A picture containing computer

Description automatically generated

Team members

*Dongseok Cho*

*Jacky Huynh*

*Ashley LaBuick*

Wash Me App Proposal

# Overview

## From Part 1

Over the course of this proposal we will review the first part of the design process, the discovery stage, where we focus on understanding the problem. The discovery stage as defined by the Nielsen Norman Group is, “a preliminary phase in the UX-design process that involves researching the problem space, framing the problem to be solved, and gathering enough evidence and initial direction on what to do next.“ We will focus on understanding the problem space in its fullest capacity, which will inhibit us from moving too quickly into the solution space.

User research or UX research as defined by CareerFoundry is, “is the systematic investigation of your users.” The purpose of UX research is to put the current system into context, and help us to understand the problem we are trying to solve and the context in which users will be using our product. The information gathered here will allow us to understand the underlying problem. Without user research we would be building systems based on assumptions, which would lead to ineffective and inefficient solutions with inherent usability issues.

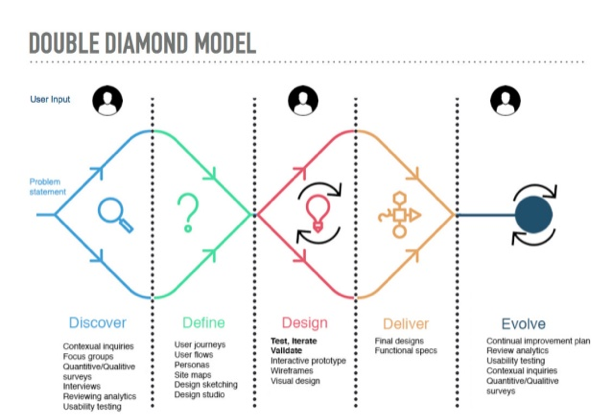
The methods we have chosen to flesh out the problem state are qualitative research methods. Qualitative research methods involve, “collecting and analyzing non-numerical data to understand concepts, opinions, or experiences.” We are using qualitative research methods to allow us to further understand the problem and help us to identify possible solutions. These methods also allow us to create a benchmark for the UX experience, with which we can compare our system to at a later date. The qualitative methods we are utilizing include: an initial semi-structured interview with Rebekah, followed by two follow-up semi-structured interviews with an ad hoc style question period. The information gathered from these interviews was utilized to help understand the current state of the problem from a users point of view and how this fundamentally relates to the UX hierarchy of needs and Nielsen's heuristics. First we gather data, then we will synthesize that data in order to create users, and further understand their goals. Effective user research will save time and help us better understand our users to build a superior system.

We then move into the more user focused section of this proposal where we will continue to hone in user goals and needs from the synthesis of our gathered data. This section will give a more indepth look at the intended users of the system, why they would use this system and what would satisfy their needs. We will continue to use a user-centred approach throughout this proposal to help capture data from the users point of view and more thoroughly understand the problem place from multiple angles. The UX goals we have written will allow users to achieve Morville’s Honeycomb, which speaks to emotions or attitudes users should have when using a particular product, or service. Focusing on these aspects from a user-centered perspective will allow us to reach beyond the base functional components of the UX pyramid and help reach the higher levels which make systems convenient, pleasurable and meaningful.

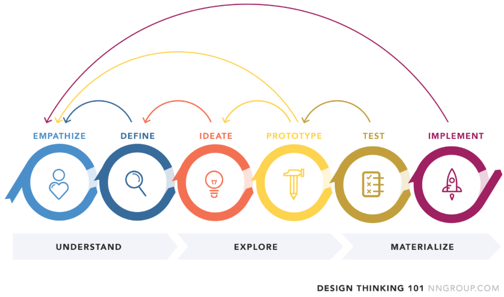
In the next section we will outline the Value Proposition Grid utilizing the previous research. The Value Proposition Grid will be outlined from both the Owner/Operator perspective as well as the Customer perspective. This grid is essential to allow us to create effective product-market fit. The Value Proposition Grid has three components: must haves/needs, performance benefits, and delighters. The must haves/needs correlate to a base threshold of the UX pyramid in terms of basic functionality. Must haves do not add value to the product, but not having them does create a negative user experience. Performance benefits are attributes which add value to the product, but not on an emotional level. With performance benefits the more we are able to meet these needs the more the customer is satisfied. Delighter benefits create a lot of customer value, because they add value at an emotional level, which correlate to the upper levels of the UX pyramid. This grid will allow us to understand where we can exceed and where we will cut back. It will help us to determine our unique differentiators, the performance benefits where we will excel and our unique delighters. To optimize the Value Proposition Grid, as outlined by Dan Olsen, we will not compete on must haves, we will have at least one superior performance benefit, and a unique delighter. As he demonstrated in his presentation, “having an outperforming benefit and a delighter is a powerful combination,” and we will seek to use this formulation to our advantage.

## Overview Part 3

As we leave the define phase, we enter the design phase of the double diamond model, which is at the center of the interaction design process, as seen in the below image. The design phase deals with the look and behaviour stage of the double–diamond model. Solutions or concepts are created, prototyped, tested, and iterated.



Before outlining the important aspects of the design phase, we will once again summarize the background surrounding the problem, and the problem itself, as a product of the define phase. The observations and information gathered in these phases shaped the solutions created in the design phase, so it is important to demonstrate the clear groundwork for our solutions. This phase focuses more on the look and feel of the product or system we are developing. It is in the phase that we will deliver wireframes, mockups and prototypes, which will allow us to gather feedback on the product we have created. In this phase, the solution we have developed will start to look and feel real, and we will be able to create prototypes that simulate the final product.   
  
The design phase moves with clear progression as each section builds on the section previous. We start first with a task flow diagram. A task flow is a series of steps a user takes to achieve a specific goal. These series of steps will be modeled through a task flow diagram, which allows us to visually communicate the steps we intend the user to take in order to complete a certain task or goal. The task flow diagram represents the complete process which a specific user may take to complete the goal, it includes a beginning, middle and end. Task flows are essential to understanding a user's behaviour as they complete a specific task.

Task flow diagrams are then used to create a type of low fidelity prototype, known as a wireframe. According to CareerFoundry, a wireframe is a basic, low-fidelity representation of the initial product concept, containing the essential elements that would feature on a webpage or app. Wireframes provide a clear outline of the page structure, layout, information architecture, and overall direction. Wireframes allow us to gather all the essential building blocks of the various screens without getting caught up in the small aesthetic details. As implied by the name ‘Wireframe’, it is just that, an outline, like a blueprint for a building, it outlines everything we will need to create the system from the ground up. Wireframes do not show complex interactions, they are generally created in black and white and depict the main chunks of content, layout, and structure.   


Once completed, the wireframes are then used to create hi fidelity prototypes. A hi fidelity prototype is a representation of the final product which is meant to simulate user interaction. A prototype looks very similar to the final product. A hi fidelity prototype is a very useful tool that allows us to test our solution with users, and get feedback on the design of the system and allow us to test and reiterate. This type of testing and iteration is essential to the design thinking process and reinforces its nonlinear nature. Feedback gathered from testing prototypes may turn us back to the discovery part of the double diamond model, demonstrating that we need to do additional research into our users.

It is important to note that hi fidelity prototypes represent an interface that is not connected to the backend development. They look and feel like the real thing, but do not function as such. This type of prototyping allows us to go through various rounds of testing, user feedback and iterations. This process saves time and money when we turn to the development of the backend.

After the development of the hi fidelity prototype we will then use a screencast walk through to demonstrate the intention behind how the user should carry out the specific goal, which the prototype is centered around.

In the Design patterns section of this proposal we will outline a few of the design patterns we implemented in the hi fidelity prototype. Design patterns are a proven solution to a common problem within a specified context. We utilize design patterns in our process because they speed up the development process and allow us to implement known, tested solutions, that meet recurring needs of users. With each pattern used in our screen design, we will outline the: name, description, problem addressed, solution and link to an example of that pattern. Implementing design patterns in our process allows us to reap the benefits of tried, tested and true solutions to recurring problems, allowing us to save time and maintain high levels of consistency in our product. Design patterns lead to a more conclusive user experience because they implement, tested, proven and accepted design solutions.

