

Link to my GitHub:

[Mckennah's GitHub](#)

Notes on GitHub Training:

- SRS is the Software Requirements Specification.
- It outlines the basic architectural elements, scenarios, customer needs, and user roles.
- Project Development: summary of project tasking, your accountability, your performance, and your future plans.
- Story: Who is the stakeholder, What is the requirement, Where is the architecture, How is the design, and When is the management.
- Epic stories are a grouping of stories that affect a feature.
- MVC known as Model View Controller, the architecture is going to be based on this.
- Frontend is the view and backend is the model and controller.
- Repositories are great for updating and managing your code. Uses CRUD.
- There is a main and there are branches that are copies of the main which helps keep us from overwriting our classmates' code by accident.
- SCI, ECP, SCCB, DO, and CCC. Remember these acronyms.
- We document what we do to help the programmer understand what we did in the code.
- ACID - Atomic, Consistent, Isolated, and Durable

To summarize we learned more about the documents we need to write for the project along with the basic architectural structure.

Sprint Meeting Minutes:

- ➔ We meet as a team on Tuesday and Wednesday of the first week to discuss what IDE we think would fit best for the project so far. Which is IntelliJ. And we talked about other tools as well. We also decided to meet later in the second week so all of us could make it.
- ➔ The second week's meetings have been more sporadic because people still have things at the new time but we have been recording the meetings for those who can't make it. So we have met nearly every day so far this week to work on the videos that show how to get and install IntelliJ and to work on our report document.

Bi-Weekly Status Report

Name: McKennah Palmer

Project Title: Acorn Nursery

Overall Status: Still on track

Summary of Project tasking

Include the following:

- **Meeting attended and minutes:** I attended all the class meetings for the last two weeks. All 4 were about an hour. I also attended nearly all of the RunTime Team meetings. We had 2 in the first week and a meeting a day practically in the second week. They are now lasting 1 hour and 30 minutes depending on what we are discussing.
- **Training notes Summary:** The first two meetings for class can be summed up in these notes. We learned more about the documents we are going to write for the project along with the basic architectural structure or MVC. The RunTime Team has focussed on creating instructions and filling out our document.
- **Stories assigned:** Think about what would work as an IDE for the project and make sure it matches the requirements. Work on the SRS document with the team and finish before the end of these 2 weeks. A personal story: Make it to as many meetings as possible to help the team stay updated. I seem to be the only Runtime Team member who is able to make it to the second class meeting of the week.
- **Project artifacts and notes identified in project tasking:** Work on the SRS document for the Runtime SW Team.

Include a self-evaluation of the rubric for the project development's rubric

Number of Hours

- **Number of Hours worked this week (expectation: 7-9 hours):** 7 hours the first week and 8 hours the second week.

- **Total Number of Hours this semester (anticipated 120/semester):** 25 hours so far this semester.

Accomplishments (since the last status report), include completed stories.

- **Number of Story Points completed:** 2
- **Number of Stories completed:** 3
- **Personal Velocity (Points/Stories):** 1
- **Project Velocity (Teams Points/Teams Stories):** 2
- **Earned Value (Actual Points/Hour - Estimated Points/Hour):** 3/15
- **Performance**

- ◆ **Last Week's Performance Average (Actual Total Points/ Total Hours):**
0

- ◆ **This Week's Performance (Actual Points/Hours):** 3/15

- **Challenges encountered and resolutions found (since the last status report):** The biggest challenge this week has been finding a time when we all can meet. We have partly fixed it by changing the time to 6 pm but we have yet to decide the days and how many times a week. I know that some in my team think that we have to meet every day, but it isn't possible with our schedules. So I suggest 2-3 times a week. Preferably on either Tuesdays, Wednesdays, or Fridays.

- **Plans/Goals/Tasking for the coming week/sprint.**

- ◆ Pick a team for the actual coding part of the project.
 - ◆ Create a basic outline of Acorn Nursery in GitHub.
 - ◆ Test branches and merging and make sure everyone is on the same page.
 - ◆ **Follow-up/Reporting**
 - No other Comments.