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.

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.

MANAGEMENT STYLE:

Our group plans to use a **hybrid waterfall + agile project management style**. This fits most appropriately with our project goals. Our project will require completion of certain stages of the project before beginning other stages—this lends itself towards the waterfall project management style. For example, we must complete the RF and ADC printed circuit board (PCB) designs before we can begin programming our microprocessor to command operation of the PCBs.

However, we will also work in parallel on many design stages—this lends itself to the agile project management style. For example, we will design and measure the RF antenna while simultaneously designing the RF printed circuit board it will work in conjunction with. Figure 1 below displays a review of the two project management styles side-by-side.

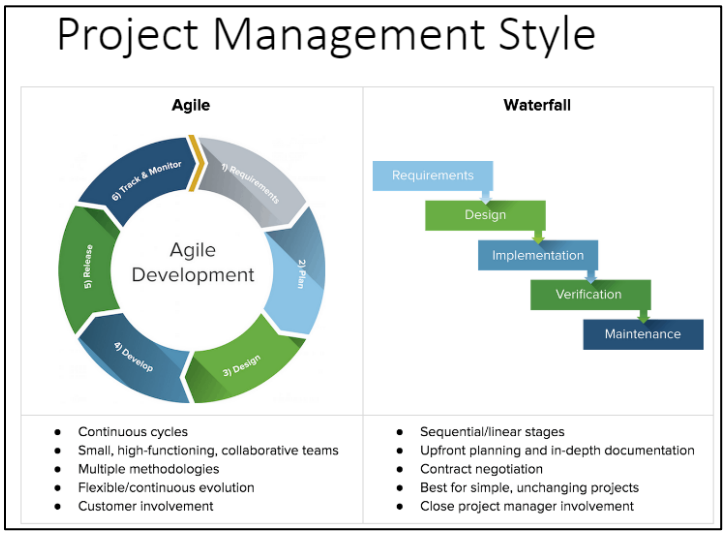

Figure 1 – Agile and Waterfall Project Management Styles[1]

[1] *Source*: EE 491 Lecture 1 Slides

PROGRESS TRACKING TOOLS:

Our group relies heavily on three primary tools for progress tracking and management: (1) Discord; (2) Google Drive; and (3) Google Calendar.

Discord: Every member of our group uses Discord daily for progress tracking and collaboration. It is our primary medium for collaboration and updating each other on our individual progress. It is also the primary medium we use to reach collective decisions, vote on contentious issues, and organize meetings. We discuss impending deadlines and long-term plans to accomplish goals and complete deliverables.

Google Calendar: We also utilize Google Calendar to schedule our many concurrent deadlines, obligations, and weekly meetings. Google Calendar is convenient for several reasons, but our primary use is to invite all group members to meetings without the need for constant manual reminders. Google Calendar allows us to create a single centralized recurring meeting and group invitation, and then it automatically reminds all of us when and where to attend. It also allows us to conveniently cancel a single meeting, if need be, without deleting the recurring meeting notice in the future.

Google Drive: Google Drive is where we house our shared work repository. It is also an effective means to monitor the progress of individual group member tasks. For example, several of our group members worked on a Bill of Materials concurrently, but they each had to research, select, and add their components for their respective project tasks independently. Google Drive allows us all to monitor the collective completion of a shared task such as composition of a Bill of Materials.

Future: We are likely to utilize GitHub when we enter the programming stage of our project. All members will have access to our GitHub repository at that time, and we will likely maintain sub-projects for the various programming tasks (low and high level/user interface display).

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.

Our RF imaging array project contains several concurrent, interdependent, and cross-compatible tasks and subtasks. The primary tasks are enumerated and explained below:

2.2.1. Computer Simulation Technology Modeling and Antenna Design

The backbone of our project is the antenna array design. 8 RF antennas will be placed adjacent to one another in a one-dimensional array of antennas. These antennas emit radio frequencies in all directions and will be studied carefully for application appropriateness prior to being procured (i.e., purchased by our team). Then, the antenna array will be assembled and calibrated in a stationary array, as shown in Figure 2 below. Assuring functionality requires use of Computer Simulation Technology (or CST for short), as shown in Figure 3 below.[2] The specific application we will use is CST Studio Suite. CST Studio Suite will enable us to determine whether our antenna array is tuned correctly, matched correctly, built correctly, and then how we can expect it to behave in real (i.e., physical) applications.