Next we will look at the Design Systems section, which outlines visual systems in place in order to maintain consistency across all components of the system. A design system is a single documented source for ensuring consistency across all aspects of design and development. It may include: a collection of documents, articles, examples, code snippets, screenshots, design guidelines, components, philosophies and other digital assets for a product design company. Design systems are essential because they provide rules and guidelines for the codes, visual elements and interactions. These systems help ensure brand consistency. In our Design Systems section we will include usage and guidelines for: brand colours, fonts, buttons, navigation and logo. These systems will be seen throughout our hi fidelity prototype demonstrating consistency and clarity across the system.

# Background

We find out information about the current system Rebekah has by using data collection (identifying the problem of a system by gathering information from its users). For this problem, we used semi-structured interviews (a style of interview where the interviewer has some prepared questions and has the flexibility to go off-script as required). The current system used by Rebekah, the owner, is one without automation. The owner first has to advertise by putting up flyers around Gibbons and relies on them, as well as word of mouth, for the car washing service to be known. The core of the system has the owner receive inquiries and requests for the car washing service through phone calls, and then if an appointment is made, the owner records the information by hand and transfers the information to an Excel spreadsheet where they can organize and sort appointments accordingly. However, the user experience (experience that the user has when interacting with a product) the owner is experiencing is a negative one because it causes her stress to maintain an extensive, manual process while balancing other aspects of her life such as her part time job.

One problem that the current system solves is that it allows customers to contact Rebekah and book a car wash. A record of the customer's information is also kept track of, although information may be lost in the manual process. It also has low advertising costs because Rebekah is spreading word by putting up flyers by herself. She also tries to leave a good impression on her current customers, making word of mouth a big part of her advertising. The current system also provides some options to the customer by having basic and deluxe car washing packages. Overall, the system solves the functional needs to act as a serviceable product, but does have issues with its reliability.

The current system is designed for the owner herself because it allows her to be flexible in the context of juggling her part-time job. By having the system done manually, Rebekah can organize customer information and appointments the way that works for her. As stated before, loss of information can happen frequently because of the manual process which may result in the wrong customer address or wrong time of appointment. It can also be inconvenient because Rebekah may be at her part-time job when a phone call is made to her. Therefore, a new system should be in place to fix these issues, as well as consider other potential users (any potential person that uses the system).

The goal state of the new system is some kind of digital solution that can track and manage the information of the clients, as well as its bookings to relieve some responsibility from the owner. The digital solution should be one that can grow with the business because one of Rebekah’s goals is to expand the business to surrounding areas, including Edmonton. This way, the owner can focus on keeping the customers happy by providing a quality service and be free from reliability issues.

Potential users of the new system will be the owner, Rebekah, to organize and manage the business. Employees are also a potential user to help provide services when the car washing service grows large enough that a single person can no longer satisfy its needs. Employees are also needed whenever Rebekah is unavailable for a booking. Another potential user is customers in not only Gibbons, but also in surrounding areas such as Edmonton. As the service continues to grow, customers will request for its service in its neighboring areas. An administrator may be a user if the system moves to a digital solution that needs constant maintenance.

# Problem

The core problem of the current system is its reliability. The problem with its reliability is that it is a manual process, and therefore, information collected may not be accurate or complete. This makes the system prone to errors, resulting in frustration in both the owner and the client. Reliability is on the second level of the UX Hierarchy of Needs which makes the current system poor because the reliability problem will extend to and affect the upper levels of the hierarchy as well, including the system’s usability, convenience, pleasurableness, and meaningfulness.

An example of a symptom from the owner’s perspective is a missed call. Rebekah could be at work, unable to pick up the call, and miss a potential customer. By missing the call, Rebekah’s goal of satisfying customers is not met. It may be a bigger problem if word of mouth says that the service Rebekah gives is unreliable, losing even more potential customers. Another symptom stemming from the system’s manual process is that information gathered from a client can be lacking or lost before recorded. Rebekah collects information through phone calls and occasionally, she can forget to ask about certain details such as the type of car, license plate, and location. This can be troublesome for both the client and the owner because Rebekah would need to call back the client, causing Rebekah to feel unreasonable stress and wasting time from both parties. Information can be lost because the data is recorded on whatever is available when the call takes place, such as the hand or a napkin. The ink can be smudged, or the napkin can be accidentally tossed away. Once again, Rebekah would need to call back to recollect the lost data by calling the client back, wasting time and lowering customer satisfaction.

A symptom from the customer’s perspective is regarding the inconvenience of the service, resulting from the unreliable nature of the current system. The manual process of booking an appointment can lead to conflicts to previous bookings, and therefore a rescheduling of the car washing service must be done. The inconvenience of the service leads to an unpleasant experience that the customer should not experience. The customers may refuse to do the rescheduling because they expect good customer service and did not get it. Another symptom of the problem from the customer’s perspective is inconvenience of contacting Rebekah. Currently, there is only one method of contacting Rebekah and that is by calling her on her cell phone. However, if Rebekah is busy and does not pick up her phone, there is no alternative for the customer to contact her. The customer has to hope that the call is returned without any way of knowing Rebekah noticed the call. That can make the customer feel anxious and uneasy about the service because they want to be sure they are heard but are met with uncertainty.

Once again, the issue goes back to the unreliability of the current system. Currently, there are too many symptoms coming from the manual process of the system. Therefore, a new, digital system with an efficient, reliable, and automatic processing is anticipated for its users to be happy.

One sample solution is to have a website like SpeedCarWash.com in the picture below (Figure 1). It allows the customer to fill out all the necessary information to book an appointment in an automated system so there will not be problems with reliability. The information will be accurate and stored safely. The customer will have more payment options, as well as an alternative to contact the owner through the website as well.

