



# User Interface Documentation

Version 1.0  
30 January 2023

To be used in conjunction with accompanying [Figma design file](#).  
Access password: SC2006

# Contents

---

<b>Colour Palette .....</b>	<b>4</b>
Design Colours .....	5
<b>Generic Components .....</b>	<b>6</b>
Menu Circles .....	7
Menu Component .....	8
Sidebar Header .....	9
Sidebar Section Headers .....	10
Floating Label Input Field .....	11
Short Text Field (Input and Select) .....	12
Clickable Elements .....	13
Hyperlinks .....	14
Checkboxes .....	15
Buttons .....	16
Toilet Overlay .....	17
<b>Mockup Screens .....</b>	<b>18</b>
Opening Page .....	19
Opening Page (Populated) .....	20
Opening Page (Selected) .....	21
Menu Expanded .....	22
Settings .....	23
Account Login .....	24
Account Creation .....	25
Account Information .....	26
Editing Account Information .....	27
Route .....	28
Set Start/Destination Location .....	29
Route Options .....	30

# Contents

---

Find .....	31
Favourites.....	32
Add Toilet Listing .....	33

# Colour Palette

**Colours used in frontend elements of NaviLoo.** Morbi pellentesque, mauris interdum porta tincidunt, neque orci molestie mauris, vitae iaculis dolor felis at nunc. Maecenas eu diam a leo porta interdum. In non massa quis odio feugiat sagittis. Quisque ac lorem. Maecenas ut sem sed ipsum suscipit malesuada. Nulla quis dui quis ante fermentum interdum. Proin eget est a augue vehicula mattis. Pellentesque sed nisi at nisi scelerisque iaculis. Phasellus orci. Nulla adipiscing. Suspendisse et leo vel elit ullamcorper gravida. Proin tempor, ligula ut tincidunt tempor, mauris nibh feugiat odio, nec tincidunt erat orci dictum purus. Etiam luctus odio. Aliquam adipiscing urna a felis. Quisque sed elit in eros aliquet ultrices. Donec at sem. Vestibulum cursus. Integer dolor. Vestibulum sit amet sem

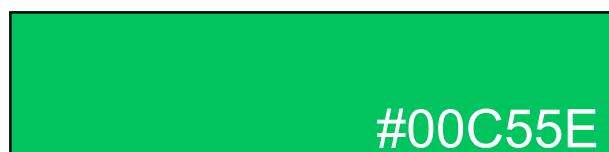
# Design Colours

---



**Rust** Interaction

Accent colour used for buttons and small interaction items.



**Green** Interaction

Primarily for buttons that result in any form of change to permanent data. TailwindCSS green-500 colour property.



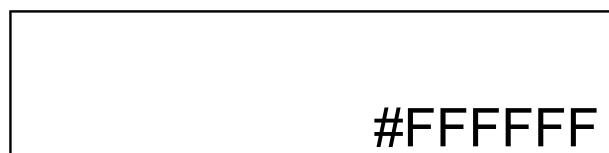
**Violet** Interaction

Main accent colour used for distinct text and interactive buttons like call to actions.



**Shadow** Interaction

Colour used for text and distinctive high contrast features against bright backgrounds.



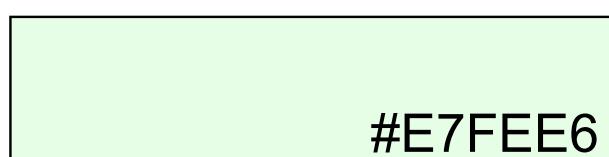
**White** Background

Default background colour for user interface elements.



**Offwhite** Background

Default background colour for map tiles.



**Beige** Background

Colour for selected elements for map tiles.