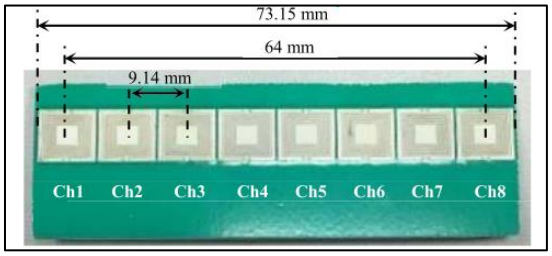

Figure 2 – One-Dimensional Antenna Array Example[3]

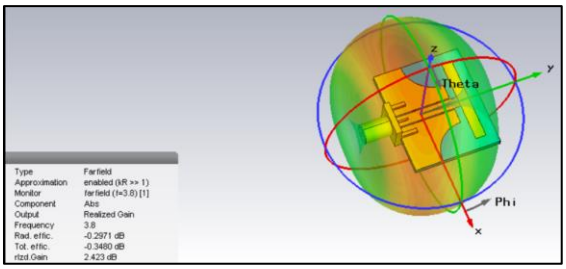

Figure 3 – CST Antenna Emission Rendering Example[4]

[2] Use of CST is a recurring sub-task we will utilize in multiple primary stages of our project including (1) antenna design, (2) antenna tuning, (3) RF PCB and switch design, and (4) microprocessor hardware-to-software programming synchronization.

[3] *Source*: https://ieeexplore.ieee.org/abstract/document/8815736

[4] *Source:* https://www.wifi-antennas.com/profile/35-sandeepv/?do=content&type=forums_topic_post&change_section=1

Figure 4 below includes an image of our preliminary schematic. KiCad was used to produce the schematic. (Source: https://www.kicad.org/). The red box signifies where the antenna array will be positioned with respect to the other interdependent components.

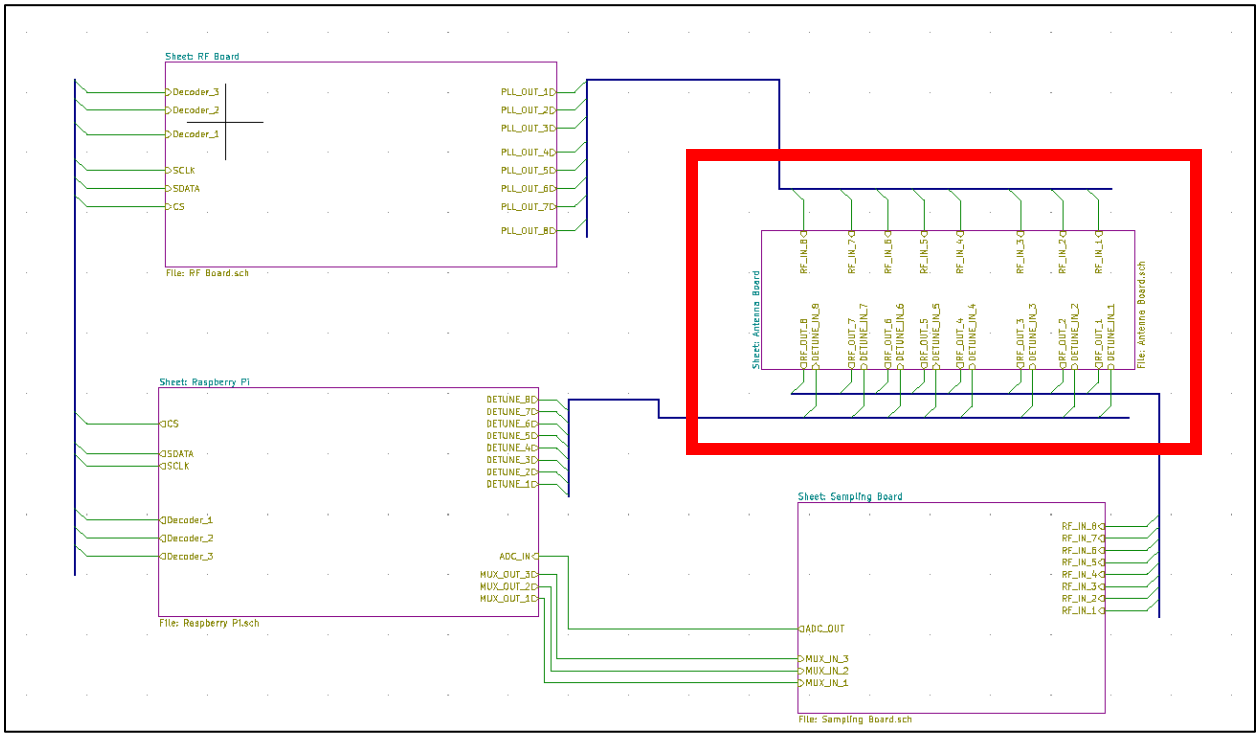

Figure 4 – Full Schematic (Antenna Array in Red)

2.2.2. Antenna Tuning

Tuning our antenna array is an essential primary task related to our project design. Each individual antenna of the eight-antenna array will be tuned to function properly on its own, and then all antennas will be re-tuned to function while stationed immediately adjacent to one another. Antennas emitting and receiving radio frequencies will inevitably interfere with each other. However, CST Studio Suite will again allow us to simulate these outcomes and design an array that is calibrated to minimize cross-interference between individual antennas. Ultimately, the antennas will be calibrated to optimize emission projection and reception.

