CSC436 Project

Conduct Requirements Engineering

- a rear-view camera with cross-traffic detection, rear collision avoidance, and smart trunk hatch;
  - Rear-view camera/monitor *page 3.128-136*
  - Rear cross-traffic collision warning/avoidance assist (RCCW/RCCA) *page5.82-89*
  - Smart trunk hatch *page 3.59-62*

Milestone 1:

System Scope, Natural Language Requirements, and Hazard Analyses for Automotive Driver Assistance Systems

Task 1

Research the Top Secret Document1 , aka “Owner’s Manual”.
Find out everything you can about:
• the blind-spot detection system with “safe-exit” feature;
• the rear-view camera with cross-traffic detection, rear collision avoidance, and smart trunk hatch;
• the smart cruise control with forward collision avoidance; and
• the lane assist system with driver attention warning.
Note that not all information might be in the same place. The manual might treat the “safe exit” feature as a driver comfort system and the blind-spot detection feature as a safety system, yet realistically, they are the same system that makes use of the same sensors. So, relevant information might be spread out across the document. Decide which system your team wants to do

Task 2

Find the IEEE 830 template on Blackboard.
Read it and the instructions therein. Then, document your findings from Task 1, 3, 4, and 5 in the appropriate sections. Be advised that IEEE 830 is merely a guideline to structure your requirements specification.

Task 3

Educate yourself which road traffic rules may be relevant for your system. Research how they work in different countries or states (e.g., US vs. Germany; New York vs. Texas). Find out about laws, procedures, or whatever else you feel is relevant. For example, the lane assist system will need to know how lanes are marked. Document all information you find relevant.

Task 4

Now that you have done the preparatory ground-work, document the natural-language requirements for your system. Do this generically, not specifically with regard to the Hyundai Santa Fe example. Specifically, you should have:
• goals, scenarios, and solution-oriented requirements

• functional and quality requirements, and constraints
• context aspects for each context facet

For each of the above, document the same properties as outlined in Homework 1 (i.e., requirement type, artifact type, quality property, etc.). Relate requirements to one another by "breaking them down," as outlined in class. Furthermore, be sure your requirements are of high quality themselves, by checking their completeness, consistency with one another, necessity, etc. Avoid ambiguity and transformational effects.

Task 5
Identify the system functions and conduct a Functional Hazard Analysis. Add the FHA results to your requirements specification in one central and reasonable section.

Hints
• To help you with this, it may help to think about the following:
    o What are external systems' expectations?
    o What are external users' and systems' expectations?
• It may also help to use the resolution strategies for "bad" requirements outlined in class.

Milestone 2:

Solution-Neutral and Solution-Oriented Requirements for your Driver Assistance System
Consider your solution for Milestone 1. Amend your solution with your solutions from this milestone and submit one PDF file, containing both.

Reminder
You are a team of Requirements Engineers tasked with documenting the Requirements Specification of a driver assistance system to be introduced into your company Blundai's new flagship
SUV. Through market research, you obtained an Owner's Manual Top Secret Document of your competitor's equivalent car, upon which you are basing your requirements.

Task 0
Find your IEEE830-compliant Requirements Specification from Milestone 1 and revise it according to the feedback you received, as necessary.

Task 1
Find your solution from Milestone 1 and create a KAOS goal diagram depicting your goals. Be
sure to correctly model obstacles, goals, AND- and OR-refinements, positive and negative contribution links, as well as conflicts. One KAOS goal diagram will probably be enough, but you

may model more, if you like. Refer to the Top Secret Document when you need to look up information. Undoubtedly, you will find that it's necessary to enhance your requirements, so feel free
to add, remove, or rectify as necessary the requirements that are already in your specification document.
Add the diagram to a suitable section to your IEEE830 Requirements Specification, with some explanation what can be seen here (one sentence is sufficient).

