AUREOS

Aureos is the first group of visual development of the UPY that Works both Android and Apple platform for the creation of augmented reality on Unity software. It arised in the middle of a programming class in order to create an outstanding product for getting a high score.

Our methodology to implement our products is based on Kanban methodology from Toyota that use techniques just in time,

The main rules of Kanban are the following three:

**1.- visualize the work:** Kanban is specialized in visualizing the situation of each task; This is done through Post-it, which are some pieces of paper that can be added to a blackboard to divide tasks,

**2.- determine the work limit:** Kanban uses a blackboard as a task manager which is divided into sections with numbers under the name of each task, as one of the main ideas of Kanban is that the work in progress should be limited.

**3. - Measure the time to complete a task:** it is practically the time it takes to complete each task, at that time called lead in time which is done since a request is made until the delivery of the product.

Since that methodology suits us to our necessities, we decided to implement this methodology that doesn’t have many rules and promoting that each member of the group has the same position in the project, nobody is better that other, so in this way we don’t have to meet us at a garage or in the house of someone to code our project instead we use a very useful platform named Meistertask that allows us to divide the tasks – the tasks are some papers used in Kanban to verify the activities that are done or not – and let us take any activity on the virtual board. Meistertask is based on Kanban board but with the difference that is a virtual board that has all the speciation that Kanban implements, Meistertask has three divisions: open (task that are not done), in progress (tasks that are working) and done (task that the team has finished). In each one of those divisions have the papers that allow us realize what activity is not done.

How do we work?

Each of the participants of the project write down on the Meistertask platform the specifications (tasks) that the programming professor wanted (client), then each of us pick one paper on the virtual board and start to work in it, the time estimated on the progress in finishing a task depends on the difficulty of the task -. When someone finishes a task has to upload it in a repository, for this reason every member needs to know about some version control system but our group needs to know specifically git, the ones who has windows must install Gitbash and having an account on the online platform Github and link his/her account with Gitbash, lastly the file is uploaded and checked for the rest of the members.

What are the advantages of working with Kanban methodology?

First of all this methodology doesn’t have many rules like crystal or scrum, and it is very suitable to work since house, and not only that as Kanban doesn’t have a person who leads the group all the members can contribute in the same way (everyone has the same role) in the project and support each other if someone couldn´t do the activity that he picked, how? Just drag the task that you chose and drop it in the part that says “open” on the Meister task virtual board.

Other advantage that did us to choose this methodology is the facility of working with Meister task, since it allows us to see in real time the activities in progress and the activities that have not been done, and as it is a free virtual platform for anyone, it allows us to work easily from our homes.

Finally the meetings that are made in the Kanban methodology were feasible for us, because we could do them every Monday in the second hour of the programming class to talk about the difficulties we had, if each one is all right in his task, or someone dislike something about the project. in this last point if someone did not like something about the project a vote was made in the group and if the others didn’t like, it was changed and if there were a majority of votes for not changing it, then it remained the same, in this way we kept the roles of the group so there are no disagreements and we all remain neutral.

Why Kanban and not Scrum?

Kanban is very easy to implement and has very few roles and basically the virtual board of Meistertask is perfect for us because we are a small group, on the other hand scrum has many more roles and also many more meetings that sometimes are not necessary, at least not for this project, and also has a role that we all wanted to avoid and was that someone lead us in the project. We think that since this project is almost 1 month to finish it and nobody was an expert in augmented reality issue, let alone know how to use Unity and Git, we didn’t need someone who works like a scrum master, but everyone learns and can contribute something to the team, also the scrum master would have a lot of work to try to fix the mistakes of others or try to do what some member of the group was supposed to do.