



Course Code and Title	SCS254 – Software Requirements Analysis
Instructors	Dr. Hanan Moussa and Dr. Iman Helal
Final Assessment Type	Project Report
General Instructions	1- Choose only ONE project 2- You can do the project individually or choose your team (max. 5 students)

First Project: Software application for COVID-19 contact tracing

Several COVID-19 mobile software applications are designed to aid contact tracing in response to the 2019-20 coronavirus pandemic, i.e. the process of identifying persons ("contacts") who may have been in contact with an infected individual.

Numerous applications were developed or proposed, with official government support in some territories and jurisdictions. Several frameworks for building contact tracing apps have been developed. Privacy concerns have been raised, especially about systems that are based on tracking the geographical location of app users. The organizations declared some conditions on governmental projects such as: the use of data would have to be limited to COVID-19 purposes, data security and anonymity would have to be protected and shown to be protected based on evidence, any sharing of data with third parties would have to be defined in law, and there would have to be safeguards against abuse and the rights of citizens to respond to abuses. Less intrusive alternatives include the use of Bluetooth signals to log a user's proximity to other cellphones. As an example of such applications: Google and Apple jointly announced that they would integrate functionality to support such Bluetooth-based apps directly into their Android and iOS operating systems <https://www.ranzware.com/2020/04/22/how-apple-and-google-will-fight-the-spread-of-coronavirus-cnet/>

As a requirements engineer for the proposed software application:

[40 Marks]

- (1) Describe the main functionalities of the application.
- (2) Provide the main use cases for the functional requirements of the application. You should provide at least 10 use cases. Explain the sequencing and dependency between the 10 use cases.
- (3) Show how the provided use cases fulfill the main IEEE quality characteristics of requirements.
- (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).
- (5) Provide use cases for the non-functional requirements: **Security and Compliance**. Describe how will they reflect on the functional requirements (perhaps also explaining the additions to the use cases you provided in questions #1).
- (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.

Based on the requirements extracted from questions (2), (5):

[40 Marks]

- (7) Label these requirements using one of the specified techniques, justify your selection.
- (8) Draw use case model.
- (9) Draw data flow diagram (DFD).
- (10) Draw activity diagram.

Notice: All models should be drawn using one the CASE tools discussed on labs (e.g. Visual Paradigm <https://www.visual-paradigm.com/>).

Second Project: Software application for a modern car

As the automotive industry is transitioning from hardware- to software-defined vehicles, the relevance of software for core technology trends is increasing rapidly. The competitive difference between different car models is today in the software. It is no longer about the car itself but it's about possibilities like remotely



opening and closing doors or start the engine with an app. The big automakers (car manufacturers) get this. They are embracing smartphones and developing apps that allow drivers to do everything from lock their doors to program exactly when their car starts. Instead of being islands in a sea of traffic, cars will communicate with each other and with the road. Automakers realize this and are scrambling to develop unique systems that offer greater utility and convenience than their competitors.

Google, IBM and Cisco also are moving into the automotive industry. This is having a huge impact on every traditional area, because all the functionalities you might have in electronics or mechanical (systems) are being shifted to software. Tesla is also a pioneer in this area <https://www.tesla.com/blog/introducing-software-version-10-0>. Some electric and plug-in hybrids have smartphone apps that allow owners to monitor how much energy they have and manage when and how their cars begin drawing power. Toyota believes effective communication between utilities and automobiles is one way to ensure fast and efficient charging. It sees a day when our cars, our homes and our phones communicate with each other, with charging stations and with the grid.

There is also a social aspect of driving, what if your car could automatically inform you where your favorite shop is, when arriving in a new city? Or connect to your social networking system and tell you where your friends are hanging out and provide driving directions to get there? Increasing functionality means increasing complexity, a challenge for automakers. Automakers have to ensure the software is 100 percent reliable. They must also ensure 100 percent security so your car can't be hacked. Researchers have identified a handful of ways to break into a car, including through the audio system. This security issue requires designing systems robust enough to prevent our cars from being stolen with a laptop. Modern cars already feature hundreds of sensors and more are coming to make our cars, and our roads, smarter. We're seeing the development of autonomous cars, predictive, personalized traffic forecasts and even roads that analyze traffic data and share that information with drivers.

Questions: As a requirements engineer for the proposed software application [40 Marks]

- (1) Propose an advanced car application (following some of the abovementioned trends) and describe its main functionalities.
- (2) Provide the main use cases for the functional requirements of the application. You should provide at least 10 use cases. Explain the sequencing and dependency between the 10 use cases.
- (3) Show how the provided use cases fulfill the main IEEE quality characteristics of requirements.
- (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: **System interface analysis, Prototyping, and Studying of existing systems.**
- (5) Provide use cases for the non-functional requirements: **Security and Reliability.** Describe how will they reflect on the functional requirements (perhaps also explaining the additions to the use cases you provided in questions #1).
- (6) Provide a prioritization table based on the criteria: **Risk, Value, Cost, and Complexity of implementation.** Assign different weights to each criterion as per your judgement.

Based on the requirements extracted from questions (2), (5): [40 Marks]

- (7) Label these requirements using one of the specified techniques, justify your selection.
- (8) Draw use case model.
- (9) Draw data flow diagram (DFD).
- (10) Draw activity diagram.

Notice: All models should be drawn using one the CASE tools discussed on labs (e.g. Visual Paradigm <https://www.visual-paradigm.com>).