Institute of Technology, Carlow

<u>Computing</u>

Course: BSc (Hons) Software Development Year 4

Author: Eamonn Gaynor

Student ID: C00197458

Tutor: Mr. Nigel Whyte

Document: Final Report

Plant Disease Identification Application

Abstract

The purpose of this report is to give a detailed overview of the project completed. It will outline changes to the project overtime as compared to what was originally outlined previous, it's specifications, the functionalities and how they interact with one another, achievements, learning outcomes both personal and from a technical perspective, reviews, future features for both applications, and the acknowledgements to those who were involved throughout its duration. The project reflects what is the finished product and will explore what may have been done differently to overcome certain challenges and obstacles that were identified and dealt with.

Contents

1. Project Idea

Plant diseases and pests cause the loss of up to 40 % of the annual harvests, a challenge for a growing world population. (The Crop Site, 2017). Dependable, accurate and precise examinations of plant disease is crucial in predicting yield loss, monitoring disease resistance and preventing epidemic outbreak. Assessments of plant disease which are inaccurate may lead to a misleading diagnosis being drawn from the examination, thus leading to poor disease management and/or loss of the plant itself. (Bock et al., 2010). Smart farming, which is the application of modern Information and Communication Technologies into agriculture, now called the *Third Green Revolution* can have a huge impact on this area of farming, thus reducing the burden for farmers globally. (Smart-AKIS, 2017)

Scientists at *EnviroCore* research innovative environmental technologies along with biotechnologies which enhance the economic and social development in an environmentally friendly manner. (EnviroCore, 2017). Currently at IT Carlow, scientists at *EnviroCore* carry out visual inspections of plants leaves and make subjective opinions on the overall health of the plant. The aim of this project is to create an application for smart devices which will allow users to identify plant disease through the means of artificial intelligence using computer vision techniques. The goal is to automate visual inspection and eliminate the ambiguity associated with a diagnosis, in turn reducing the subjectivity of the process. By using JAVA and PHP as the core programming languages both a mobile application and web application will be produced.

2. Introduction

This document reflects what is the final product which is titled '*Plant Vision'*, a plant disease identification application. The application produced as part of the fourth-year project module in the Software Development course in the Institute of Technology, Carlow. The document will describe what was required of this project, what was achieved, what has been completed from the original specification as compared to the final product produced.

Plant Vision allows the client to derive what percentage of disease a plant may have, based an image they have uploaded via the mobile application. The aim of the application is to remove subjectivity regarding diagnosis, it allows the user to track a plants deterioration or rejuvenation over time. The mobile application works in tandem with the web application whereby the user may log in to the web application to edit the leaf through analysis results, and view reports which have been comprised of uploaded data over time.

As the project progressed, ideas that were originally supplied had changed as continuous meetings with *EnviroCore*, where key functionalities and designs were discovered and discussed in which they felt would be important to the application, thus impacting specification.

3. Plant Vision explained

As previously stated in this document, '*Plant Vision'* was the chosen title of this project and its subsequent applications. The title was chosen because the project incorporates computer vision techniques, applying those techniques to the images of plants, in turn creating a diagnosis of disease, hence the name the '*Plant Vision'*.

The application allows a client to first register with the system, either by the mobile application or by the web application. They may then log in through the mobile app, from where they can then upload an image of a leaf to the system. By choosing a photo which they wish to upload, they give the leaf a name, and assign it to a group e.g. Group A, Group B etc. Once uploaded, they may log into the web application, choose the leaf they have just uploaded and view key information. Information provided by the system includes the GPS coordinates of where the image was taken, important from a user's perspective as there may several scientists using the one account, this allows the user to identify exactly where the plant was taken. e.g. laboratory, farm etc. Other information included is weather details at the time the image was taken, values such as temperate, humidity, wind speed, pressure, and the description of the day are outlined. This information allows the user to better understand the diagnosis more accurately, as weather conditions can have consequent factors on the health of a plant at a certain time. Both the GPS feature and weather feature were advised by *EnviroCore,* thus implemented.

As well as the above, the key feature is where by the user can access the results of leaf with regards to the disease. The leaf is broken down into twelve of the most common colours associated within the image. These colours are presented to the user in a percentage format, from there they can identify to which colour is the disease, and also remove and unwanted information from the analysis e.g. background colours. By doing so, they are teaching the system to remember their selection for results down the line. The user has only to do this once for every group, and it allows the system to learn from previous analysis. The next time the user uploads an image on the mobile app to the same group as previous, they have an option to automatically diagnose the disease. It can therefore identify the part of the leaf which is diseased and from the mobile app it presents the user with the updated percentage of disease coverage. Once the user has uploaded more than one leaf, they may then access the reporting too. The reporting tool uses graphs and charts to present the information of the plant group over time in a useful manner, they may also download a CSV file associated to this information.

The system must learn from the user, as explained to me by *EnviroCore,* a disease is never as clear cut as one solid colour vs. another e.g. dark brown vs. green. Most times, the disease (or their area of interest) could be concentrated on a darker shade of colour, this may be caused by something like high humidity conditions in the air. Therefore, initial automatic disease identification from the technology perspective may be deemed as inaccurate, as it may not be what the user is looking to concentrate on as per reasons above. Throughout the document, the above features will be explored in detail to explain how the application operates and how the features are achieved in greater detail.

4. Change to System Architecture

The below image (fig 1.1.) is a visual representation of the system architecture. This has been included since the specification demands have changed, as to has the functionality. The initial diagram can be found in the functional document supplied with this project.

