MARCH 27, 2017

ORDER 66

PRESENTED BY: OLIVER GIBBERD-THOMAS, JOEL HOLMES, RUBEN JARDIM, MICHAEL LOUCA AND CHRISTOPHER JONES

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 to create a monitoring web and/or mobile-based platform that views the information from smart devices and displays it to the users. Here at Order 66, 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; can add, remove and view the multiple IoF devices that the user has, 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.

RESEARCH

Our task is to program smart devices assigned to a network, so we took it upon ourselves to research devices of a similar nature. We did this by browsing telecommunication/technology company websites so that we were up to date in the most recent smart devices and their utilities. Most the team are already familiar with smart devices as they currently own a collection of different devices personally.

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, allowing it to be more accurate and specific when fulfilling the user's needs. This will be a feature of our program as our application must 'learn' from the data it receives.

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 previously mentioned 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 smartphone. This has added benefits, as you can turn your heating on/off when you are away from the thermostat, the device's software can also be synced 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.

A very popular and heavily networked smart device now is the Amazon Echo/Echo Dot. This is a smart speaker which acts as an alarm clock, personal assistant, fitness instructor as well as many other utilities. This device relies on smartphones/tablets for setup, as well as integration with networks and other apps. This makes it a hub for multiple applications and is constantly being patched to facilitate new functions. This level of functionality with secondary apps and data sources is a necessity for our program, and is somewhat of an expectation of from customers.

Team Organisation

Internal Organisation

The internal organisation of the team we have decided to use is a mixture of an 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 which the team would then discuss to find 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 different times of the development. Scaling would have to be taken into consideration because different team members would be working on larger or more difficult parts of the system.

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 at hand. Dedicated time slots would be decided with the team at each member's convenience.

The team is functioning very closely to relevant external stakeholders to make sure they have a detailed understanding of the project process. The better knowledge they have with the system, the less of a chance there is of conflicting opinions occurring between the project team and the stakeholders. Customers are the most important external stakeholders and the team takes their opinion very seriously especially during the early stages of development.

Task Allocations

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.

Team Allocations

Team Order 66 contains 5 members, 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 be aware of updates to each section. 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.

Ruben and Christopher are members that have a greater knowledge on creating the database necessary to store the user's smart devices information. Because of this, they will prioritise on that area of development. Joel, Oliver and Michael will work mostly to create the application that will run well on multiple devices and will look through the list of smart products the users have and investigate their current and past status. The roles that each member is assigned to are very flexible, so their jobs change very easily depending on the circumstances. Oliver will also supervise the development of the database to ensure that both the database and the application will work perfectly together.

Project Methodology

Technical Methodology

The technical methodology our team has decided to use is a linear methodology like that of a waterfall model with feedback. As the waterfall model alone is quite simple and hardware oriented we feel we needed additional feedback is necessary for our project model. As the nature of software is changing we need to make sure testing is an ongoing activity throughout the development process.

As we are working from a brief of fixed requirements a linear methodology suits our needs and benefits the small group we have. As we won't have to change our requirements/specification there will be no need to use an iterative methodology such as a spiral methodology.

As our project time is also a constraint, the waterfall model we will be using can be detailed on an hourly basis. This is also easier to show off than a more complex model. We have a representation of our project schedule in the form of a Gantt chart, which clearly shows all activities needed and the time it shall take.

During our design process a throwaway prototype will be used as an initial design plan used so our team all have visual understanding in what the outcome shall be and can work on building it, whereas during the software implementation we will be working with an evolutionary prototype that we will adapt and change over the project based on testing and how we feel it performs leaving our project open for tweaks towards quality.

Project Management Methodology

The project management methodology our team have decided to use an agile methodology approach to the project. Regular meetings will be made so that the team knows where the projects stands and any tasks that need to be done. If tasks need to be micromanaged, then daily sprints will be made so complete tasks. Sprints logs can then be compared to the Gantt chart so that tasks are track to being completed.

Quality assurance will be track by looking at the requirements and deciding if tasks need to be re-evaluated. Risks will also be looked over to see if the project will encounter any issues. These decisions would be made during scrum meetings.

Aspects of MoSCow will be used so that the team knows what the main objectives of the project. Prioritizing the requirements would mean that the main functions of the application would be fully functional. Knowing what requirements are a priority would mean making timeboxes more reliable to track with. Prince2 would not be a wise choice as it is complicated to use. The methodology is not very helpful with small groups and requirements would not be changing.

Project Schedule

Schedule

Our Project schedule is an important part of our overall plan. We have allocated time for each activity during our project's period. This is important as no time will be wasted or overused on each activity. The milestones are goals we will meet to have completed our project, each milestone must be completed before moving on to the next section of activities. A timeline of our schedule can be seen below. We have three major sections for each activity: Specification, Implementation and Testing and the Report.

