IST 303 Team Project

Spring 2024

Due: Various. See individual parts for instructions and what to submit.
Assignment type: Team

PROJECT REQUIREMENTS

In this project you are required to implement a web application in Python 3 using Flask. The app will interface the user to an underlying data store.

Beyond those requirements, you and your team are free to choose your own domains to create a solution for, as well as what data structures to use.

There are four parts to the project: A, B, C, and D. Below is a brief overview of what is due for each part.

- **Part A (60 pts)** : Github repository URL for your project. There should be a README in the repo with relevant documentation i.e. team members, stakeholders, requirements etc.
- **Part B (60 pts)** : Github repository URL for your project. Revise your README to reflect your answers to questions posed in Part B.
- **Part C (100 pts)** : Presentation to class, including demo of the project (Milestone 1.0).
- **Part D (160 pts; 100 for presentation and 60 for final codebase/commit history)** : Presentation to class, including demo of the project (Milestone 2.0). On Canvas, submit the URL of your repo containing your final codebase.

Part A

1. Meet with other members of your assigned team (teams are instructor-assigned with the goal of optimizing the mix among coding, GUI, and database skills).
2. Come up with a concept for your application.
3. Identify all the relevant project stakeholders.
4. Create an initial set of project requirements expressed as user stories. User stories must have estimates of completion times.
5. Meet with the instructor to discuss the concept and initial set of user stories.

> DUE: Github repository URL for your project. There should be a README in the repo with relevant documentation i.e. team members, stakeholders, requirements etc.

Part B

1. Decompose your user stories into tasks.
2. Outline what features will be in Milestone 1.0 of your project.
3. Build the iterations (at most 2) that will compose your Milestone 1.0. Record the total days of work and the time it will take for your team to complete that work. *Be sure to account for velocity before breaking into iterations.*
4. Allocate tasks to each team member and record the allocation.
5. Create a burn down chart for monitoring your team's progress.

6. Provide evidence in your github repository that you are meeting for periodic stand up meetings - your team should ideally meet at least twice a week.
7. Ensure that your development and testing environment is set up. Be sure to have some working functional (however rudimentary) and test code in your repository.

> DUE: Github repository URL for your project. Revise your README to reflect your answers to questions posed in Part B.

Part C

1. Present to your class Milestone 1.0 of your project. Each team member must present a portion of the team presentation.
2. You must present working code. In addition, your presentation must include:
    - What the code does
    - How it fulfills the user stories
    - Documentation of the use of agile methods
    - explanation of how the code was tested
3. Explain what remains to be done to accomplish Milestone 2.0.
4. Upload any presentation materials in your repository in a folder named "Part C".

> DUE: Presentation to class, including demo of the project.

Part D

1. Present to your class Milestone 2.0 of your project. Each team member must present a portion of the team presentation.
2. You must present working code. In addition, your presentation must include:
    - Succinct and clear explanation of what the Milestone 2.0 does and how it fulfills the user stories
    - Documentation of the use of agile methods, including a final burndown chart
    - explanation of how the code was tested, including details on the level of code coverage
    - highlight key successes and failures, including lessons learned about software development
3. Upload any presentation materials to your github repository in a folder named "Part D".
4. Submission of your final code repository. The final commit timestamp in your team repository must be prior to the assignment due date. Your README file must contain:
    - ALL the details on how to run the program, how to test it, and how to report the test coverage.
    - the three most important things you learned about software development working on your project.

> DUE: Presentation to class, including demo of the project at Milestone 2.0. On Canvas, submit the URL of your repo.

RESOURCES

- pytest-cov plugin for pytest
- flask documentation