

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.

|  |
| --- |
| *Fully develop a word analyzer microservice to get the length of a word, number of vowels, and number of repeated letters.* |

1. 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)  *Word Length*  As a user, I want to be able to put a command with a three-letter word in a text file format so that I get the length of the word (3). |
| (Back of index card)  **Acceptance criteria**  Functional requirements   * *Given a text file containing the command “length.dog” when the system processes a text file, then the system should return the length of the word as ‘three’.*   Quality attributes & Non-functional requirements   * *Usability: the system program should effectively read the text file provided by the user and return the results efficiently while the user enjoys the experience.* |

**Second user story**

|  |
| --- |
| (Front of index card)  *Number of Vowels*  As a user, I want to be able to use the command “vowels.xxx’ as ‘xxx’ corresponds to any word in a text file format so that I get the number of vowels. |
| (Back of index card)  **Acceptance criteria**  Functional requirements   * *Given a text file containing the command* "vowels,cat" when the system reads my text file, then the program should return the number of vowels (one).   Quality attributes & Non-functional requirements   * *Correctness: the system program must provide the correct results to the end user, as with each user input it must produce an output satisfying the specification and expected behavior of the algorithm.* |

**Third user story**

|  |
| --- |
| (Front of index card)  *Repeated letters*  As a user I want to be able put a command in a text file format so that I get the number of repeated letters of any word. |
| (Back of index card)  **Acceptance criteria**  Functional requirements   * *Given the user puts a command like “repeats.beachball” in a text file when the program runs this command then the system should return the number of repeated letters (three).*   Quality attributes & Non-functional requirements   * *Precision: the system program must reflect an accurate numerical quantity of the number of digits used to specify or express a value.* |

## Take a screenshot that shows you’ve moved these user stories into a Sprint Backlog in your task management system.

|  |
| --- |
|  |

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

|  |
| --- |
| *Text files and RESTful API* |

1. How will other programs be able to **request data** from the microservice and what request parameters will be required? Give an example call using pseudocode or actual code.

|  |
| --- |
| Request parameters (included parameters in the request call)   * **Request** with JSON body parameters using list of strings   A list of commands, where each command consists of an operation and a word, separated by a comma. |
| Detailed description (how data will be requested)   * Endpoint: POST /analyze-file * Request: Upload a .txt file containing commands * Response: JSON with results |
| Example call (pseudocode or actual code)  Example Python Request:  import requests  url = "http://localhost:5000/analyze-file"  files = {"file": open("commands.txt", "rb")}  response = requests.post(url, files=files)  print(response.json())  Example Input File (commands.txt):  length,dog  vowels,cat  repeats,beachball |

1. How will other programs be able to **receive data** from the microservice and what data will the microservice provide? Give an example call using pseudocode or actual code.

|  |
| --- |
| The Word Analyzer Microservice will use RESTful API calls to provide responses based on the commands given in text files. |
| Detailed description (how data will be provided)   * Length of the word (e.g., "length,dog" → 3) * Number of vowels (e.g., "vowels,cat" → 1) * Number of repeated letters (e.g., "repeats,beachball" → 3)} |
| Example call (pseudocode or actual code)  Pseudocode (Client making a request)  Open file "input.txt"  Read command from file (e.g., "length,dog")  Send a POST request to "http://127.0.0.1:5000/analyze" with data {"command": "length", "word": "dog"}  Receive and print response (e.g., {"result": 3})  API Call  POST /analyze HTTP/1.1  Host: 127.0.0.1:5000  Content-Type: application/json  {  "command": "length",  "word": "dog"  }  Output  {  "result": 3  } |

**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.

# Questions?

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