Nevena Young

Software Test Automation& QA

Project 2

October 16th, 2022

**To what extent was your approach aligned to the software requirements?**

This project was tedious and challenging, but I made it through! My approach was aligned with the software requirements, except for the Task class... I did not implement the logic for the requirements:

public Task(String id, String name, String description) {  
 this.id = id;  
 this.name = name;  
 this.description = description;  
}  
  
  
public void setId(String id) {  
 this.id = id;  
}  
  
public void setName(String name) {  
 this.name = name;  
}  
  
  
public void setDescription(String description) {  
 this.description = description;  
}

This is the logic for the contact class:

public Contact(String id,String first,String last,String number,String addy){  
 if(id.length() <= 10 && id != null) {  
 this.contactID = id;  
 }  
 this.setFirstName(first);  
 this.setLastName(last);  
 this.setPhoneNumber(number);  
 this.setAddress(addy);  
}

This is the logic for the appointment class:

public Appointment(String appointmentId, Date appointmentDate, String description) {  
 if(appointmentId == null || appointmentId.length() > 10) {  
 throw new IllegalArgumentException("Invalid input");  
 }  
 if(appointmentDate == null || appointmentDate.before(new Date())) {  
 throw new IllegalArgumentException("Invalid date");  
 }  
 if(description == null || description.length() > 50) {  
 throw new IllegalArgumentException("Invalid description");  
 }  
 this.setAppointmentId(appointmentId);  
 this.setAppointmentDate(appointmentDate);  
 this.setDescription(description);  
}

**How did you ensure that your code was efficient?**

The way I ensured the code was efficient was to test the code. This was the most effective way to ensure its quality. The overall goal is to ensure the inputs would validate the unit tests run.

**What were the software testing techniques that you employed in this project?**

Two techniques I used in this project were Whitebox testing and Blackbox testing. WBT allows the developer to test the current code in the early stages, even before the interface is completed. BBT was used when I didn’t know the inner workings of the system.

My mindset while working on this project was “keep the requirements in mind”. I was doing my best to keep the requirements in mind, so I don’t forget any.... But this caused me to forget about logic within the code. I am new scripting a whole code on my own, because previous classes would have most of it coded out except for what I needed to add in. This was something I need to experience, since my future is in tech. This is how biases are made and how they can affect a developer while coding.

Cutting corners while creating a code is poor practice and can be the reason why a code isn’t properly working. While scripting a code or testing one, it is not good to cut corners. This could be the very reason why an error cannot be located or kept accuring. To avoid these, take your time and have patience.