case ID:	2.1.3	
Use Case Name:	Trace location	
Actors:	User	
Pre-conditions:	User needs to enable the device's GPS.	
Post-conditions:	The user has now his own account on the application, that will trace his continuous changing locations automatically.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows two different options either "sign in" or "sign up".
	3- User may select "sign in" or may select "sign up".	
		4- Application lets the user fill in his e-mail and password and then verify this data.
	5- User has his own account now and can use the different functionalities of the application.	
		6- Application will save the user's data to the application's database and will start tracing the user's locations.
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User may not be enabling the GPS.	
		2- Application, then, cannot trace the user's location that will eventually help the pandemic to spread more. So, the user shall scan the QR code displayed in the places with no signal.
Notes and Issues:	If there is no signal to use the GPS, user shall scan QR code.	

Use Case ID:	2.1.4	
Use Case Name:	Report	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection and enabled GPS in his device.	
Post-conditions:	The user has now his own account on the application, that will let him report if he felt any kind of symptoms one day.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows the option "Report".
	3- User selects "Report".	
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User may not be enabling the GPS while reporting.	
		2- Application, then, cannot trace the user's location that will not let the user report himself.
Include:	Answer questions	
Notes and Issues:	User shall answer some questions, to officially report him as an infected individual.	

Use Case ID:	2.1.5	
Use Case Name:	Answer question	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection.	
Post-conditions:	The user has now his own account on the application, that will let him answer some questions to make sure it is of coronavirus.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows the option "Report".
	3- User selects "Report".	
		4- Application sends the user some questions to answer.
	5- User shall answer those questions.	
		6- Application will decide if the current symptoms are coronavirus's symptoms not an ordinary fever's symptoms based on some stored data.
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User answers those questions and then it appears that it is not coronavirus.	
		2- Application lets the user know and then no report will be made to that user.
Notes and Issues:	User shall answer those questions honestly without any misleading data.	



Use Case ID:	2.1.6	
Use Case Name:	Request test	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection and enabled GPS in his device.	
Post-conditions:	The user has now his own account on the application, that will let him request test if he answered the questions and it appeared that it is coronavirus.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User shall answer some questions.	
		2- Application decides that the symptoms are coronavirus's symptoms not an ordinary fever's symptoms based on some stored data.
	4- User, now, can select "Request test".	
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User may not be enabling the GPS while requesting test.	
		2- Application, then, cannot trace the user's location that will not let the user request test to his location.
Includes:	Receive notification	

Use Case ID:	2.1.7	
Use Case Name:	Receive notification	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection and enabled GPS in his device.	
Post-conditions:	The user has now his own account on the application, that will let him receive notifications from the application	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
		1- Application sends notification to the user.
	2- Close users receives a notification if they were close to an infected individual, an automated test request will be automatically sent to them and to self-isolate themselves for 14 days.	
	3- Nearby users receives a notification if they were nearby an infected individual will also be notified to take precautions to social distancing and self-quarantine themselves.	

Use Case ID:	2.1.8	
Use Case Name:	Request automated test	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection and enabled GPS in his device.	
Post-conditions:	The user has now his own account on the application, that will let him receive notifications from the application	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
		1- Application sends notification to the user.
	2- Close users receives a notification if they were close to an infected individual, an automated test request will be automatically sent to them and to self-isolate themselves for 14 days.	

Use Case ID:	2.1.9	
Use Case Name:	Receive advice	
Actors:	User	
Pre-conditions:	User needs to have an active internet connection and enabled GPS in his device.	
Post-conditions:	The user has now his own account on the application, that will let him receive notifications from the application	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
		1- Application sends notification to the user.
	2- Nearby users receives a notification if they were nearby an infected individual will also be notified to take precautions to social distancing and self-quarantine themselves.	

Use Case ID:	2.1.10	
Use Case Name:	Scan QR code	
Actors:	User	
Pre-conditions:	User needs to enable the device's camera to scan the QR code.	
Post-conditions:	The user has now his own account on the application, that will let him scan QR code when there is no signal to use the GPS.	
Flow of events:	<b>User Action</b>	<b>Application Action</b>
	1- User opens the application.	
		2- Application shows the option "Scan QR code" if there is no signal to use the GPS.
	3- User selects "Scan QR code"	
		4- Application will save the user's location to the application's database instead of using the device's GPS.
Exceptions:	<b>User Action</b>	<b>Application Action</b>
	1- User may not be have low quality camera.	
		2- Application, then, cannot trace the user's location that will eventually help the pandemic to spread more.

