**Information on Writing a White Paper[[1]](#footnote-1) and Use of this Document**

**“A white paper is a persuasive essay that uses facts and logic to promote a certain product, service, or viewpoint**.”[[2]](#footnote-2)

In this case you are promoting what you have done in your project. This means you have to report on your developed tool, technology, design or artefact. Remember **not** to write in the first person in any section of your report. The word count, not including references, appendices or diagram/figure/table legend should not exceed 7,000 words (±10%).

**How to use this form:**

This document is only a template. You **must** change the content under the headers and replace it with your own material. You may also create your own document, but make sure you adhere to the structure outlined in this document.

**Checklist before you submit your white paper:**

* Proof read your document and use the spell/grammar check and correct any mistakes
* Check the format, e.g. are any images, tables, diagrams legible etc.
* Check whether your white paper looks professional, e.g., you should probably use a font that looks professional, not something that looks like a handwritten font (or, indeed, an illegible font); there should not be large areas of white space but sufficient space to make the document look tidy; font and spacing are consistent etc.
* Make sure you refer to figures, tables and references in the body of the text. (see Appendix C for examples for the format of figures and tables)
* Include your signed Deliverables & Requirements sheet in an Appendix
* Include minutes of team meetings in an Appendix



**Smart Eco System - Future of Gardening**

Dorota Marczak, Scott Allan, Nader Sobhi

*CMP311: Professional Project Development & Delivery*

2018/2019



*Please note that Information contained in this document is for educational purposes.*

**Abstract**

**Background to the paper (brief and client) and aim of project**

With the advent of new technologies, which gradually revolutionized countless areas of human life, many employers realized that multiple activities may now be automated, or even performed remotely at relatively low costs. Amidst trending tech solutions such as 3D printing and wireless conference rooms there is one inconspicuous, yet very powerful tool. The Internet of Things, a concept which emerged barely a decade ago, turned out to be a blessing especially for small companies, which constantly compete for shares on the market while having a quite limited budget and human resources. IoT technology can be used for a variety of simple activities, which do not require advanced knowledge or complex operations, performed with users consent and according to their instructions.

Dr Lynsay Shepherd, representing Abertay Plant Systems, decided to take advantage of the opportunities created and ask the local tech start-up - DNS Team – to design and develop a plant monitoring system utilizing above-mentioned technology, as well as similar web based solutions. The client stressed that the company expects a device that could be used by anyone; not only to appeal to their existing customer base but also to “tech obsessed Millennials who care about the environment but don’t keep any plants in their home”. The idea to merge contemporary technology with the age old art of plant keeping was perceived by DNS team as a unique challenge and that is also how it was implemented.

The aim of the project, settled after closer familiarization with provided brief, was to provide an easy to use system that required no expert knowledge to operate and maintain – ideally a ‘plug and play’ device that would begin operation as soon as the end user registered the device on the companion website. Once registered, the device would start to collect data which would then be displayed via the website, as well as via a built in LCD screen. The user could then use this data to ascertain if their plant was growing in the optimal conditions.

**Methodology**

In order to deliver high-quality product within settled time, development of the hardware system and the website was conducted concurrently, which minimized the risk of a potential delay. Team members has been assigned to certain tasks according to their strengths, which facilitated breaking the work down using the Sprint methodology on a later stage. Following Scrum methodology allowed to set clear goals over the course of the development, meaning team leader could track the progress over time. Regular meetings (see *minutes*) were held to ensure the project proceeded as planned, the workload was distributed fairly and finally, that all team members have a chance to air any problems they may have faced with a certain task.

Ready to use product consists of three main elements: Arduino board, connected to a network of sensors and Wemos, database storing the readings, as well as the website, that provides constant control over wellbeing of the plants’ collection. The physical device processes the data received from the attached sensors and sends them in JSON format to the database via Wi-Fi connection, provided by Wemos adapter. The data hold in normalized form are then accessed by the website, that utilizes them to create intuitive data visualizations.

**Outcome**

Professional attitude of each team member to their entrusted tasks resulted in delivering high-quality product a few weeks before settled deadline, what may turn out to be highly beneficial for client’s business and its growth. Smart Eco System, as a user-friendly and visually attractive interface providing reliable information about plant’s conditions and its needs (based on individual settings) has a chance to become a bestseller in the gardening market, which has not fully experienced technological revolution yet. This fact, in turn, may bring common attention of gardeners devoted to their plants and ipso facto originate a mode for smart gardening devices.

Contents

[1 Introduction 5](#_Toc495312033)

[1.1 Background 5](#_Toc495312034)

[1.2 Aim 5](#_Toc495312035)

[2 Procedure 6](#_Toc495312036)

[2.1 Overview of Procedure 6](#_Toc495312037)

[2.2 Procedure part 1 6](#_Toc495312038)

[3 Results 7](#_Toc495312039)

[3.1 Results for part 1 7](#_Toc495312040)

[4 Discussion 8](#_Toc495312041)

[4.1 General Discussion 8](#_Toc495312042)

[4.2 Countermeasures (for a project in ethical hacking) 8](#_Toc495312043)

[4.3 Conclusions 8](#_Toc495312044)

[4.4 Future Work 8](#_Toc495312045)

[4.5 Call to action 8](#_Toc495312046)

[References 9](#_Toc495312047)

[Appendices 11](#_Toc495312048)

[Appendix A - Example 11](#_Toc495312049)

[Appendix B - Example 11](#_Toc495312050)

[Appendix C - Suggestions for formatting figures/tables/screenshots in the body of the text 11](#_Toc495312051)

[Appendix D - Deliverables & requirements 13](#_Toc495312052)

[Appendix E - Minutes 15](#_Toc495312053)

[Appendix F – Other 16](#_Toc495312054)

# Introduction

## Background

While it is true, that you can happily grow your plants with packet of seeds, a sunny patch of land or simple flower pot filled with soil and a water can, there are much better and faster ways to create your dreamed garden. Less than decade ago modern technology crashed into the gardening sector facing long days of developmental stagnation, interrupted from time to time with new biochemistry solutions. Despite almost unlimited opportunities brought by technological revolution, many gardeners was still confining themselves to obsolete cultivation techniques, originating from first half of twenty century and some even from the ancient times. In many cases, the reason behind it is not related to the potential costs, but rather common habit of following traditions and remaining by the things we already know. However, with the advent of GMOs people began to realize, that it is time to take control over their plant’s fate, remaining almost entirely dependent on weather conditions up till lately. That, in turn, switched on the green light for a whole range of modern solutions, including intelligent automatic watering and remote growth optimization based on cause and effect analysis. It must be noticed, that it was Millennials with their concept of a need being the mother of invention that have leaded the revolution to common success. Instead of relying on a gut feeling and mercurial weather conditions, they prefer to help the nature by in-depth exploration of plant’s environmental preferentions and delivering required mineral resources based on extensive analysis.

