Project Plan

Contents

[1. Project Details 4](#_Toc50927879)

[1.1. Background 4](#_Toc50927880)

[**1.2.**  **Business Model** 4](#_Toc50927881)

[**1.3.**  **Assumptions** 5](#_Toc50927882)

[**1.4.**  **Constraints** 5](#_Toc50927883)

[**1.5.**  **Stakeholders Analysis** 6](#_Toc50927884)

[2. Benefits to the Business 6](#_Toc50927885)

[3. Project Stages 7](#_Toc50927886)

[4. Methodology 7](#_Toc50927887)

[5. Budget 10](#_Toc50927888)

[6. Business Processes 10](#_Toc50927889)

[7. Technical Requirements 11](#_Toc50927890)

[8. Work Breakdown Structure 12](#_Toc50927891)

[9. Schedule 15](#_Toc50927892)

[10. Change Management Plan 15](#_Toc50927893)

[9.1. Technologies used 15](#_Toc50927894)

[9.2. Collaboration tools used 15](#_Toc50927895)

[11. Quality Management Plan 16](#_Toc50927896)

[12. Risk Management Plan 16](#_Toc50927897)

[13. Team Members / Human Resources (Responsibility Assignment Matrix) 16](#_Toc50927898)

[14. User Acceptance Testing 17](#_Toc50927899)

[15. Hand over Document 17](#_Toc50927900)

[16. Communication Management Plan 18](#_Toc50927901)

[17. Collaboration tools 19](#_Toc50927902)

**Project Plan Information**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | 31/08/2020 | **Release** | 1.0 |
| **Author** | Diego C., Gerardo G. | | |
| **Project Sponsor** | North Metropolitan TAFE | | |

**Revision History**

| **Version Number** | **Date** | **Reason/Comments** |
| --- | --- | --- |
| 1.0 | 31/08/2020 | Initial release |
| 1.1 | 13/09/2020 | Added sections 4 (Methodology) and 17 (Collaboration tools) |

**Distribution List (Stakeholders)**

| **Name** | **Position** | **Interest in Project** |
| --- | --- | --- |
| Diego C. | Developer |  |
| Gerardo G. | Developer |  |
| Jade Uhrbom | Lecturer | *Client* |
| Dawn Hetherington | Lecturer | *Client* |
| Delia Stanley | Lecturer | *Client* |
| Keith Critchett | Lecturer | *Client liaison* |

|  |  |  |
| --- | --- | --- |
| **Authorisation** | | |
| Project Plan approved for submission | | |
| **Project Roll** | **Name** | **Signature** |
| **Sponsor** |  |  |
| **Senior User(s)** |  |  |
|  |  |  |
| **Senior Supplier(s)** |  |  |
|  |  |  |
| **Project Assurance** | Paul Williams |  |

# Project Details

## 1.1. Background

Hairdressing lecturers and students at North Metropolitan TAFE Balga are currently using mannequins to perform their tasks. This approach has a glaring limitation: it is inflexible, in the sense that mannequins can only give a rough idea of how a person with certain features might look like with a specific hair style.

As such, it would greatly benefit everyone involved in such tasks if they could make use of an application that accurately displays a real person with various hair styles that are suitable to their needs.

According to the clients, there are three main variables to be considered for this application: **face shapes**, **skin tones** and **hair characteristics** - which include hair style, hair colour and hair length.

The application should be able to recognise such features through pictures submitted by users and then update those pictures based on user input, displaying the results afterwards. The clients also asked for 3D imagery (with rotating heads) to be implemented in the future, instead of simply working with pictures.

**1.2. Business Model**

This application targets the hairdressing industry, which generated a total revenue of $6.5 billion in 2018 in Australia and, despite a significant decline due to COVID-19, it is expected to recover starting next year.

More specifically, it is initially aimed at Hairdressing students and lecturers from North Metropolitan TAFE.

The application is a one-off deal with lecturers from North Metropolitan TAFE, who will use it to aid their teaching methodologies for their classes.

