Contents

Final Year Project Diary

Week 1: September 28th – September 30th

- Project started
- 1st meeting with Michal Konecny
- Description of meeting - Introduction to FYP:
    - Talked about what the project definition – ensuring I understood the goal of the project
    - Michal mentioned about regular meetings, this can ensure feedback can be given
    - Talked about who my clients are and how I could find them – clients would be someone suitable for the project
    - Michal mentioned that he could act like a client, but it would be more ideal for me to find clients that understand more about gardening

Week 2: October 1st – October 7th

- Began research about the few basics of gardening
- Found two clients that were willing to help with my FYP – both whom enjoy gardening
- Noted important information about gardening from the two clients
- Began research about technologies to use for the project
- 2nd Meeting with Michal Konecny
- Description of meeting – Technologies:
    - Discussed about what programming languages should be used – Go for languages comfortable with
    - Application should be easily available for clients – Has to be a web or mobile application. NOT desktop application.
    - Discussed decision to web or mobile application – Mobile Application will take longer due to learning curve but can produce better product for mobile, web application will take a shorter time and available for both web and mobile (but format for mobile isn't as reliable)
- Continued research on technologies to use for a web application
    - Single-Page Application (SPA) – Frames such as Sencha Touch & AngularJS
    - Popular free libraries to help development
        - Graphics.js – Create graphics in JS
        - D3.js – Data-Drive Documents - graphs
        - Animate.css – Animates CSS
        - Bootstrap – HTML/CSS/JS Framework for developing responsive websites
        - CreateJS: EaselJS – Makes working with HTML5 canvas easier
        - Fabric.js – Interactive object models on top of canvas element

Week 3: October 8th – October 14th

- Continued research on gardening. Looking through gardening applications.
- Started constructing a plan for the project – this would be used for the Project Definition Form
- 3rd meeting with Michal Konecny
- Description of meeting – Agile methods & UI
    - Discussed about the created plan for the project – small changes were made, changed the topic to the agile methods that should be used.
    - Discussed about agile methods to use for project – Concluded that iterative development process is best
    - Went over how UI should look for the interactive canvas the client can use
- Began researching on what libraries to use for interactive canvas
- Started learning how to use FabricJS with JQuery
- Began writing the Project Definition Form

Week 4: October 15th – October 21st

- Completed with Project Definition Form
- 4th meeting with Michal Konecny
- Description of meeting – Project Definition Feedback
    - Had a quick review on the project definition with Michal
    - Michal signed Project Definition Form
- Uploaded Project Definition Form
- Started creating a throwaway prototype for an interactive canvas for creating a garden with FabricJS and JQuery

Week 5: October 22nd – October 28th

- Finished making throwaway prototype for interactive canvas
- Continued research on gardening. Looked into more depth of gardening applications
- 5th meeting with Michal Konecny
- Description of meeting – Feedback throwaway prototype
    - Got feedback from the throwaway prototype – avoid making it too complex and keep it simple but follow the main criteria for the clients
    - Discussed about the application UI against what Michal would advise it to be like – patches instead of several different plants for the user to select. Improve tool selection. Ensure the project diary is involved.
    - Michal mentioned that use cases and activity diagrams should be made to help understand how the product would be used
- Started creating activity diagrams & use cases

Week 6: October 29th – November 4th

- Continuation of activity diagrams & use cases
- Began working on wireframes for application
- 6th meeting with Michal Konecny
- Description of meeting – Feedback on wireframes
    - Got some feedback from wireframes – avoid complexity and ensure that it's possible to develop the wireframes
    - Told to not focus too much on wireframes.
- Continued research on technologies to use for project
    - Framework for PHP called Laravel
    - Framework for Front-End development AngularJS
- Started quickly revising the basics of SQL

Week 7: November 5th – November 11th

- Started creating Entity Relationship Diagram for the database for the application
- 7th meeting with Michal Konecny
- Description of meeting – Feedback on database design
    - Got feedback on entity relationship diagrams – had room for improvement
    - Michal mentioned that the database will more than likely change in the future.
- Continued and finished an entity relationship diagram for the product – likely to change in the future