Figure 5 below is a critically important representation of the optimality for RF antenna emissions: S11 (on the y-axis) represents the amount of power reflected back from the object that the antenna's outgoing RF emissions are directed towards. For an ideal antenna, all the power would be transmitted, so the magnitude of S11 should be negative infinity. However, we will aim for anything less than -20dB as that will be sufficient for our application.

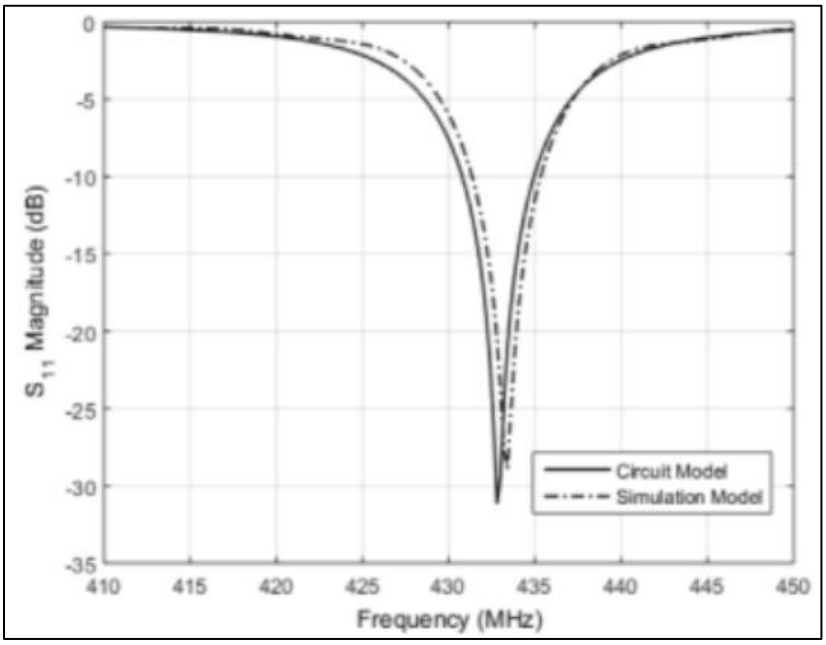

Figure 5 – Antenna Array Simulated Frequency Response Example[5]

[5] *Source:* https://www.wifi-antennas.com/profile/35-sandeepv/?do=content&type=forums_topic_post&change_section=1

2.2.3. RF PCB Design (PLL & Switches)

We will design two printed circuit boards (PCBs) for our project: **the first is a PCB for the RF antenna array**. This PCB is a primary task that will include a phase-locked loop[6] and switches to operate the various antennas in the eight-antenna array. This also correlates with enabling us to output a consistent output signal with constant frequency. This is crucial in an RF system as the system relies on a precise design frequency; changing this frequency would result in larger losses or a decrease in the expected power gain in the signal we are trying to send to the antenna to propagate out. Once the signal has been produced, we will use RF switches to selectively choose which antenna in our array we want to send the signal to which prevents interference and coupling between the many antennas.

Figure 6 below includes an image of our preliminary schematic.[7] The red box signifies where the RF PCB design will be positioned—this PCB will include both a PLL and switches. This RF PCB will interact with and control the RF antenna array while transmitting data to and from our software, analog-to-digital converter, and user-interface data display, all of which are explained on the subsequent pages.

