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.

Our group will follow an agile management style. We anticipate our project will go through multiple iterations as we develop, an agile methodology will be the best practice for this kind of development cycle. Given that our product will be interacting with 2 users (Human and dog users), we will be going back and forth between these two on comfort, utility, convenience, etc. Agile rather than waterfall has flexibility in accommodating between these 2 users criticisms between prototypes and testing.

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 manage our sprints and keep track of milestones and progress. We will divide up tasks and keep track of completed/incomplete tasks. Meeting notes and discussions are done within discord.

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.

App

- User Interface
    - Design Theme
    - Main Page

        i. Device Management Pop-up

        ii. Other Resources Pop-up

    - Activity Page

        i. Graphs

        ii. Other Analytics

- Dog Management Page

  i. Dog Manager

- Backend
  - API/Send-Receive Data
  - Connecting to devices
  - Cache Info (Flutter Hive??)
  - Database? (Firebase??)
  - Machine Learning Algorithms

Dog Device

- First Iteration
  - Device Build
    I. breadboard/jumper wires for connectivity
    II. Arduinos for central processing and control
    III. vibration motor for dog alert
    IV. bluetooth module for communication between watch and device
    V. 5 V source for power
    VI. 3D printed enclosure for structure
  - Establish connection between Flutter app, phone, and watch
- Additional Iterations
  - Change enclosure based on how fits
  - Add PCB to get rid of cumbersome jumper wire/breadboard connection

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).

Milestones

- UI allows user to reach 100% features (design/theme is beyond this milestone)
- Machine Learning algorithms are 80% accurate
- Response time from PTSD episode recognized to dog collar is less than a second
- Dogs are 99% responsive to device
- Design of sensors and hardware finalized

- Design of PCB schematic programmed
- Design of PCB housing

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

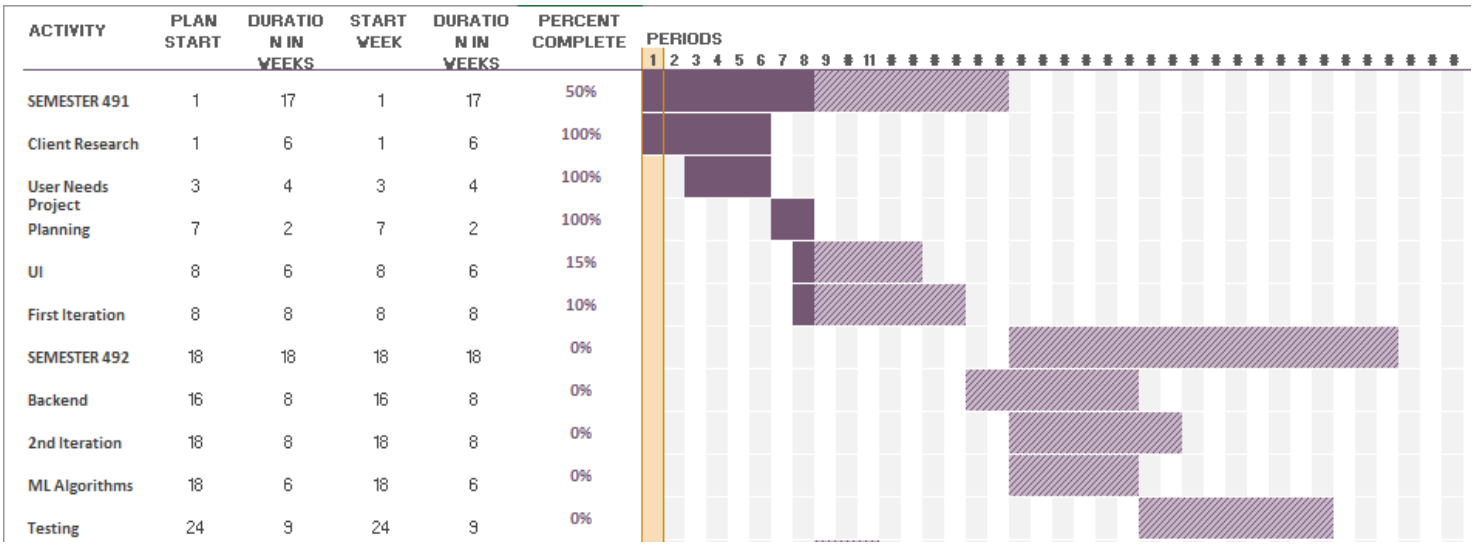
• A detailed schedule is needed as a part of the plan:

– 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.

