Team: sdmay23-15

2 Project Plan

2.1 Project Management/Tracking Procedures

Which of agile, waterfall or waterfall+agile project management styles are you adopting? Justify it with respect to the project goals.

Agile will be the primary means to our group's software development. Possessing a high-level of what is needed to be accomplished will allow several sprints to be defined. Breaking these sprints into user-stories will allow individual team members to take responsibility for specific tasks. Delegating these prompts will ensure efficiency and enable members to better understand the full scope of the project. The waterfall approach, although sequential and straightforward, is not flexible for our anticipated modifications. Using Agile will permit our client to be closely involved with the process - a characteristic not supported by the waterfall approach.

What will your group use to track progress throughout the course of this and the next semester. This could include Git, Github, Trello, Slack, or any other tools helpful in project management.

- GitHub - for scripts/codes
- Google Drive - for research papers and professional documentation
- Discord - for communication
- Asana - track agile progress

2.2 Task Decomposition

In order to solve the problem at hand, it helps to decompose it into multiple tasks and subtasks and to understand interdependence among tasks. This step might be useful even if you adopt agile methodology. If you are agile, you can also provide a linear progression of completed requirements aligned with your sprints for the entire project.

- Researching concepts and standards for CAN communication
- Setting up Virtual Machine environments in virtualBox to run CAN simulation.
- Determine what key management standard to use for the project.
- Implement and test the standard and determine if it'll work.
- Liaising and meeting with Advisor and Clients.
- Setting up a simulation environment for testing and implementation of concepts on CAN communication.

2.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

What are some key milestones in your proposed project? It may be helpful to develop these milestones for each task and subtask from 2.2. How do you measure progress on a given task? These metrics, preferably quantifiable, should be developed for each task. The milestones should be stated in terms of these metrics: Machine learning algorithm XYZ will classify with 80% accuracy; the pattern recognition logic on FPGA will recognize a pattern every 1 ms (at 1K patterns/sec throughput). ML accuracy target might go up to 90% from 80%.

In an agile development process, these milestones can be refined with successive iterations/sprints (perhaps a subset of your requirements applicable to those sprints).

- Milestones
    - Design J1939 key management protocol
    - Design message security protocol
    - Design ECU Authentication protocol
    - Implement virtual CAN Bus simulation environment
        - Set up Ubuntu Virtual Machine
        - Implement simulated control unit
        - Implement CAN Sniffer
        - Implement simulated instrument cluster
    - Implement key exchange, authentication, and message security protocols on virtual CAN environment
    - Test J1939 key management protocol
    - Test message security protocol
    - Test ECU Authentication protocol
    - Migrate to physical CAN environment
    - Test protocols on physical CAN environment
- Metrics
    - The Gantt Chart and the Asana, agile progress tracking platform are used to measure and keep in track the progress of each tasks. This makes sure that the team is consistent with their work force.
- Evaluation Criteria
    - The whole team reviews each tasks in the weekly meeting, to verify and ensure that we meet our requirements.

2.4 Project Timeline/Schedule

• A realistic, well-planned schedule is an essential component of every well-planned project

• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

• A detailed schedule is needed as a part of the plan:

– Start with a Gantt chart showing the tasks (that you developed in 2.2) and associated subtasks versus the proposed project calendar (including both 491 and 492 semesters). The Gantt chart shall be referenced and summarized in the text.

– Annotate the Gantt chart with when each project deliverable will be delivered

• Project schedule/Gantt chart can be adapted to Agile or Waterfall development model. For agile, a sprint schedule with specific technical milestones/requirements/targets will work.

Link to the Gantt chart:

Gantt Chart

Summary of the Gantt chart:

For this semester, there are five major tasks to be completed. We completed the research on the concepts and standards for CAN communication for the first two weeks. We configured a virtual machine using VirtualBox to run CAN simulations in the virtual environment. Currently, we are installing the configured VM on each of our computers. After that, we will take about three weeks to determine the key management standard that should be used for our project. Then, we will start the implementation of the standard and run through some testing till the end of the semester. The task of liaising and meeting with the advisor and client will last throughout the semester. For the next semester, we will fully focus on the implementation and testing of Simulink.

2.5 Risks And Risk Management/Mitigation

Consider for each task what risks exist (certain performance targets may not be met; certain tools may not work as expected) and assign an educated guess of probability for that risk. For any risk factor with a probability exceeding 0.5, develop a risk mitigation plan. Can you eliminate that task and add another task or set of tasks that might cost more? Can you buy something off-the-shelf from the market to achieve that functionality? Can you try an alternative tool, technology, algorithm, or board?

Having an Agile project development structure (like the one we are implementing through Asana) can actually help deal with risks and risk mitigation with each sprint, as the principles of the methodology itself were created in part to help in mitigating risk through the usage of cross-functional teams, continuous improvement/feedback, and good engineering practices in general. For example, instead of overwhelming an entire team by attempting to correct all risks at the end of development, the Agile structure allows us to incrementally reduce risk with each release or version of our product, which in turn mitigates risk in a more gradual sense, allowing for more focus on the risk contained in smaller, individual sections to decrease our own human error.

2.6 Personnel Effort Requirements

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be the projected effort in the total number of person-hours required to perform the task.

Tasks	Members Involved	Hours Needed (Week)
Reseraching concepts and standards for CAN communication	- Aayush Chanda - Baganesra Bhaskaran - Chau Wei Lim - Brian Goode - Michael Roling	**2.0**
Setting up CAN VM on everyone's computer	- Alexander Freiberg	**5.0 (only once)**
Key Management on CAN	- Alexander Freiberg - Aayush Chanda - Baganesra Bhaskaran - Chau Wei Lim - Brian Goode - Michael Roling	**3**
Testing with inputs	- Alexander Freiberg - Aayush Chanda - Baganesra Bhaskaran - Chau Wei Lim - Brian Goode - Michael Roling	**3.0**
Liasing and meeting with Advisor and Clients	- Alexander Freiberg - Aayush Chanda	**2.0**
Setting up simulation environment for testing and implementation of concepts on CAN communication	- Alexander Freiberg - Aayush Chanda - Baganesra Bhaskaran - Chau Wei Lim - Brian Goode - Michael Roling	**4.0**
Version control and professional documentation activities	- Baganesra Bhaskaran - Michael Roling	**1.0**

2.7 Other Resource Requirements

Identify the other resources aside from financial (such as parts and materials) required to complete the project.

- We might need support from the client on knowledge and some level understanding for some of the concepts and implementation (currently the client is providing some presentation on new concepts and ideas)
- Research papers and research inputs from various online sources
- Online tutorials (Youtube, Geeks for Geeks, TutorialsPoint, Code Academy, W3 Schools) to learn how to convert our conceptual understandings into code
- Reach out to our advisor (Dr Zambreno) to fill in the lapses (that we will definitely experience) in our conceptual understanding
- Virtualbox VM with our CAN Sim imported into it so we can test
- Will probably want a physical apparatus to test algorithm on