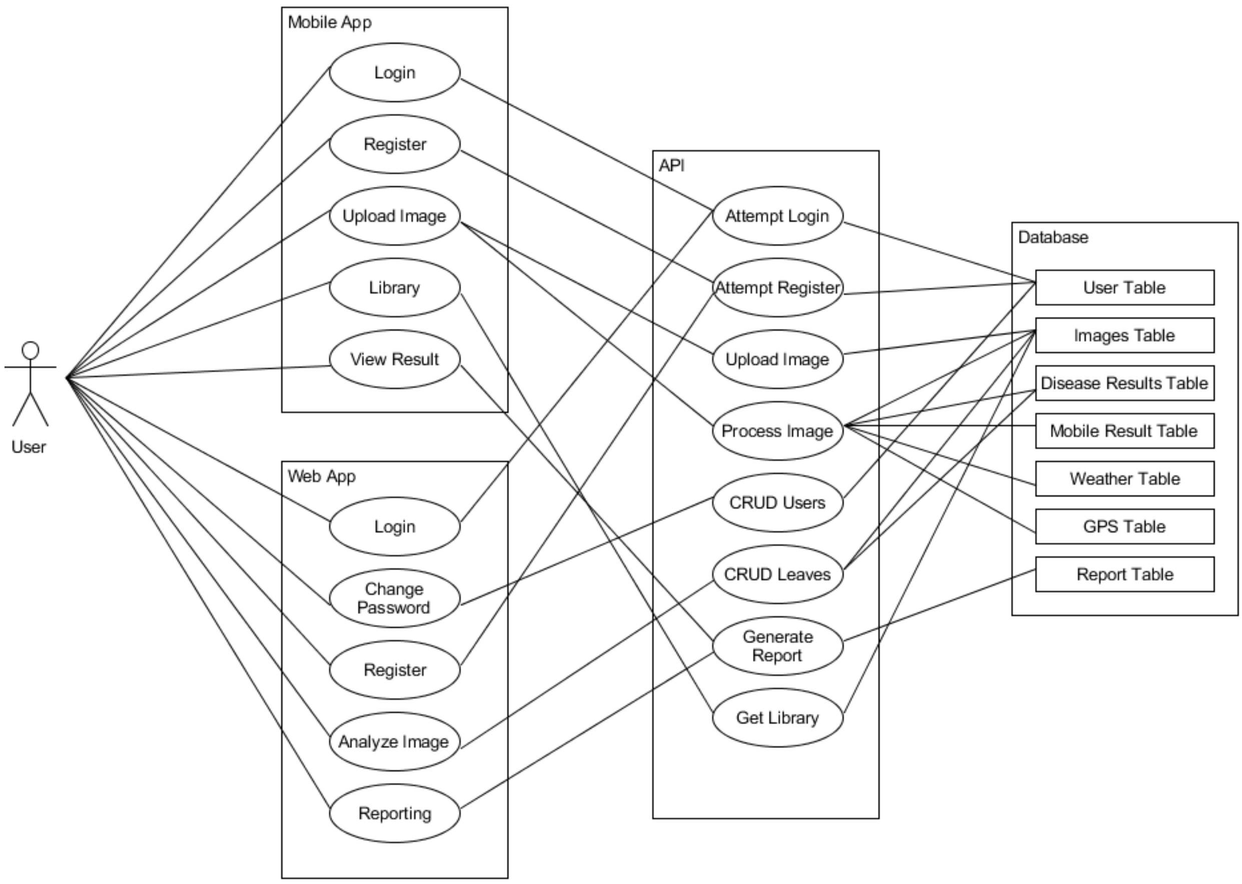


fig 1.1 System Architecture

5. Iteration descriptions

5.1 Iteration 1

The initial iteration, iteration one of the project began in October 2017. The projects initial specification was explored through discussion and critical thinking, identifying the core technologies, functions and designs. This was done by researching prior applications that may have had similar objectives to that of this project. The initial technology chosen to identify the disease was that of OpenCV, a C++ inbuilt library for computer vision. This library was explored through writing different image processing techniques such as Histograms, Fuzzy C-means, Canny Edge detection etc. These example programs were written in C++ through Eclipse using the loaded OpenCV library. However, after further discussion of the specification, for the purpose of this project where the different colours of the leaf were the main focal points, the more practical option would be to use the API which was would be written in PHP. Within the PHP language, there is a built-in function called 'colouratindex' Gelotte, K. (2018), which retrieves an index value of pixel coordinates which can then be converted to RGB values. What also drove the decision to move away from OpenCV was that the mobile application was written in JAVA, making it difficult to incorporate OpenCV within the application as it is written predominantly in C++. A MySQL database was chosen as the storage option for the data, with the images uploaded being stored on the root directory of the server.

Once the initial languages, core functions, design features had been identified, building had commenced. For the first iteration I had chosen to start on the mobile application. Applying key design techniques and building what will be the core functionality of the application as a whole. The login, register, choose photo functions were also built. Where the user may register an account on the server, from there logging in under those details. The image captured was displayed on screen, from this point I had created what was a starting foundation from which I could build upon.

5.2 Iteration 2

Iteration two began in January 2018, following on from the first iteration of the project. After completing the task of the initial design for the mobile application, work began on more advanced functionalities were built as opposed to the login and register function. As part of the second iteration, I met with the researchers in EnviroCore, I sat down with them and we discussed possible features, designs, requirements from their perspective. Some of the functionalities outlined at this point were the inclusion of the GPS coordinates, so they could view the source of the image on a google map, the inclusion of the weather conditions and also the reporting function to which they may view the leaf over a period of time with different graphs and charts. At this point the actual need for the project to identify to which disease the actual leaf is associated with, was deemed not necessary from their point of view, more so to track the diseases progression, depression over a period of time.

With regards to coding, the second iteration began by building the function to allow an upload of the image to a server. Difficult at the beginning as I had difficulties storing the photo as a 'Large Blob' datatype on the SQL database. This method is not only slow in processing time, but also takes up a considerable amount of memory on the database itself. I overcame this by saving the photos on the root directory of the sever, with the corresponding URL to the image being stored in the database alongside the user's id and the id of the image itself. This function allowed the user to enter the name of the leaf, the group to which it belongs, and the upload of that image saved to the server. Once this was completed, I then started work on the web application by creating a login and register function. The mobile and web application were now working in tandem with regards to the user table, register and login available on both devices. From there I built an API written in PHP resting on the server. This one file would soon become the focal API for all core functions that would be written in the second iteration and onwards. The API looked after the uploading of an image, and the method for which an image would be processed. Once the image was chosen to be uploaded, it would be passed through a method which identified the most common colours. Retrieved as an index value to that of the pixel coordinates, I converted these to RGB values, and then converted the RGB values to a Hash value. Once I had gathered all values, I calculated the count for each, and converted to a percentage. These percentage values were then outputted to the user via the web application once they logged in and selected the image.

5.3 Iteration 3

Iteration three began in March 2018, following directly on from the second iteration. After building the core foundations for the project in both iteration one and two, I had gathered a solid base from which to work upon.

Iteration three began by building the reporting tool incorporated within the web application. This was completed by using a PHP wrapper class called 'FusionCharts'. These wrappers were fed information by SQL query's in which information as retrieved. The charts allow the user to view the plant group over time through charts such as line, and graphs such as pie and bar. The extra feature was also added at this point allowing the user to download the data as CSV file, for their own data extraction.

