

National College of Ireland

BSc in Computing – Software Development Stream

2019/2020

Joey Tatú

15015556

joey.tatu@student.ncirl.ie

Body Branding Booking (3B)

An appointment scheduling website for artists and their clients.

Technical Report

**Table of Contents**

[Executive Summary 4](#_Toc26443733)

[Definitions, Acronyms, and Abbreviations 5](#_Toc26443734)

[1 Introduction 6](#_Toc26443735)

[2 Background 8](#_Toc26443736)

[2.1 Objectives 9](#_Toc26443737)

[2.1.1 General 9](#_Toc26443738)

[2.1.2 Databases 9](#_Toc26443739)

[2.1.3 Reviews 9](#_Toc26443740)

[2.1.4 Appointments 9](#_Toc26443741)

[2.1.5 Artificial Intelligence 10](#_Toc26443742)

[2.2 Research 10](#_Toc26443743)

[2.2.1 Personalisation 10](#_Toc26443744)

[2.2.2 Optimisation 11](#_Toc26443745)

[3 System 12](#_Toc26443746)

[3.1 Requirements 12](#_Toc26443747)

[3.1.1 Functional requirements 12](#_Toc26443748)

[3.1.2 Requirement 1: Create Schedule 13](#_Toc26443871)

[3.1.3 Requirement 2: Book appointment 15](#_Toc26443872)

[3.1.4 Requirement 3: Review Artist 19](#_Toc26443873)

[3.1.5 Requirement 4: Message user 21](#_Toc26443874)

[3.1.6 Requirement 5: Create automated appointment 24](#_Toc26443875)

[3.2 Non-Functional Requirements 26](#_Toc26443876)

[3.3 Performance/Response time requirement 26](#_Toc26443877)

[3.4 Availability requirement 26](#_Toc26443878)

[3.5 Recover requirement 26](#_Toc26443879)

[3.6 Security requirement 26](#_Toc26443880)

[4 Design 28](#_Toc26443881)

[4.1 System Architecture 28](#_Toc26443882)

[4.2 API 28](#_Toc26443883)

[4.3 Implementation 29](#_Toc26443884)

[4.4 GUI Layout 29](#_Toc26443885)

[4.5 Testing 36](#_Toc26443886)

[4.6 Server responsiveness 36](#_Toc26443887)

[4.7 Connecting 37](#_Toc26443888)

[4.8 End-user testing & Evaluation 37](#_Toc26443889)

[4.9 Scalability and performance 37](#_Toc26443890)

[5 Conclusions 38](#_Toc26443891)

[5.1.1 Advantages & disadvantages 38](#_Toc26443892)

[5.1.2 Opportunities and limits of the project. 38](#_Toc26443893)

[6 Research 39](#_Toc26443894)

[6.1 Personalisation 39](#_Toc26443895)

[6.2 Optimisation 39](#_Toc26443896)

[7 References 41](#_Toc26443897)

[8 Appendix 43](#_Toc26443898)

[8.1 Project Plan 43](#_Toc26443899)

# Executive Summary

Body Branding Bookings (3B) is an appointment and booking managing web service system, specifically designed for tattooists, piercers and body modifiers (i.e. the artists), and their clients. Features include personalisation for clients, and optimisation techniques and protocols for the system itself.

With personalisation, the client will get the likes of customised adverts to show other body modifications they may want in the near future, as well as the likes of tattoo and piercing shops nearby they may wish to go visit. This could work in the sense that if the client (e.g.) books a tattoo appointment for a large tattoo, they would see customised adverts for the likes of tongue splits, scarification and other so-called “extreme” modifications. This also depends on the frequency of when they get a tattoo.

As time progresses and the website is being used more and more, optimisation will need to be implemented at this early stage, to future-proof the website and to ensure that with large loads of data being transferred from the server to the database, messaging system and payment system, that the system will not slow down or become congested, or in the worse-case scenario, the server would crash.

This would also save on costs in the long run as less resources would be needed, in comparison if optimisation was not used. It’s important to consider as much as possible for the website to ensure it’s working with a simplistic interface and that it stays online.

.

# Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| 3B | Body Branding Bookings, the working name of the project. |
| Artist | A tattooist, body piercer or a body modifier |
| Client | A customer of the Artist |
| User | Either an artist or client. |
| AWS | Amazon Web Services |
| AWS Cloud 9 | A cloud-based integrated development system (IDE) for writing, running and debugging code. |
| AES | A type of encryption used in databases. |

# Introduction

3B is a web service system. The fundamental feature of 3B is to assist artists generate times they are available to work and have their clients book an appointment. Another feature is that artists have a page on the site for their work and shop location that clients can rate and leave comments about their experience.

The developer has many tattoos and piercings and knows first-hand how unnecessarily complicated booking a tattoo appointment can be and booking an artist that can be trusted. From dealing with deposits, changing appointment dates due to conflicts, it seems that this is far more complex than it needs to be. The project will address this.

3B will have an artist area and a client area. The artist will be able to set when they are available to work and set the likes of prices, sizes, locations (of tattoo/piercing), how long it will take to complete, and so on. Currently, there is no appointment service that fully caters to artists. With body modification becoming more and more popular, it is becoming a necessity.

Another feature includes personalisation. This will be aimed at the client. An example of this would be when the client books a tattoo, they would see artist’s pages for piercings and other body modifications, as well as shops nearby to their booked appointment.

The web application will need an Internet connection to operate correctly. However, a cached version of the appointment schedule for the artists and what appointments have been booked will be generated, so users will not have an issue viewing these, even if internet or capacity issues occur.

Users will need to create an account with email and password, or via social login (e.g. Facebook or Google) as only registered users will be able to create their work schedule and view their appointments, and use the service.

(AWS) Cloud 9 will be used to develop the prototype to create one Use Case and have it working. It will be further developed in Cloud 9 after this, but it is expected that it be transferred to its own domain and hosting before completion. Cloud 9 uses website coding such as HTML, JavaScript, jQuery. MySQL will be used for the databases. Encryption will also be used in communication to the databases.

# Background

The original idea for this project came about in early 2019. The thought process was to create a social networking Android app. A website, Inked-Up.com, was a comparison for the app. This site specifically targets adult men who are into body modification such as tattoos and piercings. With this site, it’s felt that it is dealing with a very distinctive niche which would not be ideal for everyone. The site also seems very outdated and seemingly has become a replacement platform from Tumblr, since adult content is no longer permitted. (Tatú, 2018) (Inked-Up.com, 2019)

With new social networking platforms being developed, with an example of TikTok being the newest and most popular, the app was going to be similar to the likes of Facebook and Inked-Up.com, but for everyone with an interest in body modifications. The ability to rate tattoo Artists was also to be implemented.

With this current project, it will be on the same general topic; tattoos, piercings and body modifications. Trying to book an appointment to get a tattoo or piercing is quite tedious in some situations. Contacting the artist or Artist can be troublesome, either contacting them through social networking sites or by email. There are delays in communication. It seems like a long process just to get an appointment.