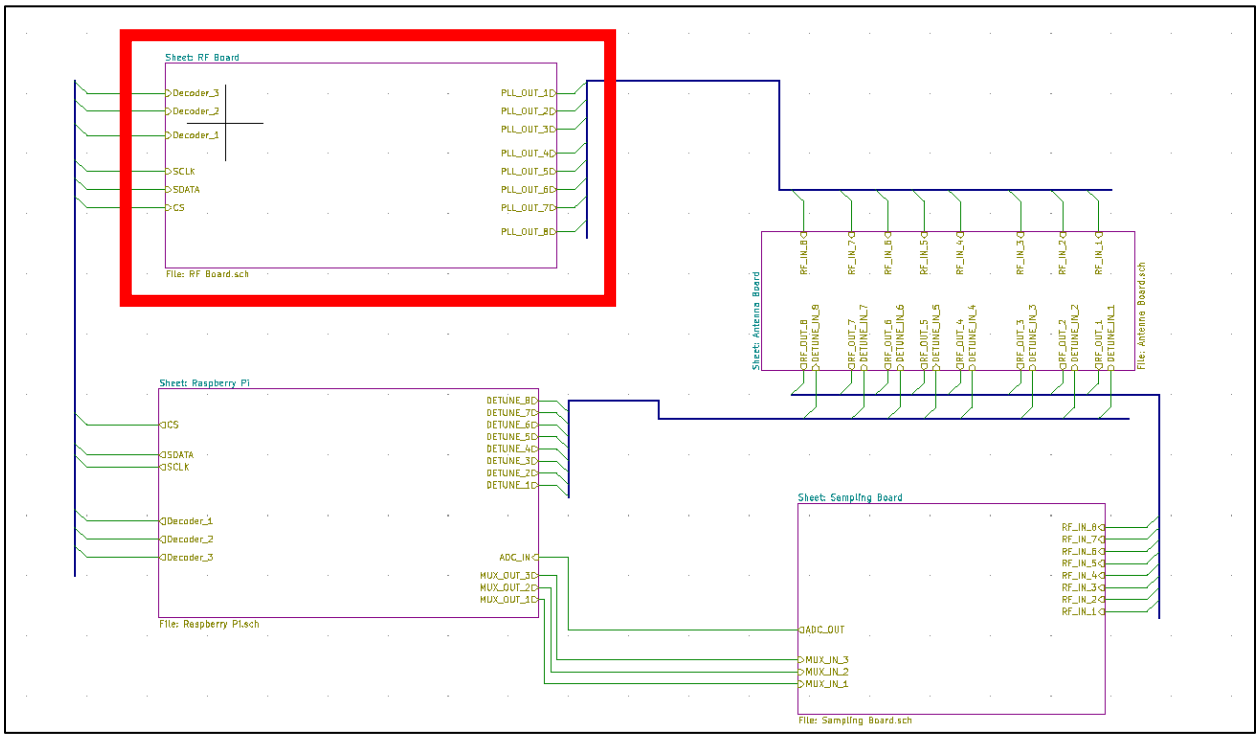

Figure 6 – Full Schematic (RF PCB in Red)

[6] A phase-locked loop (PLL) is a control system that generates an output signal with related output and input phases.
[7] KiCad was used to produce the schematic. (*Source*: https://www.kicad.org/).

2.2.4. ADC PCB Design

Another primary task will involve designing a second PCB incorporating the layout of a subsystem utilizing an analog-to-digital (ADC) integrated circuit (IC) for interpretation of the data coming from the antenna array. An ADC is a system that takes real-time analog data and models it as digital data that is usable by the microcontroller governing our overall system. This will involve utilization of layout principles set forth in the documentation for the IC, as well as documenting what the output of this IC will look like to simplify the work of our microcontroller team.

Figure 7 depicts our preliminary schematic. The red box demonstrates where the ADC PCB will be situated within the broader circuitry. The wires also demonstrate its connection to and interaction with the other interrelated system components.