*Note: because of the one-off nature of this project without medium-long term strategies to turn it into a business venture, a Business Model Canvas is not deemed necessary.*

1. **Includes**

* Hardware: Since the APIs and back-end infrastructure of the app will be deployed to a Digital Ocean droplet instance, no additional hardware will be required.
* Software:
  + For end users: Style Me app
  + For developers and admins: Admin Portal. To test the app and the Admin Portal locally, the following software will be required:
    - Laragon
    - .NET Core
    - MariaDB
    - Flutter SDK
    - Python
    - Android SDK
    - Android Studio
    - VS Code
    - Operating system: Windows 10, a GNU/Linux distribution or updated OSX

The clients will receive instructions on how to set up a development environment for this project, including the installation of all the software required.

1. **Excludes**

No hardware will be included as deliverable for this project. Aside from the code for the Style Me app, Admin Portal, APIs and database, no additional software will be supplied. No network infrastructure will be delivered, since the components of this project are managed in the cloud (Digital Ocean droplet).

**1.3. Assumptions**

It is assumed that there will be no significant interruptions for the schedule of the project plan, since no additional staff can be hired to complete the work.

Additionally, developers are expected to have access to all the software needed to work on their tasks at TAFE but are encouraged to bring their laptops in case the software that they need are not available on the classroom computers.

It is also assumed that there will no big changes in the project scope over the development lifecycle considering the limited timeframe available and the complexity of certain parts of the app.

**1.4. Constraints**

Between 25/09 and 05/10, developers will be on a break between terms 3 and 4. During that period, it is expected that there will be a brief slowdown in the development lifecycle of the project. However, that should not pose a significant impact on the delivery schedule.

Also, it is expected that any costs incurred should be handled by North Metropolitan TAFE, such as domain names and cloud infrastructure.

**1.5. Stakeholders Analysis**

|  |  |
| --- | --- |
| **Stakeholder name** | **Requirements[[1]](#footnote-1)** |
| Dawn Hetherington | * Colour swatch with shades and lightness levels * Short / medium / long hair style choices according to face shape |
| Jade Uhrbom | * Additional hair characteristics as a consultation section with sample pictures: * Texture: European, Euro-Asian, Asian, African * Porosity: Extreme, Good, Average * Density: High, Medium, Low |
|  |  |
|  |  |

# Benefits to the Business

* 1. Operational

1. The infrastructure of the project should be easily manageable, since it is not in-premises and the deployment process are well documented by the developers.
   1. Economic
      * 1. Costs should be low, since developers are using free tiers for hosting, collaboration tools and storage. Minor impact on cost is expected in case the app needs to scale to accommodate thousands or millions of users.
        2. Development cost is essentially zero
   2. Technical
      * 1. No additional hardware is required, since the APIs and database are managed in the cloud
        2. There is little risk in terms of support for the development tools used, as they are well established and used by big businesses over the world

# Project Stages

The project will be executed in two main phases once the documentation is finished and agreed upon amongst all stakeholders: **development** and **deployment**.

The development phase itself will be broken down into various tasks that will mainly involve:

* Development of a Python API containing the business logic to process pictures uploaded by users
* Development of a Flutter app as the front-end of the project
* Writing and executing tests and performance metrics

Once the development team completes the final sprint and review the work done, the project will be validated one last time by the clients to ensure that all features and measures have been implemented as agreed in the scope document.

The deployment phase will involve moving the code base to a cloud environment (AWS) and ensure that all the different parts of the project work correctly when going live.

Upon completion, all deliverables will be submitted to the clients.

For more details, see the Implementation Plan document.

# Methodology

The software development methodology adopted for this project is **Agile**. As such, it is expected (and perhaps desirable) that this document changes over time. The Agile framework that will effectively guide the progress of the project (although not strictly followed as described by its creators) is **Scrum**.

Since there are only two developers in the team of this project, we are essentially the development team. There is no product owner or scrum master as far as the personnel involved in this project is concerned, so the two developers involved will fulfill not only their roles as self-organising team members but also the other two previously mentioned (with the limited knowledge that they have on Scrum).