It’s common knowledge that, especially, for getting a tattoo, a deposit is required. This means that the client needs to go to the Artist or tattooist, give their deposit, and then return for the appointment. If the client doesn’t pay the deposit, their appointment is still there and it wastes the artist’s time. A good starting point to this project is to automate these.

To remedy this, the idea of Body Branding Bookings (3B) was generated. With 3B, the artist can select the dates they wish to work and where they are available. The clients can book one of these timeslots. The general size and/or the time it will take to complete the appointment will be set by the artist and selected by the client to determine the correct time slot needed to be allocated. Artists will have a page where clients can rate and leave reviews.

Java, JavaScript and jQuery will be revisited. These will greatly be beneficial to the project. If a new language is needed, it will be heavily considered.

## Objectives

### General

The object of this Project is to take a brick and mortar scenario[[1]](#footnote-1) of where artists and clients process appointments and payments manually, and create a website where this can be automated.

There will be two main types of accounts on the website; artists and clients. The artist account can either be a solo artist or a collection of artists.

### Databases

There will be two databases connected to the website. One database will be for the artist pages, client profiles and the actual appointments. The other database will be for the stock and the stock control.

### Reviews

Reviews of the artists will be generated by the clients. These will be visible on the Artist’s page on the website. The Artist can choose to hide a review. Clients must have at least one previous booking with a particular Artist to review that artist. This will avoid people never visiting the Artist or artist posting reviews.

### Appointments

Artists will generate what days and times the artist(s) are working. The client will book an appointment that is free with the artist. Sample sizes will be provided to generate the correct amount of time to be reserved. A deposit or full payment will be required from the client to confirm their booking.

### Artificial Intelligence

Artificial Intelligence will focus of individualisation as well as the issue of optimisation.

With individualisation, the client will have recommendations. For example, if they generate a piercing appointment, the likes of piercing jewellery or piercing locations would be recommended to them.

With the issue of optimisation, the likes of security and payment handling will be discussed and implemented.

## Research

### Personalisation

Brick-and-mortar stores are beginning to put more digital knowhows into their stores. For the clients, this is a great benefit for them by making their experience better. Some of the brick-and-mortar stores are not developing their business’ technological skills, in regard to customer interaction. (Betzing, et al., 2018) However, there are methods of doing this.

When developing from a brick-and-mortar store to a more digitised one, the likes of sensors such as cameras and facial recognition devices can figure out the basics of a person; such as their height, gender, and approximate age. A person’s facial expressions can also be considered to explore whether the client is in a positive or negative mood and whether they are enjoying the service. This data can be obtained by using smart devices on a local network. (Betzing, et al., 2018) (Webner, 2019)

This concept is similar to the physical store Amazon Go by Amazon, where Artificial Intelligence is used to keep track of the clients and store items. (McFarland, 2018)

In this project, it is hoped that a personal experience can be added for the artists and clients. It is expected that personalisation with be sought from a user’s profile, such as their age, gender.

In an example of how this could work; a male client who has just turned 18 and would like to get a tattoo. The system would ask the client to heavily consider whether they would want the tattoo, and to seriously consider avoiding visible areas such as the face, head or hands.

However, if the client is a male in their mid-40s, this message would not be shown to them as the thought process for the client would be different. It could be assumed the client already has a lot of tattoos. Instead, an upload link to share their previously obtained tattoos would be shown.

### Optimisation

Another consideration of this project is optimisation. If one takes the clothing and textile industry as an example, one can optimise what colours, fabrics and designs are in a high demand. This is completed by eliciting and gathering “Product Usage Information (PUI)”. This is obtained from clients, experts and investors. From this data, the latest styles and what the client wants are known. (Hribernik, et al., 2019)

In this project, optimisation could be used for the general size of the tattoo, the colours of the ink used. If, for example, a lot of tattoos are small and use only use 3 or 4 colours, the system would recognise this and automatically order new colour inks to the artist. On that note, the artist could input how much ink would be used for a particular size and machine learning could be used to order and maintain stock with the artist.

The same would also go for piercings, if a certain ring or stud is used on a regular basis, these could be automatically ordered. The likes of stock control for piercers could also be controlled here.

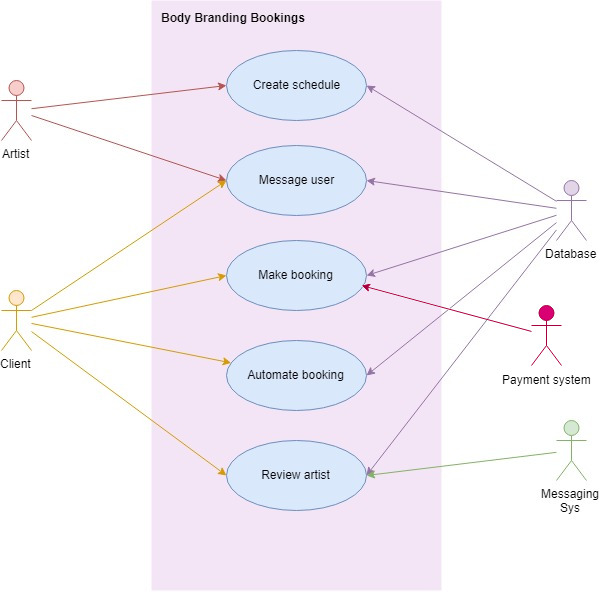
# System

## Requirements

### Functional requirements

This section lists the functional requirements in ranked order. Functional requirements describe the estimated outcomes of the System.

#### Use Case Diagram



### Requirement 1: Create Schedule

#### Description & Priority

This Use Case describes how an Artist creates their work schedule, to indicate when they are available to work.

##### Use Case

**Unique ID:** createSched

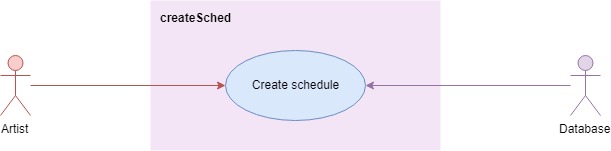
**Scope**

The scope of this use case is for the Artist to create and edit their work schedule.

**Description**

This use case describes how the Artist can set up and edit their work schedule so Client’s will be able to book time slots when the Artist is available.

**Use Case Diagram**

****

**Flow Description**

**Precondition**

* The System is active and working correctly
* The database and connection are actively working correctly.
* An Error log file is stored within the website’s files.

**Activation**

This use case starts when the Artist signs up to 3B.

**Main flow**

(Function: Sign up)

1. The Artist signs up on the System. <See A1>
2. The System shows the Artist the sign-up page.
3. The Artist inputs their email, password, password again and indicates they are an Artist, and then continues.
4. The System encrypts the data using AES.
5. The System sends the data to the Database. <See E1>
6. The System receives a response from the Database
7. The System shows the Artist the home menu for artists.

(Function: Create Profile)

1. The Artist accesses their Profile on the System. <See A2>
2. The System shows the Artist the edit profile page.
3. The Artist inputs their name, location, bio, interested in (i.e. tattoos, piercings and/or body modifications), profile photo.
4. The Artist saves the inputs to the System.
5. The System encrypts the data and sends it to the database.
6. The System shows the Artist their profile page.
7. The Artist returns to the home menu on the System.
8. The System shows the Artist their home menu.

