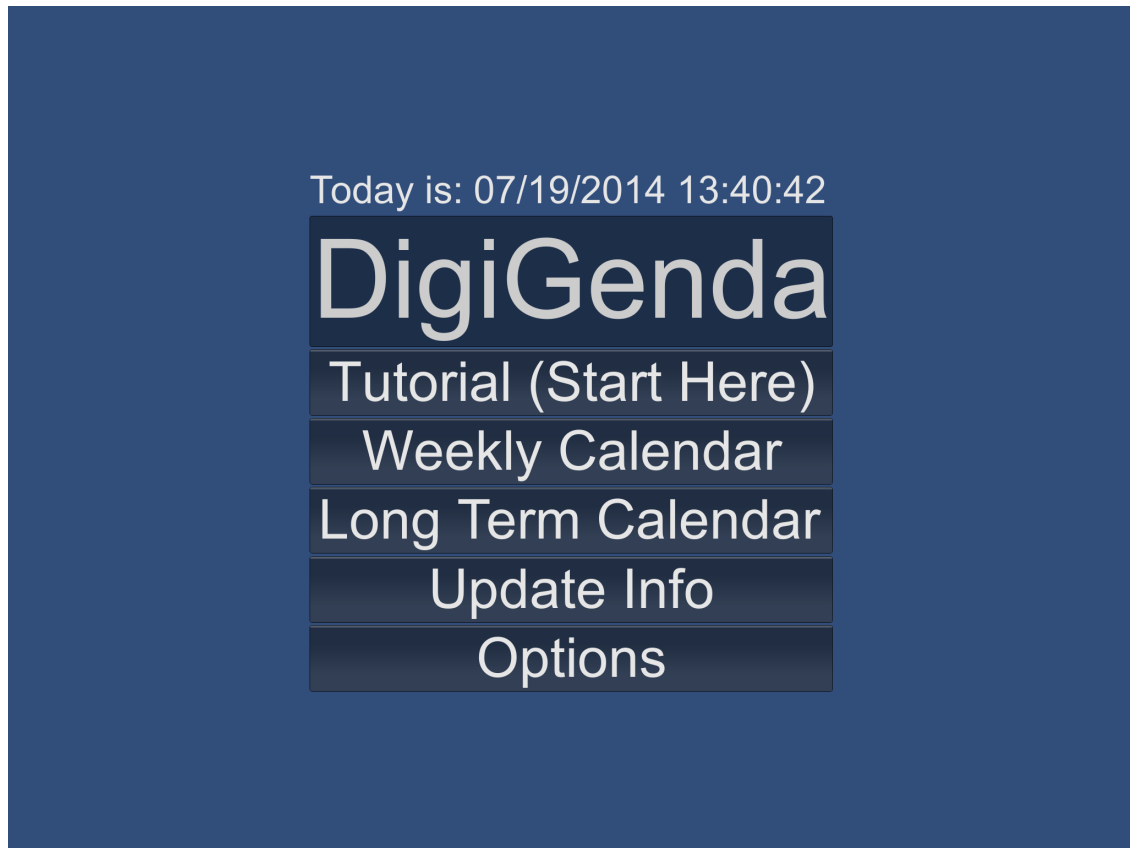


Develop an iOS Application

Jonathon Scott



Julia Grafft
Susan Canham
Wood River High School
August 15, 2014

Selecting the Project

From the beginning, I knew that I wanted this project to be an opportunity to explore the process of developing an application. I am fascinated with programming, and I am considering pursuing a career in computer science. The more I thought about creating an app, the more excited I became about the whole project. Developing an app would be a great project for me because it would allow me to do some career exploration, but there was still one major detail I had struggled to come up with. An app needs to offer its user some sort of utility. Many apps are simplistic games that are widely popular across numerous demographics. Angry Birds, Clash of Clans, and Outwitters are some well known examples. Other applications make everyday tasks more convenient for the user, a common one being mobile banking. These are designed to be simple to use and to save time.

