SAN JOSÉ STATE UNIVERSITY

CMPE 202 - Team Project

Project Group # 3

Team Name : Magicians

Week #3

CS Unplugged Activity : Error Detection

Team Member Name	Section	GitHub ID
Miao Shi	3	MiaoS
Chen Shi	3	Chen202
Carlos Martinez	4	carlo379
Jayam Malviya	4	Jayam-Malviya
Prateek Sharma	3	prateeksharmamay

GitHub Repository	**https://github.com/carlo379/Magicians**
Task Board	**https://waffle.io/carlo379/Magicians**
Kanban CFD Google Sheet:	https://drive.google.com/open?id=1K3FaFcE2PIlkH0KE9A4LqHVsYzxT27rZh8WfAZvw-gU

Team member: Miao Shi

XP core value: Communication

In the past week, the implementation of each class doesn't not go smooth. Some classes are dependent on one major class, and problems showing up during the implementation forced us to adjust out design.

Because SVCC is on this weekend, our weekly meeting to a weekday. We negotiated our meeting time in WhatsApp. I noticed the inefficiency of group chat. The negotiation went on for almost a day. There are a few reasons. First, most people don't check instant message immediately. Response can delay from minutes to hours. Once a member has a problem with suggest meeting time but he/she doesn't respond right away; other members may already think the meeting time is settled. Second, most people put the short sentences of their first impression missing important information. It can mislead people and give other members bad impression. In our case, one member doesn't want to have meetings on weekdays after this week. However, she didn't write "after this week". All of other members thought she meant this week. Some members were trying to find a meeting time in the early morning or late afternoon and other members just felt it was impossible. Because she didn't relay right away, the misunderstanding went on until the next morning. Therefore, when we write in group we should think through the words make sure that no important details are missing in order to save time on everyone avoid the time for waiting and unnecessary negotiation.

Team member: Chen Shi

XP core value: Courage

In the last week, we made a sketch of the class diagram. In this week, we tried to finalize it for the first version of our game.

We discussed together and decided to have four main classes: Grid, Card, Magician, and Scenario (Curtain). The Grid class will control the arrangement of the cards in a grid. The Card class will be in charge of displaying the cards facing up or down. The Magician will control the magician role. And Scenario class will display an animation of background show.

I was assigned to work on the Scenario class in Greenfoot. Last week, I finished the code to control a curtain moving up or down by pressing “up” or “down” keys. This week, I mainly work on adding a button to start this animation instead of pressing keys. The Scenario class will have Button, FrontCurtain, and BackCurtain three classes at this moment. Each class will have their operations, each of which starts a small animation. All the animations play together on a precise schedule, and the background show will add lots of fun to the game.

In the coming future, I am planning to improve/add more features to this scenario to make this animation process more attractive to kids. I am very excited about this task. I can image that I will be able to make many fun animation games for my son at the end of the course. One’s courage grows when he/she has more skills.

The group did a good job in weekly meeting and discussion. Each member will design a type of diagram as an aid to our coding process. Each member will also start writing code for one of the four classes in the following week. Once we have the code, we will also create test cases for them.

My suggestion to the team meeting is to send a brief written summary of each meeting via email or Whatsapp group chat. So each member can have a clear idea what happened in the meeting. For example, because of SVCC, our meeting was scheduled on weekday evening this week and I had to call in to join others since my husband comes home late every weekday evening and I have kid at home to take care of. This kind of situation could happen to anyone when there is an emergency situation. I found that a written summary with the key points/conclusions of meeting and task assignment list would be very helpful for everyone to use as a reminder later. I am willing to do this if needed.

Team Member: Carlos Martinez

XP core value: Eliminate Waste

This week was dedicated to designing our application called “Error Detection”. As a team we decided to add backlog items related to app diagrams, class coding and app test cases. This helps the team manage their time by keeping the task visible in the dashboard as a method to let every member know what’s next in the development process. Since we discussed the features of our application we all agree that the next steps were to create the different diagrams that will help us in the development process. The diagrams we identified were the Use Case diagram, Sequence diagram, Class Diagram and Object Diagram. These represent the main designs that will aid during the development process and help us avoid wasted time during coding. We also decided that after the first iteration of our design we should start coding the different classes, that were divided between the team members. This process will speed up the development by providing real experience and feedback in a short amount of time.

Another important aspect, that we agreed on, was that we should start creating test cases at the same time as we created the code for the main classes of the application. This process will ensure that we eliminate waste by finding bugs early in the development process and by notifying the developers of new code that will break the app.

Our work this week was very important since we start seeing our application as a series of objects interacting with each other. We defined the different messages that each object needed to send and the different attributes and actions that they could perform. Little by little we were able to define the responsibilities and capabilities of all the actors in our program and have a clearer idea of the path we needed to follow to complete our project.

Team Member : Jayam Malviya

XP core value : Simplicity

In third week we all have finalized upon the class, sequence, object & use Case diagrams. One of the problems we faced in this week was team members were unclear about how other classes we working & what methods they were exposing to overcome this we have to sit down and refactor some parts of our class diagrams from previous week in this week's meeting. Also after finalizing the class class diagram we were clear to go ahead & develop other diagrams for our "Error Detection" activity. While refactoring our class diagram we not only dropped one dependency from previous version but also renamed one of our entity class to resemble the actual activity. This exercise also showed the importance of refactoring and also that refactoring is not limited to code, but when applied to design can help in exploring and overcoming design flaws.

Another more important part that we refactored this week was our greenfoot project. We refactored the skeleton of all classes that are required in our activity by renaming & making the class's each property & method resemble the finalized class diagram. We are sure this would help reduce dependency among the among team members as they can refer the diagrams to overcome their doubts.

By whole and large this week was a good refactoring exercise week and as a result our diagrams are very simple and clear. We are confident that with our base clear and ready we can start putting our code to work and add the functionality.

Team Member: Prateek Sharma

XP core value: Feedback

For our meeting for the third week of project, I reviewed our progress thoroughly. As a result, I found few inconsistencies in our structure. In the meeting for third week, I requested the team to start with the feedback session.

The session proved to be very fruitful as we were able to figure out some of the flaws during it. First of all, we found out that there are some functions missing in Magician class and Grid class. We finalized to add trick(), where we will perform the actual magic. With the discussion, we agreed to change name of class Scenario to Curtain as previous name was leading to wrong interpretation of its activity. Moreover, we have added notifyGrid() to Curtain class to resolve the inconsistency between it and Grid class.

Additionally, we discussed about other UML diagrams and distributed the responsibility for drawing them. Due to some major changes in the classes structure, we have restructured our class diagram and drawn other diagrams according to that. I have created Sequence Diagram where I have depicted all the cases and represented them as a sequence while naming each function which will responsible for it.

As we have bolstered our foundation with the designs and feedbacks, we will be continuing with pending tasks of creating algorithms for the functions of classes assigned to each of us. Later we will write test cases for each of our classes to test them as a unit and then we are planning to integrate the classes with each other.