(Function: Create Artist Page)

1. The Artist accesses their (Artist’s) Page on the System. <See A2>
2. The System shows the Artist the edit page.
3. The Artist inputs the company name, address of company, bio of company, and the company’s profile photo.
4. The Artist saves the inputs to the System.
5. The System encrypts the data and sends it to the database.
6. The System shows the Artist their Artist Page.
7. The Artist returns to the home menu on the System.
8. The System shows the Artist their home menu.

(Function: Create schedule)

1. The Artist accesses their Schedule on the System
2. The System contacts the database to check if dates and times have previously been set. <See A3>
3. The System shows the Artist the calendar page with the month view.
4. The Artist selects the dates and time they are available to work.
5. The Artist specifies the dates and times they are strictly unavailable.

(Optional: The Artist selects the calendar to repeat every month)

1. The Artist inputs sizes, example photos, prices and estimated completion time.
2. The Artist saves the inputs to the System.
3. The System encrypts the data and sends it to the database.

**Alternate flow**

A1: <Artist has previously created an account>

1. The Artist logs in to the System.
2. The Artist inputs their email and password.
3. The System connects to the database and verifies the user. <See E1>
4. The database sends the reply to the System.

<returns to number 7 in Main Flow>

A2: <Artist has previously created a profile and/or artist page.>

<returns to number 24 in Main Flow>

A3: <Artist has previously created a calendar>

1. The System connects to the database
2. The System loads the Client’s bookings on the Calendar (if any)
3. The System retrieves the Artist’s calendar data and shows it on screen.

<returns to number 26 in Main Flow>

**Exceptional flow**

E1: System cannot connect to database.

1. The System cannot connect Database
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

<Returns to number 1 in Main Flow>

**Termination**

This Use Case is terminated when the Artist has successfully created or edited their calendar.

**Post condition**

The System goes into a wait state

### Requirement 2: Book appointment

This Use Case describes how a Client books an appointment

#### Use Case

**Unique ID:** makeBooking

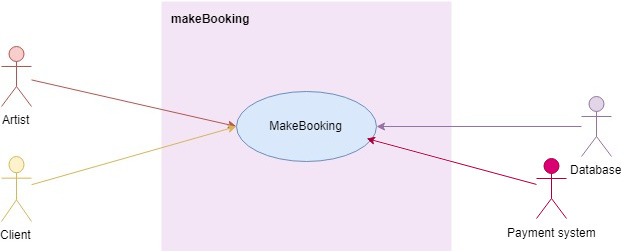
**Scope**

The scope of this use case is for a Client to book and edit an appointment, and for a user (Artist or Client) to delete an appointment.

**Description**

This use case describes how the Client can book and edit their appointments and for either an artist or client to delete an appointment. The Client can only delete their own appointments while the Artist can delete any appointments on their calendar.

**Use Case Diagram**

****

**Flow Description**

**Precondition**

* The System is active and working correctly,
* The Artist is signed in
* The database and connection are actively working correctly.
* Appointments (Artist) and bookings (Client) are displayed on a calendar on the website, but stored in the database.
* The Payments System is connecting and working correctly.
* An Error log file is stored within the website’s files.

**Activation**

This use case starts when the Client selects “Bookings” on their home menu.

**Main flow**

(Function: Sign up)

1. The Client signs up on the System. <See A1
2. The Client shows the Artist the sign-up page.
3. The Client inputs their email, password, password again and indicates they are a Client, and selects continues.
4. The System encrypts the data using AES.
5. The System sends the data to the Database. <See E1>
6. The System receives a response from the Database
7. The System shows the Client the home menu for artists.

(Function: Create Profile)

1. The Client accesses their Profile on the System. <See A2>
2. The System shows the Client the edit profile page.
3. The Client inputs their name, location, bio, interested in (i.e. tattoos, piercings and/or body modifications), profile photo.
4. The Artist selects the inputs.
5. The System encrypts the data and sends it to the database.
6. The System shows the Client their profile page.
7. The Artist selects “Home”
8. The System shows the Client their home menu.

(Function: Make booking)

1. The Client accesses their Bookings on the System.
2. The System loads the Bookings page.
3. The Client adds a new booking on the System. <See A1>
4. The System loads the Edit Booking page.
5. The Client inserts the relevant information (size and location of (e.g.) tattoo)
6. The Client selects the date and start time of the booking.
7. The Client uploads reference photos
8. The System shows the price.
9. The Client continues
10. The System sends the Client to the Payment System.

(External: Payment System put transaction on hold.)

1. The System checks if there are any overlapping appointments. <See A2, E1>
2. The System sets the booking to “unconfirmed”.
3. The System generates a booking reference number and adds it to the appointment.
4. The System inserts the data into the database.
5. The System loads the Booking Saved page, notifying the Client the booking is saved.
6. The Client returns to the home menu.
7. The System notifies the Artist of a new appointment.
8. The Artist confirms the appointment. <See A3, E2>
9. The System notifies the Payment System to process payment.
10. The System sets the booking to “confirmed”.
11. The System notifies the Client that their booking is confirmed.

**Alternate flow**

A1: <Edit appointment>

1. The Client selects a booking
2. The Client edits the booking on the system.
3. The System retrieves the booking data from the database.

<returns to number 20 in Main Flow>

A2: <Overlapping appointments>

1. The System checks for the next available timeslot with the required amount of time.
2. The System highlights the date and time
3. The System displays a message saying the Artist is booked, with the suggested new date and time.

<returns to number 21 in Main Flow>

A3: <Artist declines>

1. The Artist denies the appointment.
2. The appointment is deleted from the database.
3. The System notifies the Payment System not to process payment.
4. The System notifies the Client to create another booking.

<Goes to number 25 in A1>

**Exceptional flow**

E1: System cannot connect to the database

1. The System cannot connect to the database
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

<Returns to number 1 in Main Flow>

E2: <Artist doesn’t select either “Confirm” or “Deny”>

1. The System waits 48 hours, or 24 hours before the appointment, whichever is nearer.
2. The System notifies the Client to contact the Artist directly.

<Goes to number 33 in A2>

**Termination**

This Use Case is terminated when the Client has successfully placed a booking or has edited a booking, or a user has deleted the booking.

**Post condition**

The System goes into a wait state

### Requirement 3: Review Artist

#### Description & Priority

This Use Case describes how a Client can rate an artist and leave a comment on the Artist’s Page.

#### Use Case

**Unique ID:** reviewArtist

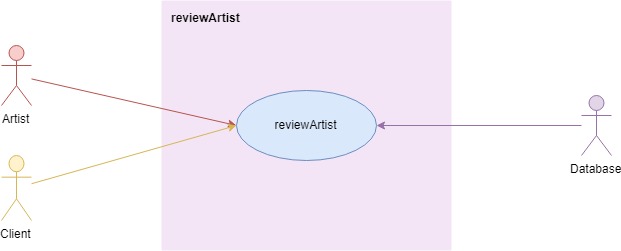
**Scope**

The scope of this use case is to show how a Client can rate an Artist.