Specification

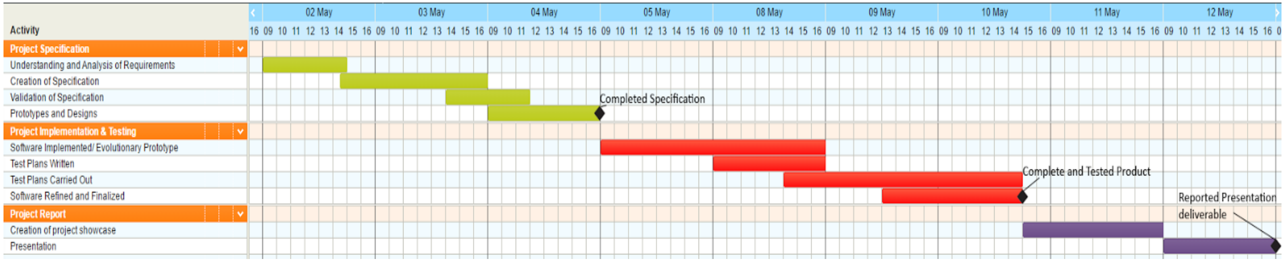
Our Specification is a crucial part of requirements engineering in our project, it will be used to validate and verify our understanding of the project, contain important fundamental designs used in the implementation and will also contain a large percentage of the information needed to show our clients the report on the final deliverable. Due to this, we feel 3 days is a good amount of time to spend working on the specification.

Implementation and Testing

This section will follow on from the design specifications and will give us a physical deliverable in the end. Testing is also crucial and will be done alongside the creation of code to make sure it is all in order. We feel four days should be enough time to meet this deliverable.

Report

The final report will collect and quality assures our project as well as create a presentation for the clients understanding and showcase our project.



Schedule Monitoring

Our Gantt chart has a schedule monitoring bar which we will move periodically to monitor progression. This will allow us to keep full track of the progress we are making and allow reallocation of time if needed IE: If we have used less time to complete the specification, we may be able to add additional features. This will ensure our project is fully scheduled and all time dedicated to the project's development will be used as such.

Risk Analysis and Contingency Plan

Risk analysis

Risks are events that, if it occurs, could have a positive or negative effect on the project's progress. Risk Management is the standard business process that monitors and assesses the reported risks.

The project manager working with the team and sponsors will ensure that risks are actively identified, analysed, and managed throughout the life of the project. Critical systems will be taken very seriously and risks will be identified as early as possible to minimize their impact.

The identification of the risks will involve the project team, stakeholders and will include an evaluation of the organisational culture and environmental factors. Here we have a Risk

Management Log to monitor all the problems that could occur during the project. Each risk is rated either ‘High’, ‘Medium’ or ‘Low’.

Risk Table:

Number	Hazard	Possible Effects/Harm	Likelihood (H, M, L)	Risk Rating (H, M, L)	Mitigations
1	Uncompleted Project	The project could be unfinished when the deadline is hit. Causing either a delay or a cancellation.	L	H	Every sprint, monitor the project progress. Hire additional workers to complete the project in time
2	Conflicting Opinions with Stakeholders	The Stakeholders could have alternative or additional specifications to complete in time for the deadline.	M	M	Constant communication with stakeholders. Compromise with the stakeholders to complete tasks that they want as well as completing the project in time for the deadline.
3	Changes in Requirements	Any changes in the project system specifications during the development process.	L	M	Have meetings with the stakeholders and project team to make changes when necessary. Project team to start a sprint to make direct changes to the project to meet the new system requirements given.

4	Conflict within the Team	Conflicting opinions of how the project should go from then on	L	M	Have meetings with the individuals that have the conflicting opinions Have a meeting with the stakeholders and the whole project team to make direct changes to the system
5	Scope Creep	Project grows in complexity as clients add to the requirements	M	H	Have discussion and plan on meeting the requirements that can be met before the deadline Delay the project to meet the new user requirements given to the project team
6	Unexpected Delays	System requirements haven't been fully met so delays occur to meet the user standards	L	M	Discuss the situation with the team and make the delay if necessary. Delay the project immediately and set up a new deadline to meet
7	Software and Hardware Defects	Software and hardware problems occurring during the project that hinders the team's progress	L	H	Have backup hardware and software ready in case a problem occurs Remove defected hardware/software and use backup products for the rest of the project
8	Insufficient Funds	Running out of money		M	Monitor the project budget constantly and make changes to the budget if necessary

		halfway through the process	L		Set up a meeting to increase the budget given to complete the project
9	Skill Related Risk	Skills that the project team has no knowledge of but is required to complete the project	M	H	Set up a meeting with the stakeholders to adjust the project team Hire additional people that has the skills required to complete the project successfully
10	Customer Requirements	The requirements provided by the customers are not well documented.	L	M	The project team will document the requirements and whether any of them have been added/edited/removed recently.
11	Lack of Understanding	The project team may lack a full understanding of their roles to develop the system.	L	M	Project team are communicating with each other constantly. If any member has issues, that can be resolved very quickly.

