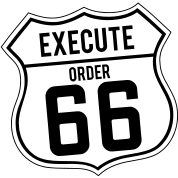
Presented by: Oliver Gibberd-Thomas, Joel Holmes, Ruben Jardim, Michael Louca and Christopher JOnes

[company name]

[Company address]



Team Organisation Plan and Project Plan

March 18, 2017

# Introduction

## Purpose of This Plan

The Internet of Things (IoF) is a term used to describe connected devices across the internet, the ability for physical appliances to communicate to users, applications and even each other. The vision of The Internet of Things constantly changes as devices and networking advances.

The start-up company has a strong interest in smart technology and wants to utilise that technology to create a monitoring web and/or mobile-based platform that views the information from smart devices and displayed it to the users. Here at Team Order66, we want to assist the development of that web-based product, making it as functional and productive as possible.

The product needs to be able to meet the user requirements given; being able to add, remove and view the multiple IoF devices that the user has and viewing trends that occur with smart devices and act based on them that benefit the user. If the Smart Fridge knows that you buy milk every 2 weeks, the web-based platform could view that trend and purchase the milk for the user.

# Team Organisation

## Research

A popular function among existing smart devices, is this apparent technological learning curve. It is an expectation that data should be used not only for regular functionality, but to create/recommend extra functionality. This collaboration of data can make it seem like the device/system is ‘learning’ about its user, which allows it to be more accurate and specific when fulfilling the user’s needs.

In anticipation of this task, we took it upon ourselves to research a variety of different networked devices, such as the Amazon Echo and Jawbone Up. However, a similar device to the Smart Fridge, is the Nest Learning Thermostat. This device connects to the boiler to regulate the temperature of your home from a personal device such as a smart phone. This has added benefits, as you can turn your heating on/off when you are away from the thermostat, and the device’s software can even be synched with third party apps such as IFTTT. This allows extra functionality, such as using a personal device’s GPS to trigger the boiler on/off when you leave or enter your home.

## Internal Organisation

The internal organisation of the team we have decided to use is a mixture of egoless and chief programmer. We came to this decision because members of the team have different skills which can benefit the project, however some team members are more proficient in certain areas. For example, the more experienced web/app programmers would be assigned with front end development, whereas those familiar with databases could work on data management etc.

For this reason, we have decided to adopt some the attributes of a chief programmer team. Depending on the time of development, one member would oversee the team and would make rapid decisions on what to implement into the software. When a decision has been made on what to implement, the rest of the team can research the best method to implement the function.

Having an egoless organisation would mean that everyone would have the opportunity to say what they think. When implementing a function, the team would put forward their ideas. The team would then discuss the best method to implement the function.

Having a hybrid of egoless and chief programmer means that everyone will be able to contribute towards the team equally. Team member’s skills will be considered, and a decision would be made on who should be leading the team at times of the development. Scaling would have to be taken into consideration, because different team members would be working on a part of the project at different times.

## External Organisation

The external organisation of the team we decided to use is a Functional Organisation Methodology. Using a Functional methodology would mean that there isn’t a dedicated project manager. This wouldn’t be a problem for us, as the team would be led by different members depending on the task in hand. Dedicated time slots would be decided with the team at each member's convenience.

# Task Allocations

The allocation of tasks is based on the organization methodologies previous mentioned. Sections of the actual programming code for the system will be supervised/checked by one allocated member who is the most comfortable with the given programming language. Even though one member will supervise the code writing to check for good practices/consistency, every member will contribute to the code. This fulfils the task specialisation of the chief programmer methodology, as well as adding the equal share of work identified in the egoless methodology. Like the Agile methodology, each part of the project is planned and implemented individually so that the member responsible for that section can supervise without conflicting with other sections.

## Planning

The planning of each part of the project is undertaken whilst every team member is present. This means that the best ideas from each member can be formulated into one solid piece of work. To ensure that every team member has a fair share of the work, tasks are allocated from the offset based on the skillset of each member. If someone cannot fulfil a piece of their allocated work, then another member can assist/swap a task to ensure that workflow between each member is fair.