

|  |
| --- |
| Agile Methods for SW Development  2018 |
|  |
| September 18  Project 4: Project Sprint Report  Authored by: Ayana Perry, David Hubin & Rakshith Varadaraju |

Project Sprint Report:

1. Team page

a. Team member names, email, and GitHub user name



b. GitHub repo URL



2. Backlog page:



a. Enough stories for whole project



b. All stories valid

c. Consistent with other pages

3. Burndown page:

a. Accurate counts of stories

b. Accurate velocities

c. Consistent with other pages

4. Sprint page for just-completed sprint

a. Actual sizes consistent with source code

b. Updated Sprint Review results at bottom of page

i. No references to pizza that demonstrates that the Sprint Review hasn’t been updated

5. Sprint page for next sprint



1. Stories chosen, assigned and estimated



1. Reasonable division of labor

* Each team member works on their own user stories individually.

GEDCOM test input file

Each team should include at least one GEDCOM test file that demonstrates all user

stories from the current and all previous user stories. Recall that you’re delivering a

product and each sprint releases new features but you also need to demonstrate

that the old features still work as well.



GEDCOM test output file

The output (see below for an example) should include:

1. Summary, with headers, of all individuals

2. Summary, with headers, of all families

3. Demonstration of user stories from current and all previous sprints. Each

test result should clearly identify the User Story name and include enough

details to allow a user to understand and resolve the problem. E.g.

ERROR: INDIVIDUAL: US01: 37: I03: Birthday 2060-04-13 occurs in the future

Note that this distinguishes between errors and anomalies, identifies the user

story (US01), the individual (I03), and the nature of the error.

Sprint Checklist

Team GitHub Repository

The GitHub repository should include at least the following:

1. Source code

2. GEDCOM test files

3. GEDCOM test outputs

Sample

# Purpose: Perform first sprint Assignment:

From now on all project work will be done in teams. Only one member of the team needs to submit work for each assignment.

There are 4 parts to this assignment:

1. Executing Sprint 1
2. Demonstrating the results of Sprint 1
3. Reviewing Sprint 1
4. Planning Sprint 2

**1. Executing Sprint 1:**

1. Perform the tasks and complete the user stories as identified for this sprint. Record actual sizes and times and copy those where appropriate on the Project Backlog page.
2. Update the Burndown sheet and calculate your velocities for the sprint.
3. Make sure that you create a line-oriented application that does something for demonstration to the customer.
4. Make sure that you commit the completed version of your source code to your GitHub repository.

**2. Demonstrating Sprint 1:**

1. Execute your system on the acceptance test file.
2. Capture the output in a separate Test Results file. This is the demonstration for the customer.

**3. Reviewing Sprint 1:**

1. At a team meeting review the results of the sprint.
2. Make 2 lists:
   * things you want to keep doing
   * things that you want to avoid doing in the future
3. Record the lists at the bottom of the Sprint1 sheet

**4. Planning Sprint 2:**

1. At a team meeting decide which user stories will be completed in the next sprint.
2. Update the Backlog sheet and the Sprint 2 sheet to show this, including ownership and estimates of all tasks and user stories for Sprint 2.

Submit your Project Sprint Report, your acceptance test file and your Test Results from Sprint 1 as a single zipped archive file to the Project 04 Canvas assignment. Make sure that the name of your GitHub repository is included on the Team sheet of your Project Sprint Report. The instructor needs this to review your source code.

Working effectively on a team is a critical skill.   Please list all of the members of your team, including yourself, and assign a score, from 0-100, on each person's contributions to the team for Sprint 1 (Project 04).   The sum of the scores for all members should add up to 100 based on your assessment of each member's contributions.

**Ayana Perry: 30, David Hubin: 50, Rakshith Varadaraju: 20.**

You cannot assign a score of 100 to yourself!

**Deliverable**: N names and N numbers adding up to 100 for each of the N members of your team.

Your score on this assignment will consist of a combination of your teammate's assessment of your contributions to the team and your code contributions in GitHub.