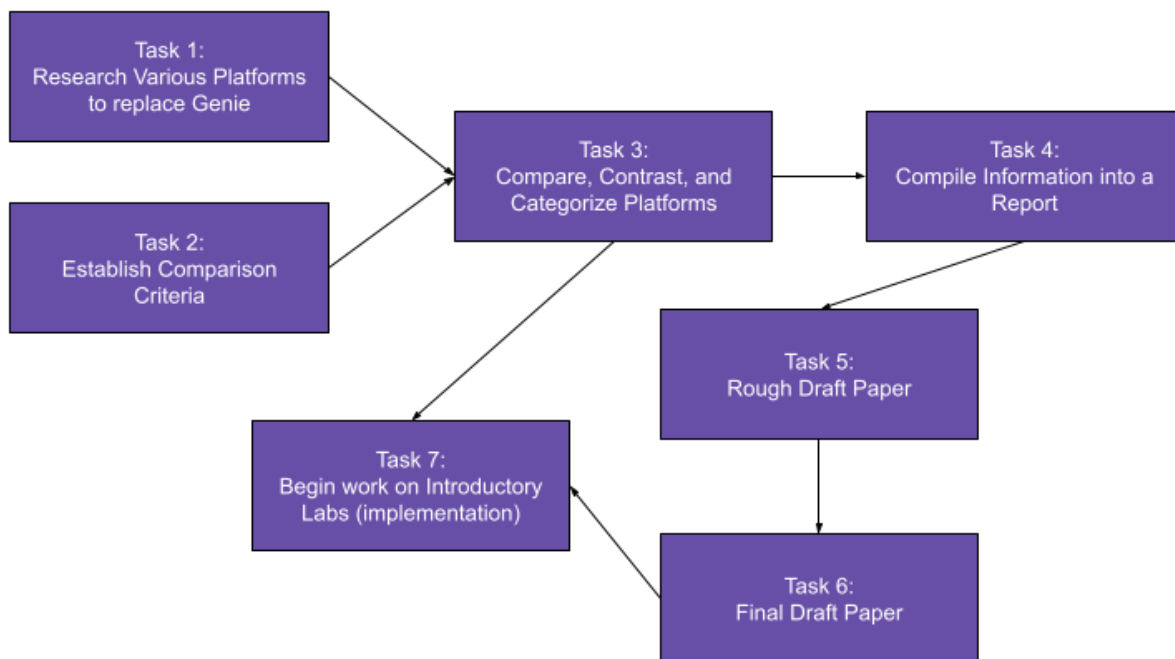


## 2 Project Plan (sdmay24-20)

### 2.1 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. At minimum, this section should have a task dependence graph, description of each task, and a justification of your tasks with respect to your requirements. You may optionally also include



For the Task Decomposition, we broke it down into big-ticket concepts and tasks. While there are submodules (not listed here), these are the big milestones with the dependency associated with them.

**Task 1:** This task is a huge part and time-consuming milestone. Researching and getting a firm understanding of the capabilities and restraints of the platforms is crucial to achieving our end-of-project goal.

**Task 2:** Taking our research and begin to critique it and categorize the platforms to get a better understanding of what we have at hand. Since the research of each platform is done by a single team member, this step allows for collaboration and discussion based on our findings.

**Task 3:** Task 1 and Task 2 need to be completed in order for us to pit the platforms together and directly see disadvantages and advantages of each other.

**Task 4:** Taking our findings, we will now compile the research and information into a report and begin writing our paper.

**Task 5:** Take the paper and complete a full and tensive rough draft.

**Task 6:** This stage will be completing our final draft of our paper and submitting for publication.

**Task 7:** Can be done in conjunction with the completion of Task 3 as we would have a good understanding of the platforms at play and can start creating the introductory labs.

## 2.2 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 has elected a *Waterfall+Agile* mix project management style to complete our project goals. We believe this approach will be most effective as we have a rather sequential approach that allows us to divide into distinct phases. We want to adopt the agile aspect in terms of flexibility and adaptability; we want to be open to changes in project requirements and priorities at any stage if any new development comes from the platforms we are researching.

With our project goals, the waterfall approach works well as we break our goals into phases: Requirements, Research, Comparison, Report, Implementation, Testing, Documentation, and Deployment.

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.

Our main form of communication and project management is currently using Microsoft Teams. Microsoft Teams allows us to communicate, plan, and create a roadmap to success. With our project advisor, we create a document that outlines the following: What we plan to accomplish within two weeks and What we did accomplish within those two weeks.

### 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 2.1. 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%.

Due to our project being heavily reliant on research and data collection of various platforms and architectures, our metric of progress is rather abstract and arbitrary.

Some metrics with our goals are as follows:

- The Scale of the Network
  - *Coverage and size it's capable of supporting*
- The Ease of Use
  - *How easy it is to use for an introductory lab*
- Capabilities and Abilities
  - *How advanced and how many tools are in our arsenal*