Once the reporting feature was completed, I concentrated my time on the learning model of the system. I did this by allowing the user to identify to which part of the leaf was diseased, and to remove any values that they did not wish to include e.g. background colours. Once that data was collected, I programmed the mobile application to look for these values upon an upload of the similar group. The mobile app would then remove unwanted colours, and pre-mark the disease. This figure was then outputted the user following upload.

Once the above was completed, I began working on the advised features such as GPS and weather. I built these into the core API, whereby once an image was upload, a fetch would be sent to a weather API (OpenWeather.com) including the users IP address to get their approximate location for weather purposes. With regards to GPS, on the Android app I included a function which asks the user for permission to access their location, from there taking the latitude and longitude values and converting them to an address on Google. This information was from there presented onto the web application alongside the weather in a google map format.

6. Application description

The project is as stated above, comprised of two applications both mobile and web. The web application being built in HTML, CSS, PHP and the mobile applicating being built in JAVA to allow for native android features. Both applications are responsive to a API on the server. This section will outline per each particular screen, its purpose and description of how it works.

6.1 Mobile Application

6.1.1 Splash Screen

fig 6.1 Splash screen

Upon launch of the application, this is the first screen (fig 6.1) the user is presented with on the Plant Vision app. This screen is a simple splash screen, with the get started button moving the user to the login screen where they may log into their account. From a technology perspective, the get started button launches the GPS method, which retrieves the coordinates of the user's device, presuming they have successfully allowed the application to access their location.

6.1.2 Login & Register Screen

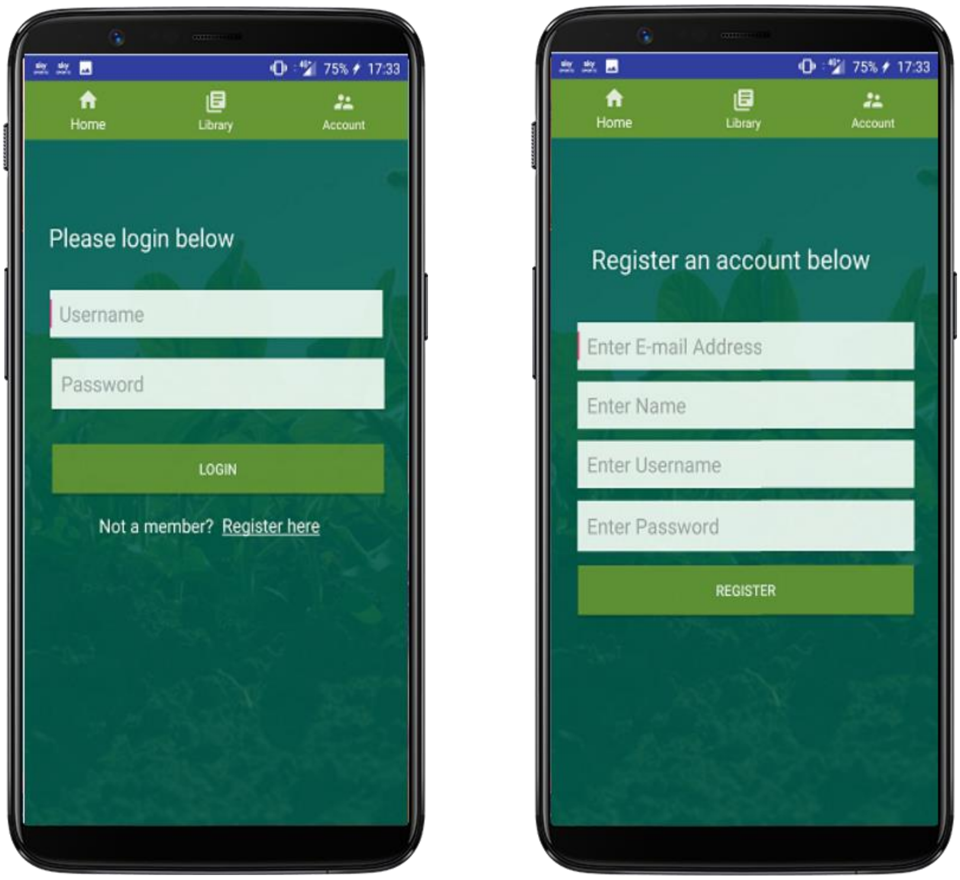


fig 6.2 Login and register screen

Once the user has passed the splash screen, they are presented with the login page as seen in fig 6.2. This login feature when populated with the correct credentials, allows the user to access the home screen of the application and use the core features of the application. They must login with their username and password provided upon registration to the system. By clicking 'Register here', they are presented with a register form, as seen also in fig 6.2. By registering their email, name, username and password, they may then log in with those details in the previous screen.

6.1.3 Home Screen

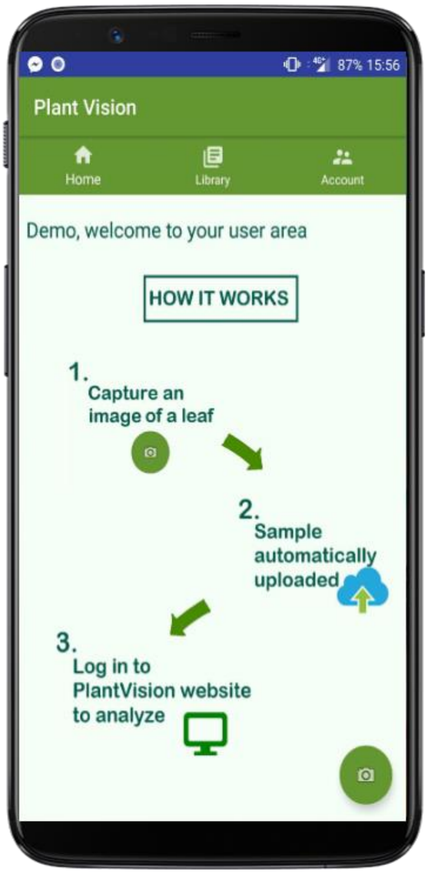


fig 6.3 Home screen

Once the user has successfully logged in, they are presented with this home screen as seen in fig 6.3. The home screen provides a quick tutorial on how to use the application, explaining how to capture an image, to upload the sample chosen, and to log in to the web application to analyse. The option to choose an image is found as a floating button at the bottom right of the screen. By clicking this button, they can then choose an image from their device and select it to upload. As can be seen from this screen, a navigation bar is also available at the top, options included here are to return to the home screen, visit the library of previously uploaded images and to change the account.