Week 8: November 12th – November 18th

- Creating a database using the entity relationship diagram
- Began learning Laravel framework for PHP
    - Followed helpful guides from online resources to create a small project to understand the basics of Laravel
- Did not have a meeting with Michal Konecny for this week, did not have anything to present

Week 9: November 19th – November 25th

- Began learning how to use bootstrap with the throwaway prototype made in the earlier weeks
    - Followed helpful guides from online resources to create a small project to understand the basics of bootstrap and applying it to the throwaway prototype
- Busy with coursework from Interaction Design
- Did not have a meeting with Michal Konecny for this week, did not have anything to present

Week 10: November 26th – December 2nd

- Busy with coursework from Interaction Design

Week 11: December 3rd – December 9th

- Busy with coursework from Interaction Design & EAT

Week 12: December 10th – December 16th

- Busy with coursework from EAT

Week 13 – 16: December 17th – January 20th

- Studying for Exams
- Researched into organising front-end code
    - AngularJS came into mind. Helpful, especially for SPA (Single Page Applications)
- Researched into combining Laravel and AngularJS together (organisation for front and back end code)
- Continued using Laraval – second thoughts of using Laravel, maybe more efficient to use Java Spring framework with AngularJS instead of Laravel with AngularJS (more familiar with Java than PHP)
- Began working with AngularJS and trying to integrate FabricJS with it

Week 17 – 18: January 21st – February 4th

- Continued working with AngularJS and trying to integrate FabricJS with it.
- Started considering the Java spring framework which involved in:
    - Researched benefits of using Java
    - Researched benefits of using php
    - Found that differences between the two does not necessarily matter for the size of project
    - Looked both frameworks and compared to make a judgement
    - Laravel's blade template and overall structure was more intriguing than Java Spring
        - Java Spring would involve JSP and using JSP in CS3160 - Enterprise Application Technology felt very restrictive in comparison to the blade template by Laravel
- Two areas were focused on, research on AngularJS and Laravel framework and trying to integrate both with throw away prototype
    - Research showed that AngularJS and Laravel complement each other well, as one handled with organizing the backend code – Laravel – whilst the other focuses on organizing the frontend code – AngularJS.
    - Worked with trying to integrate AngularJS with Laravel to see how it would all come together using throwaway prototype.
    - Issues occurred with using the AngularJS jqLite with FabricJS. They would conflict with AngularJS using its own subset of jQuery whilst FabricJS would need the whole jQuery library.

Week 19 – 22: February 5th – March 3rd (1st Iteration)

Week 19

- Had to find an alternative to account for not using AngularJS, results were as follows;
    - jQuery and FabricJS could be used together without AngularJS.
    - AngularJS provides many features that can facilitate a better product but due to possible consequences that can occur between AngularJS and FabricJS, it's decided that it's best to avoid the framework
    - Further research was conducted to find out better ways to organize front-end development with jQuery, it was found that encapsulation could be done with using **modules**. This was taken into account and will possibly be used when the development begins

- Called and emailed clients about having a meeting on February 10th to discuss about the early designs and requirements for the first development iteration.

February 10th Client Meeting

- Meeting with two clients to discuss what's happening with the whole process again on what's happening.
- Went through on trying to work out what requirements we'd like to have for this project. The requirements would be based off the description that was initially given for the project. There were several ideas given during the first meeting with the clients, however, some were just too unrealistic to develop within the time constraint – such as a system to inform you where to buy certain resources for a plant. A final agreement would be made as to what the main requirements for the first iteration cycle of the product would be:
    - User having the ability to create their own garden, this would be the garden layout. There were a few things it would allow the users to do:
        - Add your own patch
        - Can add extra tiles such as water for ponds, some sort of ground tile and grass
        - Ability to save
        - Have a way to rename all tiles (required for patches)
        - Way to edit and way to turn off editing so patches are clickable and navigate user to the activities for that patch
    - User having the ability to keep track of their own activities for all the patches they have in their garden. This would be their garden diary. This would include:
        - Their activities that are required during that day
        - Activities that are upcoming
        - Activities that have past the deadline
        - Activities that have not yet been set
        - Ability to add new activities to patches from the plant encyclopedia
    - Have a plant encyclopedia whereby there would be several different allotments
        - Allotment Categories
        - Allotments for each Category
        - Activities for each Allotment
        - Description for each Allotment
    - Have a history for the garden to see how it was like in the past. This would include:
        - How the garden was on certain days
        - What activities were being run on those days and on what patches
        - A calendar to select the date
    - Having a homepage briefly talking about gardening and its benefits. A way to make users feel more interested in starting gardening. This would include:
        - Description about gardening and its benefits
        - A quick guide on beginning your garden
        - Description to then proceed to tell the user to use the application to begin their gardening
- Requirements specification was made with Moscow after the initial set of requirements were made.
- Requirements Specification for Iteration 1 with MoSCoW:

    - Layouts of pages
        - Simplistic UI (SHOULD)
        - Consistent UI layout (SHOULD)
    - Creating an authentication system (MUST)
        - Register system (MUST)
        - Login system (MUST)
        - Change password (COULD)
        - Change email (COULD)
    - Creating a homepage discussing about gardening and the application itself (COULD)
    - Creating the Garden Layout Page (MUST)
        - Garden Editor (SHOULD)
            - Ability to save (MUST)
            - Load previously made Garden (MUST)
            - Add a patch tile (MUST)
            - Add other tiles (SHOULD)
            - Rename Tiles (MUST)
            - Warning pop-up when removing a tile (COULD)
            - Having edit on/off option (COULD)
            - Edit off would allow patches to navigate users to garden diary (COULD)
    - Creating the Garden Diary Page (MUST)
        - Ability to add activities from plants (MUST)
        - Add activity to a patch (MUST)
        - Well-designed pop-ups (COULD)
        - Organise activities to upcoming/past deadline/not set (SHOULD)
        - Ability to view plant details (COULD)
        - Ability to view plant activity details (COULD)
    - Creating the Garden History (WON'T)
    - Creating the Allotments (Plant Encyclopedia) (MUST)
        - A list of different plant categories (MUST)
        - Each plant category having a list of plants (SHOULD)
        - Each plant having plant activities (SHOULD)
        - Ability to view plant description (COULD)
        - Ability to view activity description (COULD)
- After gathering all the requirements, next day involved in quick wireframes sketches to allow for some early designs to be made for the clients
- Issue with wireframe for history garden
    - the requirements seemed too vague and difficult to create designs from.
    - Unsure if it's worth implementing for the first iteration as there's a lot of requirements already set out for the first iteration
    - Clients were fine with implementation for future iterations instead so requirements specification would be modified
- Use cases, entity relationship diagram and flow charts were created again after the wireframes and requirements were made with the clients. Previous use cases, entity relationship diagram and flow charts were helpful on what was initially going to be made but after further research

on the technologies that were going to be used, they seemed too unrealistic to follow so new use cases, entity relationship diagram and flow charts were made.

Week 20

- Week involved in setting up Laravel for this project and the setup for the login (which is already made with the framework).
- All the pages that were required were made (with no content inside them). No pop-ups were created
- Towards end of the week, began working on the Garden Layout with the knowledge that was gained from the throwaway prototype made during the previous year.

Week 21

- Created the Garden Layout for the application, but there were a few bugs that occurred with it. There were as follows:
    - Clicking on the grass tile would bring it to the front. This would then hide all the other tiles, a temp. fix for this was making a button that would send a tile straight to the back so the grass tile can be sent to the back if clicked on.
    - Patch does not navigate user to the garden layout. Too complex to implement, ate up too much time so it wasn't implemented. Could not think of an alternative for this
    - Save button not working – has not been implemented.
- After a bit of research, it was found that the whole canvas can be saved as a json and stored onto the database. Due to time constraints, this wasn't tested as there was only one page that was made during this time. This was the garden layout. There was still the need to make the garden diary, home, and allotments.