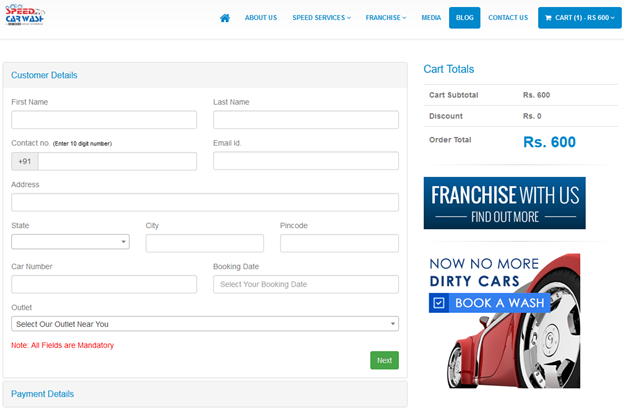
 

Figure 1 Figure 2

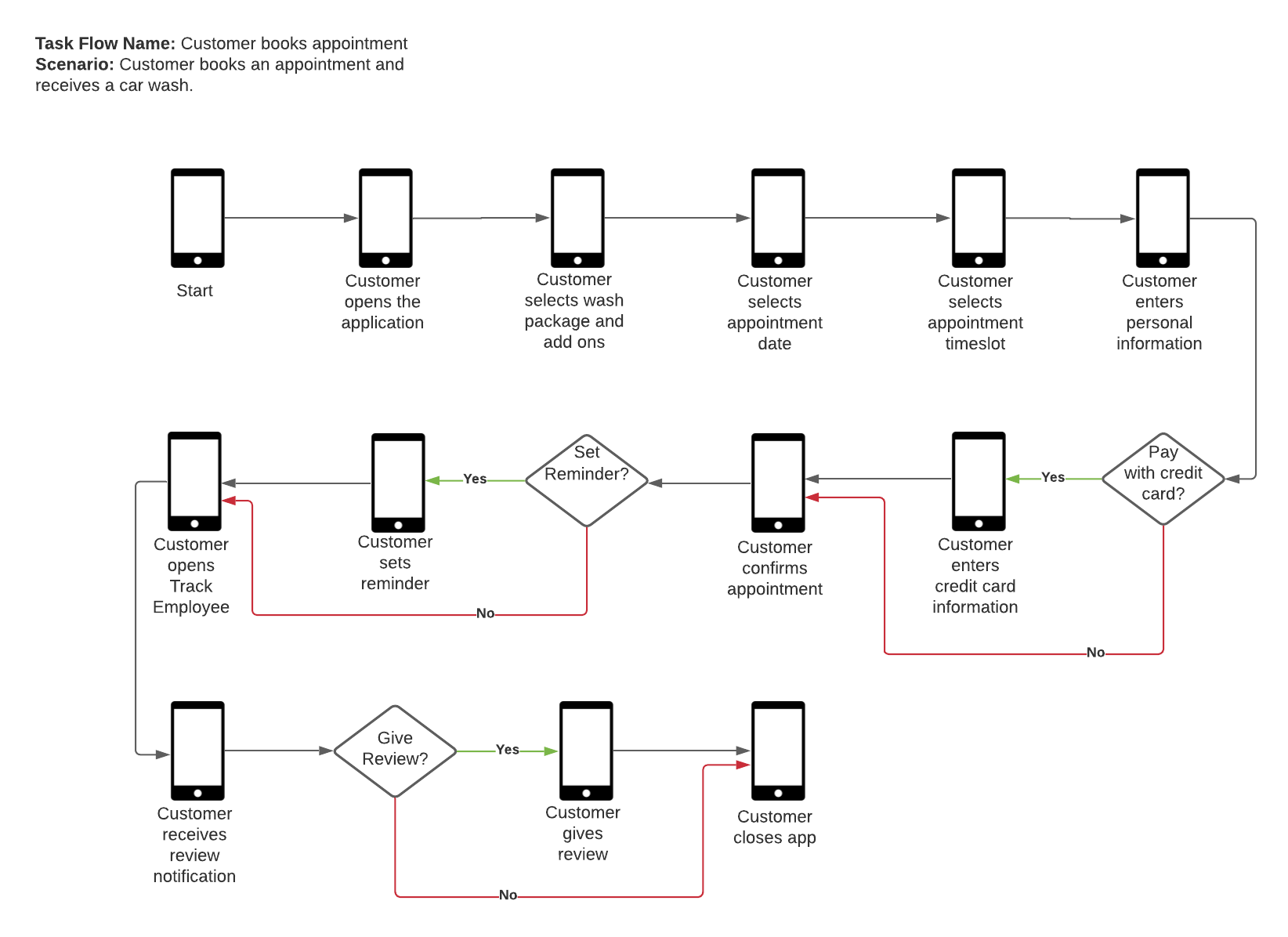
Another potential solution that other products have is a mobile application that customers can book a car wash with such as MobileWash (Figure 2). The application not only books an appointment for the customer, but tracks the when and where the car washer will arrive to the customer’s home. By allowing the customer to track the car washer, it provides a sense of security that someone is coming to provide the service they were promised and gives a good estimate of the time so they can be prepared when the car washer arrives.



# Task Flow

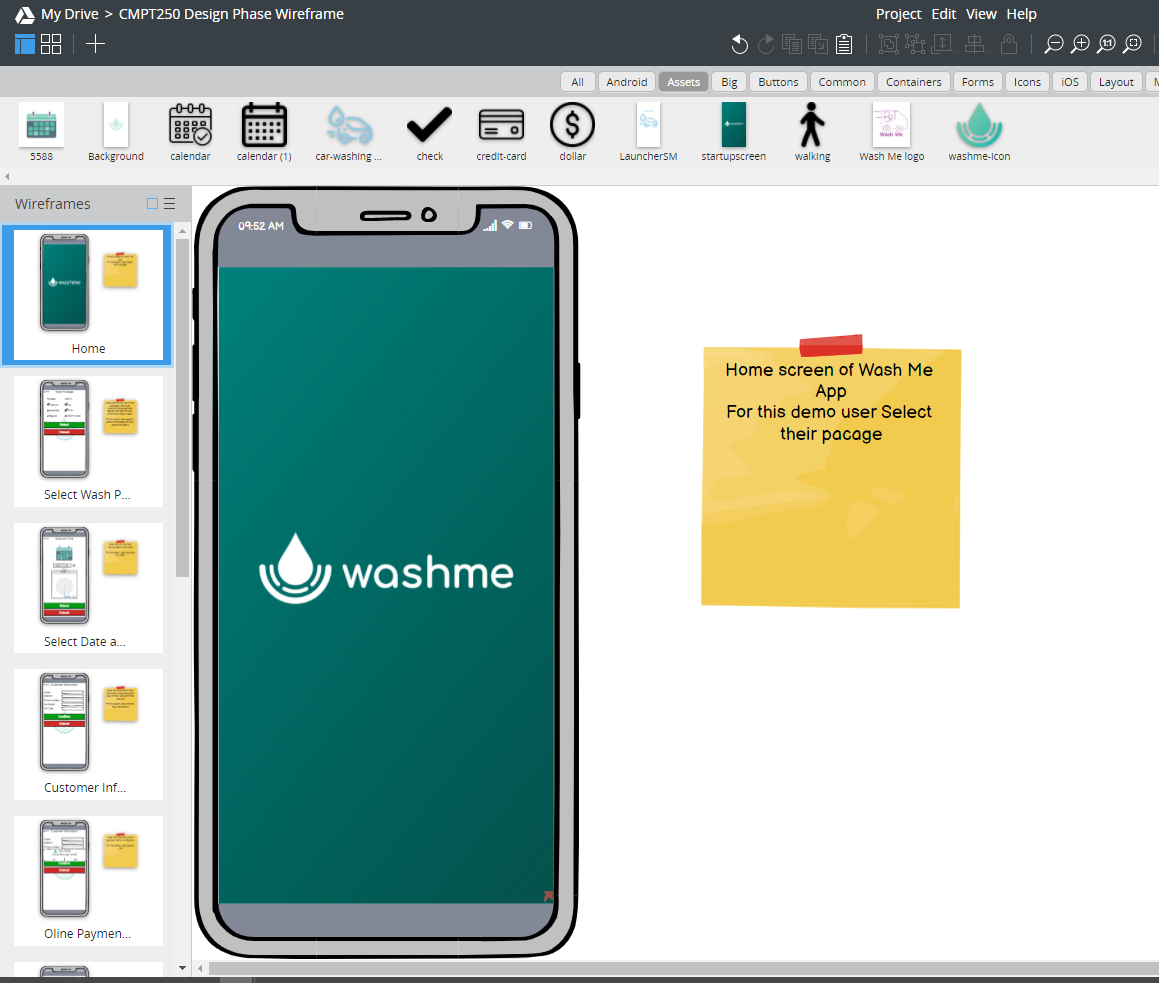
The next section of our proposal moves us to the design phase in the double diamond model (a design model that allows the team to first identify the problem, and then focus on designing potential solutions for the problem). In the design phase, we start making design deliverables such as wireframes and prototypes to model the actual product. However, to start doing that we must first create a task flow diagram to clearly lay out what steps are done by the user to achieve a specific goal. Task flows diagrams are based on use case flows that shows a specific user completing a specific goal by following a sequence of steps. A task flow is represented by containers that display what the user does at each step which may include some decision points that could put the user on an alternative path, but still reaching the same goal or post condition. Task flows are essential to the design phase because it illustrates the overall structure to achieve a certain goal for the user. They clearly label the trigger, which is a system-based event that sets the task flow in motion. Following the trigger, a task flow identifies the intermediate steps to achieve the post condition. An example can be seen in the image below.

By having the overall structure laid out visually, the design team can then start making design deliverables by adding details and screens to navigate between each step. In the first, we will use Balsamiq to create a wireframe that is quick, cheap, and easy to make. The wireframe will allow us to test our ideas by making them tangible, and eventually lead to an interactable and testable hi fidelity prototype.



