PES UNIVERSITY
AUGUST – DECEMBER 2022 SEMESTER 5
SOFTWARE ENGINEERING LAB TASKS

RAKSHIKA S – PES2UG20CS264
SHRAVYA U – PES2UG20CS326

Problem Statement – 1b: For Scrum - Agile projects

Assess the quality of your project by maintaining a sprint burndown, calculating the team velocity metric, throughput and cycle time. Further, answer the following questions:

1) While working on your project, the go to approach to debug is using print statements. Could you include these statements to highlight the location and values at the current point in the project?
2) Could you build two valid and invalid test cases for any two functions in your project?

1)Yes, we can use print statements for debugging. We can also use pdb for debugging.

Steps involved:

- Using print statements to examine the behavior of the code. This is print statement debugging.
- Find which area of the code is causing the problem by printing line numbers (and file names) in areas that we suspect may be the source of the bug.
- When the correct general area is found, read the code around that line to get an idea of which variables are important to the behavior of that code. Print the values of those variables.
- Working backwards to trace the source of the values of those variables.

Debugging using print statements:

atm.py

```python
import getpass
#import pdb; pdb.set_trace()
users = ['user1', 'user2', 'user3']
pins = ['1234', '2222', '3333']
amounts = [1000, 2000, 3000]
count = 0

while True:
    user = input('\nENTER USER NAME: ')
    user = user.lower()
    print("Debugging: atm.py, line 12")
    print("Value of user in line 11:",user)
    print("Valid users:",users)
    if user in users:
        if user == users[0]:
            n = 0
        elif user == users[1]:
            n = 1
        else:
            n = 2
        break
    else:
        print('INVALID USERNAME')
```

```
PROBLEMS  OUTPUT  TERMINAL  JUPYTER  DEBUG CONSOLE                powershell

PS C:\Users\Shravya U\Documents\sem 5\SE\code> python atm.py

ENTER USER NAME: user5
Debugging: atm.py, line 12
Value of user in line 11: user5
Valid users: ['user1', 'user2', 'user3']
INVALID USERNAME

ENTER USER NAME: user1
Debugging: atm.py, line 12
Value of user in line 11: user1
Valid users: ['user1', 'user2', 'user3']
PLEASE ENTER PIN:
LOGIN SUCCESFUL, CONTINUE
User1 welcome to ATM
**************************
----------ATM SYSTEM-----------
SELECT FROM FOLLOWING OPTIONS:
Statement__(S)
Withdraw___(W)
Lodgement__(L)
Change PIN_(P)
Quit_______(Q)
: Q
PS C:\Users\Shravya U\Documents\sem 5\SE\code>
```

Pdb:

2)

Login module:

Test case 1: Check valid username and password, login must be successful.

Test case 2: Check invalid username and password, login must fail.

Withdrawal module:

Test case 1: Check if account balance is more than or equal to withdrawal amount, withdrawal must be successful and get deducted from the account balance.

Test case 2: Check if account balance is less than withdrawal amount, withdrawal must not be successful.

Problem Statement – 4:

Let's assume you have received funding to launch your project as a start-up. Being at the nascent stage of development processes, you have been tagged under the "Initial" maturity level. Your task is to brainstorm and come up with at least 2-3 new functionality or ways to improve the quality of your project and attain higher levels of maturity according to the CMM model.

The ATM simulator that we have built is in the initial level according to the CMM model. We have received funding for this project and are willing to improvise some of the functionalities of it so that it becomes better and can be at a higher level in the CMM model.
Current level of the ATM simulator (level 1 – initial):

- No key performance areas are defined.
- Processes followed are immature and are not well defined.
- Unstable environment for further development.

To take it to level 2 (repeatable – focuses on project management) the following steps can be taken:

- Project Planning can be done in a better way. It includes defining resources required, goals, constraints, etc. for the project. It presents a detailed plan to be followed systematically for the successful completion of the ATM simulator.
- Configuration Management- The focus is on maintaining the performance of the software including all its components.
- Requirements Management- It includes the management of customer reviews and feedback which result in some changes in the requirement set. It also consists of accommodation of those modified requirements.
- Software Quality Assurance- It guarantees a good quality website by following certain rules and quality standard guidelines while developing. It will ensure that the simulator works with good site health, authority score etc.

To take it to level 3 (Well defined - organized assets) the following steps can be taken:

- Peer Reviews- In this method, defects are removed by using a number of review methods like walkthroughs, inspections, buddy checks, etc.

- Intergroup Coordination- It consists of planned interactions between different development teams to ensure efficient and proper fulfilment of customer needs.
- Organization Process Definition- Its key focus is on the development and maintenance of the standard development processes.
- Organization Process Focus- It includes activities and practices that should be followed to improve the process capabilities of an organization.
- Training Programs- It focuses on the enhancement of knowledge and skills of the team members including the developers and ensuring an increase in work efficiency.

To take it to level 4 (Analyzed, improved and managed - quantitative control) we can follow the following steps:

- Software Quality Management- It includes the establishment of plans and strategies to develop quantitative analysis and understanding of the product's quality.
- Quantitative data for our ATM simulator is collected indirectly in the form of predefined actions taken by users, such as task metrics time on task, success/fail rate, etc.
- To reach level 4, we see to it that the metrics are improved by making use of the speed score tools and improving page load time.
- The main functionality to improve is the speed. So, we will see to it that the speed score of the simulator is 90+
- We can improve it by: using a Content Delivery Network, Improving Server Response Time, Using SEMrush Site Audit as an Alternative to Page Speed Insights.
- We have to keep track of the Mean Time to Failure (MTTF), Mean Time to Repair (MTTR), Mean Time Between Failure (MTBR) etc. for the simulator to be reliable.
- The simulator should be very secure. We should make sure that the Exposure Factor (EF) will be maintained.