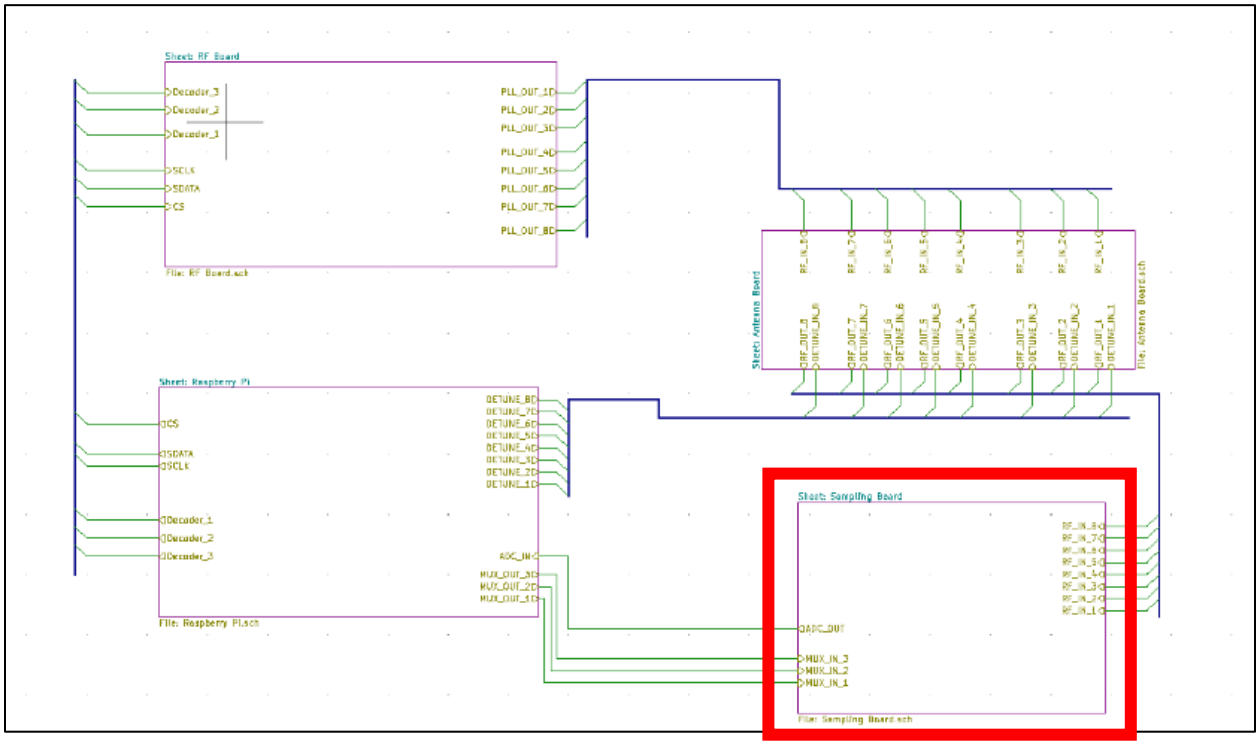

Figure 7 – Full Schematic (ADC PCB in Red)

2.2.5. Low Level Programming (Data Gathering)

A microprocessor will be programmed to control our hardware devices. We are likely to use a raspberry pi, but an FPGA or other microprocessor could also be used. This microprocessor will control several functions: (1) device commands and controls, (2) data gathering, (3) data display, and (4) user-interface communication and interaction.

The "low level" programming entails controlling devices and gathering data therefrom to provide information to the user. Specifically, data about the RF antenna emissions will be collected by the microprocessor. This data will be sent to a user interface thereby providing the user with information about emissions inconsistencies. Such inconsistencies include hidden structures in walls or biomedical anomalies that lie under the epidermis (outer layer of skin) and cannot be viewed by the human eye. An image of the raspberry pi we are considering for our microprocessor is displayed in Figure 8 below:

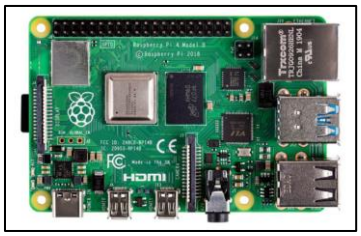

Figure 8 – Raspberry Pi Candidate for Microprocessor[8]

Figure 9 below depicts our preliminary schematic. The red box demonstrates where the microprocessor will be situated within the broader circuitry. The wires also demonstrate its connection to and interaction with the other interrelated system components. Notably, it will directly interact with the RF PCB and ADC PCB while sending commands and receiving information through those PCBs to and from the antenna array.

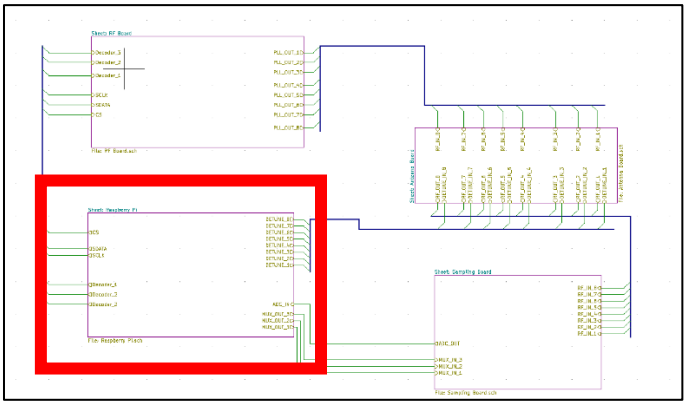

Figure 9 – Full Schematic (Microprocessor in Red)

[8] *Source*: https://www.pishop.us/product/raspberry-pi-4-model-b-4gb/

2.2.6. Data Processing and Display

The final primary task for our project will involve microcontroller configuration (as in Figures 8 and 9 above) for data processing, coupled with a user interface for data display. Our hardware devices will be implemented with compatible software that provides an easily understood visual or quantitative depiction of the information generated and collected by the antenna array working in conjunction with the other various components listed above. We plan to use Tornado (a Python web framework) to display the data on electronic devices like laptops or smartphones while a Python script or other object-oriented programming language will power the backend functionality.