(2.2) The sequencing and dependency between the use cases:

- First, user shall **sign in/sign up** to start saving his/her data to the database of the application.
- When opening the application for the first time, user will **get** a quick **snapshot** of the official status in the user's country like total cases, number of new cases, total deaths, number of new death cases and total recovered.
- User's **location** is being **traced** 24/7 when going to different locations such as home, work and transportation method using GPS, then, it is being saved to the application's database. If there is no network or location signal or high-traffic public amenities, user shall **scan QR code** displayed in those locations.
- If any user felt any kind of symptoms one day, user shall immediately **report** such symptoms. After reporting, **questions** shall be sent to that user to **answer**, to make sure that the current symptoms are coronavirus's symptoms not an ordinary fever's symptoms.
- If it was related to coronavirus symptoms, user can then **request home test** to confirm the case.
- Users who were close to the infected individual's locations will be **notified** instantly to self-isolate themselves for 14 days and, an **automated test request** will be automatically send to them.
- Nearby users to the infected individual's locations will also be **notified** to take precautions to social distancing and self-quarantine themselves.

(3) Show how the provided use cases fulfill the main IEEE quality characteristics of requirements:

- 3.1 **Relevance:** requirements are in scope as it is limited to COVID-19 purposes.
- 3.2 **Currency:** requirements are up to date.
- 3.3 **Traceability:** there is dependency between business requirements and system requirement.
- 3.4 **Clarity:** requirements are clear and is not ambiguous as the sentences are short, direct, we used the keyword "shall" and the active voice.
- 3.5 **Mandatory:** requirements are prioritized due to our prioritization table below.

(4) How will you use the below techniques for the requirements elicitation? You will have to list the stakeholders (internal and external) you will involve for each elicitation technique: Ethnography, Requirements workshop, and Interviews (provide the questions you will ask to each stakeholder):

The stakeholders	Elicitation technique	Why?	Questions
World Health Organization [WHO] (External)	Interviews via video calls	<ul style="list-style-type: none"> <li>We are far away from the WHO.</li> <li>We need to know the real information about virus from the official representative.</li> </ul>	1. What are the coronavirus symptoms? 2. What are the precautions that can people take? 3. How can people self-quarantine themselves?
User (External)	Online questionnaires	<ul style="list-style-type: none"> <li>There are many users to serve.</li> <li>We do not want to hold any meetings to help prevent spreading the virus.</li> </ul>	-
Team (Internal)	1. Brainstorming 2. Ethnography 3. Prototype	<ul style="list-style-type: none"> <li>We will use Brainstorming and Ethnography to know the requirements and how to implement those into functionalities in the application.</li> <li>We will use prototype to know the interface/framework of the application</li> </ul>	-
Lawyer (Internal)	Interviews via video calls	<ul style="list-style-type: none"> <li>We do not want to hold any meetings to help prevent spreading the virus.</li> <li>We need to know the rules and regulations to penalize the criminal.</li> </ul>	1. How can we penalize such criminal who stole the data of the users? 2. What are the official documents, rules, and regulations to authenticate such application with the WHO and with the country the application downloaded in?

(5) Provide use cases for the non-functional requirements: Describe how will they reflect on the functional requirements (perhaps also explaining the additions to the use cases you provided in questions #1):

- 5.1 **Privacy:** privacy concerns shall be raised, especially when tracking the geographical location of app users. This non-functional requirement shall be taken into consideration within the user's personal data while signing-in or signing-up.
- 5.2 **Space:** the application's database space shall be required as thousands of user's data and their continuous changing locations is stored on it.
- 5.3 **Security:** data security and anonymity shall be protected and shown to be protected based on evidence, any sharing of data with third parties(Android or IOS) would have to be defined in law, and there shall be safeguards against abuse and the rights of citizens to respond to abuses.
- 5.4 **Compliance:** our requirements had fulfilled the main IEEE quality characteristics.
- 5.5 **Safety:** the application shall be used to help prevent such pandemic to reach more and more people and keep everyone safe. Whenever a user report himself as an infected individual, a test is requested for him, an automated test is requested for those who were close to him, and advices is sent to those who were nearby him.
- 5.6 **Availability:** the user's location is being traced 24/7 when going to different locations so the system shall be available all the time.

