# Retrospect – iOs Development

## Group G: Andreas Malmström, Erik Svenningsson, Andreas Persson, Håkan Thorleifsson

## Background

The task in the course “iOs Development” was to create an app in groups of four students. Each group got to choose for themselves what the app should be about. Group G chose to try and program the old cellphone game “Snake” for iphone.

## The Basics

We started out with making a very simple game design, decided on the different views we thought we needed, the basic functionality and how we thought the mechanics would have to work. We also made some simple sketches of the basic design.

Since no one had any prior experience with programming a snake game, we had no idea how to divide the project amongst us. Everyone went through some different tutorials in order to familiarize themselves with game development in general and Swift in particular. When everyone felt they grasped the basics we finally got an idea on how to split the work.

## Initial Task allocation

Andreas P was initially tasked with animations and the storyboard (mainly constraints). Håkan decided to do the high score and everything associated with that, including the API.

Erik started with the fundamentals, initializing the game, making the snake actually move on the board and making the game navigation buttons work. Andreas M started with the logic behind generating new points, getting points in the game, and death mechanics.

## Progression

After working on our respective parts for a while, we realized that the work load was not evenly distributed, and that some tasks were easier than others. The API proved to take the longest time and be the trickiest to work out, mainly because we didn’t structure the code properly from the beginning.

All game mechanics worked out like we wanted them to, even though it took a while to get everything running as we wanted it. At first it was a little tricky since Erik and Andreas M had to work in the same files initially. Since the code we were working on was very cohesive, we decided it was easier to merge it to the same branch early on and continue from there. Some conflicts arose in GitHub since we hadn’t used it very much in the past, and we ended up having to rewrite some of the code.

We wanted to add hardware functionality to our app, and first we settled on using the camera. Since none in our group had access to an iPhone or iPad, testing the functionality became impossible. After some thinking, and realizing that the camera wouldn’t really add anything to the game in the first place, we instead decided to add shake gestures for some funny animations. At the start we also planned to have animations in quite a few of our different views, but since it caused the game to lag, we decided to remove it. Since Andreas P was already tasked with implementing animations, he was put in charge of making the hardware functionality work as well.

Somewhere in the middle of the project, Erik took over setting the constraints, since the ones we had implemented didn’t work as intended. Håkan kept working on getting the high score to work, since it wasn’t originally implemented with MVC structures, it couldn’t handle accessing and transferring data very well from other controllers.

In the closing stages of the project, Andreas M and Erik structured up the different code files and made sure we had the same syntax across the entire project. Everyone also contributed with one test each.

## What worked out well

We communicated fairly well with each other, most of the time we could help each other with different problems that occurred. It was also pretty straight forward to get the core game running, since we stuck to getting the game mechanics working instead of over designing.

## To think of in future projects

The biggest flaw of this project was that we didn’t have a real plan, and since everything worked out okay for most of the time, we didn’t really take any pause to evaluate our work in the process.

Another thing to take into consideration in the future would be to structure up everything better from the beginning, i.e. we should have made sure everyone wrote their code in the same fashion, with the same code standard, and with different files for data, views, etc.

We would also have benefitted from more frequent check-ups, to make sure everyone understood what we were doing all the time.

## In conclusion

As it would turn out, the final design of the project actually came close to our initial design, even though we changed it a couple of times during the process.

The project was a good way to learn both app development in Swift, and working in groups. Since we had no previous experience from group assignments it was a great lesson in group dynamics.

To learn how to make an app in a good way was hard, but in the same time educational. Making a game app though, was a bit harder than anticipated from the start. Had we chosen to make a more conventional app, we probably would have had more success.