According to Washington Post, Millennials began paying attention to the plants they place in their urban flats. Nowadays, they do not only serve as a live decoration or a reference to a favourite aesthetic pattern, but are also a way of expressing unique personality of their owner. As ubiquitous social media encourage young generation to share every piece of their life online, modern middle-aged society is under a pressure of being perfect in every inch and mastering every single field. However, first steps in gardening turns out to be harder than most of them expect. They browse the Internet to find the tips for geeks, but then forget to water the plants regularly and end up trying their luck again. Finally, they reach for gardening applications as a last resort in their gardening adventure. Unfortunately, instead of fully-fledged systems they find crawling applications with basic set of advices and watering reminders being a maximum of their possibilities. Having no choice, they install one of them on the phone and start from the beginning. This time they do not forget about basic needs of their plants and that is usually enough to keep them alive for next few weeks. Proud of themselves, they enlarge their little collection with a few more specimens, which turn out to be even more demanding. Irritated by unprecise advice available on the tags, such as partial shade or medium temperatures, as well as hard to estimate irrigation needs, they look for a tool that would facilitate taking care of their flowers. They are eager to pay for solution to their problem, but unfortunately none of the available products fulfill their needs.

## Aim

The fundamental high-level aim of Smart Eco System project was creating a product, that would meet the needs of modern gardeners; both digitalized Millennials just discovering the world of plants, as well as older members of society experienced in the field, but wary of new technologies.

In practice, the project’s aim is delivering a device, that would collect environmental data on a plant and place them into a database. The readings obtained in this way would then be used by a web interface to notify the user about their plant’s current health by displaying sensor data (or their averages) in the form of clear and visually-attractive graphs. The complete system itself, being the outcome of the project, will give the user the means to monitor their plants as precisely and meticulously as if they were placed in a scientific laboratory. Regarding an experienced gardener, such an opportunity could become the beginning of plant growth optimization, which could potentially save them time and money. The general objectives of Smart Eco System were defined as follows: creating a website housing all the tools needed to take care of a plant’s heath based on collected data. As well as creating and managing a database holding all the necessary information including readings and a list of devices released to the market. Finally, delivering a device that would not only provide an intuitive visualization of raw data received from the physical device, but also to suggest actions needed to take base on them.

Although the main goal of the project was to design and build the Smart Eco System along with a user interface, there were several other objectives identified. Firstly, it is expected that the system will provide a set of features, such as an accessible and responsive website, also including the system’s ability to adapt to different screen sizes that it is being displayed on. The application should be easy to maintain, as basic training provided to the members of staff will not cover the full technical specification and the staff themselves may not have technical expertise. Furthermore, it is crucial that the application was user-friendly and visually attractive, otherwise it may dissuade potential customers from purchasing the product. Another important factor is safety – malicious attacks may cause interruptions in service and these, in turn, may lead to claims of a lesser product if they occur frequently. By treating sensitive data with particular care, the company protects itself against credential leaks, which in turn ensures that they will not have to pay out of pocket to compensate for any damages they may have caused. This would also prevent the tarnishing of the image of a reliable brand.

Regarding the physical device, a vital feature must be reliability combined with a smooth flow of data, as the company cannot afford releasing a poor quality product on an already saturated market. Reducing possible data loss is a minimum requirement. While it is important from the client’s point of view, the user may not appreciate a functional device, that is hard to set up and cumbersome in the same time. Last but not least, collected data must be stored in a manner with the relevant laws of operation of the region, and its integrity should be preserved throughout the system’s life-cycle.