In other words, the duties of the development team will involve:

* Contacting clients to ask for feedback on the work in progress and additional requirements that might have to be integrated into the scheduled tasks
* Addressing concerns of the other team member, including difficulties completing their assigned tasks and other issues
* Sorting and organising the Backlog by priority, according to the Requirements Prioritisation document
* Being clear and transparent to ensure that each item in the Backlog is well defined and fits into the requirements of the project as seen in the SRS document. The team member who creates a new item should be able to explain what it is about and its purpose and reasoning within 2 minutes
* Keeping track of the progress of the tasks listed in our Asana board
* Discussing next steps and progress with the other team member during casual meetings (Mondays, Thursdays and Fridays) and Scrum meetings (exact dates to be confirmed by the lecturer), as well as through our usual communication channel online daily – which is Slack
* Ensuring that after each Increment is marked as “Done” (at the end of each sprint, which takes 2 weeks), working software is released according to what was planned for such sprint – regardless of how basic or complex it is
* Mediating Scrum meetings by reviewing topics planned in the agenda and time, as well as the preparations necessary to hold such events – such as reading each other’s report document for the past sprint
* Being proactive by investigating solutions to critical bottlenecks or roadblocks that potentially delay the progress of the current sprint before such issues are formally discussed by the end of such sprint
* Helping each other understand scope and goals of the current sprint and next steps, optimising the time required to complete their tasks
* Clarifying any questions or concerns that the clients might have regarding the scope of the project or its progress
* Taking responsibility for his own work by testing and verifying potential issues or bugs before it is committed – any caveats or downsides to certain approaches taken should be communicated to the other team member

To summarise the procedures and practices based on Scrum that will serve as basis for the management of this project:

* **Sprint**: a well-defined Scrum event that lasts 2 weeks (in the case of this project). During this period, developers will collaboratively work on their respective tasks assigned on Asana, submitting each task or subtask completed to our repositories on GitHub (see section **17** for more details). Changes to tasks assigned will only be made if absolutely necessary and will be thoroughly discussed with the other team member so that no half-measures will be taken
* **Sprint Planning**: another important Scrum event which duration depends on the duration of each Sprint. Since our Sprints will last 2 weeks, each Sprint Planning will last at most **4 hours**. During that period, a goal will be clearly defined based on the work finished on the previous Sprint and the next few items in the Backlog will be assigned to each team member. If there are any dependencies between the tasks assigned, they will be scheduled appropriately (and possibly broken down into subtasks) so that one of the developers will not be overwhelmed and pressured into finishing their tasks within an unreasonable time frame. Each Sprint Planning will be held as a meeting either face-to-face on Mondays, Thursdays or Fridays or online through Slack.
* **Sprint Review**: an event held at the end of each Sprint. Its goal is to analyse the artifacts done in the current sprint, whether there is anything missing, answer lingering questions regarding non-critical issues, assess the performance of the team as a whole, start discussing the next Sprint and estimate the timeline to complete the next possible tasks based on how well the team has made progress and the strengths of each team member. Such events will last at most **4 hours** and be held either face-to-face on Mondays, Thursdays or Fridays or online through Slack.
* **Scrum**: commonly defined as daily meetings to review the work done so far, discuss roadblocks and next steps, the dates when they will be held will be defined by the lecturer (with the first one scheduled for 14/09/2020). To compensate for the lack of such events, developers will meet informally face-to-face three times a week (during and after class hours) to discuss the topics typically addressed in Scrum meetings.

As can be seen, Scrum practices will not be strictly implemented in this project. However, the plan of action described in this section should hopefully provide a good foundation to accomplish the goals of this project on time.

A **Kanban** board was created using Asana for the developers to manage their tasks on each Sprint, as well as the Backlog of the project. See section **17** for more details.