6.1.4 Upload & Result Screen

fig 6.4 Upload screen

fig 6.5 Result screen

From successfully choosing a photo from the previous home screen, the user can then view their selected image on the upload screen as seen here fig 6.4. From here, the user can type the name of the leaf, and create a group name for the plant, as to categorize them for later analysis. As seen beside the text fields, the user is also presented with list of active groups to which they have already entered previous. The switch titled 'Automatic Disease Detection' is also provided to the user to allow them to view the percentage of disease coverage after the image has been upload. This option is available provided the user has previously analysed the original upload to that group chosen.

Upon successfully selecting the automatic disease detection option from the previous upload screen, the user is presented with a percentage of disease coverage (fig 6.5) relating to the image they had just uploaded.

6.2 Web Application

6.2.2 Login & Register Screen

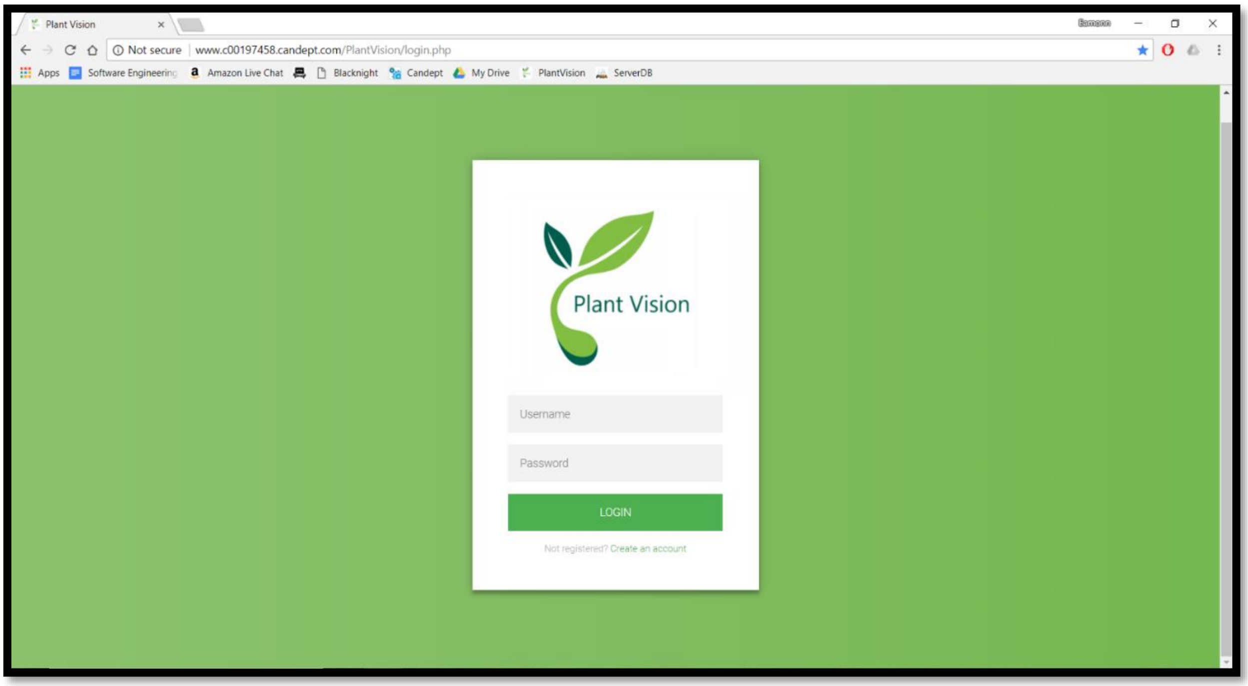


fig 6.6 Login and register screen

Upon launching the web application through the URL provided, the user is presented with a login screen as seen in fig 6.6. The user may log in with the details they had previously registered with on the mobile application, or if they have not already created an account they may do so by clicking on the 'Create an account' option seen below at bottom of this screen. Once successfully logged in the user will be presented with their home screen.

6.2.3 Home Screen

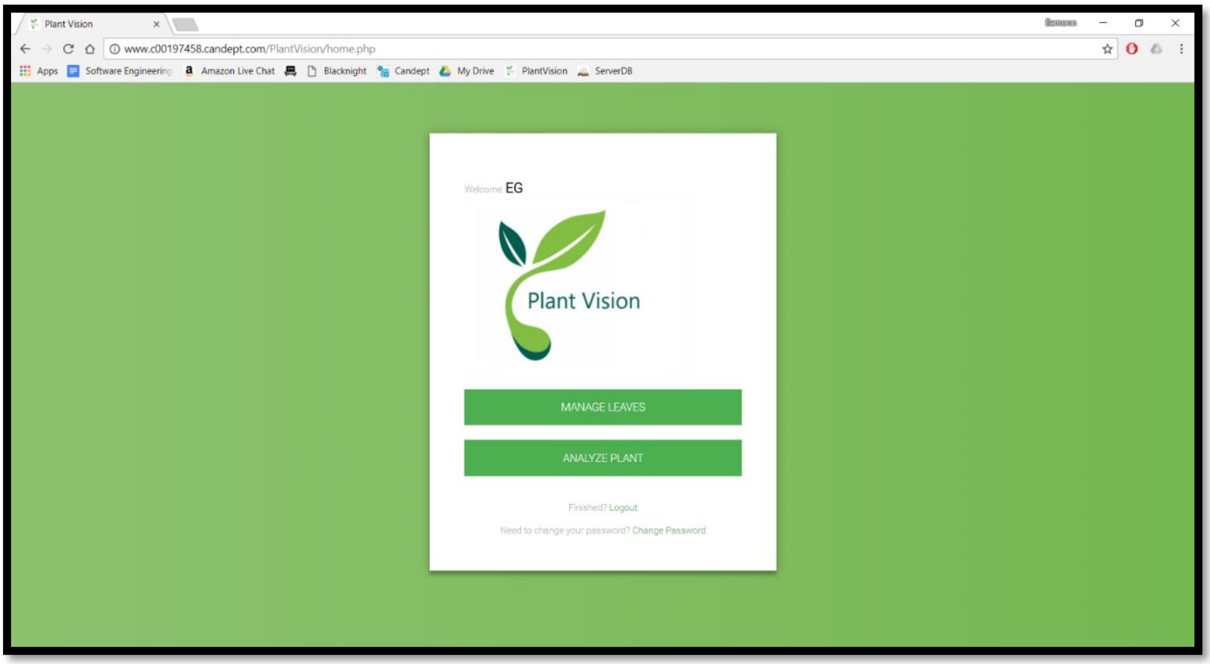


fig 6.7 Home screen

Once the user has successfully logged into the web application, they are presented with their home screen as seen above in fig 6.7. The user is given four options upon reaching this screen, 'Manage Leaves', 'Analyse Plant', 'Logout' and 'Change Password'. If the user selects the manage leaves option, they will be taken to analyse section where they may view further details about the images they have uploaded. The analyse plant option allows the user to view reports drawn from the plant groups they have uploaded, this includes charts, graphs and CSV downloads.