Users should be able to clear data pertaining to them -> FUTURE WORK

# Procedure

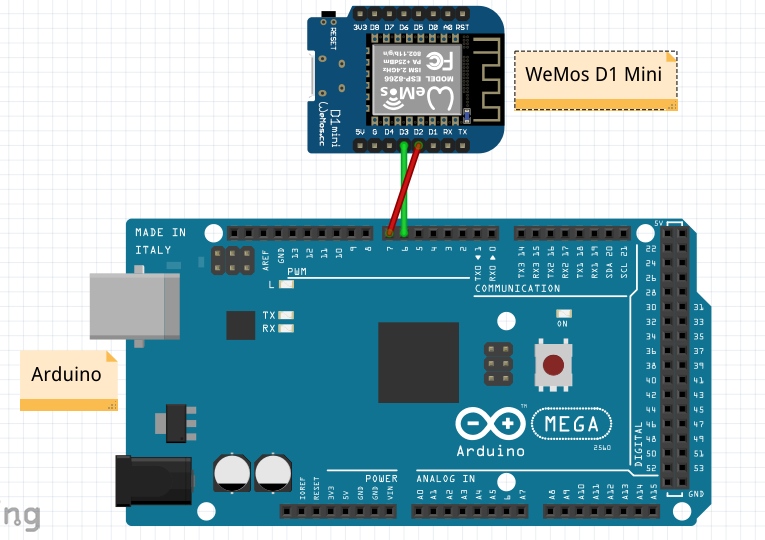
## Overview of Procedure

According to Duncan Haughey, author of Project Smart platform, the key to a successful project is a detailed planning. That is also why the next step, taken after setting the goals of the project and identification of the stakeholders, was dividing the development stage into sections corresponding to the settled deliverables. Each of them consisted of a number of features, which has been established based on an extensive list of functional and non-functional requirements, defined by the brief provided by the Product Owner. Before any development work was done on any of the other components, the organization of the database had to be completed in a way, that would enable reception of data from the physical device, as well as to allow the website to read and process the data held within. Then, the team proceeded to the prototyping stage, where each part of the system was analyzed in terms of both its inner range of tasks, as well as the cooperation with other elements of the set. A vital unit considered during the analysis was to find a way to implement a connection to the Wemos board from the Arduino Uno board. Damage to one of the above could result in the potential loss of the ability to transmit the data, which in turn would lead to the failure of the whole system. To avoid this situation, the specialists worked towards the optimization of the wiring and a special case, that would protect the device against inundation and possible breakage. Meanwhile, the sub-team allocated to the development of the user’s online platform performed broad research in order to gain inspiration based on professional, functional and visually attractive services available on the internet. They then utilized the ideas to prepare paper prototypes, which allowed them to achieve as close a match as possible to the client’s requirements. Once this was completed, the developers could estimate the server space required to host the website, so web hosting could be purchased. When both teams prepared a graphical visualization of the part of the system they were allocated to, the team leader called a general meeting in order to discuss the solutions and make corrections wherever necessary. The implementation process ran smoothly and without any major time delays, however it required some inner rotations due to certain unfavorable circumstances. The team members worked collaboratively on the features identified as more complex and time consuming.

## Procedure part 1 – arduino

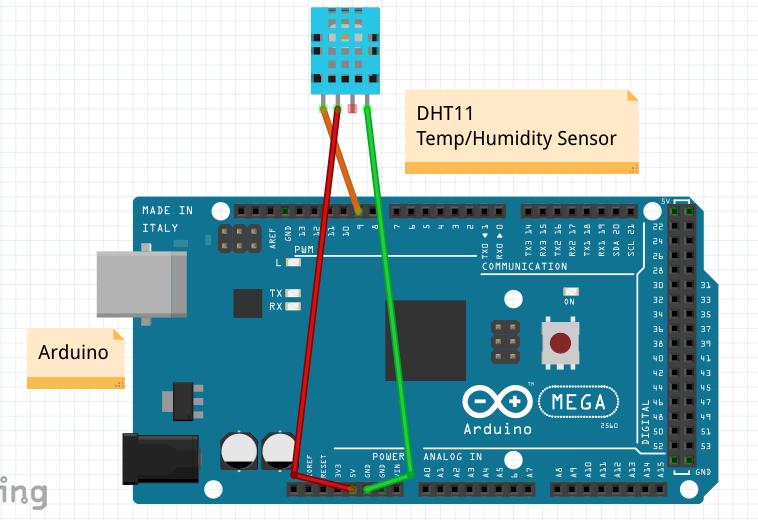
The hardware device was developed incrementally to ensure that each stage was successful and allow for appropriate testing and modification were required. By building the device in small stages, it allowed the team to track progress as well as served as a troubleshooting guide should an error occur. For example, if the device was behaving normally before a sensor was added, only to misbehave upon integration, it would seem the cause of the error lay with incorrect installation of the new sensor. This section will look at how each stage was carried out and how the overall solution was reached. Connection Diagrams will be provided for each stage, with a complete Diagram incorporating the whole hardware system included at the end of the section.

**Connect arduino and wemos**

****

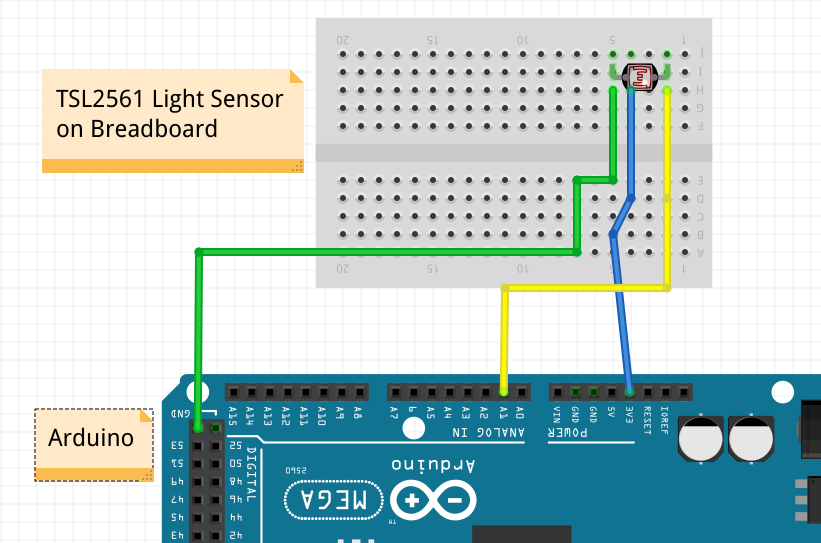
Before we could gather the required data from the sensors to send to the database, we had to give the Arduino the capability to connect to the Internet. This was achieved using a WeMos D1 Mini board. The WeMos is powered by a microUSB connection and uses two pins to send and receive data from and to the Arduino, making it relatively simple to connect and setup. The code running on the WeMos board consists of establishing a connection with the database, listening for the data sent from the sensors via the Arduino, placing the sensor values within a JSON string and finally querying the database and inserting the sensor values in the appropriate table. Once the WeMos has established a connection with the database, it will construct and send a JSON string every thirty minutes. Before any sensors were connected, dummy data was used to ensure that a WiFi connection could be set up between the device and the database server, as well as to ensure that the data was being stored in the database correctly. Once this stage had been complete, the next step was to connect a sensor and obtain useful data.

**Connect temp/humidity sensor**

****

The DHT11 sensor was selected as it enabled the team to take readings on the temperature and humidity in one unit. This sensor requires three pins to operate. It is powered via a 5V pin, requires use of a GND pin and transfers data via a digital pin. A DHT library is also required for successful operation and enables the use of functions specific to the DHT11. The dht.begin() function starts up the sensor, whilst the dht.readHumidity() and dht.readTemperature() functions translate the data received from the sensor into a human readable format. These values are read as floats before being converted to String objects for use with the JSON string.

**Connect light sensor**

****

The TSL2561 was chosen for use as the light sensor as it is very simple and provides easy to use data. The sensor reads the amount of light falling on it in a given environment and outputs this data on a scale from 0-100. This allowed the team to save the data values as an int and display it to the user as a percentage with 0% meaning fully dark and 100% fully bright. The sensor requires three pins to operate: 3.3V to supply power, GND to ground the circuit and an analog pin to provide the data, in this case the A1 pin. A breadboard had to be used as the sensor is very basic and is not attached to a board to provide the pin out functionality. The breadboard can easily be replaced with female-to-male connection wires for a full production model, but the breadboard was kept in place as it allowed the team to daisy chain power connections for other sensors.