**Description**

This use case describes how the Client can view and rate the Artist via the Artist’s Page.

**Use Case Diagram**

****

**Flow Description**

**Precondition**

* The System is active and working correctly,
* The Client is signed in
* The Artist is signed in
* The Artist has previously created an Artist’s Page.
* The database and connection are actively working correctly.
* An Error log file is stored within the website’s files.
* Users cannot edit or delete ratings.
* Artists need to approve comments before they’re publicly visible.
* If an Artist does not approve a comment, it is deleted.

**Activation**

This use case starts when the Client selects “Rate Artist”.

**Main flow**

1. The Client selects “Rate Artist” from their home menu.
2. The System retrieves and lists all the Artists that the Client booked with in the past. <See E1>
3. The Client selects one of the Artists.
4. The System loads the Artist’s Page.
5. The System calculates the average star ratings and displays it.
6. The System loads the comments that are marked “checked”.
7. The Client selects a star-rating out of 5 stars. <See A1>
8. The Client enters a new review (comment).
9. The Client posts the comment on the System.
10. The System puts the rating into the “Ratings” array.
11. The System puts the comment into the “Comments” array and marks it unchecked.
12. The System notifies the Client that their comment is pending.
13. The System reloads the page and repeats numbers 4 – 6.
14. The Client returns to the home menu.
15. The system notifies the artist that there is a new comment on their Artist’s Page.
16. The Artist approves the comment. <See A2, E2>
17. The System changes “unchecked” to “checked” on the comment.

**Alternate flow**

A1: <Client is editing comment>

1. The System shows an Edit button beside the Client’s comment.
2. The Client edits their comment.

<returns to number 9 in main flow>

A3: <Artist declines>

1. The Artist denies the comment.
2. The comment is deleted from the database.

**Exceptional flow**

E1: System cannot connect to the database

1. The System cannot connect to the database
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

E2: <Artist doesn’t select either “Confirm” or “Deny”>

1. The System waits 24 hours
2. The System notifies the Artist of a new comment on their Artist’s Page.
3. The System waits another 24 hours.
4. The System deletes the comment. (No reply from Artist)

**Termination**

This Use Case is terminated when the Client has successfully rated the Artist.

**Post condition**

The System goes into a wait state

### Requirement 4: Message user

#### Description & Priority

This Use Case describes how an Artist can message a Client, and how a Client can message an Artist.

#### Use Case

**Unique ID:** messageUser

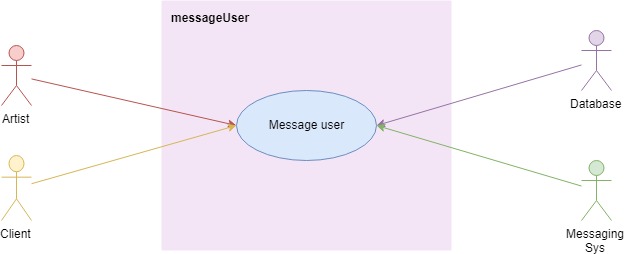
**Scope**

The scope of this use case is to show how a User message another User.

**Description**

The scope of this use case is to show how a User message another User. Artists can message Clients and Clients can message Artists.

**Use Case Diagram**

****

**Flow Description**

**Precondition**

* The System is active and working correctly,
* The Client is signed in
* The Artist is signed in.
* The database is active and working correctly.
* The Messaging System is working correctly.
* An Error log file is stored within the website’s files.
* Users cannot edit or delete ratings.
* Artists need to approve comments before they’re publicly visible.
* If an Artist does not approve a comment, it is deleted.

**Activation**

This use case starts when the selects “Messages” from their home menu.

**Main flow**

<Artist messages Client>

1. The Artist accesses messages on the System. <See A1>
2. The System displays client names that have had an appointment in the last month or have an appointment in the next month.
3. The Artist selects one.
4. The System records the user ID, the type of user (whether artist or client.
5. The System generates a messaging ID.
6. The System send the user ID, type and messaging ID to the database. <See E1>
7. The System sends the user ID, type and messaging ID to the Messaging System. <See E2>
8. The System communicates with the Messaging System and displays messages as they are sent and received.
9. The Artist sends messages as required (external)
10. The Artist closes the messaging dialog box.
11. The System requests to stop the messaging session to the Messaging System.
12. The System loads the Messages page.

**Alternate flow**

A1: <Client messages Artist>

1. The Client accesses messages on the System. The System displays Artist names that they have or had booking with
2. The Client selects one.
3. The System records the user ID, the type of user (whether Client or client.
4. The System generates a messaging ID.
5. The System send the user ID, type and messaging ID to the database. <See E1>
6. The System sends the user ID, type and messaging ID to the Messaging System. <See E2>
7. The System communicates with the Messaging System and displays messages as they are sent and received.
8. The Client sends messages as required (external)
9. The Client closes the messaging dialog box.
10. The System requests to stop the messaging session to the Messaging System.
11. The System loads the Messages page.

**Exceptional flow**

E1: System cannot connect to the database

1. The System cannot connect Database
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

E2: System cannot connect to the Messaging System

1. The System cannot connect to the Messaging System
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

**Termination**

This Use Case is terminated when the User has successfully sent and received messages.

**Post condition**

The System goes into a wait state

### Requirement 5: Create automated appointment

#### Description & Priority

This use case describes how the System creates automatic appointments for the User.

#### Use Case

**Unique ID:** autoBook

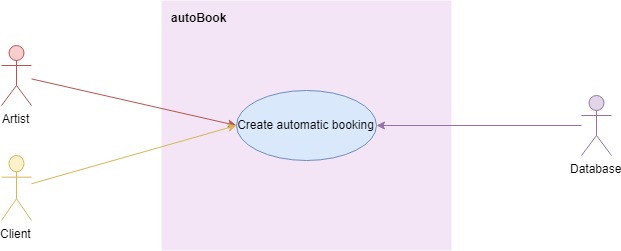
**Scope**

The scope of this use case is to automate a new booking for the Client.

**Description**

The scope of this use case is to automate bookings for the Client. Depending on the number of previous appointments (i.e. more than 4), the system will generate an appointment to be confirmed by the Client and the Artist.

**Use Case Diagram**

**Flow Description**

**Precondition**

* The System is active and working correctly,
* The Client is signed in
* The Artist is signed in.
* The database is active and working correctly.
* An Error log file is stored within the website’s files.
* The Artist and Client need to approve appointment before it’s confirmed
* The Artist and Client are able to turn on this feature

**Activation**

This use case starts every morning at 1am.