I decided against creating a game because of the formality of this project, and thus began thinking about how I could create a product that would improve the lives of myself and my peers. Thinking back to the previous school year, I remembered how difficult it was for me to feel at ease as Sunday night rolled around. Had I done all my school work? Had I forgotten some major assignment? I had experimented with various organization techniques because organization promotes relaxation, success, and free time ("Top 5 Benefits of Getting Organized"). However, the organization tools available hadn't worked for me. I wouldn't write in my agenda because I can be forgetful, and my handwriting is difficult to read. I usually relied upon my memory, but when my life became busy and stressful events and responsibilities would slip my mind. I needed a new system. One Sunday night, as I lay in bed going through my mental checklist of To-Do's and tomorrow's Will-Do's, I thought about how great it would be if there was a digital version of my mental checklist. Wouldn't it be great if there was an app for

the iPhone or iPad that could help me manage my school work? And suddenly, I had my project idea! I could be that someone. I could develop an app that would not only solve my own organizational issues but be useful to other students as well. I just needed to answer one simple question; “How can an organization application improve a student’s time management skills?”

Before this project, I possessed a basic understanding of a few programming concepts. I had taken a couple of introductory classes with Mr. Schrader, and I learned the elements of game design and the characteristics of the gaming industry. At first, I tinkered with a twelve year old game, called Warcraft III, that offered its users the ability to “create” their own maps and mini-games. I didn’t write a single line of code, but I was able to get a solid grasp on the concept of using “if statements” to trigger events within a program. The next semester my attention was directed towards learning how to use Unity, an intuitive modern game development engine. The software, which I used under their free license, allowed me to write my first scripts and implement them into a game world. Through the development of two different projects, I was able to become familiar with the work flow in Unity and the process of scripting in C#.

The project became a learning stretch when I decided to write code for the iOS operating system which I had never had the opportunity to work on before. I needed to discover new scripting techniques. I needed to learn how to push my creation from my laptop onto an iPad or iPhone. I needed to experiment with saving data. I needed to manage my time during the summer and create an organizational system for the development of the project.

I chose the global context of Scientific and Technical Innovation, and I will explore the context through the lense of products, processes and solutions. My project will allow me to experience the process of creating a product that can provide the solution to my homework and time management issues.

Planning the Project

My project didn't require a large amount of planning because I knew that the majority of my time would be spent interacting with code. With that said, the biggest part of planning consisted of opening up my weekly schedule to work time. One exciting aspect that I couldn't plan ahead for was the actual system that would handle the functions of the app. I could plan the functions and I knew where to start, but the actual process, the journey, had to be allowed to happen organically. To start the organic process, I made a couple of sketches that I could strive towards making the interface look like. I knew the professional development process took a lot more time than the three months I was given to produce my product so it was essential that I got to work immediately and left plenty of breathing room at the end of the project to take care of any bugs and receive feedback. I submitted the following steps in my project proposal:

1. I will research and acquire the necessary software required to develop the app.
2. I will research the process of iOS development.
3. I will learn how to use the software.
4. I will collect information about agenda use and organization tactics in students.
5. I will research good behaviors that promote good organization/memory.
6. I will create an outline of the desired features in the application.
7. I will develop the application.
8. I will ask friends, family, and anyone else to test my app for bugs and improvements as needed.
9. I will submit the app to Apple for review and approval.

At the beginning of the project I had already a general idea of the whole development process. I needed to collect information about all the tools I was going to use and research possible

alternatives and select the best option for me. From there, I wanted to learn from the mistakes of those before me and take their advice and so I decided to research the development process.

During the planning process, I didn't know exactly which tool set I would be using, but no matter which one I used, I needed to become familiar with it through personal exploration and the associated manuals. If I really became stuck, I planned to follow a YouTube tutorial to get me out of the rut and back on track. Next, I wanted to learn about my peers so that I could tailor the app to suit their needs as well as my own. I planned to collect this information through a short online survey. I wanted to include more research on established organizational habits so that I would better understand the need for the product and allow those habits to be integrated intuitively within the application. I knew that the timeline was too tight to wait until all the research was completed to start on the project, so I planned to start the development process as soon as my project proposal received approval from Julia Grafft. I wanted to start immediately by figuring out the software and filling in where I got stuck with research.

