GEORGE MASON UNIVERSITY
Department of Computer Science
Volgenau School of Engineering

SWE 760 - **Software Analysis and Design of Real-Time Systems**

Fall 2016 Tuesday 4:30-7:10 PM Location: IN 208

Term Project

The goal of the term project is to analyze and design a real-time embedded system for an autonomous car, which is equipped with smart sensors (for sensing its environment) and actuators (for controlling the environment).

There is much written about autonomous cars, such as the articles below:

https://en.wikipedia.org/wiki/Autonomous_car

http://robohub.org/how-do-self-driving-cars-work/

http://www.mouser.com/applications/autonomous-car-sensors-drive-performance/

https://www.google.com/selfdrivingcar/

Note that the Wikipedia article describes different levels of automated vehicle classification. The degree of automation should be at level 3 of the National Highway Traffic Safety Administration (NHTSA) classification:

Level 3: The driver can fully cede control of all safety-critical functions in certain conditions. The car senses when conditions require the driver to retake control and provides a "sufficiently comfortable transition time" for the driver to do so.

Assignments for this term project are:

Phase 1:

a) Write a high-level requirements document listing the main features of the autonomous car and the scope of the real-time software control system.

b) Develop a software system context model depicting the boundary between the software system and the external environment. Identify the external classes on the context diagram and specify the interface between each external class and the software system.

c) Develop a use case model and use case descriptions for the embedded software system.

d) Develop a state machine for the control system, which controls the motion of the autonomous car.

Phase 2:

a) Develop interaction diagrams (one for each use case), using either communication diagrams or sequence diagrams, depicting the sequence of interactions among the objects participating in each use case. Identify the object structuring criteria used.

b) Develop a state machine for each state dependent control object showing all states, events, and actions. Make sure that each state machine is consistent with the appropriate interaction diagram(s), including all alternative paths on the interaction diagrams.

c) Write an overall description of the analysis model, stating all design decisions made and any assumptions you made.

Phase 3:

a) Develop an integrated set of communication diagrams showing all subsystems and objects in the system, as well as all message communication.

b) Develop the software architecture (depicted on a concurrent communication diagram), showing the distributed subsystems. Define the message communication interfaces between the subsystems.

c) Develop a component based software architecture depicting the components and connectors, including ports and interfaces

d) Design the task architecture (depicted on concurrent communication diagrams) showing the concurrent tasks in each subsystem and the interfaces between them. Describe the criteria used for task structuring. Define the message communication interfaces. Design the internal structure for each composite task, depicting the passive objects it contains.

e) Conduct a performance analysis for the real-time design, assuming that the tasks are assigned (a) to a uniprocessor system (b) a multicore processor system.

f) Write an overall description of the design model, stating all design decisions made and any assumptions you made.