# Balsamiq wireframe

A wireframe is a low-fidelity way to show the design. It is a graphical representation of an app or website that allows designers to understand the most essential elements and the content easily. This wireframe demonstrated the task flow of a customer making an appointment. This cleared the actions described in words and gives the designer an example of structure, layout and overall direction of the flows. However the wireframe has limitations, such as it cannot show the complex interactions, limited usefulness for usability and doesn’t show the visual design. Therefore, the wireframe is good for initial setups.

**

# Hi fidelity Prototype

A hi fidelity prototype is like a wireframe in that it shows the overall structure of the product, but is much more interactive and simulates the finished product. Basically, it is the wireframe but upgraded from being low fidelity to hi fidelity. The hi fidelity prototype uses graphics similar to a finished product and necessary interactions to allow the user to achieve their goal. Stakeholders can then give feedback before settling on the final product. There are multiple ideas the prototype can help us test by letting the user test it. One example is the look and feel of the product to see if it matches the user’s mental model of the product’s design. UX goals and usability criteria are other things that the hi fidelity prototype tests because it has many of the final interactions to be in the product. Prototypes are also useful for sharing with the client to see if it matches their expectations. If there is anything that displeases the user or client, the design team can still edit and change the prototype at a reasonable time and cost.

Below is the link to the hi fidelity prototype of the Wash Me app. It follows a scenario where the customer’s goal is to book an appointment, track an employee, and leave a review after receiving the service.

**Link to Prototype:** <https://indigodesigned.com/share/xbab5wndhhfa>

# Screencast walkthrough

A screencast is a digital video recording that captures actions taking place on a screen. The screencast contains voice-over narration, which helps demonstrate how to use specific operating systems, software applications or website features. The purpose of the screencast is to show the users what kind of actions are required and what kinds of alternative solutions are possible on an operating system. And also the screencast can help filing reports of bugs and look for potentially unclear written explanations.

**Link to Screencast**: <https://spark.adobe.com/video/LbWKCCpXq6dOS>

# Design Patterns

**Name: Scrolling date picker / Scrolling time picker**  
**Description**: Allows the user to scroll through the available dates and times on the screen and input the desired date and time.  
**Problem Addressed**: The user wants to request a booking for a certain date and time.   
**Solution**: Scrolling through the available date and time slots allows the user to communicate the desired date and time which they wish to request for a booking.  
**Usage:** We use this solution in our prototype on the screen where the user is requesting a booking date and a booking time for their services.   
**Link to Example**: <https://developer.apple.com/documentation/uikit/uidatepicker>  
  
**Name: Selection Controls**  
**Description**: Allows the user to input data into the system, by making choices or selections from options which are predefined for them. The user can communicate their choices by selecting from the predefined options provided for them.   
**Problem Addressed**: The user wants to communicate desired choices to the system, in a consistent format.   
**Solution**: Utilizing selection controls allows the user to easily view all the options available from the system, and communicate which ones they desire, in a consistent format.   
**Usage:** We use this solution in our prototype when the user is requesting a booking and selecting the type of service they wish to book.  
**Link to Example**: <https://material.io/components/selection-controls#usage>  
  
**Name: Dialogs**  
**Description**: A modal window that appears in front of the app, this allows the system to communicate critical information to the user or force the user to make a decision that the system requires.   
**Problem Addressed**: The system needs to communicate a message to the user or get the user to make a certain decision before continuing to interact with the system.   
**Solution**: Interrupting the users flow with a modal window forces them to read/interact with the necessary information, or make a decision before continuing to interact with the system.   
**Usage:** We use this solution in our prototype when the user is asked if they would like to pay online, it forces them to make a decision about payment before the booking is approved. We also use it when we ask the user for a rating/review, this forces the user to make a decision about reviewing their experience and calls attention to this choice.   
**Link to Example**: <https://material.io/components/dialogs>

**Name: Bottom Navigation**  
**Description**: A bar at the bottom of the screen that displays 3-5 options for the user to select from. Each option is represented with a label, icon or both. When the user makes a selection they are taken to the top level screen of that item.   
**Problem Addressed**: The user wants to be able to easily access certains main areas of the system.   
**Solution**: Bottom navigation allows the user to consistently access main areas of the system. These options remain visible at all times when the system is running in order to make navigation clear, consistent, simple and easy to use. Locating the navigation at the bottom of the screen makes it easily accessible to users on handheld devices, so it can be utilized while on the go.   
**Usage:** We use this solution in our prototype throughout all the screens, except the launch screen. The navigation along the bottom allows easy quick access for the user to the main screens utilized in the application. These options are apparent and visible at all times so they can be easily and quickly accessed.   
**Link to Example**: <https://www.bbc.co.uk/gel/guidelines/bottom-navigation>

# 

# 

# Design Systems section

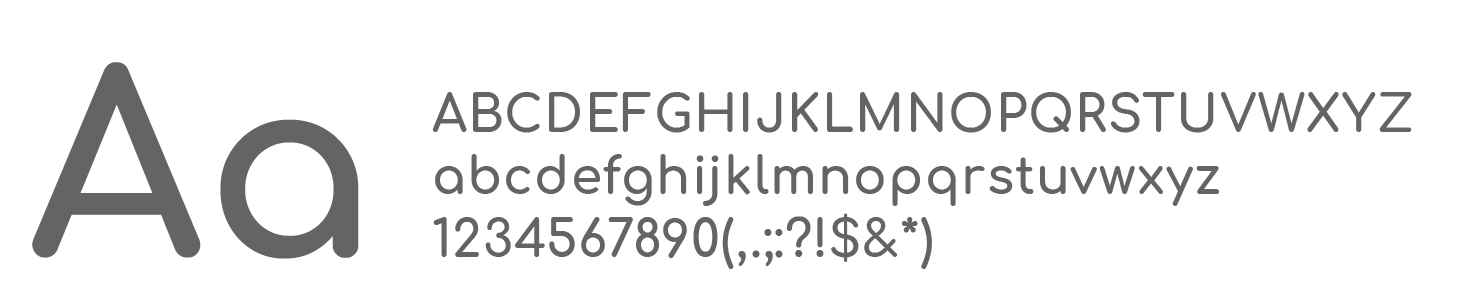
**Colors**

Wash me uses a clean modern colour palette, we utilize gradient colour swatches to show movement and flow through our brand and brand icons.

# 

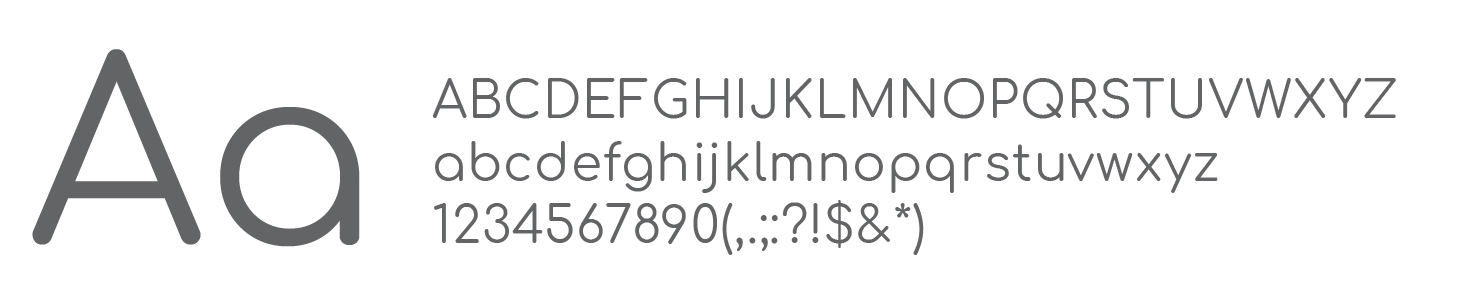
**Fonts**

Our main font was selected for its simplicity and clean look. The type choice reflects the approachable, feel of the brand, as it is rooted in community values. Wherever possible, it should be used for all forms of content, ensuring a consistent look and feel across all branding assets.