Completing the Project

In the first step I needed to acquire all the equipment and software necessary in order to develop an app. I already had Unity and MonoDevelop installed on my computer from school, so I investigated Unity and found the option to "build" projects onto iOS devices. From there I visited the page on Unity's website that laid out some steps necessary to begin development ("Getting Started with IOS Development"). One of the steps included the required purchase of an Apple Developer membership. The one hundred dollar yearly membership granted access to Apple's support program and full suite of tools. However, the most important tool was the iOS specific compiler, Xcode. Unity possessed the capacity to export a project into the Xcode format, which can then be compiled and run independently on mobile devices. When I signed up for the

membership, I had to contact the Apple Developer's customer support line, and talk with a representative to solve an legal liability issue because at the time of registration I was under 18 years of age. The issue was solved by creating a new account and applying for membership under my mom's name. The final program I needed to acquire was GIMP. GIMP is a free open source alternative to Adobe's Photoshop that would allow me to create some essential artwork like the app icon and logo.

My second goal included the research of the iOS development process. I was able to define the whole process in five generalized steps ("How to Make an iPhone App"):

1. I needed to set up the development environment.
2. I needed to create a plan or outline for the features of the app.
3. I needed to follow that plan to create the app.
4. I needed to test the app.
5. I needed to release the product to the intended audience.

I had envisioned a similar process when writing my project proposal so I was very pleased when the research steps nearly aligned with my proposed steps. Once I had the process defined, I began to search for development advice. One source gave me an overview of how to use Unity to specifically develop for iOS. Most of the information I already knew, like how to manipulate the user interface, but I did learn about some interesting and useful shortcuts available only to those who had the expensive Unity Pro License. One of these time saving features was the ability to wirelessly sync with the mobile device and deploy the development build remotely (Abernathy). On a more general side of the process, I came across an article that cited a common major issue being the communication barrier between the client and the programmer. It explained that many non-programmers have difficulty expressing ideas into words that a programmer can easily

understand and develop into the client's envisioned product. I wouldn't have to deal with that issue because in this situation I am both the client and programmer. The same article emphasized the importance of selecting the correct mobile platform. I could choose between either iOS or Android, and I choose to develop for iOS for the sole reason that I didn't have access to an Android device (Erickson).

My third goal required that I became familiar with the tools of the project. I like to learn from trial and error. Making mistakes is great for me because it makes me think critically about how to solve them. This became my initial approach to learning the new software. I was already pretty comfortable using Unity, so the major part of new learning involved actually figuring out Apple's Xcode and how it could be paired with Unity to run the app on a mobile device.

Eventually, after many unsuccessful attempts to figure out the process on my own, I turned to a YouTube video that walked me through the entire process (Hutong Games). I made sure I was comfortable with this process before proceeding to ensure that I would be time efficient once I began to write code. This step was very important because it taught me how to test each of the one hundred and twenty development builds on the correct device. I already knew how to use MonoDevelop and Unity, so once I figured out Xcode, I was able to play around in GIMP to figure out how to make a simple icon that could represent my app.

In order to complete my fourth goal, I collected information about agenda use from my peers through a simple 2 minute survey I created using Google Forms. I had first been introduced to Google Forms from my AP Biology course, so I knew them to be quick and reliable. After creating the survey and making minor adjustments, all I had to do was send my participants the link. I did something that was slightly out-of-character for me when I realized that I needed a larger pool. I normally only use Facebook for its messaging function, but for the sake of the

project I typed up a status update requesting anyone who read it to take a minute and help me out. I was pleasantly surprised when I received about 30 responses, most of which portrayed positive support for the DigiGenda concept. The results of the survey are found in Table 1.

Table 1: Research Results

Agenda Use	
User	16
Non-User	14

Due Date Management	
Memory	18
To-Do List	8
Non-School Agenda	2
School Agenda	9
Electron Device	10
Homework Folder	1

Those Who Didn't Use Agenda and Reason	
Students who felt they weren't needed	4
Students who didn't like to write	7
Students who would lose the agenda	7

Use DigiGenda on iPad	
Students who didn't own an iPad but said they would if they owned one.	22
Students who said no	3
Students who said yes	5

Grade in School	
9th	0
10th	4
11th	5
12th	15
College	5
Middle School or Younger	1

Number of Average Classes	
6 Classes	19
7 Classes	12
8 Classes	5
9 Classes	1
10 Classes	2

The fifth step involved researching common organizational/memory techniques and advice for establishing and improving them. One source claimed that the most important part of being organized is the acceptance that it is a continual process that is never complete (Andi). Others emphasized the importance of routines, to-do lists, color coding, calendars, and student planners (Gracia). I even found some advice for parents of ADHD children that recommended big tasks should be split up into smaller sequenced tasks and that the planning horizon shouldn't be set month to month, but instead week to week (Goldberg). After this research I was also able to answer the project question. An organizational application can improve a student's time management skills by incorporating helpful strategies. I directly implemented the week to week calendar and color coding to create a digital student planner that help can help establish a weekly routine.

