SAN JOSÉ STATE UNIVERSITY

CMPE 202 - Team Project

Project Group # 3

Team Name : Magicians

Week #5

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**
Sprint Task Sheet:	https://drive.google.com/open?id=1JN3y5G12EQSm2q9p0jQpo4-95bcFwT67hmc98oagL9k

Team member: Miao Shi

XP core value: Communication

In the past week, our team was working on improving the user experience. We added better looking buttons, cards, and backgrounds, and roughly designed demo view and theory view, which we hope can help players to understand where and how our “computer trick”, error detection, is used. I worked on button graphic design and worlds navigation.

Our communication keeps getting smoother. In the passing week, whoever in our team updated anything in master source, he/she requested a pull request from waffle.io and also sent out a message in team chat group with a brief description. Therefore, all member was able to be pull the latest updates in hours. For the graphic design, we communicated a lot trying to make sure each scene in our project has the same style. Since we chose our team name as “Magician”, we picked our game theme to be “magic poker card tricks”.

In the most recent weekly meeting, we brainstormed the several ways for apply multi-player functionality to current game and their possible implementation solutions. In-person meeting definitely expanded our imagination and helped us to discovered our weakness.

Good communication now is making good progress to our team; project workflow is transparence to all the team members; and all members understand the project in the same stage.

Team member: Chen Shi

XP core value: Courage

This was an exciting week. The grid class, magician class and card class are now ready to use. We made the first integration of our code. The curtains and the card grid can work together now! Every time the curtain moves down and hides the card grid, one of the cards will be flipped randomly, which is exactly what we expected! I worked on integration of code and the design of the user interfaces for Theory world. Next week I will be working on the navigation function within Theory world. We are so encouraged by the progress of the team!

With the growing courage in every team member's heart, we started the part 2 of the project, which is Scrum method based. One big difference between Kanban and Scrum is that Scrum has sprints. In Kanban, we just plan new tasks and move them forward according to our progress, with a limitation of WIP. But in Scrum, the tasks are selected into a sprint first. Then we focus on delivering this sprint. In industry, it always takes years for a group to fully master Scrum method. We have so little time comparing with that. The biggest problem is that even we know the principles of Scrum and even we feel that we are practicing Scrum by following those principles, we are probably still making mistakes such as "write lengthy requirements documents, engage in waterfall-like issues, follow a plan instead of responding to change, and focus on delivery dates instead of product quality" (Rodríguez et al, 2016) without noticing it. Meanwhile, we are overwhelmed by tons of new software development tools thrown to us since last Saturday: Docker hub and cloud, Restlet, Jackson, JSON, design patterns… We have to learn how to use all of them as soon as possible so that we can submit a nice final demo.

With all those obstacles ahead, we can not let our courage blind our eyesight. We need to work harder to obtain as much as possible from this short and fast-paced Scrum learning experience.

Reference:

Rodríguez, G., Soria, Á. & Campo, M. (2016). Measuring the Impact of Agile Coaching on Students' Performance. IEEE Transactions on education, 59(3), 202-209. doi: 10.1109/TE.2015.2506624

Team Member: Carlos Martinez

XP core value: Eliminate Waste

The work performed this week helped us to have a working application with most of the elements laid out and working properly. The core functionality of the application was completed and several test classes were developed to ensure future changes will not break what was achieved to this point. As a team we discovered that multiple people pushing code to a repository can be challenging and produce many conflicts in code. To avoid or reduce these conflicts we decided to implement frecuents commits, with at least one commit per work session. These frequent commit will ensure that we do not wait too long or having too many changes to push, since that can create more conflicts with other people's code. Along with this, we agreed that it is important to pull from the repository before starting any new work session. Doing this will ensure we have the latest code base before adding any work to it. By following these rules, the conflicts in code are reduced, and if occur, they are easier to solve, without creating too much disruption. Using this strategy helped us to reduce waist, by working more efficiently and using our time in coding and not in solving code conflicts.

One important element we worked on was the presentation of our game and how the user was going to interact with it. We decided that a menu with the basic game options was important, in order to have an starting point and a reference point to come back when a game was over. For this, one of the team members developed a menu by creating an additional world; then we linked our game world to one of the options of the menu. This is giving our game a structure that is easier to understand and to follow by new users.

In summary, this week we achieved Important milestones and we planned ahead for the next features that we wanted to include in our game. We used the new burndown spreadsheet to plan ahead for the backlog items and task that we need to complete in the following 4 weeks.

Team Member : Jayam Malviya

XP core value : Simplicity

This week was very productive, we integrated our game end to end. And it was a wonderful demo, in which we ran our game for the first time. It was also very pleasing to see the simple design of the classes that helped us in integrating the code.

Initially we had to struggle a bit while integrating the code, as each team members had their own worlds to test their code. but after spending some time we were able to integrate the whole project. With our core functionality part working and doing some more attractive UI changes we were able to complete the basic part of our game. After demo each team member was very happy and motivated to complete the project.

We also commit our changes into the repository and each team member faced some difficulty after that, team members started getting conflicts on their changes and it was a chaotic moment. We actually learned a lesson from this small mishap. To avoid such problems in future we all together decided that, whoever will commit, will inform the team members about the change. in this way we ll will be sure that we have to take the latest code before pushing our changes. By following this we can be sure that we will have a healthy development master always up & running.

There was also lots of discussion in the team about how the kids are going to learn to play this game. Team members concluded that it will not be feasible to teach the kids the game manually. So we all decided that we have to give some kind of demo part within the game, which students can go through before starting to play the game. In this way we can automate the training part of the game and it will be fun for kids also to learn the game on their own. we also brainstormed on how we are going to achieve this, after lots of brainstorming team decided that we will create separate worlds for the kids to navigate between and in this way they can switch within the game between demoing part and actual training part. On the whole I feel this separation of worlds is a great idea and was an outcome of teams brainstorming. Also this design is fairly simple and too complicated to implement.

Team Member: Prateek Sharma

XP core value: Feedback

The project is taking up a good shape. In the past weeks, we have completed the core of the project. The core included creating the grid with cards, randomly flipping the cards and the most important aspect of our project, the Magic. The functionality of magician to find the flipped card has been completed.

Our strive for perfection ensures us to look for better. We review our work on every meetup and refine our work according to the feedbacks. As a result, we reviewed our implementations and found it to be less interactive with the audience. Their as planned while brainstorming, we planned to have a menu page that offers user to see the demo or start playing the game. At the end of the game, we would be having a button to let user see the flipped card, which will be displayed as a highlighted card.

The other issue we encountered during our feedback session was of code integration. As the team was working closely on modules, there was time when there was inconsistency in the Git commits. Therefore, it was decided that we will follow the basic rule where before starting coding, we will take a Git pull and once finished we will push it on repository so that there is minimum inconsistency.

The upcoming tasks includes hosting the application on Docker container and make the game a multiplayer game. We have to use RESTful web API framework to expose the services that will receive requests from users and all the functionalities are implemented on server. As the project second part has asked to follow the Scrum approach; we have planned the tasks for upcoming four weeks. We have distributed the responsibilities and started working on our next tasks. We will keep monitoring the burndown chart and create our sprint tasks accordingly.