**Headings Text and Main Logo Font**

**Comfortaa**

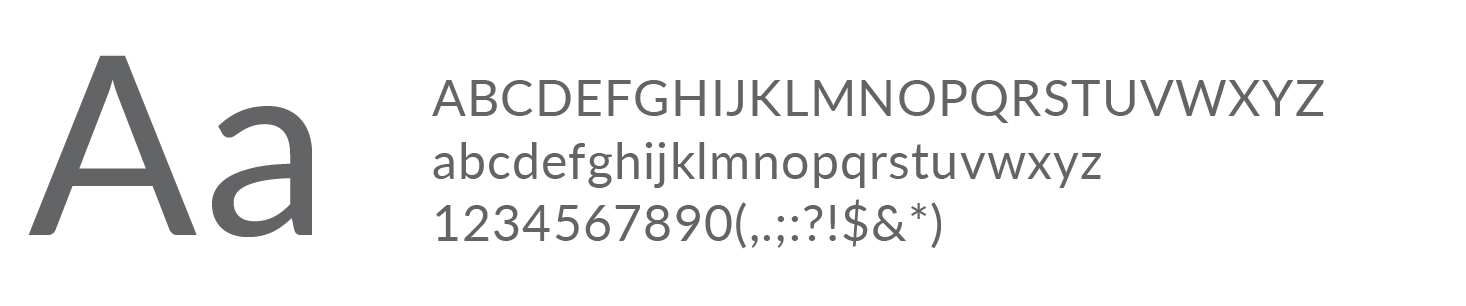
**Font Weight: 700  
Style: Normal**



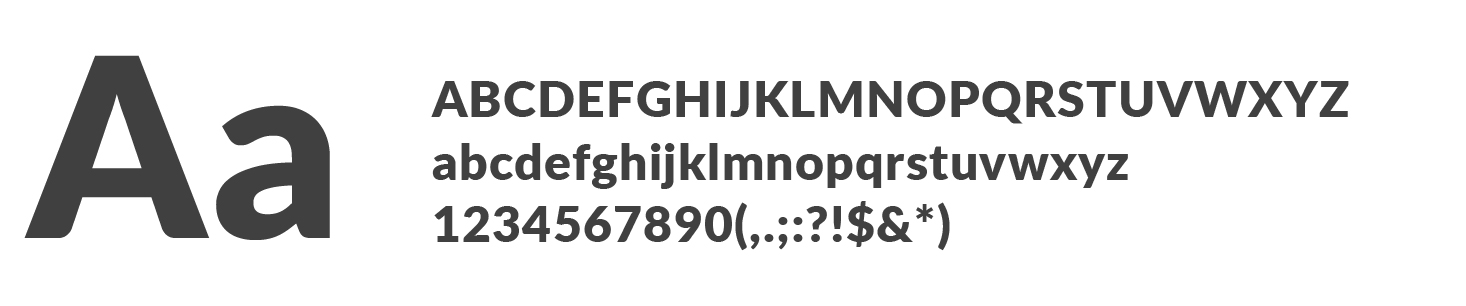
Comfortaa

Font Weight: 400  
Style: Normal

**Body Copy & Paragraph Text**



Lato

Font Weight: 400  
Style: Normal

Lato

Font Weight: 900  
Style: Normal

**Usage**

Google typefaces were chosen for accessibility to the public, They are available for free download at the below links. The css snippets are also provided below.

**Comfortaa**<https://fonts.google.com/specimen/Comfortaa>  
font-family: 'Comfortaa', cursive;

**Roboto**<https://fonts.google.com/specimen/Lato>  
font-family: Lato, sans-serif;

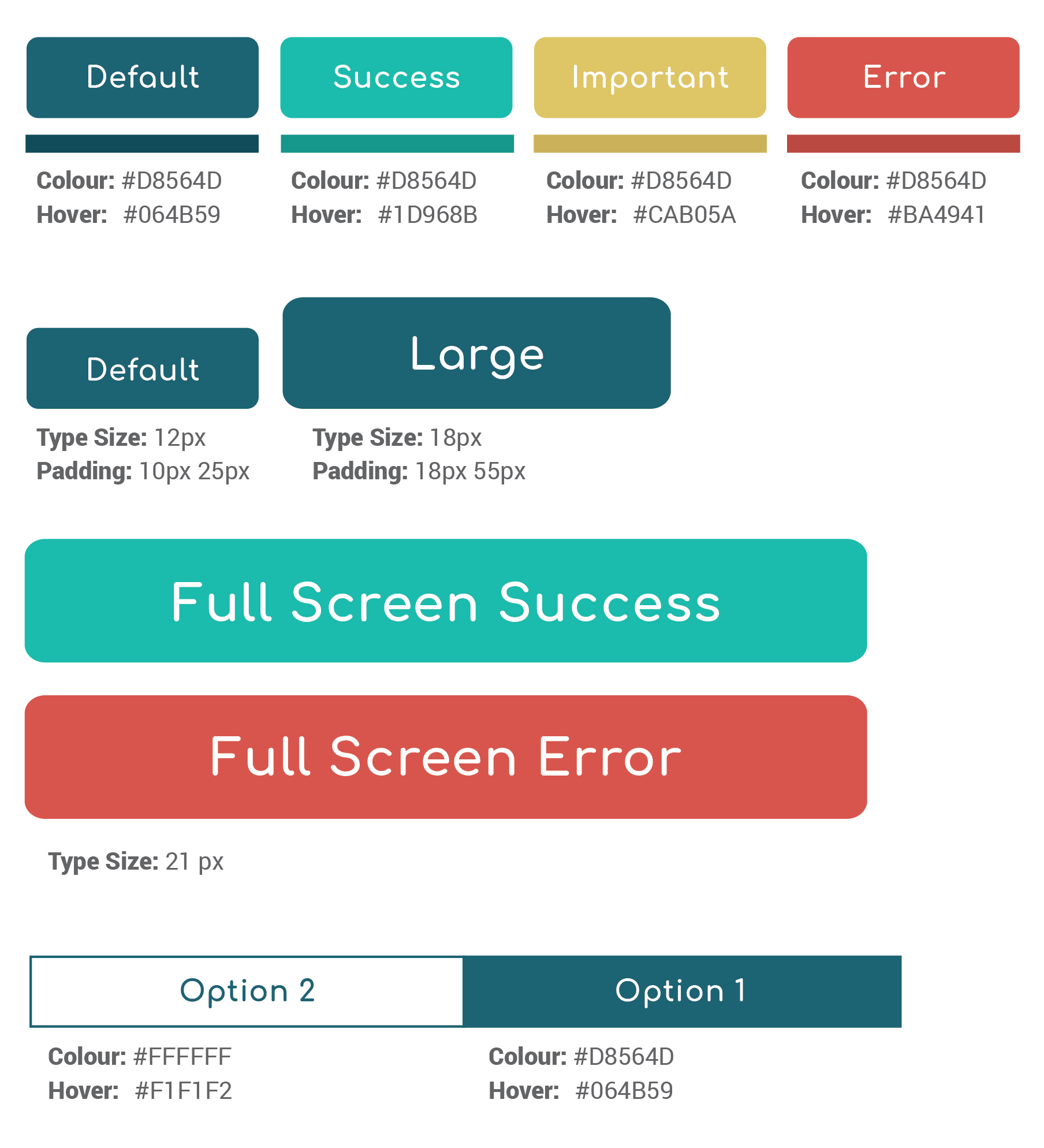
**Type Scale and Weights**

|  |  |  |
| --- | --- | --- |
| **Mobile Header**  **Comfortaa Font Weight: 700 Style: Normal**  **Size: 16 px  Colour: #008481** | **H1**  **Comfortaa Font Weight: 700 Style: Normal**  **Size: 26 px  Colour: #29BBAC** | **H2**  **Comfortaa Font Weight: 400 Style: Normal**  **Size: 21 px  Colour: #166273** |

