Oregon State University
Ecampus

CS361: Assignment 6: Sprint 2 Plan (for Microservice A)

Overview

Plan the microservice you'll make for your teammate(s). That includes defining how to **request** and **receive data** from the microservice.

Instructions

Complete each item below by replacing the highlighted text (**Usability note**: double-click the text to select it).

1) What is your **Sprint Goal**? (e.g., fully implement the spell-checker microservice) The Sprint Goal must clearly communicate what the microservice will do.

Sprint Goal: I will create microservices that write and read txt files, that will be used in our project. It will be able to generate a random number, that will then be used to send to the main file that will determine if j= {1,4,7) plays rock, if j= {3,6,9} plays scissor, and if j= {2,5,7}, plays paper.

2) Define **at least three user stories** for this Sprint. Provide your user stories and their functional and non-functional acceptance criteria (and associated quality attributes).

Requirements for Microservice A:

- You must implement at least three user stories.
- Each user story must have a name.
- Each user story must use the "As a… I want to... so that…" format.
- Each user story must have at least one functional acceptance criterion.
- All functional acceptance criteria must use the "Given… when… then…" format.
- At least one of the user stories must have an associated quality attribute and non-functional acceptance criterion.

First user story

(Front of index card) Random number generator As a user, I want to input a string "run", so that it will be read and create a random number to send to back to user to play rock, paper, and scissor.
(Back of index card) **Acceptance criteria**

Functional requirements

- Given that the function gets the result “run”, it will then start a random number generator to write and send to user to play rock, paper, and scissor.

Quality attributes & Non-functional requirements

- Testability: If a string “run” is read, then it will start the software to generate a random number.

Second user story

(Front of index card)

Reading strings in text file

As a user, I want to be able to read strings in a text file, so that it can set as a variable to run the current program.

(Back of index card)

Acceptance criteria

Functional requirements

- Given a text file exists, it can print out what is in the file.

Quality attributes & Non-functional requirements

- Reliability: The software will be able to constantly read what is in a text file, so that it will be able to re-write new strings inside.

Third user story

(Front of index card)

Writing new strings in text file

As a user, I want to be able to write strings in a text file, so that it will be used in other programs to continue the function of the project.

(Back of index card)

Acceptance criteria

Functional requirements

- Given when we can read a text file and set our new variable, it will generate a random number, and then write the generated number in the text file to replace the previous string inside.

Quality attributes & Non-functional requirements

- Usability: If the program can open and read the text file, it can write in it to replace what is in the file.

3) What kind of **communication pipe** will the microservice use? (e.g., text files, REST API)

Text files

4) How will other programs be able to **request data** from the microservice? If possible, give an example call using pseudocode or actual code.

We will be using txt files to communicate the implementation of our program. Others will request by writing in the text file a specific string, then it will go through the other program then come back with results and that result will be used in the main program to display its result.

5) How will other programs be able to **receive data** from the microservice?

We will be using txt files to communicate the implementation of our program. While writing and reading text files, it will allow different programs to use what is in the file to run their program, until it is complete.

While True:

Open text file and reads

Sets variable to what is in file

Closes file

If variable equal “run” start a random number generator

Opens same file, but instead of reading, it writes the new value (random number)

Closes file

This would be a good time to make a new repository to house the microservice.

Submission

PDF or Word format via Canvas.

You must follow instructions at Modules > “Attach a Document to "Text Entry" Field”.

Grading

You are responsible for satisfying all criteria listed in the Canvas rubric for this assignment. You will be able to revise this assignment if you miss points.

Questions?

Please ask via Ed so that others can benefit from the answer.