6.2.4 Chosen Image Screen

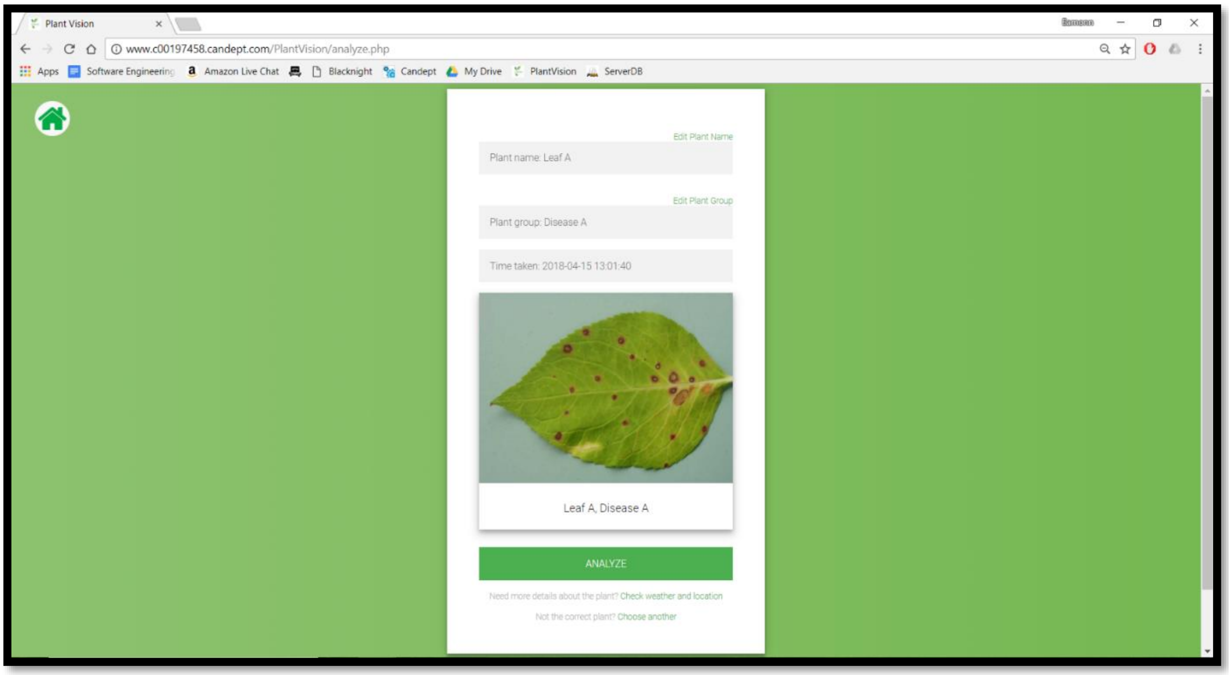


fig 6.8 Choose image screen

Upon selecting the analyse option from the home screen, the user is then presented with the details of the chosen image, as seen in fig 6.8. The user is given details such as the name on the image, the group to which it belongs and the time of which the image was taken. Both the name and group of the plant are editable options where the user can change the values by selecting the ‘edit’ option at top of the input boxes. Below are options for the user to view more detailed information about the plant such as weather and the location, or the option to select another plant.

6.2.5 Extra details Screen

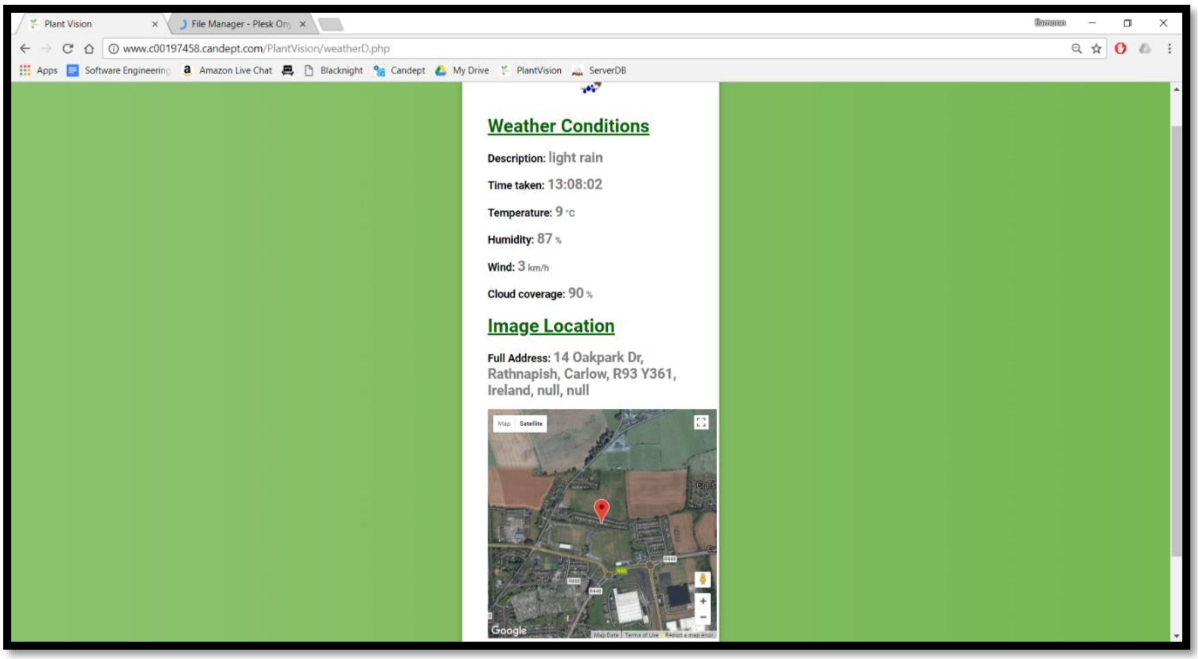


fig 6.9 Extra details screen

If the user has selected from the previous screen to view more detailed information regarding the image, they are then therefore presented with the screen as seen above in fig 6.9. This screen provides the user with both weather and location details associated with the image selected. From the image above, you can see weather conditions displayed such as humidity, temperature, wind, and cloud coverage. As stated previously in this document, this information can be key to understanding a diagnosis of a plant as these variables can prove to be huge factor in a plants overall health. Below the weather details, a google map can be viewed with a coordinates marker set at the longitude and latitude of the time the image was taken.