(6) Provide a prioritization table based on the below criteria: dependency (based on your answer in question #2), urgency, and complexity of implementation. Assign different weights to each criterion as per your judgement:

**\*Note: the requirements mention below are the same as question number 2.1:**

Requirements	Criteria of prioritization, weight and total									
	Dependency	Weight	Total	Urgency	Weight	Total	Complexity	Weight	Total	Grand total
2.1.1	5	5	25	1	3	3	5	4	20	48
2.1.2	3	5	15	1	3	3	2	4	8	26
2.1.3	5	5	25	5	3	15	5	4	20	60
2.1.4	5	5	25	4	3	12	4	4	16	53
2.1.5	3	5	15	3	3	9	1	4	4	28
2.1.6	4	5	20	4	3	12	2	4	8	40
2.1.7	3	5	15	3	3	9	1	4	4	28
2.1.8	2	5	10	4	3	12	2	4	8	30
2.1.9	1	5	5	2	3	6	1	4	4	15
2.1.10	5	5	15	5	3	15	4	4	16	46

This means the prioritization of our requirements will be:

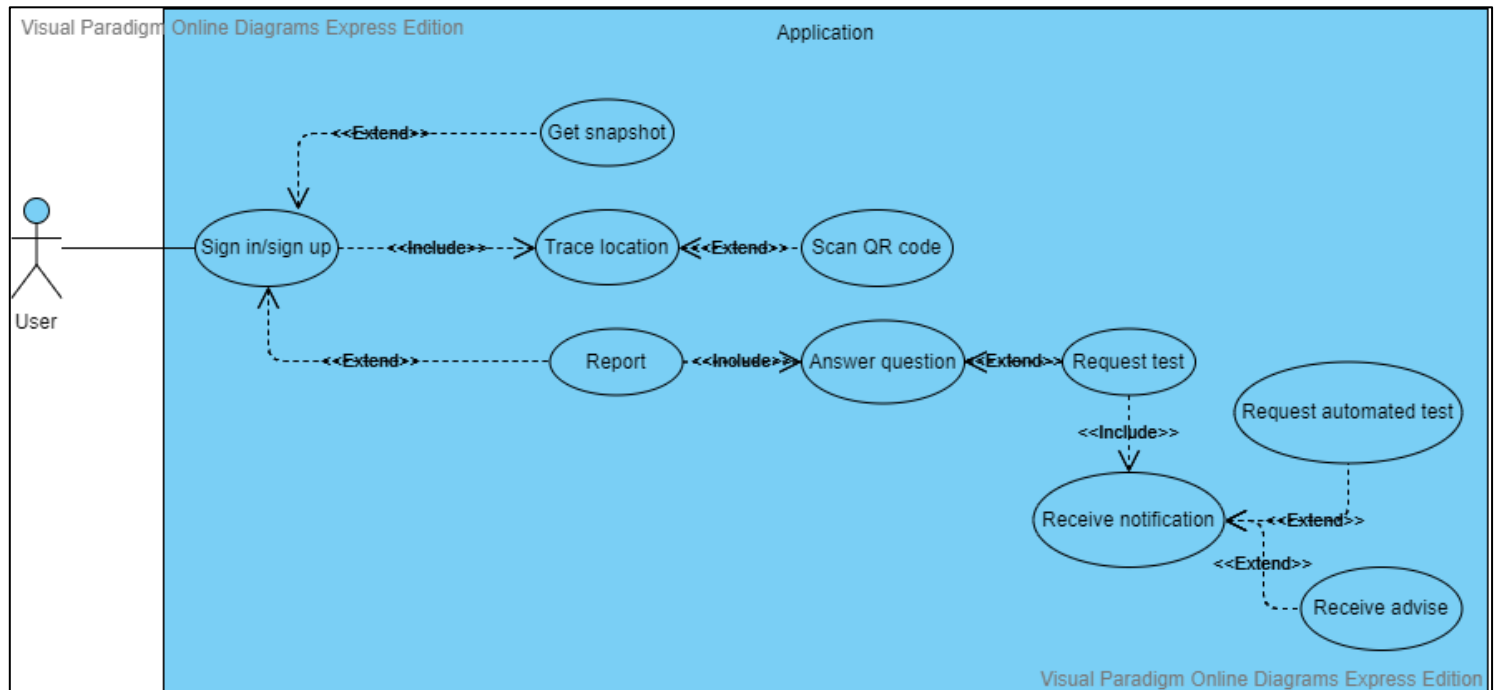
1. Trace location
2. Report
3. Sign in/sign up
4. Scan QR code
5. Request test
6. Request automated test
7. Answer question
8. Receive notification
9. Get snapshot
10. Receive advise



