# **PRACTICLE TEST FOR SOFTWARE DEVELOPMENT POSITION**

**Name: Peter Muturi Wairimu**

**ID no: 36491048**

**Github Profile:** [muturipeter (github.com)](https://github.com/muturipeter) /muturipeter

**Question attempted: Q 2**

Contents

[**PRACTICLE TEST FOR SOFTWARE DEVELOPMENT POSITION** 1](#_Toc116385640)

[**SOLUTION**. 1](#_Toc116385641)

[**SOFTWARE DEVELOPMENT METHODOLOGY USED** 1](#_Toc116385642)

[**APPROACH:** 1](#_Toc116385643)

[**FEATURES AND FUNCTIONALITIES:** 2](#_Toc116385644)

[**THE CODE:** 2](#_Toc116385645)

[**SIMPLE USER INTERFACE AND NAVIGATION** 3](#_Toc116385646)

[**DATABASE:** 4](#_Toc116385647)

## **SOLUTION**.

**SOFTWARE DEVELOPMENT METHODOLOGY USED:** AGILE SOFTWARE DEVELOPMENT METHODOLOGIES.

**Reason:** Agile is a test based development methodology and this would be relevant to the solution I’m offering since the program will be subject to planning, coding, testing & debugging, implementation and launch of each and every phase of the project.

**Tool used:** In my experience I would start the project by defining the **Sprints, stories and tasks** using a software project management tool such as [**JIRA**](http://www.jira.com)**.**

### **APPROACH:**

This specific problem can be solved using simple custom code in different languages such as *Java*  and *python.*

SD Language to use: PYTHON

Data Structure and algorithm: Arrays and dictionary as used in Python

**Reason:**

* Python would perform more efficiently in achieving the (regex) match from the dictionary since it just needs to import some pre-defined variables.
* Python is also compatible with the Data structure in use for the project.
* Well, python will achieve the goal with lesser code and lesser libraries as compared to other languages and is easy too.

### **FEATURES AND FUNCTIONALITIES:**

**System requirements:**

1. Python version 3:9 +
2. IDE –In this case Pycharm but I would prefer to use Visual Studio.
3. Working Internet connection when launching the system for the first time.
4. Any Operating system be it Mac, Linux or Windows
5. Figma or any other design tool for UI and user navigation

**Features and functionality of the system;**

1. User input.
2. User Output
3. User registration
4. Database connection

## **THE CODE:**

# This is a sample Python code snipet to store and match the custom 'regex' strings  
#Developer: Peter Muturi Wairimu  
#NB:No interpreter was found on the pre-installed IDE  
#Date last git commit:11/10/2022  
#Start be importing libraries in use  
import re  
dict= {}  
dict[re.compile('actionname (\d+) (\d+)')] = method  
dict[re.compile('differentaction (\w+) (\w+)')] = appropriate\_method  
 def alphabet (str) args[x]:  
  
#Create Variable X and assign it a value 'a'  
x='a'  
  
def execute\_method\_for(str):  
 #Match each regex on the string  
 if  
 matches = (  
 (regex.match(str), f) for regex, f in dict.iteritems() #items stored as dictionarries on the project Data structure  
 )  
 else  
  
 #Filter out empty matches,  
 matches = (  
 (match.groups(), f) for match, f in matches if match is not None  
 )  
 #Get to apply the functions above  
 for args, f in matches:  
 f(\*args)

#With more time and resources I would have linked the program with a database, and included user input so that the alphabet is picked from the user

*The exported database is located at* ***C:\Users\Bridging\Desktop\Peter Muturi Wairimu\code***

## **SIMPLE USER INTERFACE AND NAVIGATION**

Output (Enter any alphabet

Results :

SUBMIT

**Current Page / Tab Title**

## **DATABASE:**

Managed to create just a simple relational database with 2 tables so that in the program the user can track their interactivity with the system, store results as well and query them for future references.

DB, INFO;

Database name: peter\_muturi\_wairimu

User:’root’

Password: ‘ ’

*The exported database is located at* ***C:\Users\Bridging\Desktop\Peter Muturi Wairimu***