**Main flow**

1. The System checks the completed appointment schedule for the day and locates the first appointment. <See E1>
2. The System checks the Client has not opted out for automated appointments. <See A1>
3. The System checks the Artist has not opted out. <See A1>
4. The System adds an appointment for the Client for the Monday 4 weeks away at the same time.
5. The System sets the appointment to unconfirmed.
6. The System checks the Artist’s schedule. <See A2>
7. The System notifies the Client
8. The Client confirms <See A3>
9. The System notifies the Artist.
10. The Artist confirms <See A3>
11. The System sets the appointment to confirmed.
12. The System goes to the next completed appointment that day.
13. The System repeats numbers 2 – 12 until there are no more completed appointments for that day

**Alternate flow**

A1: The Artist and/or Client have opted out of automated appointments

<returns to number 12 in Main Flow>

A2: Artist’s Schedule already has appointment booked for that date and time>

1. The System finds the next available date and time that is not booked

<returns to number 7 in Main Flow>

A3: The Artist and/or Client select deny or do not respond after 24 hours.

<returns to number 12 in Main Flow>

**Exceptional flow**

E1: System cannot connect to the database

1. The System cannot connect Database
2. The System displays a message to the user sating that technical difficulties are occurring.
3. The System stores what happened into an error log.

**Termination**

This Use Case is terminated when the System successfully books new appointments for the Clients and Artists

**Post condition**

The System goes into a wait state

## Non-Functional Requirements

This section specifies other particular non-functional attributes required by the System. Examples are provided below

## Performance/Response time requirement

The System must be able to provide a response to the User in less than 2 seconds. When the System cannot connection to the database or the Messaging Service, a response to the User must be presented in 6 seconds of less.

## Availability requirement

3B must be available 24/7. To ensure this is the case, 2 back-up connections to the database will be developed.

## Recover requirement

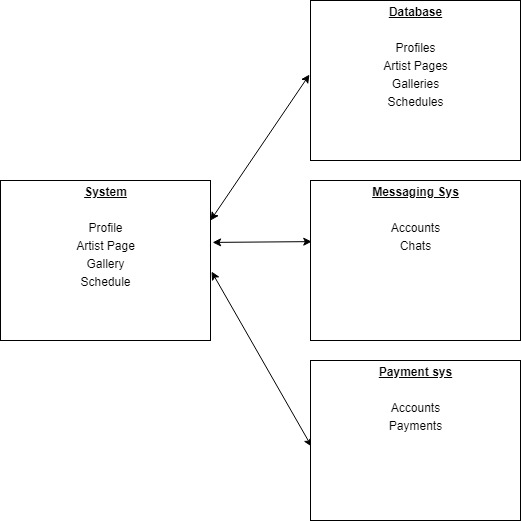
At least 2 backups in separate physical locations must be available. If the database or the Messaging Service is unable to connect to the System, the system will notify the developer at once in the form of an email.

## Security requirement

No plain text passwords are stored anywhere. Encrypted passwords using AES will be stored in the database. Session keys and cookies will be used.

# Design

## System Architecture



## API

Input from User

When the user is inputting text, a text field will be used. This is recognised by the System using the variable name.

Database connections

An example of how tables in the database would be used is the Users tables. Tables used in the database include User, Account, Profile, Pages and Messages.

For the user to see their profile, the System will read the information in the database using API and JSON (a machine-readable format). This ensures (e.g.) all profile layouts are the same for each User.

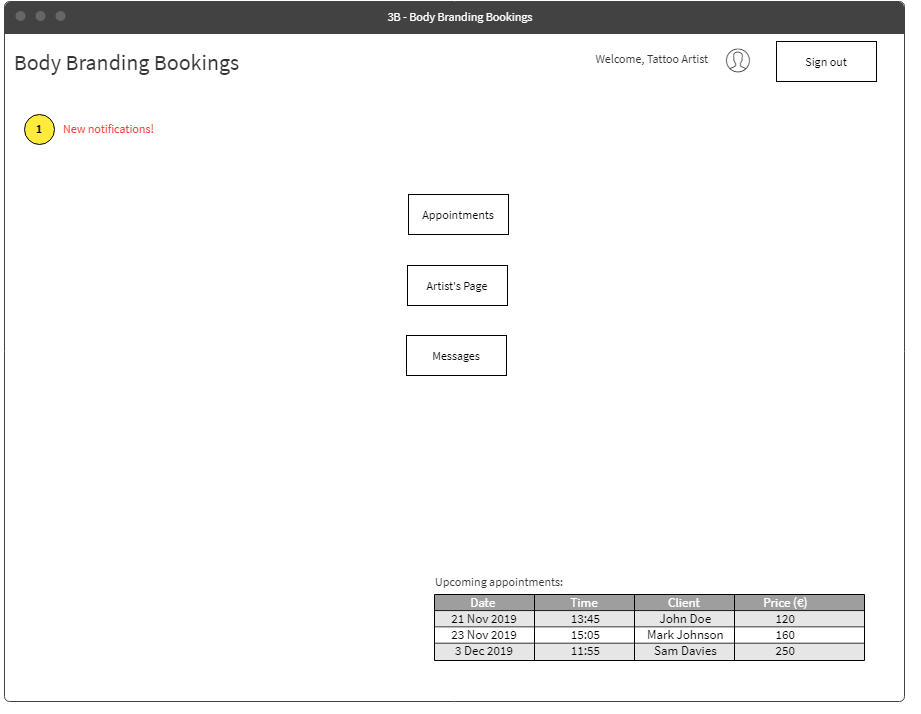
## Implementation

[Working document, will be completed after the prototype]

