Slopes: Reflections

Although our design process went fairly well, now that we can look back on our experiences, there is a lot to reflect on. Our initial planning and use cases did not change very much as the project progressed, so we feel that our planning was quite effective. We thought that having planned out a design document and a requirements document really helped in our development process, as it helped us stay focussed on our finished product goal. We were constantly learning about features that would work well in our app and as such it was easy to go off on a tangent, and start putting in more and more functions. However, having the design document and requirements document to refer to kept us focused on what our finished product goal should look like.

After reading Cockburn’s article, “The Interaction of Social Issues and Software Architecture”, we decided to take Cockburn’s advice on assigning an owner for each deliverable. We split our team into three different sub teams, all working on different functionalities. We had a couple people working on making the database, a couple working on the ability to parse and display information from the database, and a couple working on gathering information from available APIs. This was an effective plan for us because each of these functionalities are controlled by separate controllers. Since each controller has its own defined functions and scope variables, sub teams were able to produce their own functionalities, while rarely having to deal with code written by other team members.

Looking back, we could have started thinking seriously about testing requirements at an earlier point in the project. It would have been very helpful to have the tests ready before the functions were. This did not have serious impact on the project, but we would have saved time on refactoring the code if we were more organized at an earlier point. But also ties into our choice of framework,

We chose to develop on the Ionic platform because this allowed us to release a unified app on both the Android and the iOS market with ease. Because the majority of our team was new to app development, this allowed us to build our app without having to learn two new native languages. Furthermore, Ionic comes with built in UI components, making our job of presentability a lot easier. However, our choice of framework did definitely come with some drawbacks. Because Ionic is still a fairly new framework, it is constantly changing, and different functionalities are constantly being added and removed. It was difficult to quickly determine whether the cause to our code not working was due to incorrect implementation or to it no longer being supported in Ionic.

If we were to build another app, we would definitely choose to build natively. We believe that choosing to build using either Android or iOS development would make for a more reliable framework. Because so many people develop on both Android and iOS, choosing to develop natively would also allow us access to a more available documentation.