Week 22

February 25th – February 26th

- The first two days of this week involved in creating the Garden Diary. The garden diary didn't have any database involved with it so this was all front-end programming with dummy data (data that did not exist from the database, instead just created in the frontend).
- Contacted clients about having a meeting on the 3rd March

February 27th – 1st March

- The next three days involved in creating the allotments page. Part of this section had dummy data and it was the description of the allotment and the activities for the allotment. However, all the allotments were from the database. So, there was some backend running as well. There wasn't enough time to do it all properly due to the time constraint but this was enough to allow the clients view exactly what was going on.

March 2nd

- The next day involved in creating the pop-up for the Garden Diary when they add an activity. This would use data from the database to return the plants categories and the plants. As there were no activities added in the database, there were no activities added to the pop-up.
- The pop-up didn't follow the wireframe exactly because there were problems with trying to follow it with 3 rows. This is how the pop-up worked and the problems that occurred:

    - Each plant category can hide and show their plants. This was done in JQuery, although this was not adaptable. If another plant category were to be added, the frontend code would have to be modified to work as intended – this was an issue.
    - Each plant would have its activities shown if it was selected. There were no activities for the plants so this wasn't implemented.
    - Even though the activities were not implemented, the problem came in again with the inability to hide/show the activities if new activities were added as code from the frontend would have to be modified.
- An alternative has been thought of, and this is that the activities would be joined with the plants. An example of one would be "Lavender: Watering" and "Lavender: Harvesting", this would remove the need of having the ability to hide and show all activities – far too time consuming to do manually.
- The second popup was not created either where a patch would be selected for the activity. There was a second thought on implementing this.
    - From what was learnt from HCI and Interaction Design, it's not great to have many clicks to do something.
    - Having multiple pop-ups are not ideal.
    - An alternative idea came into mind by having a button in the button selection of the left of the garden diary called "Patches". When this button is selected, you'd be able to add an activity from here instead of having the ability to add an activity at all points. This would then remove the second pop-up required.

March 3rd

- Had a meeting with the clients
- Went through a UAT for the garden layout feature
    - All tests passed for garden layout
- Showed all other features and discussed the ideas I had during the previous day. The discussions and results from the meeting were as shown:
    - Change the names "Allotments" to "Plants". Preferred using the name Plants
    - Looked at the allotments that were available, and there were some plants they had in their garden that didn't exist in the system.
        - We then came up with the idea of having the option to add your own plants and activities.
        - I wasn't too sure if it was worth having a plant-encyclopedia AND your own plants because implementing both would take too much time. I preferred having just to add your own plant.
        - Alternative was made to just have users create their own plant
    - We came to an agreement that having the ability to add your own plants and removing the plant encyclopedia was just fine.
    - Changes with how the activities are handled in the Garden Diary were fine to proceed with except for the buttons.
        - Didn't understand what "Activity Looped" meant.
        - How do you remove activities? Doesn't seem possible.
        - What activity is it? No name for it.
        - When does the activity have to be completed? No date shown.

    - Not understanding why there was a load button for the Garden Layout.
        - Is it necessary to have?
        - Shouldn't it load straight away?
        - If it is loaded straight away, what's the need for having it later?
    - Change the images.
        - Too big and not understanding why there was a flower for the patch tile.
    - No implementation for home page.
    - No paging added to the pop-ups.
        - Further discussed as I did try to add paging but it proved to be difficult to implement.
        - Overall, it wasn't an issue and having a scroll bar is completely fine.
        - If it was for a mobile application it would be an issue, but it's a web application so it wasn't a problem.
    - Don't like the buttons being joined together. Prefer them to be separated like shown in the wireframes.
    - History page has not been implemented but I came up with another possible feature to have which is still beneficial and allows users to view their past activities.
        - Have a calendar to select a date and show all the activities that were completed during that day.
        - Have option to view the garden to see the state it was in when the activity was completed.
        - Discussed with the clients and they seemed very happy with this being implemented. However, due to time constraints. I suggested this be made in the final iteration instead if there's enough time.
- Requirements specification was made with Moscow after the initial set of requirements were made.
- Requirements Specification for Iteration 2 with MoSCoW are:
    - Creating a homepage discussing about gardening and the application itself (COULD)
    - Updating the Garden Layout
        - Remove loading button (SHOULD)
        - Implement back-end for garden layout (MUST)
    - Creating the Garden Diary
        - Remove loop activity (MUST)
        - Have a new button called Your Patches and only add activities should exist for these patches (MUST)
            - The above causes only one pop-up being necessary
        - Inactive activities should have Start and remove buttons fully implemented (SHOULD)
        - Other activities should have Completed, Reset, Stop and Remove buttons fully implemented (SHOULD)
        - No dummy data should be used, back-end could should handle all data (MUST)
    - Creating the Garden History Page (WON'T)
        - Create a calendar, when a date is selected from the calendar show all activities that were completed on that date (WON'T)