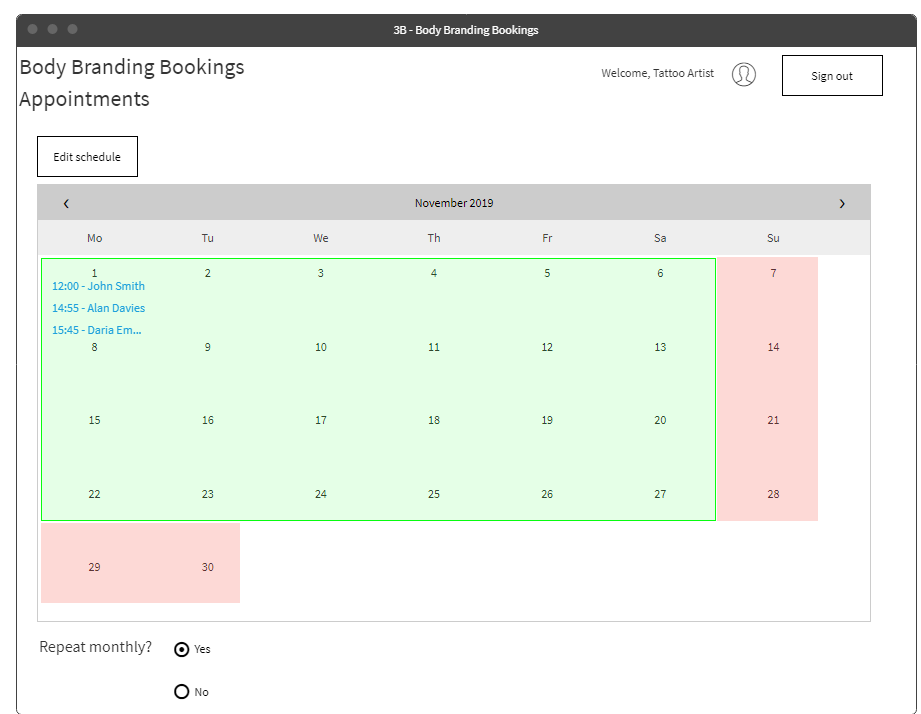
## GUI Layout

These mock-ups show what 3B will look like

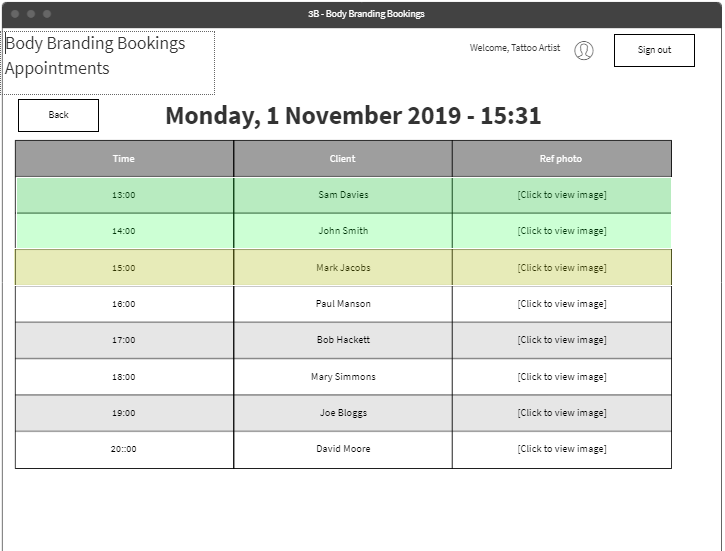
1. This is the main Home Menu for the Artist. It is similar for the Client.



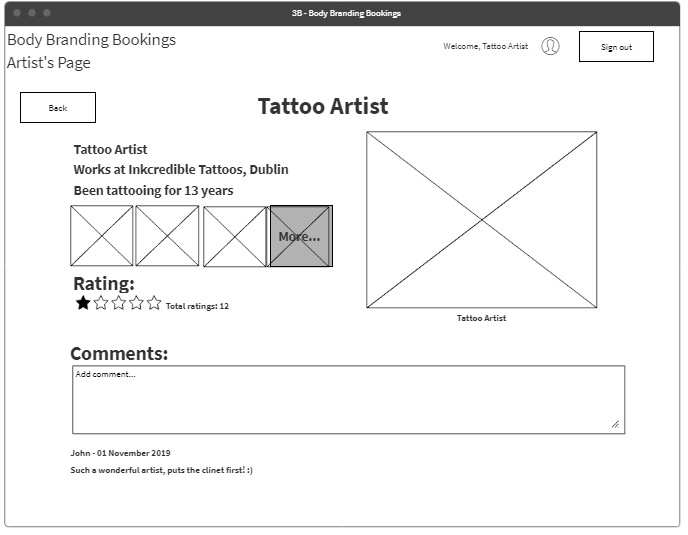
1. The calendar for the Artist to select their work schedule. This example shows how appointments would look on a month-view. The green indicates when the Artist is working and the red is when the Artist is not working.



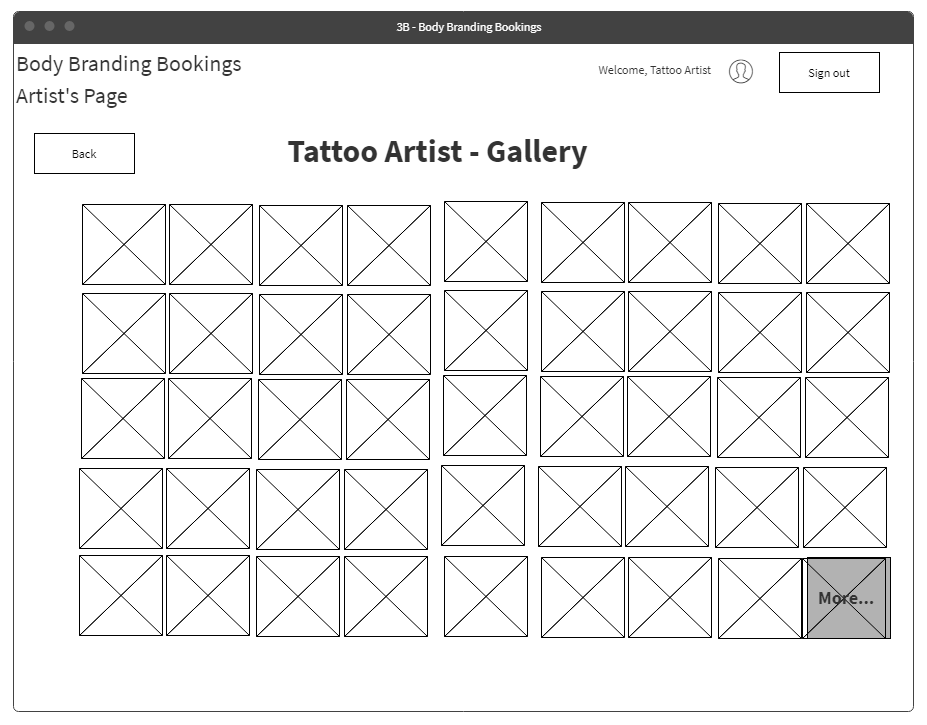
1. An example of the Artist’s calendar showing the appointments for the day. The green shows the appointments that are completed and the yellow shows appointments in progress.



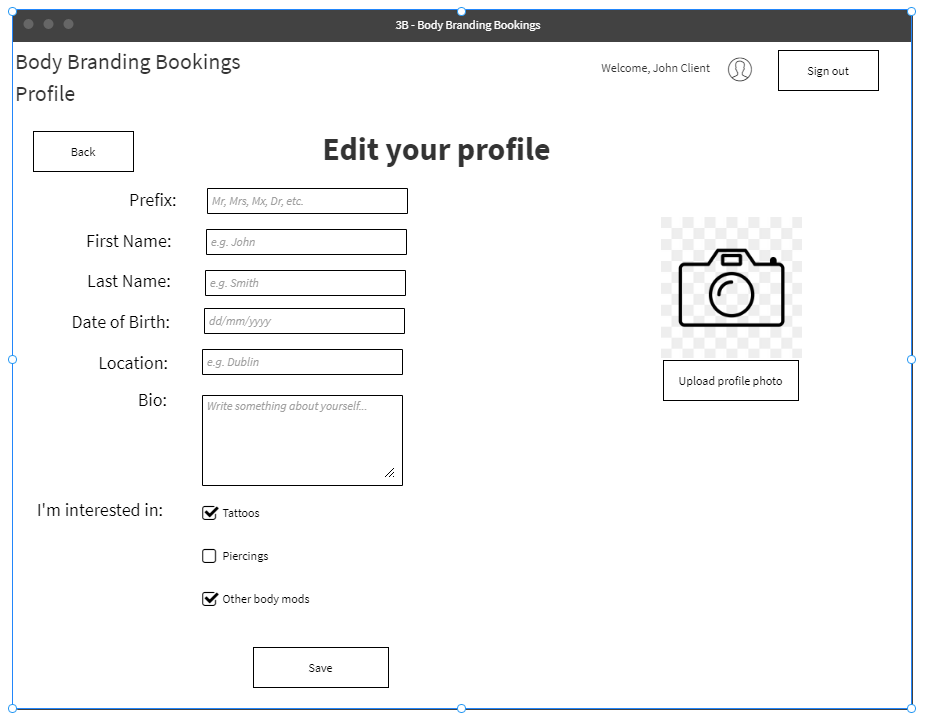
1. This is the Artist page, the “More…” link goes to the Artist’s Gallery.