We are also considering displaying this information on an LCD screen with an HDMI port. Figure 10 below depicts an image of a candidate we are considering for our user interface display monitor—however, we may also use a wireless connection that enables use of any laptop as the display.

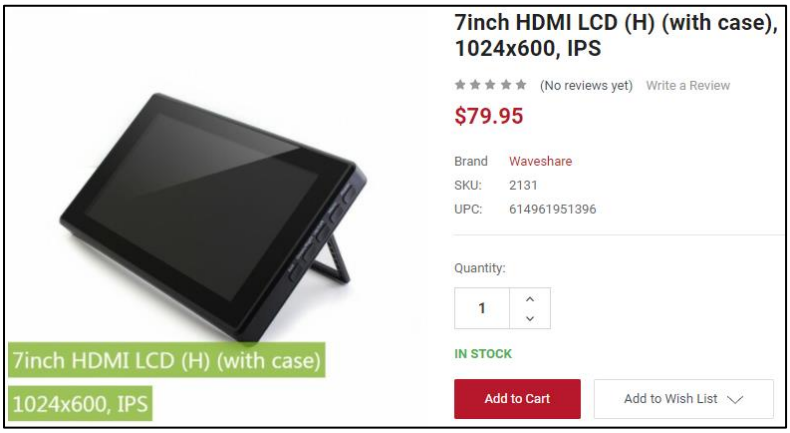

Figure 10 – User Interface Display Monitor Candidate[9]

[9] *Source:* https://www.pishop.us/product/7inch-hdmi-lcd-h-with-case-1024x600-ips/

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 sprint).

2.3.1 Milestones and Metrics: *Antenna CST Modeling, Design, and Tuning*

Milestones:

1. Design antenna without detuning or measuring circuity in CST; verify desired resonant frequency of 400 MHz with the antenna design
2. Design antenna with the detuning circuit along with variable capacitor on the matching network to allow for variation in components upon assembly and to tune the boards once they are fabricated
3. Place antenna design onto a PCB and have it manufactured to be tested with Vector Network Analyzer (VNA)[10] to confirm simulation parameters for components were accurate and design is valid
    a. Expect to see S11 magnitude to be around -20dB or less
    b. When compared to simulation, should see fairly close match but not exact due to manufacturing limitations of PCB and respective components
4. Using design from previous step, place 8 antennas in series and modify the tuning and detuning circuitry to achieve lowest possible interference among antennas
5. Design (and then fabricate) PCB with the antennas, matching network, and detuning circuitry on it and have it fabricated
6. With fabricated board, use VNA to tune all antennas individually and verify functionality at the design frequency; once again, seeking S11 magnitude of around -20dB or less

Metrics:

1. Overall S11 (reflections back to the source) of about -20dB or less if possible
2. Simulation to match real world within about 5%
3. Coupling between probes we want to be around -30dB represented by S21

[10] "VNA" stands for "Vector Network Analyzer." A VNA sends out signals and sees how they are reflected, or absorbed/transmitted.

2.3.2 Milestones and Metrics: *RF PCB Design (PLL & Switches)*

Milestones:

1. Identify viable components for vital integrated circuits (ICs) such as phase locked loop and RF switches
2. Simulate behavior of ICs in advanced design system (ADS) to ensure proper signal behavior such as the following:
    a. delivering sufficient power to the antenna as required
    b. Matching networks to ensure transmission line effects are minimized
3. Identify required supporting components (passive devices and other ICs)
4. Complete schematic design and PCB layout in CAD software
5. Create Bill of Materials (BOM) and purchase components for RF PCB
6. Solder components onto the board
7. Confirm (via testing) successful operation of board with expected results

Metrics:

1. PLL correctly produces a signal of 400MHz
2. Power transfer across the switches is at least 10dB
3. RF switches correctly turns on/off and sends the signal to the right destination

2.3.3 Milestones and Metrics: *ADC PCB Design*

Milestones:

1. Vital components for ADC PCB will be identified
2. Vital components for ADC PCB will be purchased
3. Supporting components for use of ADC IC will be identified and purchased
4. Total sub-BOM for ADC PCB assembled, with PCB layout identified
5. Layout for PCB used to order custom PCB from supplier
6. Components for ADC PCB hand-placed and visually verified
7. ADC PCB tested with a waveform generator and a digital oscilloscope
8. ADC PCB implemented and requirements verified within the top-level design