        - Display the plant, patch, activity names and the date of completion (WON'T)
        - Have a way to check the state of the garden when this activity was completed (WON'T)
    - Recreating the Allotments page
        - Rename to Plants (MUST)
        - **Remove the plant encyclopedia** (MUST)
        - Have user Add their own plant (MUST)
        - Have user View their own plant (SHOULD)
        - Have user Modify their own plant (COULD)
        - Have user Delete their own plant (COULD)
        - Can add activities to each plant as well (MUST)
        - Can view activities (SHOULD)
        - Can modify activities (COULD)
        - Can delete activities (COULD)

Week 23 – 24: March 4th – March 17th (2nd Iteration)

A lot of time was spent during these two weeks and got a lot done for the project. The tasks were spread out in the week, so there were not specific days working on a feature of the project. For this reason, this is sectioned out as what was done for each feature.

Garden Layout

- Removed the load garden button. The garden was loading a default dummy data JSON every time the Garden Layout is loaded. This is just a dummy data, but it was later updated. Plan was just to make sure that it was even loading the JSON file.
- Changed the image sizes so they're all the same size and changed the patch tile image.
- Implemented the ability to save the garden. Although several issues arose when implementing this
    - An error came up when loading up a JSON in a canvas with the algorithm that was used in place. The problem was not in the algorithm, but rather a bug with the FabricJS library. As the library was downloaded during the previous year, it wasn't updated and fortunately there was an update during December that fixed this bug. So the library had to be updated.
    - The library was updated and another bug arose with the algorithm that was used. It just did not work. Unable to resize the image or else several errors would occur. A new algorithm would have to be implemented to fix this bug.
    - A new algorithm was implemented and the errors were removed. Although with this new algorithm in place, another bug came into issue. When canvas was converted into a JSON and saved. When loaded, it was unable to read the extra attributes that were added for each tile. This meant that there were bugs with loading the JSON
    - The problem that occurred was identified. The problem was that in the FabricJS library, the extra members that were given for the rectangle object did not exist. The solution would be to create a sub-class within the FabricJS library with the extra members. This way, it would load these members right after the library has been loaded and when loading the JSON canvas, it would be able to read the extra members that were added.
- Patches that are created in the Garden Layout can now have activities applied to them.

Garden Diary

- Modified the description for each activity. Now has **four** sections. Patch Name, Plant Name, Activity Type and the Deadline Date.
- Added two new buttons that opens two new content. “Your Patches” and “Patches with no Activities”. “You Patches” changes the content to show all the patches of the user. The “Patches with no Activities” was specifically added from myself as I felt it may be helpful to have. It allows the user to see which patch has no activities in it. This way it can help them ensure they have activities running in all their patches.
- Modified the options the user has for their activities. For active activities:
    - They can click “Completed” if it’s completed - it would be added to the activities history.
    - They can click on “Reset”, this would reset the deadline date.
    - They can click on “Stop”, this would stop the activity – not added to the activities history.
    - They can click on “x” to remove the activity from the patch.
- For inactive activities:
    - They can click “Start” to start the activity
    - They can click on “x” to remove the activity from the patch.
- The deadline can be modified if the plant’s activity days is modified. If days was originally 1 day, the deadline would be the next day. The user would can modify the days to something like 2 days, the deadline would then be the day after the next day.
- Pop-up for adding an activity has been modified.
    - Only the user plants are now displayed.
    - The plants and activities are now combined.
    - Unsure of whether to keep the “type of plant” section. May remove it and just have one column. The reason for this is because it’s too difficult to try to manage all of the users plant in JQuery and it may not even be necessary. We’ll discuss with clients during the next meeting.