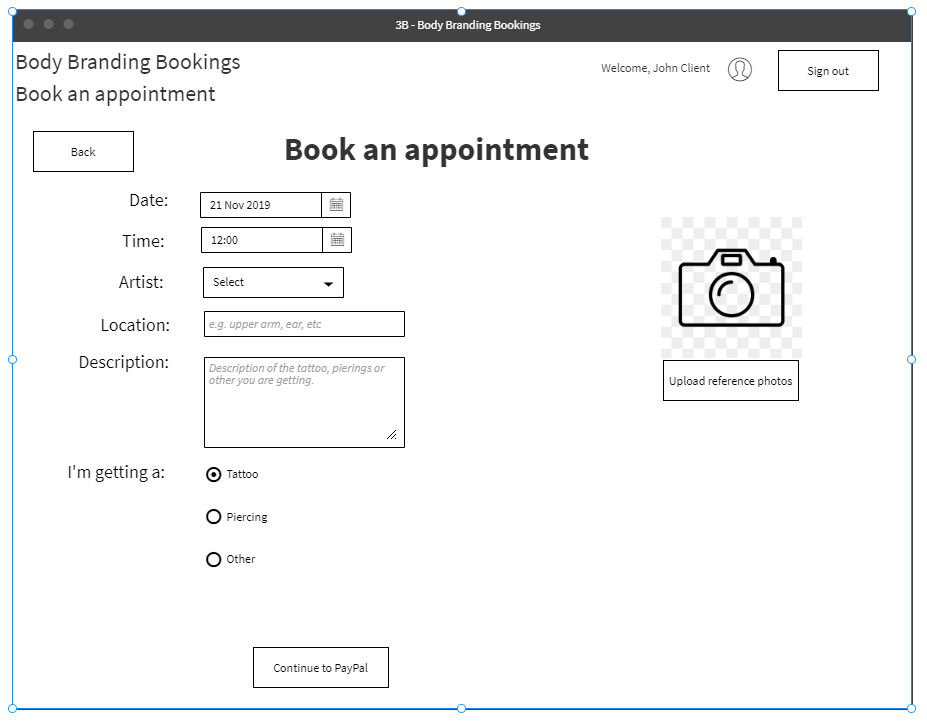
1. This is the Artist’s Gallery, where a Client can view the Artist’s work.



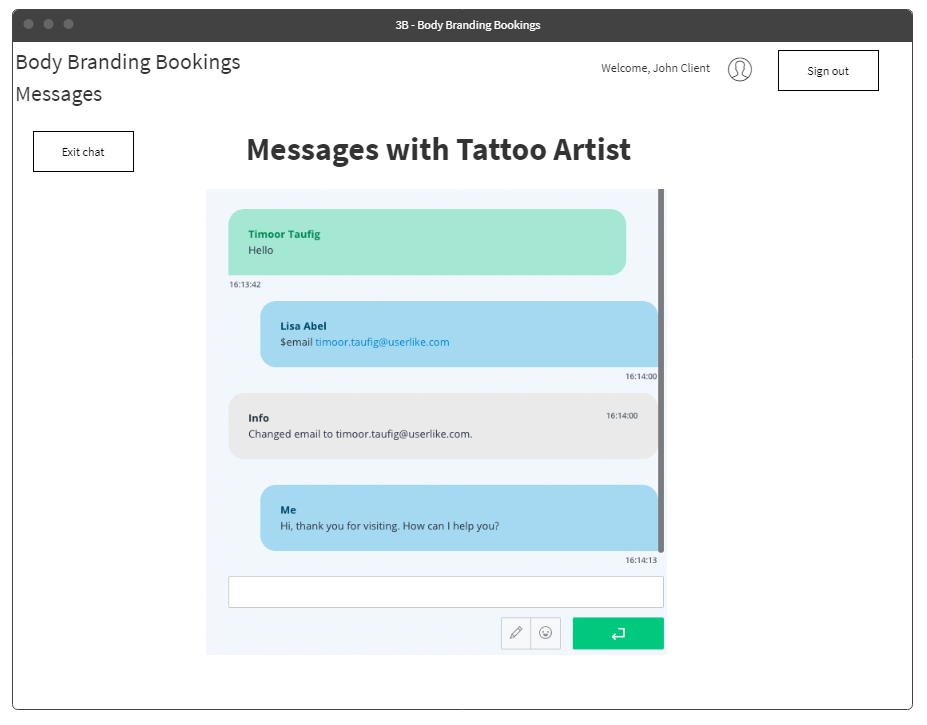
1. This is the Edit Profile page. This is the Client’s view. The Artist’s view has some additional features like where they work and the length of time they’ve been doing their profession.



1. This is how a Client can book an appointment. The Artist field is a drop-down menu of all the Artists listed on the site. The Client can also start typing to select the Artist quicker. Payment for deposit is managed by an external payment system.



1. This is an example of a conversation dialog in the Messages section



## Testing

Testing will be completed by the developer only. Testing will include:

* How fast the server responds
* How fast the server connects to the database and messaging system.
* How relevant is the article that Chatbot returns to the User?

## Server responsiveness

This will be completed in NetBeans using JUnit testing.

## Connecting

This will also be done in NetBeans where possible and will gauge the connection times from when the server makes a request to (e.g.) to the database, and how long it takes to retrieve the information.

## End-user testing & Evaluation

This will be done by the developer. The developer will go through the app and see how well everything feels, as well as measure the ease-of-use and the responsiveness of the website

The developer will also look at the good points and bad point, such as if the app seems clunky or possibly too simple.

## Scalability and performance

It’s important for the social network to handle scaling up or down where needed. Dependant of the number of users, the app may become slow. Testing of this will be done by sending the server and Firebase an increasing number of requests and see how many requests it handles until the server crashes. This will also gauge performance, as if the requests become slower with the increasing requests, the performance will be weak.

# Conclusions

### Advantages & disadvantages

The advantages of this project are that artists and clients will not need to rearrange their appointments manually. This project could potentially be developed further to use any sort of medium, not necessarily body modifications. Artists and clients would also be less stressed in relation to this. Artists will have more time to focus on their skill and/or business as less time responding to messages and emails in regard to appointments.

A disadvantage of this project is that it will be competing with popular calendars like Google Calendar and Outlook Calendar & To-do.

### Opportunities and limits of the project.

The limit of this project will be that for the initial iteration, personal calendars will not be included. However, this project is not intended to replace a calendar service not to be exported to another calendar.