# Budget

|  |  |
| --- | --- |
| Staff | Cost |
| Diego C. | On average $ 21 / hour |
| Stefan S. | On average $ 21 / hour |
| Gerardo G. | On average $ 21 / hour |

Total cost (estimated): **$ 28,350**

(75days, 6 hours/day, $21/hour)

(To be rectified to take additional costs into account)

# Business Processes

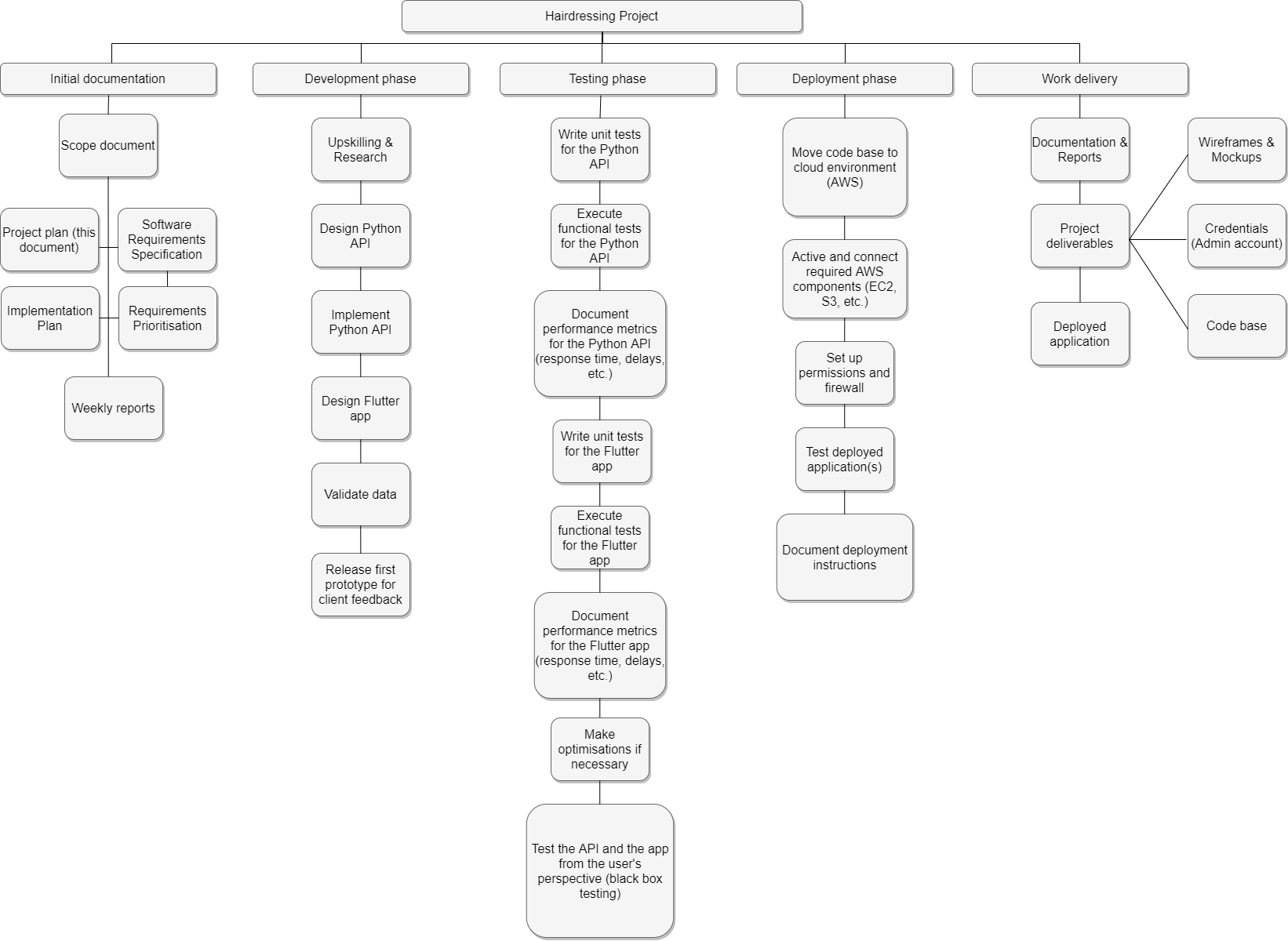
|  |  |  |  |
| --- | --- | --- | --- |
| **Product** | **Business Process** | **User Training Required** | **OSH Requirements** |
| **Style Me app** | Changing Hairstyle | Hints:  Choose a face shape to improve realism \*(actively links back to face shapes) |  |
| **Style Me app** | Changing Hair colour | Hints:  For darker hair select stronger colours |  |
| **Style Me app** | Taking Portrait Selfie | Hints:  Clean background  Bright, Soft Lighting  \*A square crop outline for the face | Be careful of your surroundings when taking a picture (especially outside) |
| **Style Me app** | Browsing Photos from camera on phone | Hints:  Navigate to the “Upload picture” section and tap “from my gallery” |  |
| **Style Me app** | Selecting A Face Shape | Displaying face shapes & possibly examples |  |
| **Style Me app** | Photo Gallery | Hints:  Navigate to the “My pictures” section to visualise previously uploaded pictures |  |
| **Style Me app** | Results Gallery | Hints:  Select 2 styles to compare |  |
| **Style Me app** | Logging in | As soon as the app loads for the first time, users will be presented with a login screen. They should input either their username or their password where indicated on each field. |  |