Metrics:

1. Sampling rate of ADC PCB will be verified to be 100kHz
2. Resolution of ADC PCB will be verified to be 16 bits

2.3.4 Milestones and Metrics: *Low Level Programming (Data Gathering)*

Milestones:

1. Code shell
2. Purchase of microcontroller
3. First draft of code for transmission and reception of all outgoing/incoming data
4. Testing phase
5. Consistent data transmission, reception, and interoperability

Metrics:

1. 80% accuracy for commands output to PCBs

2.3.5 Milestones and Metrics: *Data Processing and Display*

Milestones:

1. Code shell
2. Purchase of microcontroller
3. Interpretation of incoming data for transmission to user interface display
4. Incorporation of wireless data display tool (e.g., Tornado)
5. Consistent data transmission, reception, display, and interoperability

Metrics:

1. User satisfaction survey at 80%

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

Table 1 below displays a Gantt Chart. This chart sets forth the timeline for our project tasks and associated subtasks (developed in 2.2).

Table 1 – Gantt Chart for Team 23 Task Completion Schedule

TEAM 23 GANTT CHART - TASK COMPLETION SCHEDULE										
Task | Month	Aug ('21)	Sept ('21)	Oct ('21)	Nov ('21)	Dec ('21)	Jan ('22)	Feb ('22)	March ('22)	April ('22)	May ('22)
Antenna CST Modeling, Design, and Tuning										
RF PCB Design (PLL & Switches)										
ADC PCB Design										
Low Level Programming (Data Gathering)										
Data Processing and Display										
Technical Writing										

Table 1 makes clear that many of our primary and subtasks must be completed concurrently. However, some of the tasks are dependent on the completion or near-completion of others. For example, low level programming cannot begin until after antenna modeling, RF PCB design, and ADC PCB design are nearing well-established enough to begin producing representative data samples. Data processing and display is necessarily the last task we will complete; it is dependent on the completion and consistent functionality of all other technical tasks.

Technical writing will remain a consistent responsibility throughout the duration of the project. As such, its start date maps to the start of the class while its end date maps to the end of next semester's course. Finally, we have set ourselves an early deadline for completion of all technical aspects of the project. We intend to complete the project as quickly as possible to provide ample time for troubleshooting and presentation preparation.

2.5 RISKS AND RISK MANAGEMENT/MITIGATION

Consider for each task what risks exist (certain performance target may not be met; certain tool 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?

Agile project can associate risks and risk mitigation with each sprint.

1. **Risks and Risk Mitigation**: ***Antenna CST Modeling, Design, and Tuning***

Risk	~ %	Risk Mitigation Strategy
Rate of reflections back to source > -20dB	15%	
5% simulation-to-real world correlation tolerance	50%	Constant tuning and re-tuning
Destabilizing antenna crosstalk interference	70%	Error isolation and independent antenna tuning
Faulty probe coupling	30%	

2. **Risks and Risk Mitigation**: ***RF PCB Design (PLL & Switches)***

Risk	~ %	Risk Mitigation Strategy
PCB schematic design mistakes	40%	Meticulous schematic review
PLL failure to produce 400 MHz signal	20%	
< 10dB power transfer to antennae	25%	Switch re-design simplicity and energy conservation
Faulty automated RF on/off functionality	30%	

3. **Risks and Risk Mitigation**: ***ADC PCB Design***

Risk	~ %	Risk Mitigation Strategy
ADC PCB sampling rate < 100kHz	15%	
<< 16-bit ADC resolution	40%	FIR filters to filter out quantization noise

4. **Risks and Risk Mitigation**: ***Low Level Programming (Data Gathering)***

Risk	~ %	Risk Mitigation Strategy
< 80% accuracy for commands output to PCBs	50%	Careful quality control of PCB design; re-coding and constant quality assurance if necessary

5. **Risks and Risk Mitigation**: ***Data Processing and Display***

Risk	~ %	Risk Mitigation Strategy
Non-intuitive graphical display of data	25%	Objective user surveys for constructive feedback
Delay in processing of data	40%	Simple and efficient coding structure; careful calibration of hardware components

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 total number of person-hours required to perform the task.

Table 2 below sets forth a detailed estimate of the projected effort per project task in total number of person-hours.

Table 2 – Team 23 Peron-Hour Effort Requirements (Per Task and Total)

Task	Person-hrs./wk. (est.)	~ # Wks.	Total
Antenna CST Modeling, Design, and Tuning	15	20	300
RF PCB Design (PLL & Switches)	10	16	160
ADC PCB Design	12	12	144
Low Level Programming (Data Gathering)	12	14	168
Data Processing and Display	10	16	160
Technical Writing	8	40	320
Total Person-Hour Effort Requirement			**1252**

The table above makes clear that some tasks will span a shorter calendar duration, but more time must be dedicated during the weeks where the task will be completed. For example, ADC PCB design will only run 12 weeks, but it will take more time during those weeks than programming to complete the data processing and display task.