## Procedure part 2 – WEBSITE

Designing an online platform, which would provide the means to monitor the plant’s growth, began with establishing a list of functional and non-functional requirements, that it was expected to fulfill.

First and foremost, the website aimed at the presentation of the collected readings in a graphical manner, which would allow the users to perform a precise analysis. That, in turn, would facilitate growth optimization, as environmental conditions play a major role in the gardening world. Also, the platform is required to provide a way to register the device along with the plant it is placed in. Purchasing the product, the user gets access to the activation code assigned to that device. Then, logging into the system and registering the device is all it takes to begin a gardening adventure. Main requirements regarding the website included also accessibility, basic human right by virtue of UN Convention on the Rights of Persons with Disabilities, as well responsiveness, protection of the stored data against malicious attacks and visually attractive, user-friendly interface.

Regarding the complexity of Smart Eco System, as well as the differences between the presentation and functional part of the website it was decided, that the most favorable solution would be splitting the platform into pre- and post-login version. By separation of those logically independent parts, the company got the chance to make the most of the marketing opportunities created. Therefore the front page, being to some extent a business card of The Abertay Plant Systems, was expected to provide basic information about the company and its innovatory product released recently as a part of its pioneer project. In order to organize the development process, all the requirements were converted into an extensive list of features. ADD HERE

Once the features has been allocated to certain sprints,

# Results

Put your results in here. Any tables or results should be included here unless there is a large amount of data. Appendices should be used for large amounts of data and referenced in the text. Only important details should be included in this section, i.e. material that convinces your client about the (hopefully fantastic) performance of your design/tool/etc. Consider what is important here to make your project look good.

## Procedure part 3 - Database

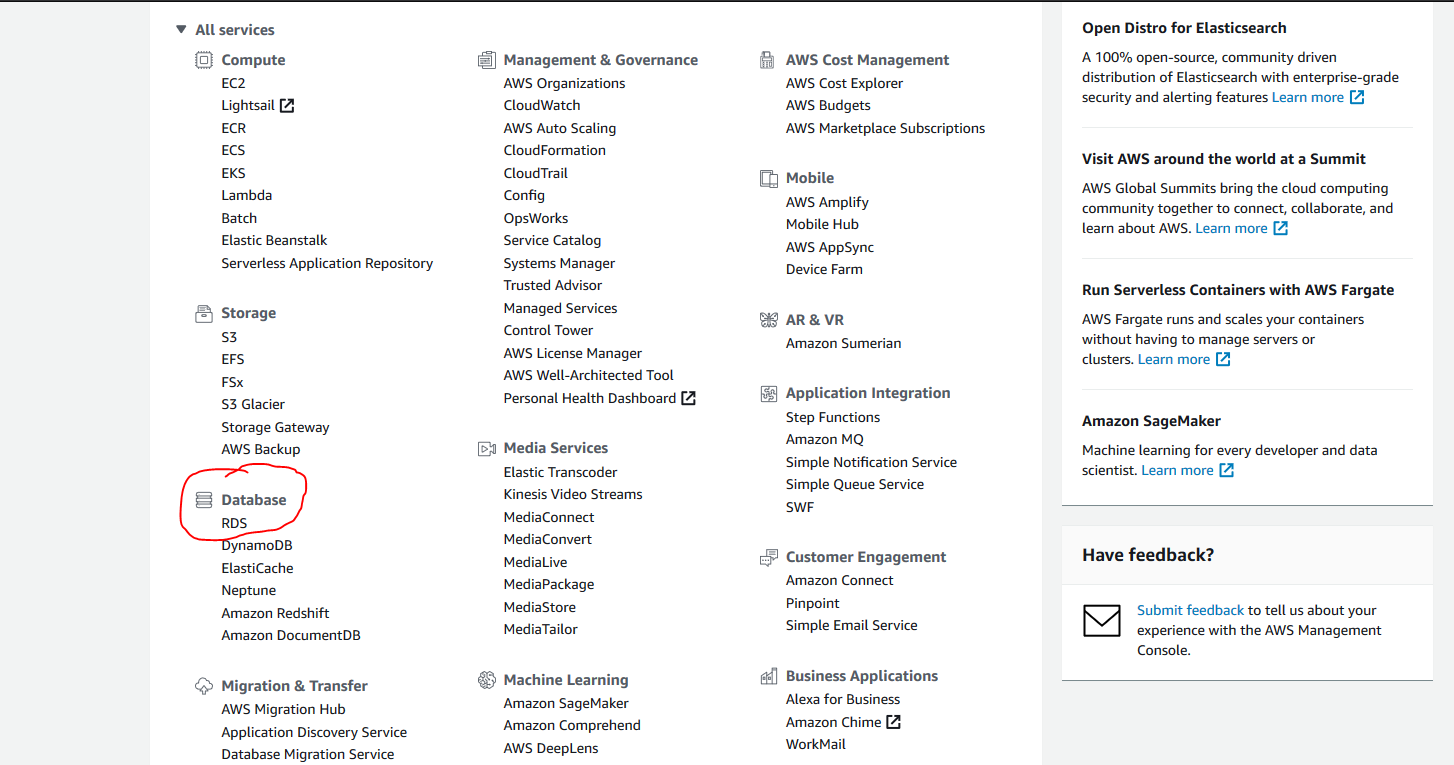
Storing the data that is produced and used by the system is one of the main foundations that the application is built on. This is due to the fact that the application relies heavily on the ability to store information for later use, the storage of data will mainly be performed by the Arduino System and the retrieval will be performed exclusively by the website. The format in which the data is stored is combination of a Relational Database (RD) Model and a NoSQL model. The reason for this being that while some of the data will be highly structured the rest will be of varying size and types (i.e. integers, text, real numbers). In order to simulate the potential conditions that will be in place once the system goes into production it was decided to use a platform that could host the potentially considerable amount of data that will be held. The platform that was deemed most appropriate was Amazon Web Services (AWS) which allows its users to lease all kinds of servers and server space ranging from simple RDs to servers setup to perform well with machine learning code. In our case a simple server setup to host a RD was all that was necessary. The setup process followed to lease the server space, create the database, and create the tables will be shown and described below.