ACTIVITY	PLAN START	DURATION IN WEEKS	START WEEK	DURATION IN WEEKS	PERCENT COMPLETE
SEMESTER 491	1	17	1	17	50%
Client Research	1	6	1	6	100%
User Needs	3	4	3	4	100%
Project Planning	7	2	7	2	100%
UI	8	6	8	6	15%
First Iteration	8	8	8	8	10%
SEMESTER 492	18	18	18	18	0%
Backend	16	8	16	8	0%
2nd Iteration	18	8	18	8	0%
ML Algorithms	18	6	18	6	0%
Testing	24	9	24	9	0%



Light Purple - Planned Duration for each task

Dark Purple - Completed

Gantt Summary: The majority of our requirements gathering with users and clients has been completed, the bulk of the project that is left is developing and iterating a device and app. We have a UI prototype and a bill of materials for the actual device, but in the coming weeks the UI will begin to get implemented (approximately a week per page, including a developing a theme/design)

and the first iteration of the device (this will be an arduino with a bluetooth module) will be developed in conjunction with the app. Second semester will be primarily focused on getting the backend piece implemented on the app, this includes connection to device, database work, and machine learning implementation. A 2nd iteration will be developed in the second semester, this will be more of a blueprint based off of our 1st iteration's design. Finally, testing (testing will be completed throughout, but the last weeks will be for finding any tweaks needed).

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?

Agile project management can associate risks and risk mitigation with each sprint. When implementing this style, we will be able to adapt to issues that come up in real time and retarget more efficiently.

Risks (and mitigation):

- Machine learning algorithms will have to be unsupervised (unless further research suggests otherwise) so there is a chance the algorithm will not find anything helpful. To mitigate this risk, we will not completely rely on the algorithm, if vitals are clearly suggesting a possible PTSD attack we will assume it is. (Risk Rating 0.6)
- Users may feel our system is intrusive. To mitigate this risk we are making our system as configurable as possible, allowing them to turn features off and on, even customizable, so the device is comfortable and non-intrusive for all users and dogs. (Risk Rating 0.7)
- Dogs may find the device too uncomfortable or may not respond to the stimuli we provide. This would hinder the dog's ability to respond and be trained using the vibration stimuli. To mitigate this risk we are including buffer room for multiple iterations of the Dog's device design as well as time for testing on dogs. Additionally, the configurability of the device should allow the user to adjust the vibration motor to best suite the individual dog. (Risk Rating 0.3)
- Hardware may not be capable of meeting our requirements. To mitigate this risk we our ordering multiple versions of our hardware modules so that we can figure out which is best fit for our system (Risk rating 0.3)
- Devices may incorrectly detect PTSD attacks (false positives). To mitigate this risk, we are offering the user the ability to turn off and on the device if they anticipate doing an activity that may raise their heart rate (exercise, roller coasters, etc). (Risk Rating 0.7)

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.

Personal Effort Requirements	
Task	**Time (hours)**
UI-Main Page	2.5
UI-Activity Page	2.5
UI-Dog Management Page	2.5
Backend-Main Page	4-6
Backend-Activity Page	4-6
Backend-Dog Page	4-6
Backend-Machine Learning (Learning Curve)	5-10
Backend-Caching User Info	2.5
Create Schematic and find compatible parts	5
Order Parts For Device	0.5
Connect Core connections to Arduino (Power and communication)	0.5
Program Arduino for core functions	5
Test Core Connections/Functions	3
Connect peripherals to Arduino (vibration device)	0.5
Program Arduino for Peripheral Function	0.5
Test Peripheral Function	2
Create Enclosure and Holding Device	10
Test on Dog (Multiple types of service dogs with different size vests)	20

(After First Iteration)	
Create PCB Add on To Arduino	8-15
Reconnect Devices	0.5
Create Enclosure for Device (CAD)	2
3D Print Enclosure and Integrate (Hours include printing time)	8-15
Reiterate Enclosure Design for Comfortability and Utility	Indeterminate
TOTAL	~105

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 have access to persons with both PTSD and a trained service dog. They will serve as a knowledge base for questions we wouldn't be able to answer about the relationship between the animal and the person. In addition, we will also be able to test out our device and ask questions of the human about what their preferences would be about how our device will function. We will need arduinos or raspberry pi in order to control rumble packs and bluetooth radios to communicate with the watch on the user.