Plants

- Only user plants are shown in the plants section. It’s no longer a plant-encyclopedia.
- Able to select your plant on the left-hand side (will more than likely remove as it seems useless to have and is a waste of space)
- Ability to now add a new plant, add a new plant with activities
- View plants created
- Edit plants created (including their activities)
- Description for plant and activities have now been joined together in one page
- Ability to remove plants

Garden History

- The database has been set for the garden history. There’s a table that holds all the information required for activities that were completed in the past. If an activity is completed, the garden canvas JSON is obtained and stored in a table along with all the details for the activity and the date it was completed.
- The feature is ready to be implemented but not yet has due to time constraints and coursework from other modules.

Home

- Nothing has been done for the home page. Going to discuss with the clients about the home page. I personally don't believe it's necessary to have. The application is aimed for busy individuals with interests in gardening, not new gardeners.

March 17th – Client Meeting

- Had a meeting with the clients
- Went through UATs for all features implemented
    - One test failed with new images not loading for one of the clients
    - Every other test passed.
- Further discussions made about the prototype and discussed the ideas I had during the second iteration. The results from the meeting were as shown:
    - Garden Layout
        - Bug occurred with the images in the Garden Layout for one of the clients. Was not updated. Explained that it's different and tried to explain how it looked. They were fine with the changes.
        - Garden Layout was mostly fine except for the way you save. Prefer it automatically save instead of manually saving.
        - Could have repositioned the name of the tile. Possibly on the tile? Unsure of where the location should be, not a big deal but a different location would be nicer.
        - Forgot the reason for having the "Switch to Garden Diary", doesn't seem to be needed as long as there's some navigation to the garden layout.
        - Way to modify grid size? The current size could be max, but height and width could be adjusted to fit more of a natural garden?
        - Fence tiles? Another tile that's useful to have.
    - Garden Diary
        - A lot better. A lot more understandable, possibly separate the buttons a bit more? Format of buttons are better too.
        - Have a way to specify which activity button has been selected? Not sure if today's activities has been selected, upcoming activity, etc.
        - Name doesn't match its purpose, it's not a diary, more of an activity handler. Rename to Garden Activities?
        - Always goes to todays activities, have it change to the most important activities? I.E. Overdue>Todays>Upcoming>Inactive
        - Looped option, when activity is complete. It's reset and recorded in the database. Reduces number of clicks for repetitive tasks like watering.
    - Plants
        - Nice feature to have. Mentioned about removing the buttons on the left and having just the main content area being shown. Clients were fine with this. Only issue they had was suggesting to centre the "Add New Plant" button.
        - Would like to still have the plant encyclopedia as well as having your own plants.
    - Home
        - Fine with removing this altogether.

    - Garden History
        - Could be called Activities History or better, Diary.
        - Would have liked to see this implementation but understand the situation, they said it's dependent on how my situation as to whether I should implement it or not. It's a nice feature to have but it's up to me since I may not have the time to implement it.
    - Overall UI could change a bit, there may be a slight difficulty in understanding what you should be doing. May not be important.
- Requirements Specification for the Final Iteration with MoSCoW are:
    - Reformat layout of the whole application to make it easier to use and understand (COULD)
    - Garden Layout
        - Fix bugs that are occurring with images (MUST)
        - Automatic saving system instead of manual saving (SHOULD)
        - Reposition where name is tile appears (COULD)
        - Remove Switch to Garden Diary button (MUST)
    - Garden Diary
        - Space out buttons (SHOULD)
        - Indicator of the activity type (todays/upcoming/overdue/inactive) (SHOULD)
        - Change name to Garden Activities or something more relevant (COULD)
        - Show the most important activities first, overdue>todays>upcoming>inactive (COULD)
        - Add loop button. When loop is true, when activity is completed it will reset the deadline and add it to the database for the garden history (if implemented) (SHOULD)
    - Plants
        - Align Add New Plant to the centre (SHOULD)
        - Re-add the Plant Encyclopedia (COULD)
    - Home
        - Remove page (SHOULD)
    - Garden History (COULD) if not created, tasks below are not acquired.
        - Rename to Activities History or something more relevant (COULD)
        - Create a calender, when a date is selected from the calender show all activities that were completed on that date (SHOULD)
        - Display the plant, patch, activity names and the date of completion (MUST)
        - Have a way to check the state of the garden when this activity was completed (COULD)

Week 25 – 28: March 18th – 21st April (final iteration)

Week 25 – 26 (first half): March 18th – March 27th

- Meeting with Michal, showing the 2nd iteration application.
    - Few problems occurred. Image still not updated, and a few other bugs occurred such as patches not updating.

