Dr. Mohan's Senior Project Schedule

Senior project is structured to be a demonstration of your ability to investigate the feasibility of the proposed project, identify the requirements, produce a design, develop and test the final solution. As such, you and your team are expected to manage the project. The senior project instructor is more akin to the role of a project manager and tracks your progress to ensure you can meet the client's expectations on time. I give my teams a fair bit of leeway in how they choose to organize themselves and how they manage the relationship with the client. I do expect my teams to meet certain standards and models of performance. The schedule and expectations are provided below.

Schedule:

	Week	List of Things to be completed
1.	Before first meeting (Week 0)	1. Setup Meeting time with advisor 2. Setup Meeting time with client 3. Setup time for daily standups with team 4. Setup GitLabs repo(see below) for team. Add advisor to Gitlabs repo 5. Setup Trello (see weekly sprints) for task management. Add advisor(srmohan@gmail.com) to Trello.
2.	Week 1 Meeting	1. Progress on problem statement 2. Progress on class diagram/architecture diagram and method for documenting design 3. Progress on project plan 4. Setup any and all tools/VM's that are needed for the project. 5. Identify testing strategy
3.	Week 2 Meeting	1. First version of problem statement 2. First version of design and architecture 3. First version of project plan 4. First sprint backlog including identification of features for first sprint
4	Starting Week 3	1. Commence Weekly sprints 2. Deliverables include all artifacts identified below under weekly sprints.
5	Week 9	1. Senior Project Review by either Dr. Hays or Dr. Hewner 2. Dr. Hays will review 2 teams and Dr. Hewner will review 2 teams. 3. The teams are responsible for signing up for a meeting 4. See section of Senior Project Review for more details.

Source Control:

1. I expect all my teams to use continuous integration (CI) and continuous delivery (CD). Unless instructed by the client otherwise, I expect you to utilize the rosebuild system available at ada.csse.rose-hulman.edu
2. All artifacts including source code, tests, documents generated for the class (these can be maintained as wiki's), meeting minutes and other artifacts identified by the team and the client.
3. The team must setup their GitLabs repo before the first meeting.

Client and Advisor Meetings:

1. Meet weekly with the project advisor during the agreed upon meeting time. Attendance at this meeting is mandatory.
2. Meet weekly with the client during the agreed upon meeting time. Attendance at this meeting is mandatory.
3. The team must send an agenda ahead of the meeting with the project advisor and the client. The agenda at all times will include the features that are being worked upon during the current sprint and the top 4 features for the next sprint.
4. Minutes must be captured for these meetings and posted on the team's GitLab repo.

Weekly Sprints:

1. We will do weekly sprints. Sprints will begin week 3.
2. The sprint backlog will always be kept up-to date
3. At the start of each sprint, I expect to see rough use cases or user interfaces for the features that will be implemented during the sprint. If the feature being implemented does not involve use cases, a paragraph description is required.
4. Responsibility for each shall be clearly identified.
5. The team will hold daily standups in one of the department labs.
6. At the end of each sprint, the team will demonstrate the features to the advisor. Along with the demonstration, an updated copy of the design (class diagram will suffice) should be shown to the advisor. All the use cases must be updated to keep up with the actual behavior of the system. The team will also document the tests that were performed on the implemented code and the current status of bugs if any.
7. The team will also mention any and all feedback received from the client.
8. The team will use Trello to identify and manage all tasks. I am open to other software suggested by the client for task management but would prefer Trello.

Problem Statement

- Should have a problem statement agreed to by your client by midterm.
- Should be updated with changes at end of term.
- Normal format is for the problem statement to be about 2-3 pages, to include a *short list* of key features, quality attributes, and other pertinent info the client agrees to, like how the system relates to or replaces some existing system. The Function Form Economy Time model used in 333 would be a good template.
- **Preliminary Version: 2nd Week Meeting Time**
- **Final Version: 5th Week**

Project Plan

- Description of your Agile processes[1] (e.g., pick a lifecycle, explain how you will use it, and why you expect it to be a good choice)
- Sprint
    - How do you plan to determine features for each sprint?
    - How do you plan to assure quality of code /application at end of every sprint?
    - What is the duration of each sprint?
    - Will you be releasing a working version of the system after each sprint?
- Risks
    - Updated and added to (includes all risks)
    - ID
    - Type (risk/opportunity)
    - Probability
    - Impact
    - Mitigation strategy with revisit date (for all risks)
    - Outcome
    - Resources responsible
- Configuration Management Plan
    - What sort of version control are you planning to use?
    - How are you planning to manage it?
    - What items are going to be under source control?
    - Are you planning to have any hooks for source code?
- Overall completeness
    - Includes structure for subsequent quarters (we will provide our expectations ahead of time, toward this end)
    - Uses good documentation standards
- **Preliminary Version: 2nd Week Meeting Time**
- **I expect my teams to launch their first sprint Week 3.**
- **I expect my teams to have daily standup meetings (preferably in F217/F225).**
- **Living Document**

Requirements Document

- Requirements will normally be captured in a format specified by the client. If the client has no specific requirements, we require you to include detailed use cases/user stories and a supplementary specification, based on input from your client and from other sources. Occasionally, other pieces also are important. (E.g., if you talk to other systems, perhaps charts of those interactions.)
- **Expect to have an updated version of this at the end of every sprint**
- **Living Document**

[1] Please determine this based on the project and any requests from the client.

Document the Design

a. I leave this decision to the team/client. But please continue to document any and all design decisions that the team makes. I will expect to see this at the end of every quarter.

Since we are recommending that an agile/iterative technique, a preliminary architecture/design/test document will be very useful. It will also help the team to finalize and document the architecture/design for the complete system by the end of the first quarter.

Senior Project Review:

Dr. Mark Hays and Dr. Michael Hewner have agreed to serve as reviewers for my senior project teams. During Week 8 or Week 9 of each quarter, each team has to sign up for a meeting with either Dr. Hays or Dr. Hewner. Each of them will review 2 teams. It is upto the team to reach out and schedule this meeting. If one of them has their slots filled up, you will have to sign up with the other person. The feedback from this review will have a significant impact on your grade.

During this review, you will have to present your requirements, architecture and current design. They reserve the right to ask questions about the choices you made and the right to review any code that they see fit. Depending upon the nature of the feedback, the team might have to make changes to their code base, design and architecture.