6.2.6 Analyse Screen

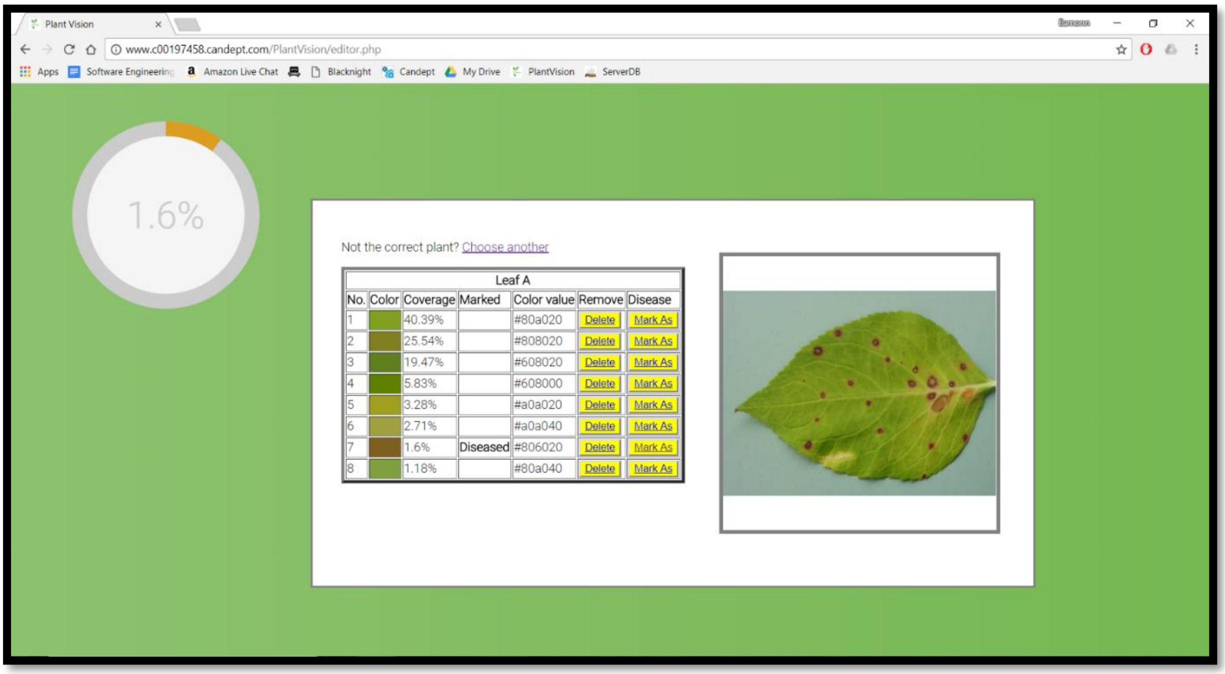


fig 6.10 Analyse Screen

After successfully choosing the image they wish to view from analyse option, they are then presented analyzation screen as picture above in fig 6.10. From here they can choose to remove values that they do not wish to include e.g. background colour and to mark the value to which the disease belongs. The disease is therefore marked as ‘Disease’ and a visual representation of the percentage is shown the top as seen in the sample above. Once this is completed by the user, the following time they proceed to upload an image to the same group as previously analysed, it will allow the mobile application to automatically determine to which part of the leaf is diseased based on prior selections.

6.2.7 Report Screen

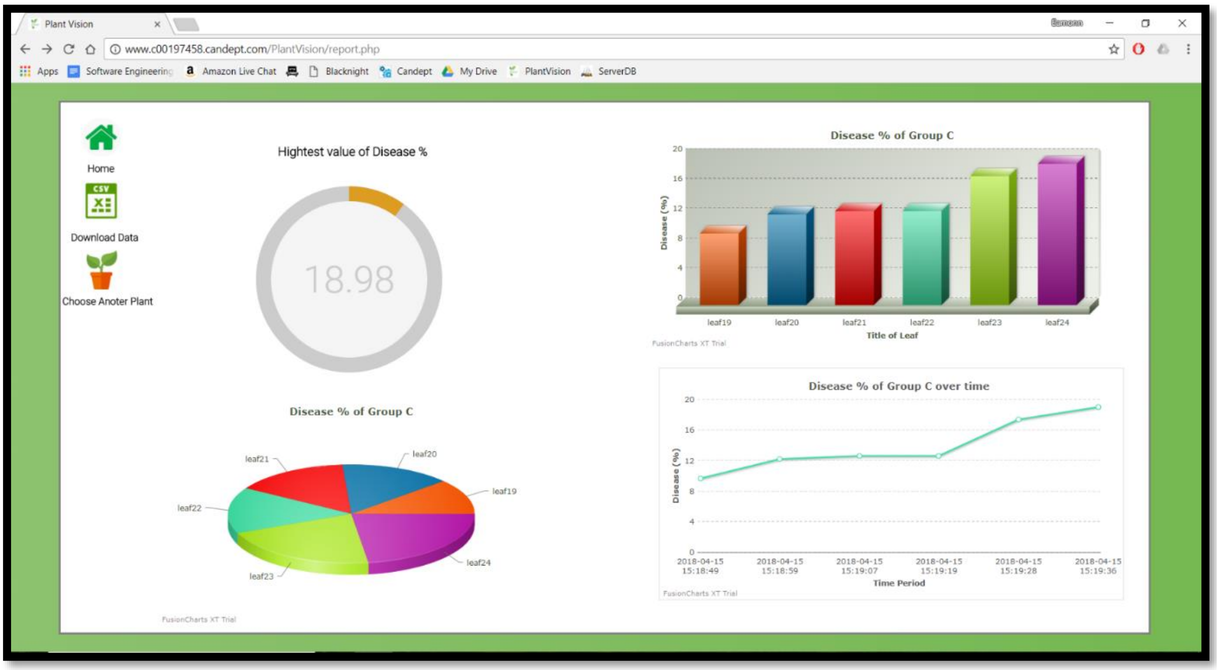
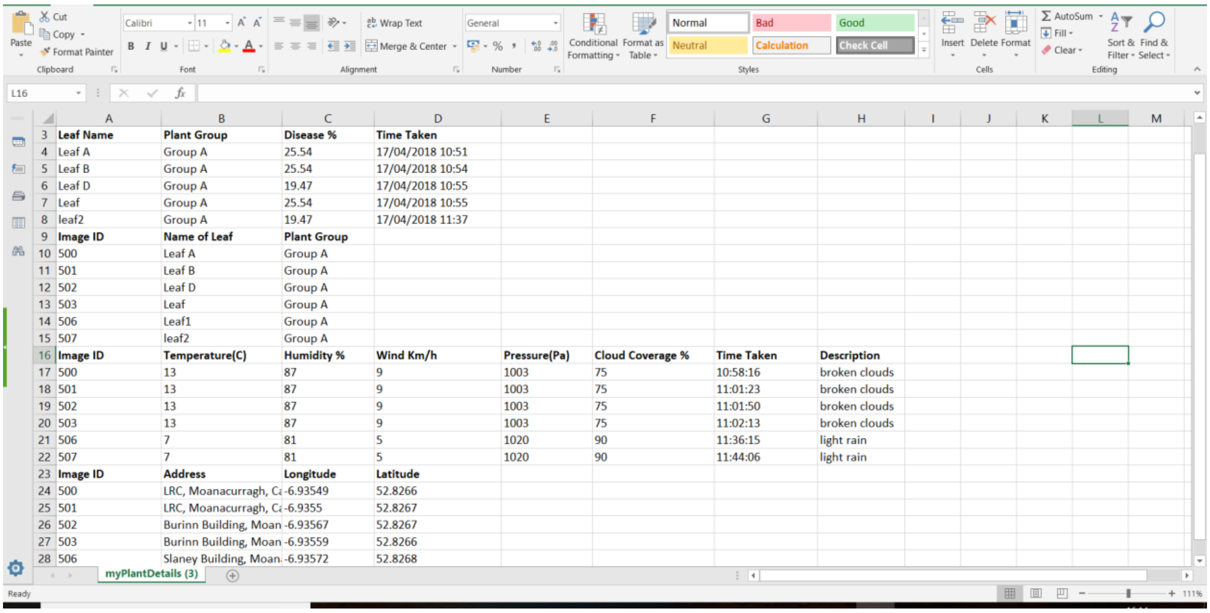