CONTINGENCY PLAN

Growing a business can be a very risky practice because there is no guarantee that the project will not only be successful with providing profit but to even run properly. Because of this, a contingency plan will be created to lay out the course of the business future actions, serving as an alternative plan when the team is expecting a failure during the project. After viewing multiple sources, we decided to have an ongoing plan that will be added and amended during the development process.

The contingency plan coordinator, as well as the management, will determine which members of the project team are responsible for each function during each of the processes. As tasks are assigned, additional responsibilities will be created to address functions during specific phases. The table above describes the 11 biggest problems that could occur during the development process.

Having an unfinished product when the team meets their deadline could happen due to multiple reasons. The way our team will deal with that situation is by having a formal meeting that will include the stakeholders to decide on the project's future which will be to either end the project or to give the team a new deadline to meet the user requirements they were given.

Scope Creep refers to the uncontrollable growth in a project's scope at any point. This problem can be a result in poor change control, poor communication between teams and a lack of vision in the project development. Our team believes that this is one of the most harmful problems that could happen because it could cause multiple issues to occur, such as overspending or miss the deadlines that were given.

Finally, software and hardware defects could become a big problem and could occur at any point in the development process, which makes it especially important to have a plan in place if such event happens. Using a revision control software to back up the team's work on a regular basis is a standard procedure that all members of the team will follow. Having backup hardware available in case of a hardware failure will be very costly. However, we believe it will be worth the money as it will protect the team from problems with the hardware that could affect the whole project.

Quality Assurance

Strategies

The aim of Quality Assurance is to use a systematic methodology to evaluate the quality of our software, making sure that it adheres to the requirements and is free of any compromising bugs. Quality Assurance isn't just enacted at the end of the life cycle of our product, it is something being done constantly throughout the creative process. This will ensure that the product adapts to being consistently changed, as well as providing a more stable and bug-free finished product. Quality Assurance is also the constant revision of matching the actual product to the team's/clients shared vision of the outcome.

Prototyping is an effective process for continuous improvement. As our group is only working on this project, resources can be easily divided into specific roles. Prototyping also allows us to identify problems within the system at an early stage so that they can be amended. We could use prototyping in this instance, as creating the backbone of our software could act as a template for future development, as well as providing the guidelines to develop on other platforms (android rather than web etc).

A Test Case is a set of steps that a developer will create to show how the software should function as it moves from one state to another. Test Cases are vital to ensure that all of the requirements of an application have been fulfilled, as each state could represent a function. Test Cases are usually segmented into 3 stages; Inputs, Outputs and Order of Execution. Input Tests check the response of user input (such as keyboard data entry) to register how data is manipulated, as well as showing how the system responds to errors. Output Tests show the actual function appears to the use, as well as registering the functions activity with other data sources (databases etc). Order of Execution Tests is based upon a cascading style, where one the progression from one state to another is tested. For example, if a function is required to progress to a different screen, this testing format ensures that that eventuality is consistent.

The requirements of the software should be as close to the desired functionality as possible. Outdated requirements are specifications which the stakeholder has asked for which, at the time of implementation, are no longer possible/required. Communication between the developers and the stakeholders is key to prevent outdated requirements, and to maintain to good mutual understanding of the software that is being created.

References

Mind Tools. Contingency Planning Developing a Good ‘Plan B’ [Online]. Available from: https://www.mindtools.com/pages/article/newLDR_51.htm

Project Smart. Project Risk Management: it’s either contingency planning now or emergency relief later [Online]. Available from: https://www.projectsmart.co.uk/project-risk-management-its-either-contingency-planning-now-or-emergency-relief-later.php

TechTargets, September 2015. Contingency Plan [Online]. Available from: http://whatis.techtarget.com/definition/contingency-plan

Total Success, 18th September 2012. Project Management Skills Training – Contingency Planning[Online]. Available from: https://www.totalsuccess.co.uk/project-management-skills-training-contingency-planning/

CNET - Smart Devices Website - Best Smart Home Devices of 2017 [Online]. Available from: https://www.cnet.com/uk/topics/smart-home/best-smart-home-devices/

Amazon.co.uk - Supported Smart Devices for the Amazon Echo [Online]. Available from: https://www.amazon.co.uk/gp/help/customer/display.html?nodeId=201749230

PCMag - Consumer Electronics - The Best Smart Home Devices - Eric Griffth [Online]. Available from: http://uk.pcmag.com/digital-home/85/feature/the-best-smart-home-devices-of-2017