Task 2
For each goal from your KAOS goal diagram, create a UML sequence diagram depicting an exemplary fulfillment of that goal. You don't need to create exactly one sequence diagram for each
goal – you may combine multiple goals into a single diagram. Be sure to consider main scenarios and alternative ways to fulfill the same goal. It is VERY likely that there are multiple ways
for the driver to achieve the same goal.
Also consider model exception scenarios for certain things that could go wrong. Your Hazard Analyses from Milestone 1 will help you to find your exception scenarios.

Task 3
If you already defined some scenarios in your solution of Milestone 1, find the goal in your goal diagram they are associated with. Document this dependency somehow, e.g., through a sentence somewhere near the goal and scenario.
If you cannot find a suitable goal, this is an indication that you're missing some goals. In this case, revise your natural language goals in Milestone 1 as well as your KAOS diagram from Task 2 accordingly.
Now check that all goals in your KAOS goal have scenarios associated with them. If you have a goal without a scenario, this is an indication that
a) the goal is superfluous or
b) you're missing the scenario.
Choose to remove the goal or add the scenario.

Task 4
Now take another look at your solution-neutral requirements from above. Are all diagrams you created across Tasks 1-3 consistent? Fix possible inconsistencies. If you are uncertain how to model something, refer to your context from Milestone 1 to see if some context entity (i.e. a human, a document, etc.) can help you elicit the right information. Document your additional findings using UML notes with natural English.

Milestone 3:

Solution-Oriented Requirements, FMEA, & Safety Argumentation
Find your solution for Milestone 1 & 2. Append your solution of this milestone to your solutions from the previous milestones and submit one PDF file, containing all requirements, revised diagrams, and analyses.

Reminder
You are a team of Requirements Engineers tasked with documenting the Requirements Specification of a driver assistance system to be introduced into your company Blundai's new flagship
SUV. Through market research, you obtained an Owner's Manual Top Secret Document of your competitor's equivalent car, upon which you are basing your requirements.

Task 0
Find your IEEE830-compliant Requirements Specification from Milestone 1&2 and revise it according to the feedback you received, as necessary. In particular, make sure that diagrams are
consistent to one another.

Task 1
Given your findings from the previous milestones, document the solution-oriented requirements for your system. The purpose is to take your exemplary fulfillments of high-level goals with scenarios and refine them into system structure, functions, and externally observable states that make up your system. Specifically, you should have:
• at least one UML class diagram describing your system's data in the static-structural perspective
• at least one UML class diagram describing the architecture in the static-structural perspective (hardware and software, but focus on the software, and don't forget the interface specification!)
• at least one UML activity diagram describing the functional perspective
• at least one UML state machine diagram describing the behavioral perspective

Task 2
Now take another look at your solution-neutral requirements from Milestone 1. Are all diagrams consistent to each other? Fix possible inconsistencies. If you are uncertain how to model something, refer to your context considerations from Milestone 1 to see if some context entity (i.e. a
human, a document, etc.) can help you elicit the right information. Document your additional findings using UML notes with natural English.

Task 3
Reconsider your solution-oriented requirements, specifically the operations of each class and specified activities. Conduct a Failure Mode and Effects Analysis (FMEA) for each method in the codebase relevant to your project. Document your findings in the FMEA template provided on Blackboard and add it to your specification.

Task 5
Now create a Safety Case using the Goal Structuring Notation. You will not need to create a confidence case, but you may include it anyways, if you want). Be sure to argue by means of the
Hazards identified during FHA in Milestone 1 and Failure Modes identified during FMEA in Task

3. Include the hazard-mitigating requirements you added/augmented as evidence.
Add your GSN Safety Case diagrams to your Requirements Specification into an appropriate section.

Task 6

At the very end of your IEEE810-compliant requirements specification, include an appendix containing a team reflection essay on your project. Roughly one page in length (i.e., less than 750words / 10 sentences or so), this essay should detail from your very own perspective as a team
about the requirements engineering process and how it informed your implementation.
Don’t write about team dynamics, i.e., that you acknowledge the detriment of procrastination or the need for better teamwork, unless it pertains to how you engineered your requirements.