    - Discussed about whether the activities history is worth implementing. It seemed like a nice feature to have but due to the time constraints, may not be worth implementing.
    - Would be nice to click on the patch to navigate the user to the patch activities.
    - The UI could be improved. Judging by how Michal interacted with it, it seemed to be confusing to understand what was going on. This was due to how uneasy it was to understand what you had to do, a tutorial would be helpful and/or a change in the UI.
    - Michal said that the main thing to focus is ensure that the application is more robust.
- After the meeting, the bugs that occurred in the previous meetings were fixed. The whole UI for the application was modified. It had a similar format as before but the navigation bar at the top was removed. It was then allocated to the left-hand side with icons. The names for each pages were changed as well to something more understandable.
    - Garden instead of Garden Layout
    - Activities instead of Garden Diary
    - Plants remained the same
    - Diary instead of Activities History
- There was enough time as well to implement the Activities History (changed to Diary) so this feature was now implemented. It has:
    - A date picker which allows the user to pick a date.
    - The results then come up with all the activities that have been completed for that day. The description are as follows:
        - Patch Name
        - Plant Name
        - Activity Name
        - Completion Date
    - Able to "Check Garden" which allows the user to check how the garden was when this activity was completed.

Week 26 (second half): 28th March – 31st March

- Meeting with Michal, showing how the application is right now. There was a need for some sort of guide as it may be difficult for a first-time user to use straight away.
- Discussed a bit more about the structure of the dissertation and how it should be written.
- **Extra modifications made for the requirements**
    - **Help Page to be added to guide the user on what to do** (SHOULD)
        - Video guide on using Garden Layout
        - Video guide on using Garden Diary
        - Video guide on using Garden History (if implemented)
        - Video guide on using Plants
        - A user manual to use
- Began working on the user manual for the application
- Been busy with coursework deadlines so the whole project hasn't been worked on as much for the second half of week 26

Week 27: 1st April – 7th April

- Final meeting with Michal, discussed a bit more about the structure of the dissertation and what happens with the demo.

- Been busy with coursework deadlines so the whole project hasn't been worked on as much during week 27
- Set goal to create a user guide and put a line on the project from then on. This will be the final iteration. Get an evaluation done and have the clients view the final iteration for the last feedback on it.

Week 28: 8th April – 21st April

- Finished creating the user manual guide for the application
- Finished creating the Help page which involved in four different videos for the four different sections. Garden, Activities, My Plants and Diary.
- **Finished with the final iteration development.** A few requirements were not completed, but based on the MoSCoW, all the MUST requirements were completed

17th April (Last Client Meeting)

- Last client meeting for last feedback on final iteration goes well. Overall the clients were impressed with the final deliverable. There were possible improvements that could have been made for the application and these were the notes:
  - Garden
    - Ability to change patch color? Another way of differentiating between different patches
    - More tiles being added for a better match for a garden
    - A way to tell if there's patch has todays/upcoming/overdue activities from the garden page
    - Modification to the size of the garden
  - Activities
    - Show most important activities first
    - Start activity **from the pop-up** – save number of clicks
    - Have option to have activity restart when complete
  - Plants
    - Add plant-encyclopedia
    - Ability to add categories for user plants
  - Diary
    - Have blank page show an image? Especially when no activities exist for a date.
    - Highlight which patch the activity existed in
  - Help
    - No issues, well made.

18th April (User Acceptance Tests and feedback)

- 18th April (Evaluation Stage) The application was tested by 9 university students. Results were collated in a user acceptance testing results document.