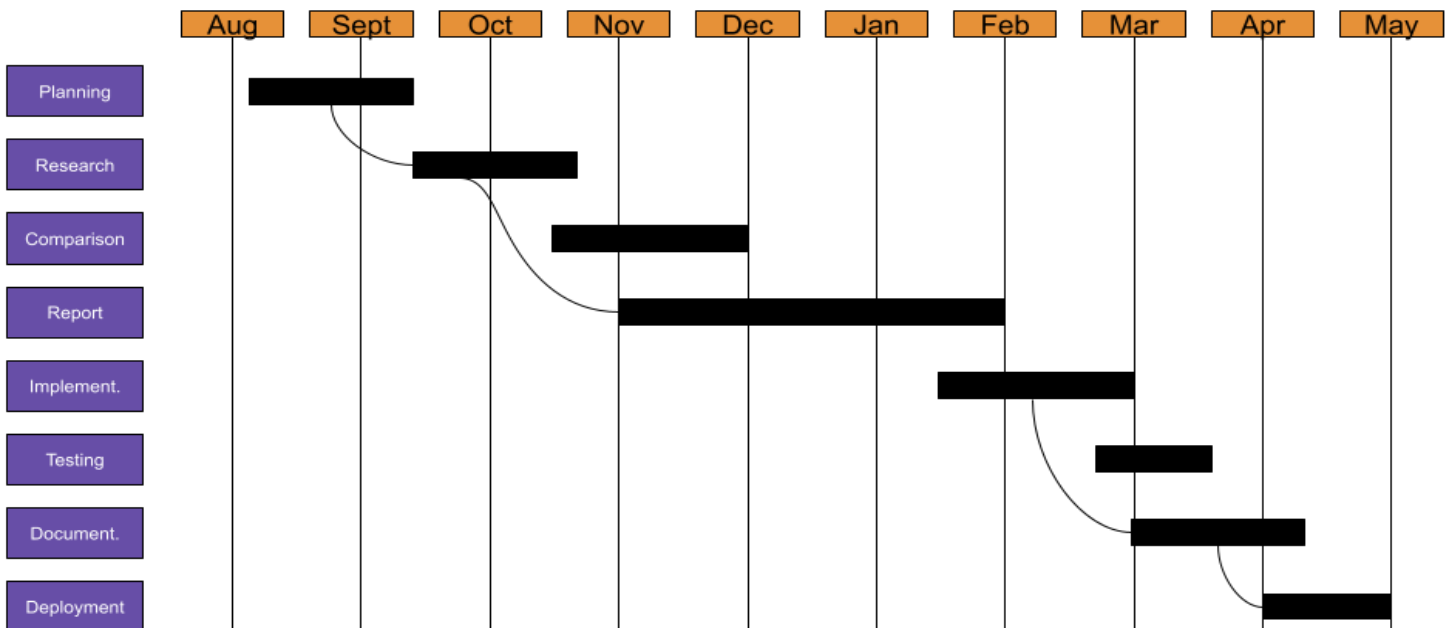
#### Milestones (\*Subject to change)

- 1) Fully research 8+ various platforms to allow the team members to have 80% knowledge of public websites and infrastructure.
- 2) Categorize the platforms and aim for a scalability of 75% rating
- 3) Categorize the platforms and aim for a usability of 60% rating
- 4) Categorize the platforms and aim for a capability rating of 80%
- 5) Select the best platform(s) that score the highest in our rating criteria based on our arbitrary metrics
- 6) Create lab assignments that have a rating of 60% or higher for usability

## 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

### GANTT CHART: Rough Schedule for SDMAY24-20



**Planning:** We will look at defining the project's scope, objectives, and timeline. We'll compose a comprehensive project plan to ensure uniformity and consistency.

- Submodule(s): Timeline Building, Project Objectives, Project Requirements/Standards

**Research:** We will look at researching and building ethos on various platforms and architectures to help us gain information and data.

- Submodule(s): Break down each platform into separate categories and define our metrics on what we want from each system.

**Comparison/Analysis:** Using the research gathered from each member, it's crucial to create categories and comparisons between the platforms to ensure the most efficient and effective.

- Submodule(s): Pit the platforms together to see which one would create the most practical and accessible use case for university labs.

**Report:** Compose and compile all researched and collected data into a formal paper.

**Implementation:** Take the research above and begin to create introductory labs for Cyber/Network related courses.

**Testing:** Test the lab designs, implementations, and overall effectiveness to ensure it's appropriate for the class.

**Documentation:** Create lab documentation that upholds the standards of the university and specific class.

**Deployment:** Public and release all findings and documentation.

## 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?

There are two places where risk might become a concern in this project. The first is during the information collection phase. The risk involved here is being able to find the information we are using for comparison and whether we can get a timely response from platforms when we are unable to locate that information ourselves. There is not much that can be done to mitigate this risk except to try and exhaust all other forms of information before having to reach out to the developers of the platforms. This first risk should not exceed .5 as the teams running the platform are supposed to respond to questions in a timely manner. The second area is during our implementation phase. During this phase the risk of not being able to fully implement our experiments is of concern. This risk can be mitigated by contacting the teams in charge of the platforms and working together to come up with solutions.

## 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.

<b>Task/Goal</b>	<b>Description/Explanation</b>	<b>Estimated Person-Hours</b>
Project Planning	Define the scope, objectives, timeline, dilemmas, roadblocks, and requirements	40 hours
Research & Analysis	Research various platforms, gather information, and analyze their capabilities	80 hours
Analysis & Comparison	Take the research and begin to compare and assess the strengths & weaknesses	60 hours
Report Writing	Take our final research and compile it into a research paper for publishing	210 hours
Implementation	Begin to create introductory labs based on the platforms researched	120 hours
Testing	Conduct various tests, collect data, and ensure the project description is met	30 hours
Documentation	Create student-friendly lab documentation to accompany the implementation	60 hours
Deployment	Release and publish research, labs, and other activities	40 hours
		<b>640 Hours</b>

## 2.7 OTHER RESOURCE REQUIREMENTS

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

For our project we do not have any major resources required to complete our project. Our project is heavy on research and collecting data and information. Your standard resources include internet, computer, and time. Platform resources and costs.