(7) Label these requirements using one of the specified techniques, justify your selection:

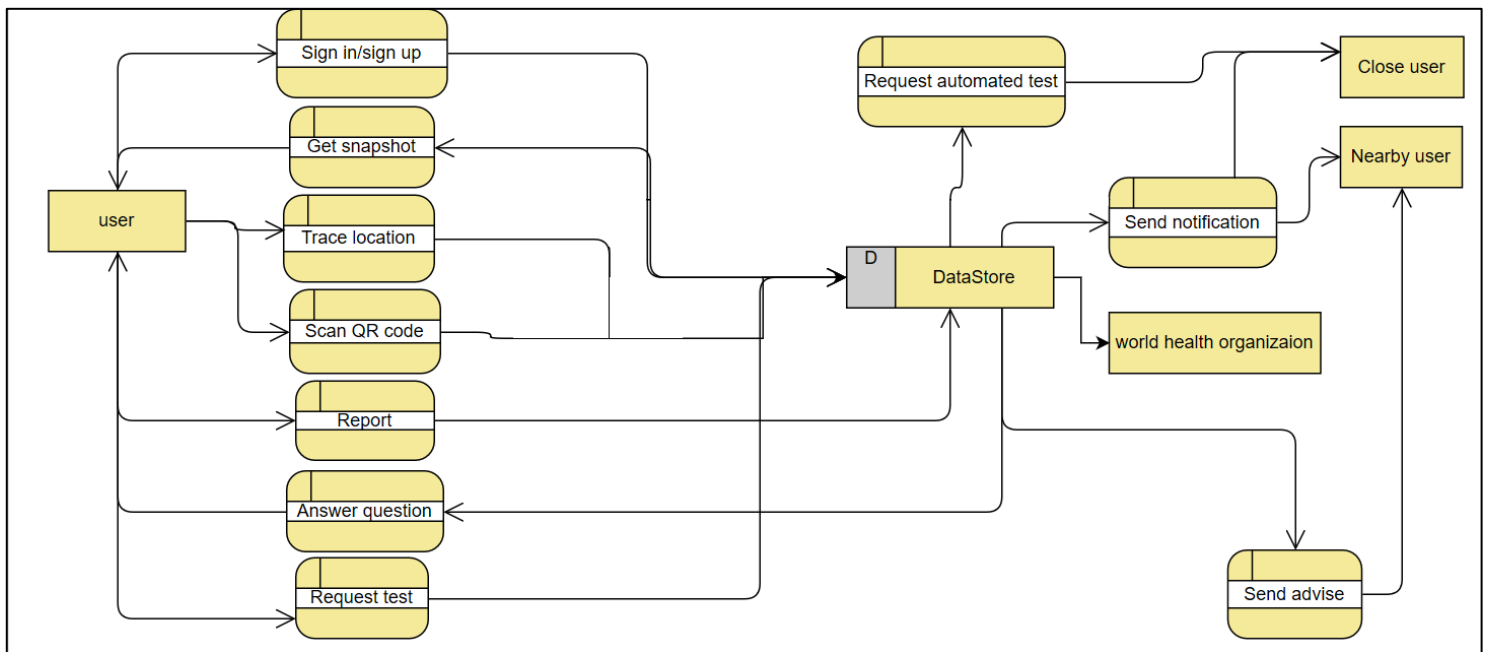
We used the **hierarchical numbering** when labeling our requirements, as it is one of the most used methods when listing requirements. We used a number in each section, and every requirement under that section begin with that number and then more digits indicate a more detailed requirement, so you know that this requirement is under that section.

(8) Draw use case model:



(9) Draw data flow diagram (DFD):

**\*Note: the story mention below is the same as question number 2.2:**



(10) Draw activity diagram:

**\*Note: the story mention below is the same as question number 2.2:**