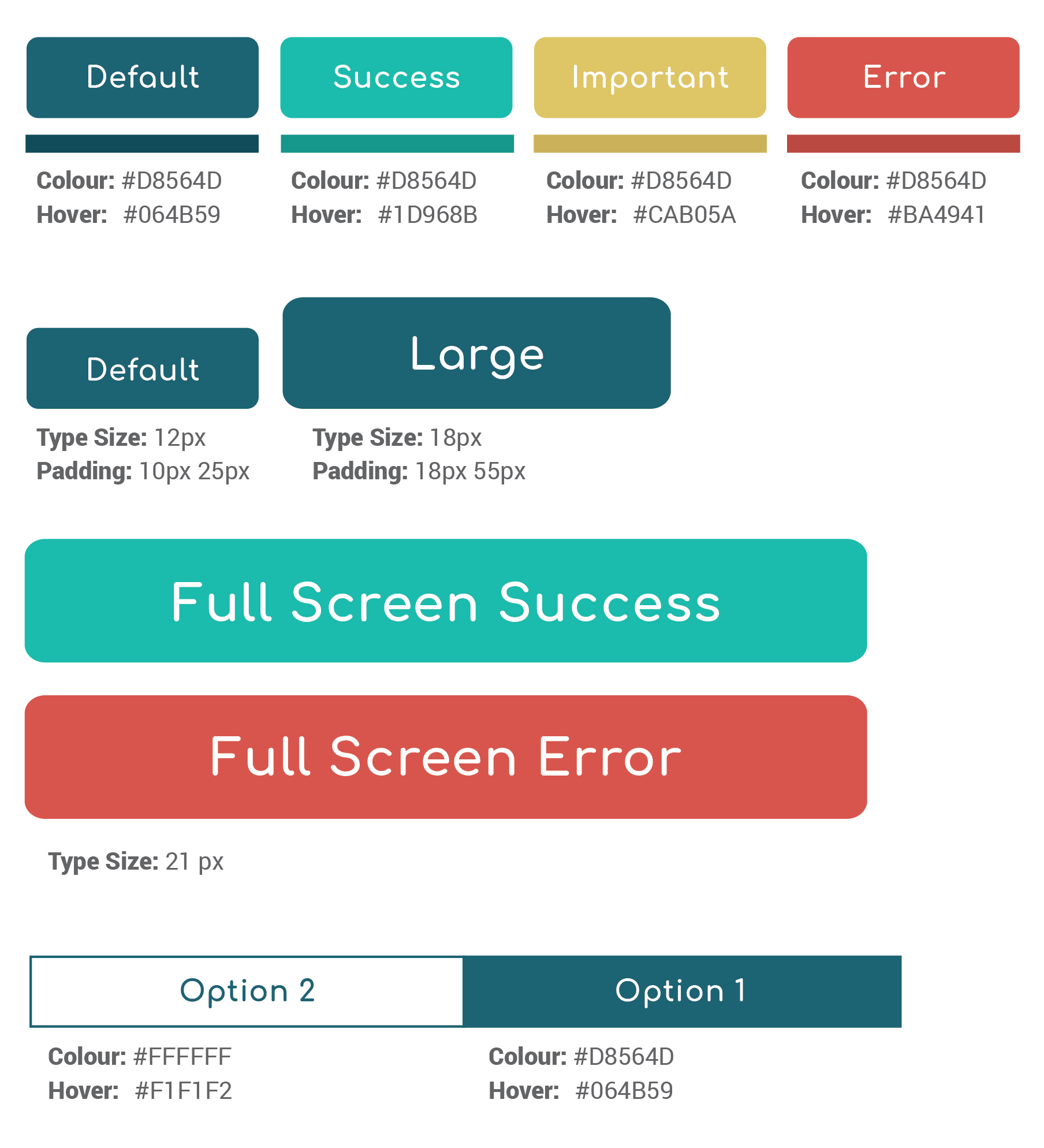
|  |  |  |
| --- | --- | --- |
| **H3**  **Comfortaa Font Weight: 700 Style: Normal**  **Size: 18 px  Colour: #29BBAC** | Body Type  Lato Font Weight: 400 Style: Normal  Size: 12 px  Colour: #3F3F3F | Emphasis  Lato Font Weight: 900 Style: Normal  Size: 12 px  Colour: #3F3F3F |