A user manual will be provided with more in-depth instructions.

# Technical Requirements [[2]](#footnote-2)

* 1. Network infrastructure.
     1. None
  2. Inventory management
     1. Adding hairstyles to the app globally via the app with an admin account
  3. Security or privacy provisions
     1. An admin account is provided for managing the app
     2. Most security measures and best practices will be implemented to ensure that users’ data is kept safe

# Work Breakdown Structure



# Schedule

See the attached Implementation Plan.

# Change Management Plan

Several changes must or could happen to our project. The ones described below are most likely to occur according to our projections:

## 9.1. Technologies used

Since not everyone in the development team is familiar with the choices of technologies outlined in the project specification, upskilling will be needed for the following:

**Flutter**: mobile framework to be used to develop the Style Me app that uses Dart.

**Fast API**: server-side web framework that uses Python, which will handle image uploads and the business logic to process such images through Machine Learning libraries.

**Cloud Firestore, S3**: storage solutions for the images processed by the app.

**GNU/Linux, Digital Ocean, AWS**: production environment of the APIs consumed by the app.

* Scope (impacted parties): development team.
* Plan: train developers who are not familiar with these technologies with a series of tutorials gathered by other developers over **5 days**. Teach them the main concepts of these technologies, provide examples and sample code for relevant use cases for this project (e.g. image upload, sending API requests).
* Benefits: By the end of this upskilling period, developers who were previously not familiar with these technologies should feel comfortable writing code using them and satisfactorily complete their tasks.
* Milestones: After **3 days**, developers should be able to create simple applications from scratch using these technologies. After **5 days**, they should be able to contribute to our project and work on their assigned tasks.

## 9.2. Collaboration tools used

It is possible that the collaboration tools currently used by the development team (Asana, GitHub, Trello) are not sufficient for organisational purposes.

For instance, the free tier of Asana does not allow using timelines, which could be important to more easily visualise schedules and milestones outlined by the scope document of this project.

In such cases, **1 day** should be enough for the developers to either familiarise themselves with the additional features introduced by the “premium” tier of the collaboration tools used or to learn a new collaboration tool altogether.

# Quality Management Plan

Quality management will most likely not prove to be a great concern considering the way that the project is structured, its scope and engagement of the developers.

The hair styling app is only supposed to prove that as a team we are capable of making a functional mobile app, it does not necessarily have to be of high quality (only passable quality as it is our first app).

Quality is also not much of an issue due to the over-perfectionistic nature of the team members. Diego often rewrites systems to make them more logical, Stefan is visually obsessed with interfaces, & Gerado uses software that is much higher level than required.

As a team we also vet each other’s work when possible through **at least** one other team member before moving on to our next task.

# Risk Management Plan

Each member in the team is knowledgeable in different pieces of software, to minimize risk each team member should have a basic understanding of each-others’ work. This should be established through meetings/voice-calls/screen-sharing. Thus, if one team member is no longer able to continue their work another can pick-up/understand their work more quickly than a newcomer and continue it.

Only one member contains the credentials for the Digital Ocean account being used to host the app, however the code for this droplet is backed up on a Git repository with a setup guide & can be relatively easily deployed to another account.