For the sixth step, I needed to create a basic outline that I could follow during the development process. I made some simple drawings of the basic interface I wanted to use on the app. The two sketches I drew became the "Weekly Calendar" and the class specific assignment creation screen. I drew up the idea of displaying small snippets of information about the assignment type in little bubbles on the calendar screen. It wasn't anything super technical, but it

gave me something to aim for while I focused on working with the code to make the product run. After these drawings, I brainstormed a list of possible functions that I could put into the app. Possible functions included a dynamic class schedule, assignment tracking, integration with skyward, simple calculator, fitness tracker, community service, and a college brag sheet.

In the seventh, and most important step, I developed the app. The first thing I programmed for was the basic interface that I drew up in the outline. I carefully measured the grids and placement of each button as well as tested the size of the text. Next, I focused on creating a system to save user data so that any information that the user input wouldn't be lost upon the mobile device closing the app or shutting down. Once I established a system of saving and recalling the data, I worked on creating the menu that would allow the user to input their name, the names of the classes, and the classes' starting and ending times. I wanted the user to be able to change this data at anytime in order to cope with a changing schedule throughout a school year. Accurate time updates were the next step, and I was able to create a script that updates the numerical dates on the "Weekly Calendar" top banner every Monday. Dealing with dates required that I look up some C# specific code on the Microsoft website ("DateTime Structure"). Then, I moved into creating the system of creating assignments and displaying them on the calendar. I allowed the user to pick the type of assignment (homework, study session or exam), name the assignment, and input a few key details. I assigned colors to each assignment type to create a color scheme that, at a glance, gives the user an idea of the kind of work they need to do for the night. Green became homework, yellow became study sessions, and red became exams. There was some space left over on the assignment creation, so I added the functionality to keep track of longer term assignments like projects and essays. Throughout the entire process, I often would need to double check how a certain line of code should be used and just as often I needed

help translating function into code. For both situations I made extensive use of Unity's scripting guide ("Scripting API: UnityEngine - 'Classes'") and the Unity Community's forums ("Scripting").

The development process wasn't always smooth sailing. I would often hit major roadblocks. I spent a lot of time trying to figure out how to save user data and tried multiple techniques before one worked for me. My programming speed was heavily influenced by processing power on my computer. Unity's coding companion, MonoDevelop, would often freeze up and crash and I would lose precious progress. This was a common occurrence, until I found a workaround. I wrote new lines of code into a word processing document, which I would then copy and paste into the coding software. Besides some minor formatting issues, I was able to vastly improve my working efficiency in this manner.

The eighth step of the project required that I show my app to others and receive feedback. This actually was a continuous process during development. I highly valued the feedback from my friends because they weren't concerned with sugar coating their responses. I kept a good friend updated on my progress throughout the summer with many video demonstrations. The same friend actually suggested I include the assignment details section which vastly improved the usefulness of the app (Rinckwald). When I felt I had nearly finished the development process, I took the app wherever I went, and I showed it to anyone who would listen. In addition to receiving immediate feedback through conversation, I would send them a follow up message on Facebook to another quick survey. Not as many people responded to the survey as I had hoped, but those who did helped me fix issues and try new features. Some even suggested features that I could add to the app in updates. These features included: a longer length calendar, capabilities to handle non-school related activities like community service, jobs, and sports, and

an even more mobile version of the initially iPad exclusive app. Due to the restricted time frame of the project, I was only able to tackle one of the suggestions, and so I created the iPhone compatible version. This version is slightly more difficult to use because of the significantly smaller screen size. However, I found it exciting to listen to my “customers” and improve the product directly to their request.