**Buttons**

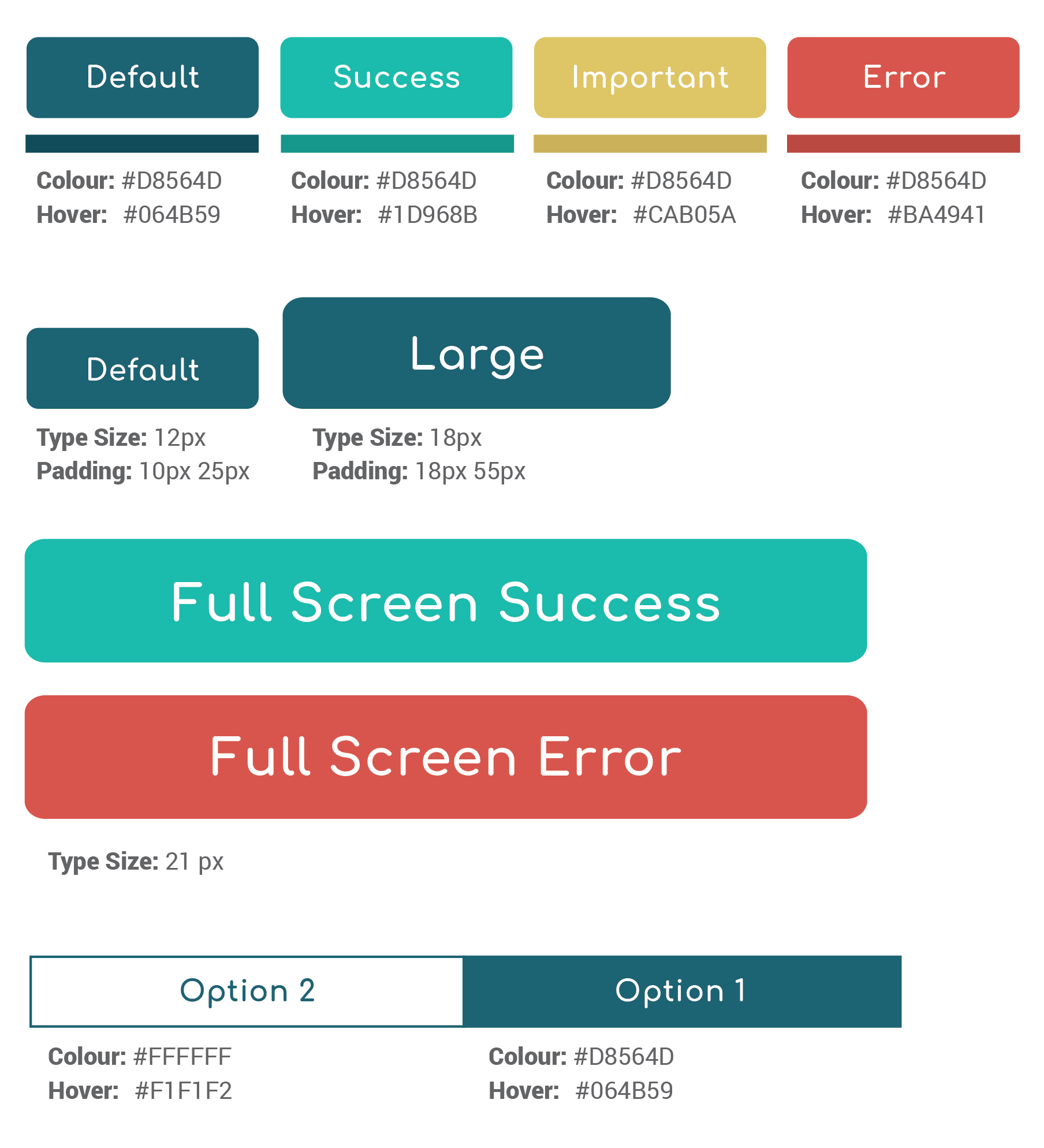
**Primary Buttons**

****

**Button Size**

****

**Choice Buttons**

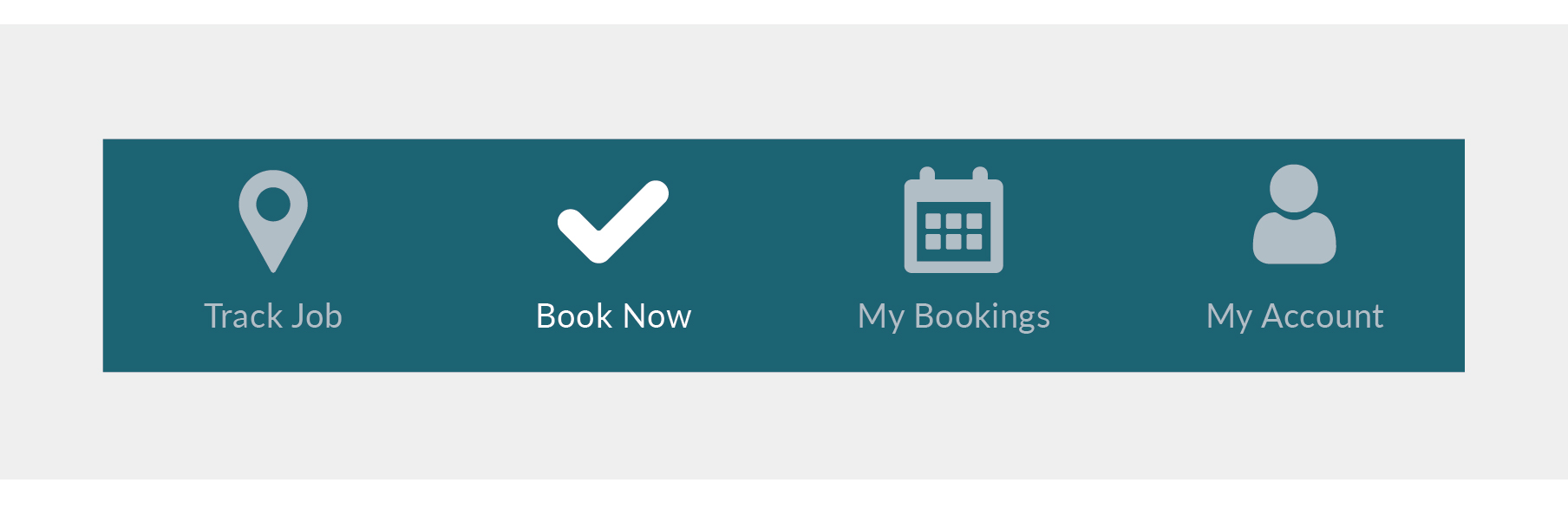
****

**Navigation**

**Primary Navigation - Bottom Navigation**

The primary navigation form is bottom navigation. It is used to jump between main sections of the application. Each option sends the user to the top level navigation of each section.

|  |  |
| --- | --- |
| **Default State:** Type Size: 10px Font: Lato Font weight: 400 Menu colour: #FFFFFF Transparency: 70% | **Active State:** Type Size: 10px Font: Lato Font weight: 400 Menu colour: #FFFFFF Transparency: 100% |

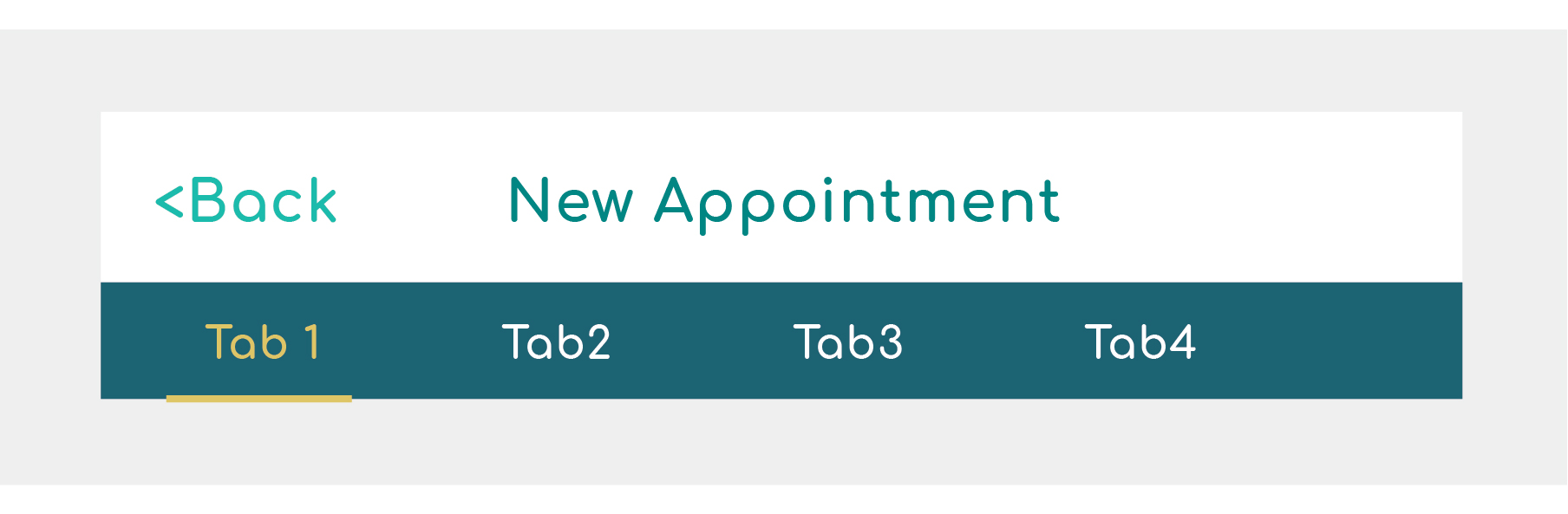


**Secondary Navigation - Tab Navigation**

Text labels should clearly and succinctly describe the content of the tab they represent. Tab content should contain a cohesive set of items that share a common characteristic.

Tab labels appear in a single row. They can use a second line if needed, with truncated text. Alternatively, you can use scrollable tabs to allow room for longer titles.