# Team Members / Human Resources (Responsibility Assignment Matrix)

|  |  |  |
| --- | --- | --- |
| **Name** | **Skills Required** | **Training Required** |
| Diego | Dart, Python | Flutter, FastAPI |
| Gerado | Dart, Python | Flutter, FastAPI |
| Stefan | Dart, Python | Flutter, FastAPI |

# User Acceptance Testing

The app will be tested with reasonable expectations by ourselves, fellow students, and family members. Testing will focus on whether the app is logical and simple to use. Forms will be given to testers, so that they can more easily provide feedback & outline which parts of the application were difficult to use. Once a functional prototype it is working feedback will be gathered multiple times a week & changes to the app will be made weekly, based on the provided feedback.

Asking our clients to test the application is almost impossible due to their schedules. Our clients will not be a large influence during the testing, though their perspectives are import for the design of the app, so they will be contacted for issues surrounding major design changes.

# Hand over Document

A scope document: a basic contract of the work we will complete, with deadlines. This will include internally or as attached documents; a summary of the research we have done for the software/APIs/hosting solutions we have chosen, and the wireframes of the app.

# Communication Management Plan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key Stakeholders** | **Stakeholder Engagement Action (select 1)**   * Inform * Consult * Involve * Collaborate * Empower | **Aim of the communication. Specific topics**  Content, format, level of detail | **Tools to be used**  Meeting, reports, emails. | **By Whom**  Position responsible | **Frequency** |
| **Diego C.** | **Collaborate** | **Organising tasks, reviewing peer work, completing tasks assigned on each sprint, submitting prototypes and other deliverables to get feedback from clients** | **Meeting: Blackboard collaborate, Slack, face-to-face at North Metropolitan TAFE**  **Reports: Word document, Slack, E-mail** | **Developer** | **Weekly** |
| **Gerardo G.** | **Collaborate** | **Research libraries, organize dependencies** | **Meetings, slack, email** | **Developer** | **Weekly** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Collaboration tools

The collaboration tools used in this project for development purposes are mainly **GitHub** and **Asana**.

A single GitHub repository will store all the source code of this project and additional (personal) repositories will be created by each team member for tasks that are not directly related to the project, but still contribute to the progress of this project. As an example, the repository linked below was created by Gerardo G. when he was working on upskilling tasks for FastAPI:

<https://github.com/ggornes/FastAPI-test>

Regarding the main repository of this project, the process in which the code related to each task will be submitted will be defined as follows:

* + - 1. Developer commits his changes and opens a new pull request on GitHub
      2. Another developer reviews his work and either approves (by merging) the pull request or rejects it
      3. In case the pull request is rejected, the author of the pull request should address any issues as scheduled for the task and submit a new pull request when done
      4. A gracing period of 48h will be provided for the author of the pull request in case there are still any pending issues to be addressed
      5. If all options are exhausted, developers will have an emergency meeting and break down the pending task into subtasks and work together to help the original author of the pull request

As for our Asana board, it is inspired by the **Kanban** methodology and contains six columns, listed from left to right: User Stories (gathered according to requirements obtained via communication with the clients), Backlog, TODO, In Progress, Pending and Done.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Items from the Backlog will be added to the TODO column after each Sprint Planning, with an assignee and appropriate deadline according to our Implementation Plan. Subtasks may be added to each task with different assignees, so the assignee indicated in each task as illustrated in the pictures above may not accurately reflect which person is currently working on what. See the picture below as an example.

A screenshot of a cell phone

Description automatically generated

As can be seen in the picture above, Diego C. was assigned to install all dependencies in the new cloud environment, whilst Gerardo G. was assigned to set up a database and permissions, even though the main task itself (Cloud migration) indicates that Gerardo G. is the assignee.

**Links:**

* GitHub repository of this project: <https://github.com/HairdressingProject/styleme>
* Asana board: <https://app.asana.com/0/1187175073096657/board>

1. Requirements may include applicationbusiness, database, network, people in the organisation, platform, system [↑](#footnote-ref-1)
2. Bandwidth, hardware problems, network growth, network security, network traffic congestions, new technologies, power usage [↑](#footnote-ref-2)