Figure X1

As seen in Figure X1 the user will be presented with this page once they sign up or sign in to AWS, from there they can select the type of service they want to use. In the case of this system an RDS instance will be created. Following this the user will then select the “Create Database” option, shown below in Figure X2, and follow the instructions displayed by the Setup Wizard.

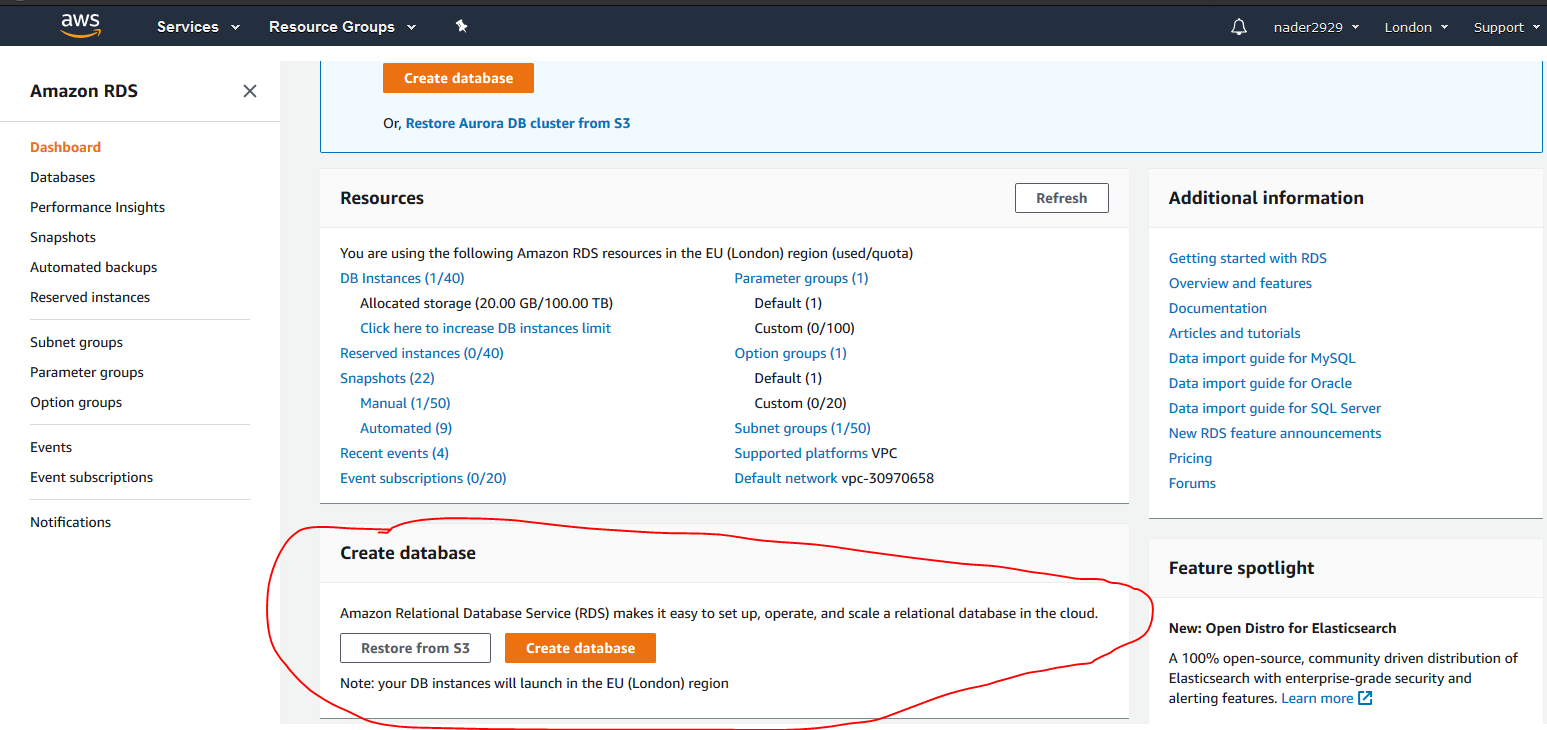


Figure X2

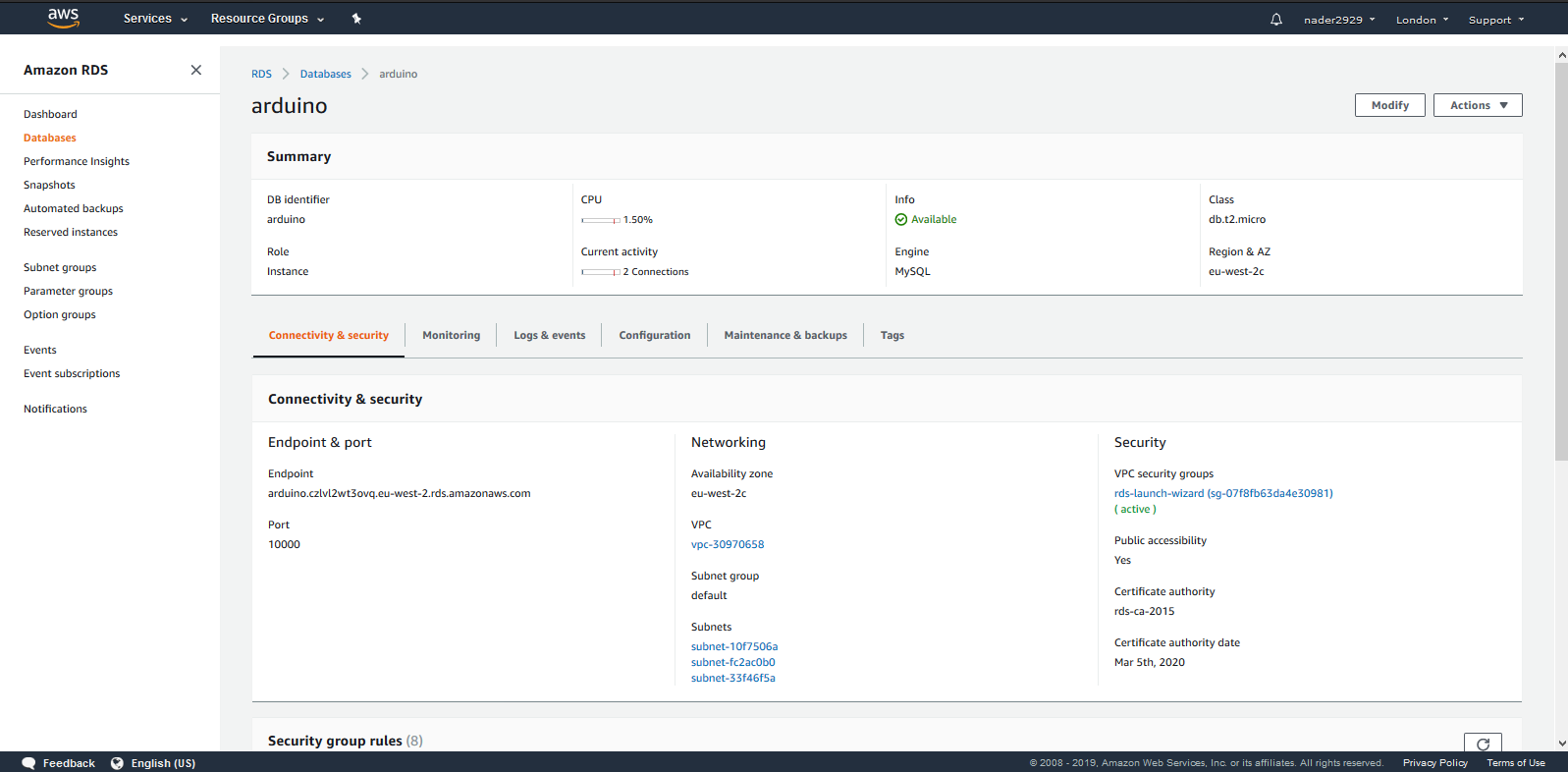
Once the setup was complete the user could select the instance they create and change any settings they wish using the panel provided, Figure X3 showcases this panel.

