2 Project Plan

2.1 Project Management/Tracking Procedures

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

We will use a hybrid management style. After some research, we came to the conclusion that waterfall development is better for a single, large project like ours. So, we are going to use mostly waterfall, with some key changes. We are going to meet with our client once a week - more often than is normal for waterfall development. We are also not going to provide a deliverable to the client at each phase. After some phases, the product will not appear much different to an average consumer because that phase was for more back-end details that are not fully functional because they take several phases to complete. We will be using the waterfall method of "the next phase cannot start until the current phase is finished."

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.

We will use Github to store our code, Github Issues to keep track of which tasks are assigned to which people, Discord for team meetings, and instant messaging for communication.

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.

Task 1- Design Research:

- Cell Tower
  - Height? Dimensions
  - Construction cost?
  - Range?
  - Speed/bandwidth?
- Electricity
  - Research Mesecon's mod and how it can interact with other features
- Obstacles
  - How much do different materials block 5G signals? (dirt, stone, metal, etc)

Task 2- Implement features:

- Cell Tower
  - Model Design
  - GUI
  - Functionality
- Electricity
  - Design changes
  - Implement functionality
- Satellites
  - Model Design
  - Research
  - Resource cost
  - Functionality

- Obstacles
  - Signal transmission

        - Environment
        - Power supply
    - Fiber-Optic
        - Functionality
        - Wavelength/Bandwidth
        - Cost

Task 3- Test Features

- Test each of the features implemented in Task 2 as they are finished

Task 4 - Build Installer

- Make sure it is easy to use
- GUI development

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%.

Milestone 1: Finish Research and Begin Development

Milestone 2: Implement first feature

Milestone 3: Connect 2 features

Milestone 4: Finish all features and begin testing

Milestone 5: Finish testing and confirm the mod works

Milestone 6: Finish Installer

Metric: How many features are implemented

How to determine progress made for each feature:

Can you interact with it in the game?

Is it designed well and is it aesthetically pleasing?

Does it function correctly 100% of the time?

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

2.4 Project Timeline/Schedule

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

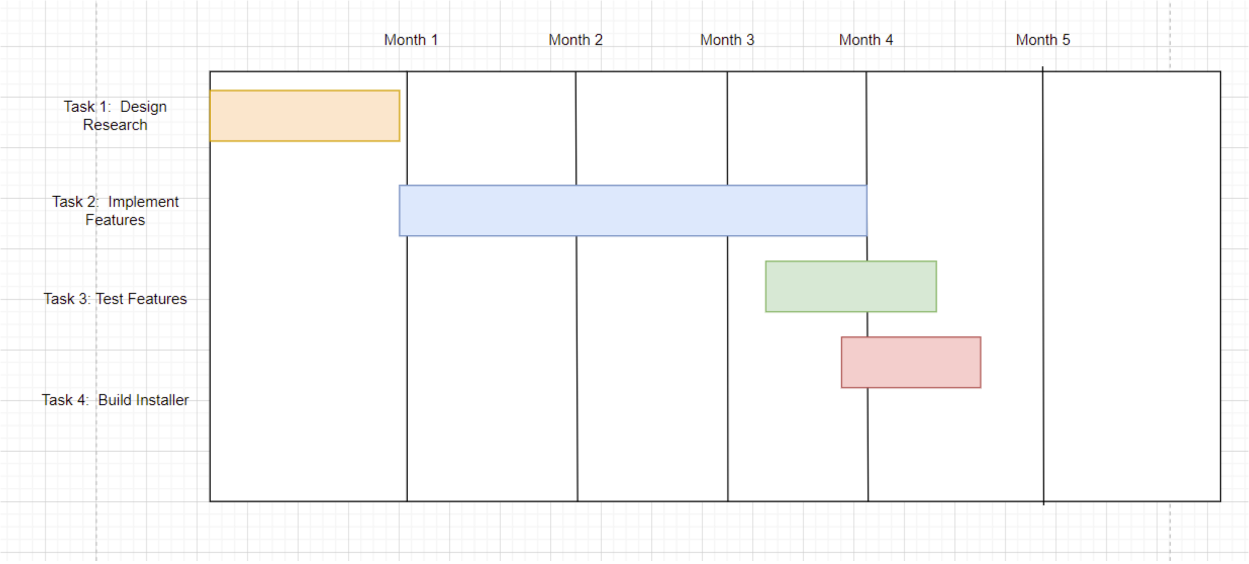
• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

• 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. 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 models. For agile, a sprint schedule with specific technical milestones/requirements/targets will work.



We want to have the first Task done and begin development by the end of the first month

Developing the features will be the longest task we hope to have that done by the end of the fourth month so 3 months to complete the 5 features

Testing can start before developing the features is completely finished it should only take about 3 weeks less

We will start working on the mod installer before testing is complete it should only take about 3 weeks as well

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?

Possible risks:

-Project not done in time: 20%
-Pre-made mods do not work as expected: 60% Risk Mitigation plan: Ask the rest of the team for help figuring out the mod, or use a different mod. This risk would be possibly occur in task 2

-Security vulnerability discovered after final deadline: 65% Risk Mitigation plan: Implement new and improved security protocols. This risk would possibly occur in task 3
-Issues with wireless connectivity: 60% Risk Mitigation plan: Communicate within the team to find the most stable method. This risk would possibly occur in task 2

Similar to an Agile project, we will associate risks and risk mitigation periodically throughout development.

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.

Task #	Hours needed	Explanation
Task 1: Design Research	10 hours per team member	Answering questions will lead to even more questions, but this should be enough to account for that.
Task 2: Implement features	50 hours per team member	Implementing features will be the single greatest time sink, so we want to give it plenty of time.
Task 3: Test Features	20 hours total	We estimated this amount of hours due to having to test all features to confirm functionality
Task 4: Build Installer	10 hours per team member	Building the installer will basically be connecting all of our features with the software and see if it is overall compatible

2.7 Other Resource Requirements

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

We will need to network to reach out to the clubs and organizations using the mod

We will need fiber optic cables - this includes hardware components

We will need software to design the models and code the mod