Antenna CST Modeling, Design, and Tuning is the backbone of our project. RF PCB design is likely the second-most integral component of our RF imagining array. As such, they both require considerable commitments in time and calendar duration.

2.7 OTHER RESOURCE REQUIREMENTS

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

PARTS (BILL OF MATERIALS AS OF EARLY OCTOBER 2020):

Table 3 – Bill of Materials for Printed Circuit Boards

RF Circuit						
Part Number	**Description**	**Digikey Link**	**Quantity**	**Unit Price**	**Extended Price**	
ADF4360-7BCPZRL7	Voltage Controlled Oscillator	https://www.digikey.com/en/products/detail/analog-	3	7.74	23.22	
HMC253QS24	SP8T RF Switch	https://www.digikey.com/en/products/detail/analog-	2	17.39	34.78	
NXB0104PWJ	4-Channel Level Shifter	https://www.digikey.com/en/products/detail/nexperi	2	0.69	1.38	
AS213-92LF	SPDT RF Switch	https://www.digikey.com/en/products/detail/skywor	16	0.98	15.68	Second Option RF Switch
SN74LVC2GU04DBVR	2-Channel Inverter	https://www.digikey.com/en/products/detail/texas-in	16	0.41	6.56	
BGS18GA14E6327XTSA1	Cheaper RF Switch	https://www.digikey.com/en/products/detail/infineor	2			
ADC/Sampling						
Part Number	**Description**	**Digikey Link**	**Quantity**	**Unit Price**	**Extended Price**	
ADS8866IDGSR	IC ADC 16BIT SAR 10VSSOP	https://www.digikey.com/en/products/detail/texas-i	2	5.41	10.82	
MAX4020ESD+	IC OPAMP VFB 4 CIRCUIT 14SOIC	https://www.digikey.com/en/products/detail/MAX40	4	8.57	34.28	
	Raspberry Pi 4	https://www.amazon.com/Vilros-Raspberry-Compl	1	99.99	99.99	
ADG1606BRUZ	Multiplexer Switch ICs 2 Ron, +/-5V,	https://www.mouser.com/ProductDetail/Analog-De	2	8.83	17.66	datasheet

Table 4 – Bill of Materials for Antennas

Antenna					
Part Number	**Description**	**Digikey Link**	**Quantity**	**Unit Price**	**Extended Price**
TC74VHC238FK	3:8 Decoder	https://www.digikey.com/en/products/deta	3	0.72	2.16
SMS7621-040LF	Shottky Diode	https://www.digikey.com/en/products/detail/sk	12	0.66	0.45
Still Needed:					
Op-amp					
Variable capacitor					

Table 5 – Bill of Materials for Diode Test Circuit

Part Number	**Description**	**Digikey Link**	**Quantity**	**Unit Price**	**Extended Price**
ADT1.5-122+	Balun		4		
SMS7621-040LF	Schottky diode	https://www.digikey.com/en/products	20	0.66	

TOOLS:

We require several tools and utilities—including both hardware and software—for completion of our project. These include at least the following:

- **KiCad** – for schematic design
- **CST Studio** – for simulation of RF antennas
- **Virtual Network Analyzer** – to facilitate antenna testing
- **Soldering** – to join different types of metals on PCBs together by melting solder
- **Python** – to project visual user interface display of data onto electronic devices

FACILITIES:

We require use of the Applied Sciences lab at 1925 School Road in Ames, Iowa. The facilities in lab will allow us to test our radio frequency antennas as well as solder components to our printed circuit boards. We will also be able to produce a 3D printed custom bracket to hold all of the components of our RF antenna imaging array system, and we will likely finalize the construction in the applied sciences lab.

FUNDING SOURCES:

While there are specific monetary resources we need—and the instructions suggest we should not list those here—we also need to find ***sources*** for funding. We are exploring several avenues for corporate funding. We have considered and contacted several national and local companies.[11] We are in the process of corresponding with these potential corporate sponsors. If we fail in obtaining corporate funding, we will self-fund the project.

[11] The complete list of companies we have reached out to is as follows: MOSIS, TI, MICRON, Walmart, Costco, 3M, Verizon, Taco Bell, Coca-Cola, Google, Microsoft, Danfoss, Barilla, Mid-American Energy, MISO, John Deere, and Harman.