Figure X3

From there the user could use their choice of Database Management Software to create the tables they need or want to create. In the case of this system the developer decided to use MySQL Workbench to implement the design they had in mind. MySQL Workbench allows the developer to write SQL code that will perform the desired actions, such as create tables, insert data, or form relationships between the entities. The first table that needed to be created was the one responsible for holding the readings recorded by the Arduino devices, the code used to achieve this is shown below in Figure X4.

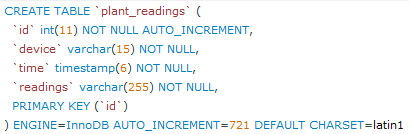


Figure X4

The table was then populated by the Arduino once it was operating correctly and could be used in the future by the website to present the readings in a graphical manner. The ‘readings’ column of this table is where the NoSQL aspect comes into play. Due to the fact that the readings produced from the devices could in the future be extended, i.e. in the future more sensors could be added to the device and this would not affect the database. Therefore no changes will need to be made to the database.

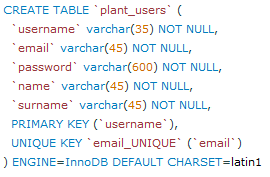


Figure X5

In Figure X5 the code used to create the user table is shown. Once this table was created and the website’s code that was responsible for populating the table was properly setup and functional users could easily login and register on the website.

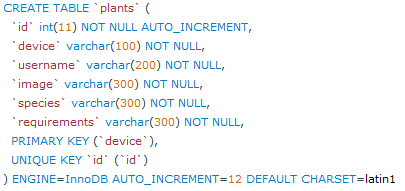


Figure X6

In Figure X6 the code used to create the plants table that would hold the relations between the devices and users. This table would be populated by the users once they login to the system and add their devices to their accounts.

## Results for part 1

* Include results for part 1
* Use as many sub-headings as necessary.
* If you are using software, web designs prototypes etc. then screenshots of important features should be included here – with descriptions. If it is a hardware project then pictures / diagrams should be included here. These should be clearly labelled (for example “Figure 1 list of wireless LANS”) and referenced in the text. For example “see Figure 1 for an example of the results at this stage in the project”
* Note that a reader should be able to re-create your work from this description. When you write this section, keep this in mind.

**Results for database**

The database used to hold the data used by the system acts as a type of bridge between the website and the device so naturally the developers needed to ensure that it was working correctly and securely. Luckily, the testing and the subsequent results for the database are minimal as all that needs to be ensured is that the database cannot be accessed from the backend i.e. an unauthorized user being able to login to the database using a Database Management System. The other factor that could be tested was the prevention of SQL injection type attacks although this was mainly mitigated on the website and not directly through the database. In this section only the backend security of database will discussed and shown.

The actions that the developers took was to set a unique username and password for the database that differed vastly from the default options, as well as switching the port that the database operates under from the default 3306 to port 10000. These changes are shown below in Figure Z1 with the password being redacted by the setup screen of MySQL Workbench.

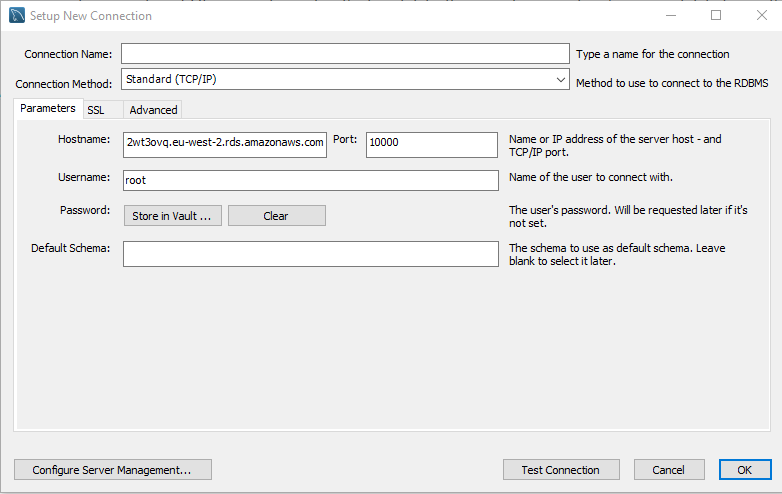


Figure Z1

In order to showcase the databases ability to hold data in an organized manner screenshots of some of the data held in each table will be shown below.