fig 6.11 Report screen



	A	B	C	D	E	F	G	H
3	**Leaf Name**	**Plant Group**	**Disease %**	**Time Taken**				
4	Leaf A	Group A	25.54	17/04/2018 10:51				
5	Leaf B	Group A	25.54	17/04/2018 10:54				
6	Leaf D	Group A	19.47	17/04/2018 10:55				
7	Leaf	Group A	25.54	17/04/2018 10:55				
8	leaf2	Group A	19.47	17/04/2018 11:37				
9	**Image ID**	**Name of Leaf**	**Plant Group**					
10	500	Leaf A	Group A					
11	501	Leaf B	Group A					
12	502	Leaf D	Group A					
13	503	Leaf	Group A					
14	506	Leaf1	Group A					
15	507	leaf2	Group A					
16	**Image ID**	**Temperature(C)**	**Humidity %**	**Wind Km/h**	**Pressure(Pa)**	**Cloud Coverage %**	**Time Taken**	**Description**
17	500	13	87	9	1003	75	10:58:16	broken clouds
18	501	13	87	9	1003	75	11:01:23	broken clouds
19	502	13	87	9	1003	75	11:01:50	broken clouds
20	503	13	87	9	1003	75	11:02:13	broken clouds
21	506	7	81	5	1020	90	11:36:15	light rain
22	507	7	81	5	1020	90	11:44:06	light rain
23	**Image ID**	**Address**	**Longitude**	**Latitude**				
24	500	LRC, Moanacurragh, Ca	-6.93549	52.8266				
25	501	LRC, Moanacurragh, Ca	-6.9355	52.8267				
26	502	Burinn Building, Moan	-6.93567	52.8267				
27	503	Burinn Building, Moan	-6.93559	52.8266				
28	506	Slaney Building, Moan	-6.93572	52.8268				

fig 6.11.1 CSV Extract

The following image (fig 6.11) is the report screen, this appears once the user has selected the reporting option from the home screen and has successfully selected a plant group to which they wish to view reports. As seen in the above sample, the user is presented with a bar chart displaying information regarding the disease percentage based on the different leaves uploaded. Below the bar chart, a line graph was drawn to demonstrate the rejuvenation or deterioration of the plant based over time. At the bottom left of the screen a pie chart was drawn to give a visual representation of the volume of leaves within that particular plant group. The user is also given the accessibility to download this data in CSV format for their own data analyzation as seen in fig 6.11.1.

7. Conformance to Specification and Design

This project conforms to many of the specifications and designs originally laid out in the project specification. One of key aspects of the specification was to implement the interface and functionality of application with the researches, those being the scientist in EnviroCore.

Shortly after iteration one, a meeting was set up with researchers involved in EnviroCore. I sat down with Dr. David Dowling and Dr. Carloalberto Petti to discuss ideas and visions for the application in which I would build. It was at this meeting that some of functionalities as mentioned above were created.

Both David and Carloalberto suggested the idea of adding in the GPS and weather function to application. They explained how this type of information is very important when diagnosing the health of plant as certain weather conditions might alter the perception of a plant. The GPS function was also needed, as several researchers may use the one account on the Plant Vision application, and this will allow them to view exactly where the photo was taken as to ensure they know what sample is being analysed. e.g. lab, farm etc.

As well as the above functionalities, we discussed some key design features that would be implemented. The option to view each particular colour as a percentage on the web was identified as being quite important. As previously stated, and explained by David and Carloalberto, it is not always as easy a detect as one colour being completely different to that of another e.g. green vs. brown. In most cases, the researchers would be exploring the different shades of one colour to which they would like to devote attention. This design feature was implemented as seen in 'Analyse Screen' (sec 6.2.6) where the percentage for each particular colour was displayed to the user and allowing them to pick which particular colour represents the disease.

Both David and Carloalberto were extremely helpful in the designing of the application, as to get first hand ideas and thoughts of the people who would eventually be using the application, could only lend itself to being beneficial to the overall success of the project.

8. Learning Outcomes

9.1 Personal

Throughout the duration of this project, I feel there has been considerable scope regarding personal development and learning. Through its nature, the fourth-year project is quite an undertaking with regard to workload, and it was crucial that time management plays a vital role. I felt that through the development of agile development, where deadlines and goals were set out for each of the iterations definitely eased the pressure when developing the application. By having clear objectives to complete on a gradual timeline ensured for myself that motivation and concentration levels were kept high, as opposed to attempting to complete the entire project as a whole. Asides personal motivation, this also greatly benefited the application as each mini project and feature was given ample time to be both efficient in design and functionality.