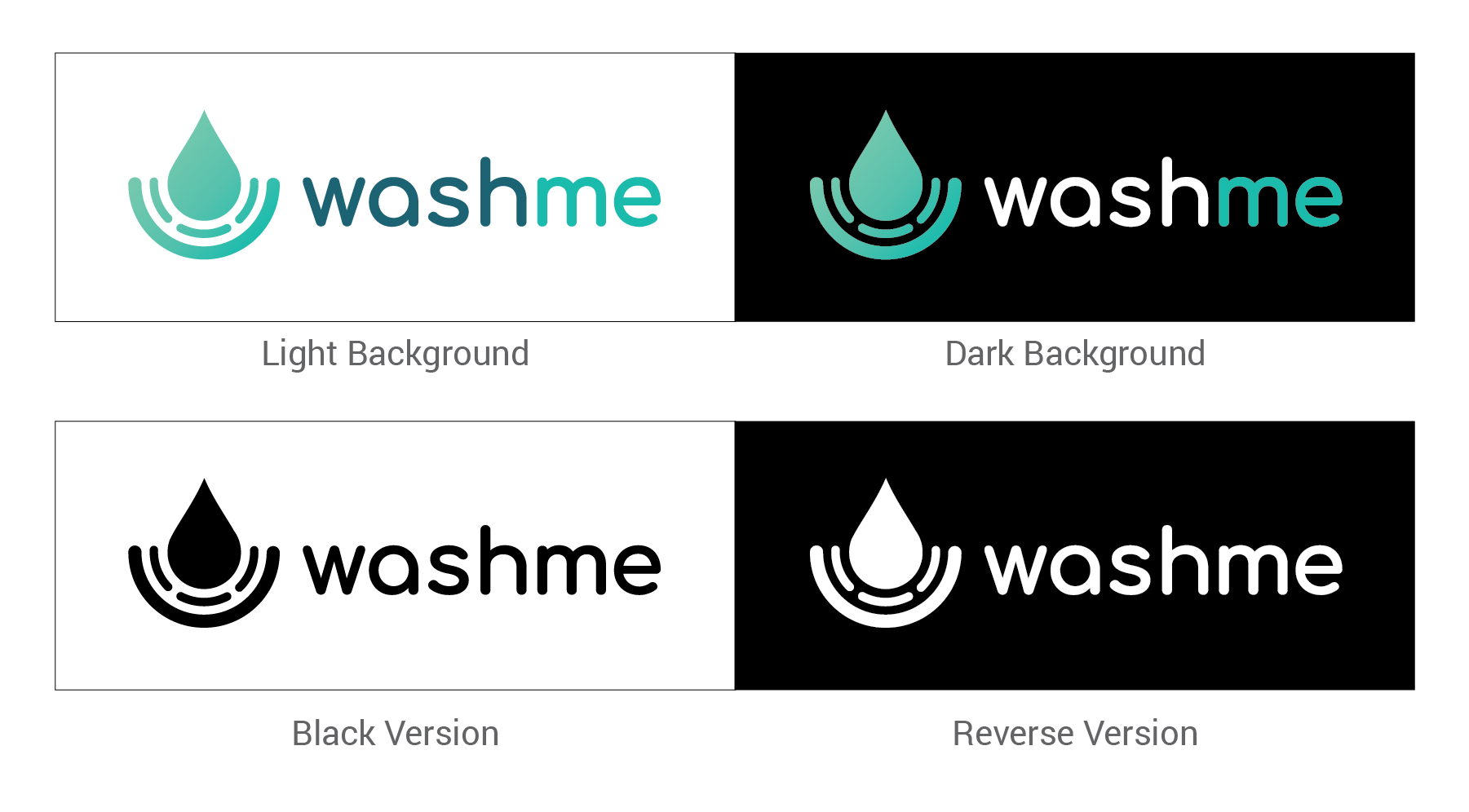
|  |  |
| --- | --- |
| **Default State:** Type Size: 12px Font: Comfortaa Font weight: 700 Colour: #FFFFFF Transparency: 100% | **Active State:** Colour: #DEC566 Border Bottom: 2px Border Bottom Colour: #DEC566 |



**Logo**

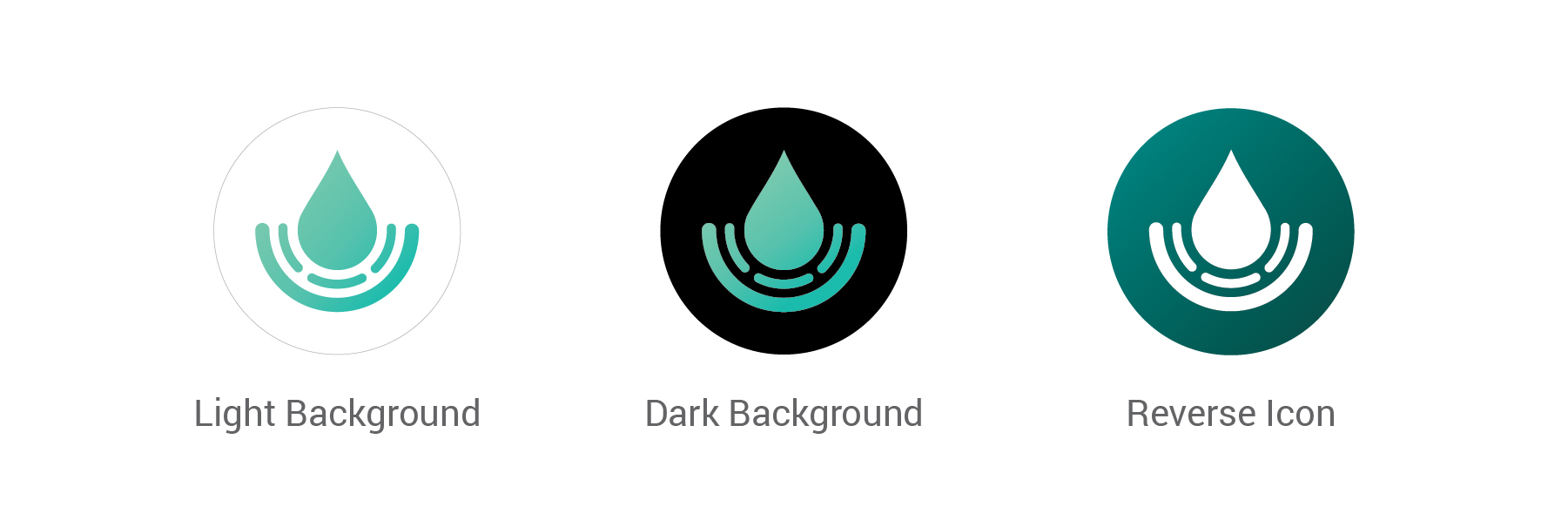
**Full Logo**

When possible, always use the logo in its entirety. Keep the spacing between the icon and type consistent, do not alter. The full colour light background is the preferred version of the logo, this should be used when possible.

****

**Icon**

Only use the icon in cases where there’s an established familiarity with the Wash Me brand and/or space is too limited to use the entire logo.

****

# Summary

The original system used by the owner is one without automation which leads to the core problem of being unreliable. Its reliability is poor because of the manual processing of information, making the system prone to errors such as lost job details and missing addresses. Reliability is on the second level of the UX Hierarchy which means it will also cause problems to the upper levels of the hierarchy as well. Some symptoms include losing customer information and losing a potential customer because an employee did not pick up the call to book their appointment.

In the first steps of the discovery phase, we identified the root of the problem by using qualitative research methods which include an initial semi-structured interview with the owner, followed by two more semi-structured interviews with an ad hoc style question period. To further enhance our understanding of the problem, we use the Value Proposition Grid to outline the must haves, performance benefits, and delighters that the system should have to achieve a clear product-market fit. Combining these methods allowed us to identify that reliability was the core problem of the current system.

In order to solve this problem and identify the system requirements, we move onto the define phase of the double diamond model. We used various methods such as persona, scenario, happy path, use case diagram, and use case text flow to find both functional and non-functional requirements. Personas allow us to understand potential users that would use our product, enabling us to make decisions based on realistic users. The second method is the scenario which helps us identify the contexts of when and where users decide to use the current system, as well as figuring out the current system’s successes and failures. The third method is the happy path which is based on the scenario with additional features. Happy paths allow us to fix and identify any missing functionalities from the scenario, highlighting all functional and non-functional requirements to easily recognize and communicate them to the rest of the team. It also helps us identify some of the steps the user has to take to achieve their goal. The fourth method is the use case diagram which helped the design team determine functional requirements at a glance and determine the associations between them. Use case diagrams also help us determine any secondary actors that the system needs to function such as Google Maps for the GPS. Finally, the last method is the use case text flow which allows the design team to identify the preconditions, trigger, and the sequence of steps to achieve a goal for a specific use case. Using a combination of these methods covers each individual method’s weakness which allows us to be confident in our identification of the requirements for the system.

We then move onto the design phase where we create deliverables such as prototypes for the final product. These prototypes are based on the requirements identified in the define phase to be a potential solution to the problem found in the discovery phase. Prototypes are made to help the stakeholders envision the final product. Stakeholders which include the development team, client, and end users then help us evaluate the prototype by providing feedback on its design and interactions. Some features in our design will fix symptoms identified from before, such as a process where the customer can book their own appointment, and customer forms which ensures all necessary information is obtained. However, some features may miss the mark such as navigation or the look and feel of the product. Therefore, the next step in the development process is to allow the end users to test out the product, receive feedback, then evaluate the design, and adjust it accordingly to best fit the users’ needs. When the design is deemed good enough to be released, we move into the delivery phase of the double diamond model where we commence the final testing of the product and finalize the design.

# Credits

Ashley:  
- Overview  
- Design Patterns  
- Design Systems Section

Jacky:

- Task Flow  
- Hi Fidelity Prototype  
- Summary

Dongseok:

- Wireframe

- Screencast