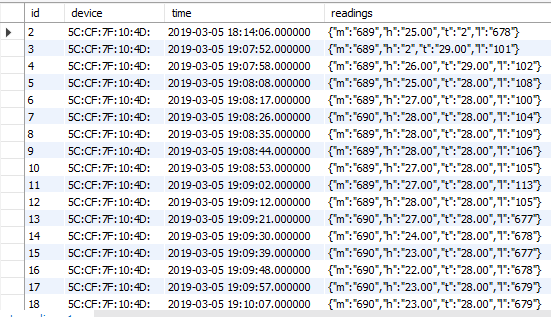


Figure Z2

Figure Z2 shows the *plant\_readings* table columns as well as some of the records held in the table.

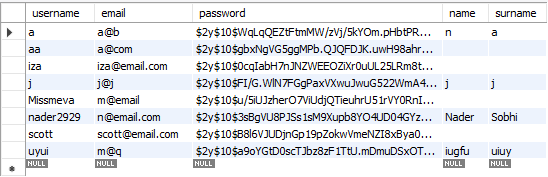


Figure Z3

Figure Z3 shows the *plant\_users* table columns as well as some of the records held in the table.

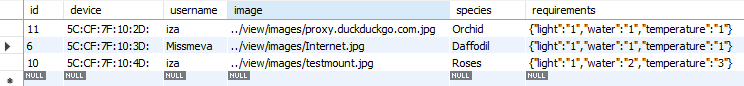


Figure Z4

Figure Z4 shows the *plantrs* table columns as well as some of the records held in the table.

# Discussion

## General Discussion - Scotty

Here, you want to discuss your results/outcomes.

* What does it all mean? Discuss anything of interest. How do findings relate to other work in this area?

* Relate the findings back to your aims - **how well have you met your aim**? Consider performance indicators.
* Discuss, for example, return on investment (ROI), usability, adherence to standards, speed of implementation, exploits found etc. Ensure that you make clear you understand your client’s pain and can relieve it.

## Conclusions

Smart Eco System is a unique product, designed to help the client attract not only gardeners actively looking for the newest solutions of plant growth optimization, but also Millennials, who currently constitute a minority in the gardening market. Having access to the readings presented in the form of visually attractive graphs, as well as the indicators showing plant’s demand for water, all they need to do is to regularly check it’s status by logging into the online platform.

Moreover, such a technology could potentially raise the prestige of the company and significantly contribute towards the business’s development, and thus increase the number of connections in the industry. It would change the perception of the company from a small, local shop to competition others have to reckon with. That, in turn, would translate into bigger profits due to their new-found reputation as a pioneer of smart home equipment.

However, if the client, for some reason, decided not to release the product, the aforementioned competitors would definitely take advantage of the company’s stagnation and introduce interesting technological solutions. By doing so, customers would be enticed to purchase their gardening tools from their stores and not from Abertay Plant Systems.

## Future Work

As with any project created to fit a client’s specification there are some features or aspects of the system that were identified as potentially beneficial but were not implemented due to some extenuating circumstances. Starting with the database a NoSQL solution could have been implemented instead of the current model which currently consists of a mix of Relational Database Scheme (RDS) and a NoSQL scheme. The reasoning for this was that the information stored currently would only consist of some basic user information, the sensor readings from the device(s) and nothing else. However, in the future it may be prudent to switch the storage method used for this data to a NoSQL solution as it would allow for the easy expansion of any of the data centric features. An example of this would be allowing multiple people to use the same account and therefore expanding the users table. This is really the only change, identified by the developer, that could be implemented with regards to the database.

As opposed to the database the physical device responsible for collecting readings on a plant’s surroundings could be improved in a multitude of ways. These are mainly quality of life changes and therefore were not implemented at this point in time. Beginning with the implementation of some sort of interface for the Arduino Collective so as to allow the user to perform a few actions that are currently unavailable or only available through the web application. As far as the developer can identify there are only 3 ways that an interface could be implemented to allow the user to control the device’s functionality. The first would be a custom-built application that would interface with the device once it is connected to the user’s personal computer. This version would allow the user to make changes to the device’s functions through the interface and then the appropriate instructions would be loaded onto the device. This iteration would be the simplest to implement as the Windows OS can already recognize when an Arduino device is connected. The second way to implement an interface would be to use a display and a number pad that operates in a similar fashion to the keys found on older mobile phone devices. This would allow the user to directly control the device without the need for an intermediate medium. Although a functional and novel idea for an IoT device, the implementation of this idea was deemed as too difficult to complete within the allotted time. Finally, another method that could be used to implement an interface would be to use the website to allow the user to set some options that would control the functionality of the device and then load those the appropriate device instructions onto the device. This is probably the hardest, of the three models described for device interfaces, to implement as it requires that two-way communication is possible between the device and the website. Additionally, it would require more storage for user preferences as well as more data to be collected on each individual user. Continuing with the device, another aspect that could be implemented in the future would be to allow the user to sustain or nourish the system through the device itself, mainly watering and providing light to the plant. This feature ties in with the previously mentioned future improvement of implementing a device interface as it would allow the user to not only monitor the devices status but to react by activating a watering or lighting mechanism to provide sustenance to the plant. The reason this was not implemented was that the developers felt that this would be fairly trivial to implement at a later time when some type of interface had been designed and attached to the device. Lastly and least importantly at this stage would be the design and construction of a plastic 3D printed case for the device to be housed. This was done due to the extremely limited access the developer had to such a printer.

The website is the final component which holds potential for future upgrades or modifications. Beginning with the addition of a feature that would allow the user to regain access to their account should they forget their password. There are plenty of methods that can be used to implement such a feature, including email reset links or other two factor authentication methods. This was, again, not implemented due to time constraints and not due to difficulties in implementation. Another security feature that will be implemented in the future will be the use of the HTTPS protocol to encrypt connections between the users and the website. This was not implemented because the web hosting used did not support the developers preferred method of adding the required certificate to the website and thus would require more work and time than available at that point in time.