As aside from the above, I felt a personal sense of achievement upon completion as I have never constructed such a project of this scale before. Initially, the thoughts of creating an entire project, unique only to myself was quite daunting. From the time I received the original specification, reading the complexity of the requirements I felt it was an unachievable task. However, once I began working on the project I became more and more enthused as the ideas and designs I had on paper were coming to life through my programming skills. Through learning, development, and constant feedback from both Nigel and the researchers at EnviroCore I was able to complete the project in a timely manner. From what was originally a daunting project, became something that I can now look at with a personal sense of pride and achievement in the work that I have produced.

9.2 Technical

From a technical stand point, I gained valuable knowledge into the technologies I used throughout this project, which I am now confident of adopting into the working environment after college.

By using the experience I have gained from my previous three years in the course, along with personal self-learning, I was able to grasp a far better understanding of the languages I used. The mobile application was written in JAVA, using Android Studio as the IDE. I had used Android studio once before in the software engineering module last year, however very much at a primitive level. By building a fully functional mobile application by the end of this project, I now feel I have a far greater understanding in both the JAVA language and android applications. I would be very confident in building another application through these technologies and I believe the learning curve I experienced this year would bode well in future projects where similar challenges may occur.

As well as the above, building an API (Application Programming Interface) this year was something I had never experienced before. The API although written PHP, a language I have been familiar with from previous modules, programming the API to act as a middle ground between both the Web and Mobile app with the main focal point for processing the images proved to be quite challenging, but something which I can most definitely draw on in future developments.

10. Project Review

10.1 What went right?

I would consider the overall project to be of a success, many of the requirements initially outlined have been completed and of quality. One of the major difficulties I had was to process an actual image to identify the different colours. However, as seen in fig 6.10 I believe this, as the core functionality worked out quite well. What was of notable interest was that David and Carloalberto (Researchers in EnviroCore) thought the implementation of this function was beneficial to both that of a practicality and design sense. They believed that this type of feature was exactly what they had hoped for and would be of great use to researchers alike. This had given me great confidence as being the core function of the application, it was important that it was of value and quality.

As well as the above, the feature to upload an image to a server alongside relative information (name, group etc) was definitely something that proved to successful. Before this project I had never experienced working with file uploads to a server, it was more so numerical and text datatypes. Although challenging originally, as I had explored the possibility of uploading the image straight onto the database table as Large Blob datatype. However, this proved to be not only inefficient as it slows down the system, but also heavy on memory consumption taking up huge amounts of space on the database. After researching and consideration, I opted for the more efficient manner of storing the images URL to the database, with the actual image stored in a separate location on the directory of the server. Again, being one of the core features of the application, it was imperative that this was a success as the application would not operate without.

10.2 Challenges encountered

In its nature, a project of this size will experience difficulties and challenges to of which some are eventually overcome, and some which may require different methods to be implemented.

Within this project I experienced several challenging moments, some regarding GPS, weather API's, image uploading, colour counting, sessions, all of which although time consuming in their own right, were overcome eventually through several scenarios and outcomes. However, one of the main sticking points I had experience with was that of the camera function. Although I had programmed the use of the camera, where the user can click the camera option, take a photo within the app, and said photo be displayed as an ImageView on screen, I found it difficult to then upload that said photo to my server. The user can of course upload any photo they wish through their gallery (e.g. photos taken from native android camera), but to take a photo within the mobile application itself was an issue I unfortunately did not have sufficient time to implement. The problem occurred where the image being taken was saved under 'example.plantvision' under data in the mobiles external storage. To retrieve the image for upload, it returned a null pointer where the image being uploaded was returning blank. Android by its nature does not allow for the images taken within the application to be stored on the DCIM directory of the phone. After building the camera function, I left the uploading section till iteration three as I was using the gallery option for testing. Something now in hindsight if I had spent more time on I would have got it successfully implemented.

11. Future Features

11.1 Interface, Functionality, Platforms

For future developments there a number of different features I would like to add to the overall project. First off, I would have liked to add more functionality to the mobile application. Such as editing the images through the mobile itself, rather than having to log into the web application. A feature whereby they could select from a library of plants that they have already uploaded and from the mobile it would output the image with associated details to match. As well as extra functionality to the mobile, I would have liked to also add to the web application. By adding a feature such as an upload image section, where from the web app they could upload an image stored on there PC. I feel with more input from David and Carloalberto over time more features and designs would be discovered through user testing and these could also be implemented to help make the application more efficient and increase productivity.

As well the above, I would have liked to incorporate lighting as an aspect to be taken into consideration when uploading an image. As lighting can take a huge effect on the results of an image regarding colour, I would have liked to explore the idea of reducing these effects by increasing brightness or colour density through the processing of the image.

One of the other aspects I would have liked to explore would have been to launch the mobile on more than one platform. At the moment the mobile is only available on Android, however it would have been more practical to have it launched on IOS as well. Unfortunately, I could not have built it as a responsive web application as so to work on any mobile device, as I needed to access native mobile applications such as the camera and the gallery function. I feel given sufficient time, all the above would have been more than achievable to incorporate.

12. Acknowledgements

Mr. Nigel Whyte (Project Tutor)

First, I would like to thank Nigel as my project supervisor who originally came up with the project specification. His experience, guidance, ideas and visions throughout the duration of the project made all tasks seamless to which ultimately helped shape the outcome of my project.

Dr. David Dowling & Dr. Carloalberto Petti (EnviroCore Researchers)

I would also like to say a great thanks to both David and Carloalberto, whose input as users to the project was extremely beneficial. There thoughts and ideas expressed through our discussions helped a great deal in the designing of the application.

Lecturers at IT Carlow

I would like to say thanks to the lecturers I had experience with over the course of the four years throughout this programme. Through various learning techniques involving many different technologies I gained a skillset required to attempt such a project and the ability to bring with me what I have learned into the working world.

Fellow Students

Finally, I would like to say thanks to my classmates who were always on hand to help with whatever challenge/problem I may have encountered throughout the year with little or no complaint.

13. References

- **(Envirocore., 2017).** Research - Envirocore. [online] Available at: http://www.envirocore.ie/research/ [Accessed 4 Nov. 2017].

- **Smart-AKIS. (2017).** *What is Smart Farming? - Smart-AKIS*. [online] Available at: https://www.smart-akis.com/index.php/network/what-is-smart-farming/ [Accessed 5 Nov. 2017].

- **Gelotte, K. (2018).** *Image Color Extract Tool - Cool PHP Tools*. [online] Coolphptools.com. Available at: http://www.coolphptools.com/color_extract [Accessed 16 Apr. 2018].