# Smart Rockets

Have you ever thought about how machines learn without being explicitly programmed? What about how a machine learns about the environment that it is placed in. We are able to give a problem to a machine and actually watch the progress on how the machine thinks and learns while trying to find a solution. I used a Genetic Algorithm to show how a machine can follow an object, find the optimal route or even dodge obstacles.

Built using nothing else but Processing, you can see just how easy it is to create a Genetic Algorithm. Genetic Algorithms are based off biology. Given the same characteristics as evolution we can see the progress of the algorithm evolving generation by generation.

Using a custom-built fitness functions, the machine is able to get the distance from the current target and use it to gather the information needed to try and find an optimal path and speed to try and reach it.

# Day-By-Day

Day-by-day is no ordinary calendar application. This is an Android app that will save you the most time for your day. This is a to-do/note taking application integrated with Google Maps API system. Our goal is to try and find you the best route possible throughout your day.

It’s a beautiful minimalistic app that will help you be more efficient around your daily tasks. You will enter your task for the day and you will be prompted with a location feature that will allow you to find your way around the tasks in putted.

Built entirely using Android Studio this application takes full advantage of the Android operating system and combines two applications that millions of people use throughout their day.

# Typography project

This project is it illustrate just how I have developed my design side. Using only text, black and white contrast and CSS skills, this two-page newspaper allowed me to show just how I have managed to come to terms with typography.

It was made with just HTML5 and CSS3, this doesn’t have any of your fancy elements like JavaScript that you might find in your average website. The creation of each element, heading and paragraph was chosen for a reason. Showing the user just what I thought about and why I chose the font, style and weight.

Giving the user a feeling of a newspaper but the modern approach as a news website. The process of building this involved studying newspapers and their approach to design. It also holds some key elements such as line breaks that new modern websites are using

# Implementing a Reliable Data Transfer Protocol

I built a fully functional simple reliable data transfer protocol. This protocol is able to send and receive data using the Layers that are in the current world wide web. Sending data between layer 5 (Session Layer) and passing the data down to Layer 3+4 (Protocol Layer).

Built in the Java framework this application uses NetBeans. Sending a message from one machine (port) and receiving the message on the other machine (port), in the background it has an acknowledgement function that lets me know that the data has been send, a check sum to let me know that the message hasn’t been lost or altered in any way and timer so if the package gets lost in transmission you will also be notified about it.

# Game built using Phaser and JavaScript

A custom built clone of ‘Flappy Bird’ built using Phaser and JavaScript. This game was taken from the concept of ‘over-night’ sensation Flappy Bird. Using a JavaScript Framework called Phaser, this game takes assets such as sprites and background images and places them into a preloader that is then used to display the screen.

Using the inbuilt physics engine that Phaser is well known for, this game wasn’t too difficult to build, the point of this project was too allow me to knock another Framework out of the park. Some basic math’s was used to calculate the jump and hit detection of Spiderman.

# Quality Assurance Testing

Throughout the summer I took up work as an Intern with a company based in Dublin. This gave me an insight into how a company uses agile development management and I was put on the Quality Assurance team.

Building tests for the company using Calabash and Cucumber (two library’s built off Ruby-On-Rails) and testing them thoroughly making sure that they will up to standards allowed the team to fasten up the testing process.

QA testing on Linux distros gave me the full advantage to learn not only QA testing and the Linux distro but also about properly merging Git files with a full 20-man team. This knowledge is something that you cannot learn in college.

# Two Player Simon Says game using Arduino

This project was built throughout the last year of college. Using an Arduino Uno board and processing a two player Simon Says game was made. Hooking up one Arduino board and programming the Simon Say game was the first step.

After that the next step was hooking up two Arduino’s to talk to each other. This was done through an application called Processing. Getting the two boards to talk through OSC (Open Sound Control) messages was an easy step as we had open source library’s already out there to connect the boards.