Apart from security features there are some functional aspects that have not been implemented due to time constraints as well as the fact that they were deemed as small quality of life changes and therefore would not improve the functionality of the system by a significant amount. That being said, they will be discussed here as potential improvements for the system in the future. Starting with the implementation of a feature that would enable users to discuss plant care with plant experts or members of the user base who have proven to be extremely adept at plant care. This would allow for a kind of social network that would enable users to interact with other people who share the same passions as them. Another quality of life change that would improve the overall usability would be to add a feature that would allow users to upload a picture of their plant so that a trained AI model that would identify the species of the plant and suggest methods to improve the plants health. This feature could also be extended to utilize the portion of the website that asks the user to enter what they think are the ideal environmental conditions for their plant. While this is a novel and potentially useful idea, to build such an artificially intelligent model would require a large amount of development time as well as a huge dataset of pictures of all the most likely types of plants to be used with the system. Meaning that it was not be possible to implement at this time. Finally, another feature that could be implemented in the future would be to add a “wiki” type section of the website that would host all kinds of information about plant care. While very easy to implement this can be very time consuming as it would require the developer in charge of the task to do extensive research so as to ensure that the information hosted on the website is accurate. This is also the reason this feature has not been implemented as of yet.

## Call to action - scotty

To find out more about how you can gain the power to track and optimize your plants lifecycle with the Abertay Plant Monitoring System, visit [www.abertaypms.com](http://www.abertayps.com) today to arrange a free, in-depth demonstration. The first fifty people to sign up for a demonstration session will receive 25% off the total cost of purchasing the system and six months of ongoing training and technical support. The first hundred people to sign up will receive six months of ongoing training and technical support. The only way to take advantage of these offers is to visit [www.abertaypms.com](http://www.abertaypms.com) today and register your interest in the Abertay Plant Monitoring System!

References

**For URLs, Blogs:**

Bremer, J. 2012. *x86 API Hooking Demystified*. [blog]. 2 July. Available from: [http://jbremer.org/x86http://jbremer.org/x86-api-hooking-demystified/api-hooking-demystified/](http://jbremer.org/x86-api-hooking-demystified/) [Accessed 15 April 2016].

**Ours:**

**https://www.smallbusinesscomputing.com/News/Networking/what-the-internet-of-things-means-for-small-business.html**

<https://www.w3.org/standards/webdesign/accessibility>

<https://www.projectsmart.co.uk/project-planning-step-by-step.php>

# Appendices

## Appendix A - Example

Put any large amounts of data here (e.g. code).

**Hello.Php**

<html>

<head>

<title>PHP Test</title>

</head>

<body>

<p>Hello World</p>

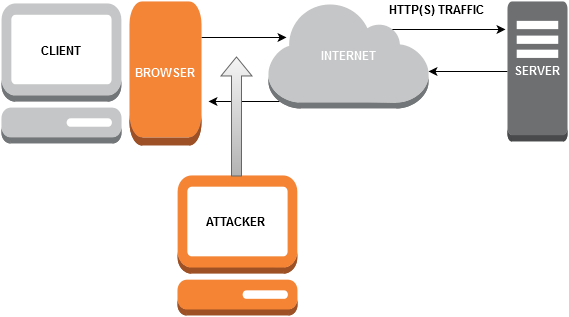
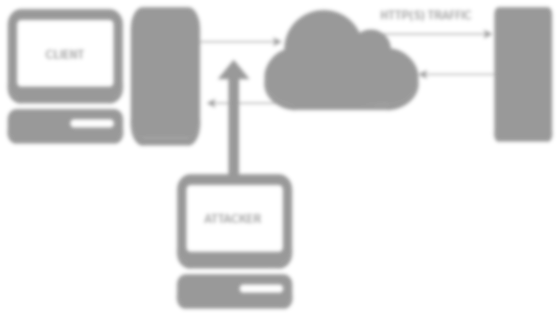
</body>

</html>

## Appendix B - Example

And here….

## Appendix C - Suggestions for formatting figures/tables/screenshots in the body of the text



**Figure 1-1** A diagram of the system that was used in the security test.

**Table 1-1** Advantages of the X versus Y

|  |  |
| --- | --- |
| **Advantage Description** | |
| **1** | Can be programmed easier. |
| **2** | More flexible that the traditional system. |
| **3** | More secure than the traditional setup. |



**Figure 1-1** Snippets of code can be formatted like this.

## Appendix D - Deliverables & requirements

**Agreement Form: Project Deliverables**

|  |  |
| --- | --- |
| **Group Number, Names of Team Members, and Programme** |  |
| **Programme specialist’s Name** |  |
| **The deliverables listed below will be submitted by the team by the due date.** | |
| **Part A deliverables** | **To be agreed by programme specialist and team, for example:**   * Executable code/investigation report * User/reference manual |
|  | * Requirements Specification, signed off by the programme specialist (see overleaf) * Software Design specification or investigation methodology * Testing * …. |
| **Programme specialist’s signature** |  |
| **Team members’ signatures** |  |

**Agreement Form: Requirements**

Group Number:

Team members (print):

Project Title:

Please refer to the attached documentation for full details on the project. The requirements are listed in Table 1. The signatures below indicate that the requirements for this project have been agreed by the project stakeholders.

Any changes to the project documentation should be made using the correct change authorisation procedure agreed with the programme specialist.

Table 1

|  |  |
| --- | --- |
| **ID** | **List of Agreed Requirements (fill in)** |
|  |  |

|  |  |  |
| --- | --- | --- |
| **Stakeholders** | **Signatures** | **Date** |
| Team members |  |  |
| Programme Specialist |  |  |
| Client (if applicable) |  |  |

## Appendix E - Minutes

This section should contain **detailed** minutes of meetings.

This should comprise:

• Date, time and venue of the meeting

• Who is present

• Any absences and apologies for absences received

• Approval or amendments of previous minutes if available (review)

• A description of what is discussed and agreed upon (or not) in the current meeting

• Any challenges flagged up (in the team) and how were they addressed

• Actions for next meeting

• Date, time and venue for next meeting

There should also be a brief section on the atmosphere/challenges within the team, how team dynamics were addressed etc.

## Appendix F – Other

1. Document prepared and revised by Natalie Coull, Colin McLean, Andrea Szymkowiak [↑](#footnote-ref-1)
2. Graham, G., 2005. *The White Paper FAQ (Frequently Asked Questions)/That White Paper Guy – Gordon Graham*. [online] Available at < <https://www.thatwhitepaperguy.com/white-paper-faq-frequently-asked-questions/#what_is>> [Accessed 9 May 2016]. [↑](#footnote-ref-2)