# Research

## Personalisation

Brick-and-mortar stores are beginning to put more digital knowhows into their stores. For the clients, this is a great benefit for them by making their experience better. Some of the brick-and-mortar stores are not developing their business’ technological skills, in regard to customer interaction. (Betzing, et al., 2018) However, there are methods of doing this.

When developing from a brick-and-mortar store to a more digitised one, the likes of sensors such as cameras and facial recognition devices can figure out the basics of a person; such as their height, gender, and approximate age. A person’s facial expressions can also be considered to explore whether the client is in a positive or negative mood and whether they are enjoying the service. This data can be obtained by using smart devices on a local network. (Betzing, et al., 2018) (Webner, 2019)

This concept is similar to the physical store Amazon Go by Amazon, where Artificial Intelligence is used to keep track of the clients and store items. (McFarland, 2018)

In this project, it is hoped that a personal experience can be added for the artists and clients. It is expected that personalisation with be sought from a user’s profile, such as their age, gender.

In an example of how this could work; a male client who has just turned 18 and would like to get a tattoo. The system would ask the client to heavily consider whether they would want the tattoo, and to seriously consider avoiding visible areas such as the face, head or hands

.

However, if the client is a male in their mid-40s, this message would not be shown to them as the thought process for the client would be different. It could be assumed the client already has a lot of tattoos. Instead, an upload link to share their previously obtained tattoos would be shown. This is developed further in the Requirement Specifications.

## Optimisation

Another consideration of this project is optimisation. If one takes the clothing and textile industry as an example, one can optimise what colours, fabrics and designs are in a high demand. This is completed by eliciting and gathering “Product Usage Information (PUI)”. This is obtained from clients, experts and investors. From this data, the latest styles and what the client wants are known. (Hribernik, et al., 2019)

In this project, optimisation could be used for the general size of the tattoo, the colours of the ink used. If, for example, a lot of tattoos are small and use only use 3 or 4 colours, the system would recognise this and automatically order new colour inks to the artist. On that note, the artist could input how much ink would be used for a particular size and machine learning could be used to order and maintain stock with the artist.

The same would also go for piercings, if a certain ring or stud is used on a regular basis, these could be automatically ordered. The likes of stock control for piercers could also be controlled here. This will be developed with the Requirement Specifications.

.

# References

Anoki, 2018. *Inked-Up.com.* [Online]   
Available at: http://inked-up.com/  
[Accessed 3 December 2018].

Betzing, J. H., Hoang, A.-Q. M. & Becker, J., 2018. *In-store Technologies in the Retail Servicescape,* Münster: University of Muenster.

Hribernik, K., Arabsolgar, D., Canepa, A. & Thoben, K.-D., 2019. Applying Product Usage Information to Optimise the Product Lifecycle in the Clothing and Textiles Industry. In: *Product Lifecycle Management (Volume 4): The Case Studies.* Basel: Spinger, Cham, pp. 73-95.

Inked-Up.com, 2019. *Inked-Up.com.* [Online]   
Available at: http://inked-up.com  
[Accessed 18 October 2019].

Kim, S. & Um, N.-H., 2016. RECOGNITION IN SOCIAL MEDIA FOR SUPPORTING A CAUSE: INVOLVEMENT AND SELF-EFFICACY AS MODERATORS. In: *Social Behavior & Personality: an international journal.* Seoul: Hongik University, p. 1867.

Krishna, R., 2018. *TikTok users are complaining that in-app harassment and a general wave of dislike toward them are resulting in online bullying..* [Online]   
Available at: https://www.buzzfeednews.com/article/krishrach/tik-tok-users-bullying-abuse-complaints

[Accessed 3 December 2018].

McFarland, M., 2018. *I spent 53 minutes in Amazon Go and saw the future of retail.* [Online]   
Available at: https://edition.cnn.com/2018/10/03/tech/amazon-go/index.html  
[Accessed 16 October 2019].

Tatú, J., 2018. *Social Modifications,* Dublin: National College of Ireland.

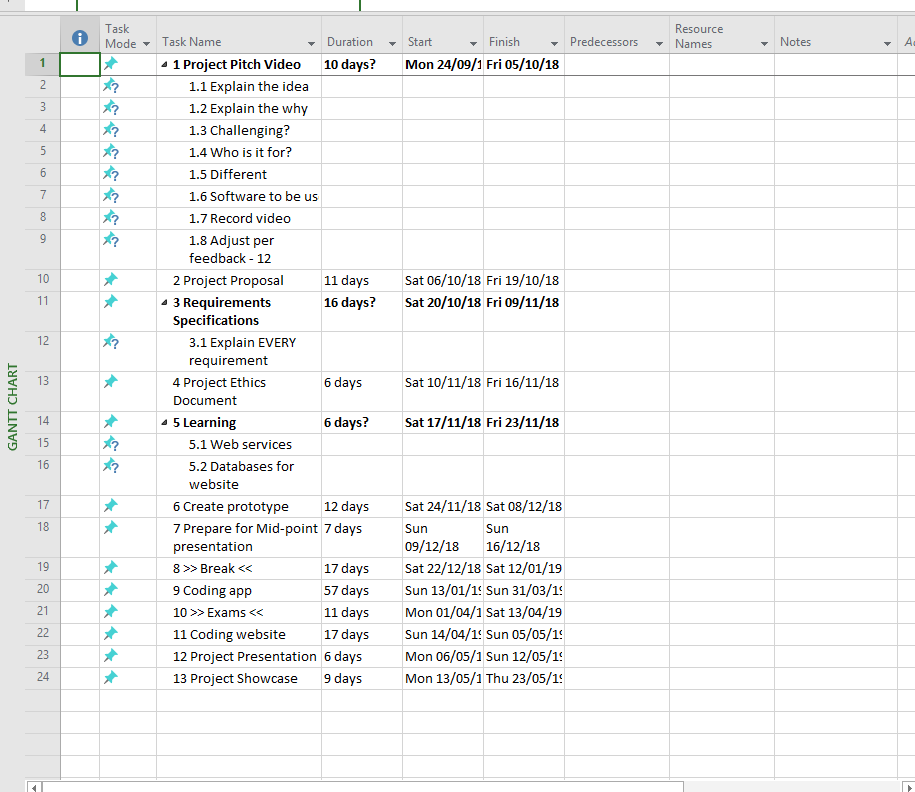
Tenbarge, K., 2018. *Can Tik Tok Be the Next Vine, or Are Seconds-Long Streaming Apps Over?.* [Online]   
Available at: https://www.inverse.com/article/47817-what-is-tik-tok-the-app-flooding-social-media-with-ads-and-partnerships

[Accessed 3 December 2018].

Webner, F., 2019. Streaming Analytics—Real-Time Customer Satisfaction in Brick-and-Mortar Retailing. In: *Cybernetics and Automation Control Theory Methods in Intelligent Algorithms.* Essen: Springer, pp. 50-59.

# Appendix

## Project Plan



1. A **brick-and-mortar store** is a traditional store that you find (e.g.) in your local shopping centre. [↑](#footnote-ref-1)