For the final step of the project, I submitted the iPad version of my app to Apple for review and approval. The whole process was a little bit confusing at first, but I was able to find all the assistance that I needed in the submission guide (“App Distribution Guide”). When I prepared the submission build, I had to archive it into a format that could be easily sent to the App Review team in binary. From there, the team will be able to review the app and determine if it meets their quality standards so that it can be distributed on the App Store. The average response time is about 6 days. I submitted the app on August 5, 2014. Over the next seven days I received two emails updating me on the status of the review. Finally, I received an email confirming that the app has been approved on August 12, 2014. The iPad version of DigiGenda made it through the review process and is now available free of charge on the App Store!

Reflecting on the Project

I was very pleased with the outcome of the project. The amount of time and thought I put into achieving the goal of creating an iOS application taught me about the extreme detail and work that goes into creating even the simplest of apps. The product itself is a great representation of my individual efforts, the research, and the advice from others. I didn’t, however, plan to spend as many hours as I did developing the project as I did. I suppose it was a good thing because the majority of that time was spent pondering a problem and thinking of potential solutions.

The global context, Scientific and Technical Innovation, and the exploration, products, processes, and solutions, directed me to learn a great deal about the amateur app development process. I experienced the whole process of iOS development from idea conception to debugging and testing to product submission. My final product, DigiGenda, is my virtual response to the project question of, “How can an organization application improve a student’s time management skills?” In English, the response would be, “An app can help a student establish a routine in which they utilize color coded, week by week calendars to effectively manage their time.”

This project made me realize a lot about myself. I learned that I don’t mind taking risks. At the start of the project, despite a feeling of confidence and excitement, I felt a twinge of uneasiness. I was so new to programming that I felt slightly overwhelmed at the prospect of undertaking such a complicated and important project in this manner. Now that its nearing its end, I am happy that I did. I learned to balance the summer’s activities of camping, sports, college visits, online classes, social events, and Personal Project. With a lot on my plate over a short amount of time, I excelled at juggling so many components by taking the initiative and prioritising each aspect in order of importance while still setting aside time to have fun with friends. I learned that I can be resourceful and imaginative when presented with problems.

This project helped me explore a career interest. Overall, the experience was pleasant, and I still possess the same curiosity that I started with at the beginning of the project. I am now more seriously considering pursuing a career in computer science. As Confucius said, “Choose a job you love, and you will never have to work a day in your life.” If I am doing what I love, I wouldn’t mind never working a day in my life.

Works Cited

- Abernathy, Christine. "Beginning Unity 3D for IOS: Part 1/3 | Ray Wenderlich." Web log post. *Ray Wenderlich*. 9 Jan. 2013. Web. 28 May 2014.
- Andi. "What Does Being Organized Mean to Me?" *Orderly*. OrderlyBCS, 13 Jan. 2014. Web. 29 July 2014.
- "App Distribution Guide." *IOS Developer Library*. Apple Inc., 2014. Web. 05 Aug. 2014.
- "DateTime Structure." *Microsoft Developer Network*. Microsoft, 2014. Web. 14 July 2014.
- Erickson, Christine. "How to Avoid the Common Pitfalls of Mobile App Development." *Mashable*. Mashable, 28 Mar. 2012. Web. 06 June 2014.
- "Getting Started with IOS Development." *Unity Manual*. 27 Feb. 2013. Web. 09 June 2014.
- Goldberg, Donna. "School Organization Tips for Students with ADHD or Learning Disabilities." *ADDitude Magazine*. New Hope Media LLC, 2013. Web. 6 Aug. 2014.
- Gracia, Maria. "20 Ideas to Help Students Get Organized." *Get Organized Now*. GON, 2014. Web. 6 Aug. 2014.
- "How to Make an iPhone App." *WikiHow*. N.p., 6 Aug. 2014. Web. 6 Aug. 2014.
- Hutong Games. "01 - From Unity3D to Your IOS Device." *YouTube*. YouTube, 11 May 2012. Web. 29 May 2014.
- Rinckwald, Joel. Personal interview. 1 Jul. 2014.
- "Scripting." *Unity Community: Forums*. Unity Technologies, 2014. Web. 14 July 2014.
- "Scripting API: UnityEngine - 'Classes'" *Unity Documentation*. Unity Technologies, 2014. Web. 14 July 2014.
- "Top 5 Benefits of Getting Organized." *The Paper Tiger Blog*. Paper Tiger Blog, 6 Feb. 2012. Web. 6 Aug. 2014.