**Salt** Background

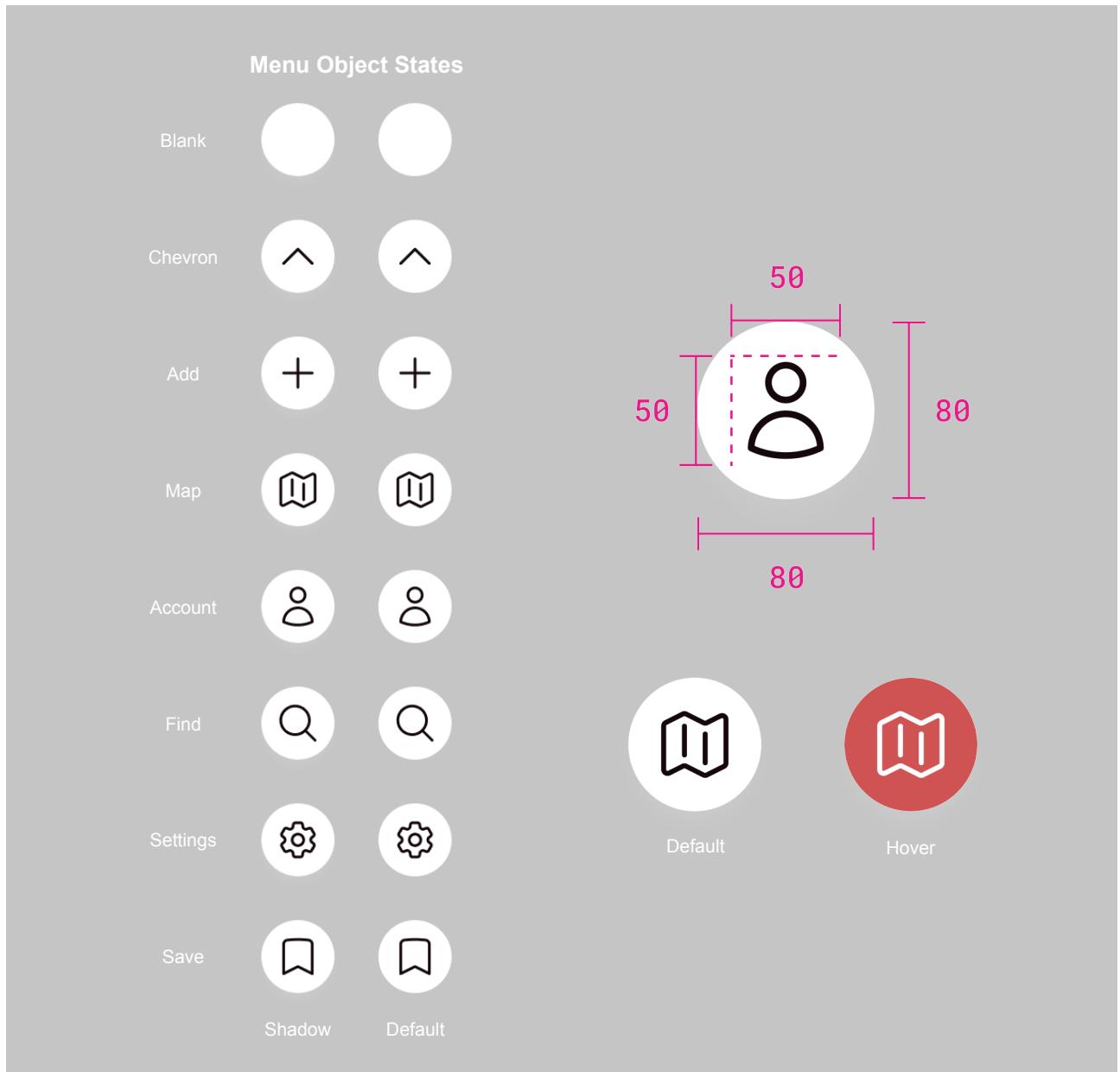
Colour used for roads and drivable paths.

# Generic Components

**Documentation of the reusable components used. This section includes the possible states the elements can take on as well as general use case examples.**

Morbi pellentesque, mauris interdum porta tincidunt, neque orci molestie mauris, vitae iaculis dolor felis at nunc. Maecenas eu diam a leo porta interdum. In non massa quis odio feugiat sagittis. Quisque ac lorem. Maecenas ut sem sed ipsum suscipit malesuada. Nulla quis dui quis ante fermentum interdum. Proin eget est a augue vehicula mattis. Pellentesque sed nisi at nisi scelerisque iaculis. Phasellus orci. Nulla adipiscing. Suspendisse et leo vel elit ullamcorper gravida. Proin tempor, ligula ut tincidunt tempor, mauris nibh feugiat odio, nec tincidunt erat orci dictum purus. Etiam luctus odio. Aliquam

# Menu Circles



## Style

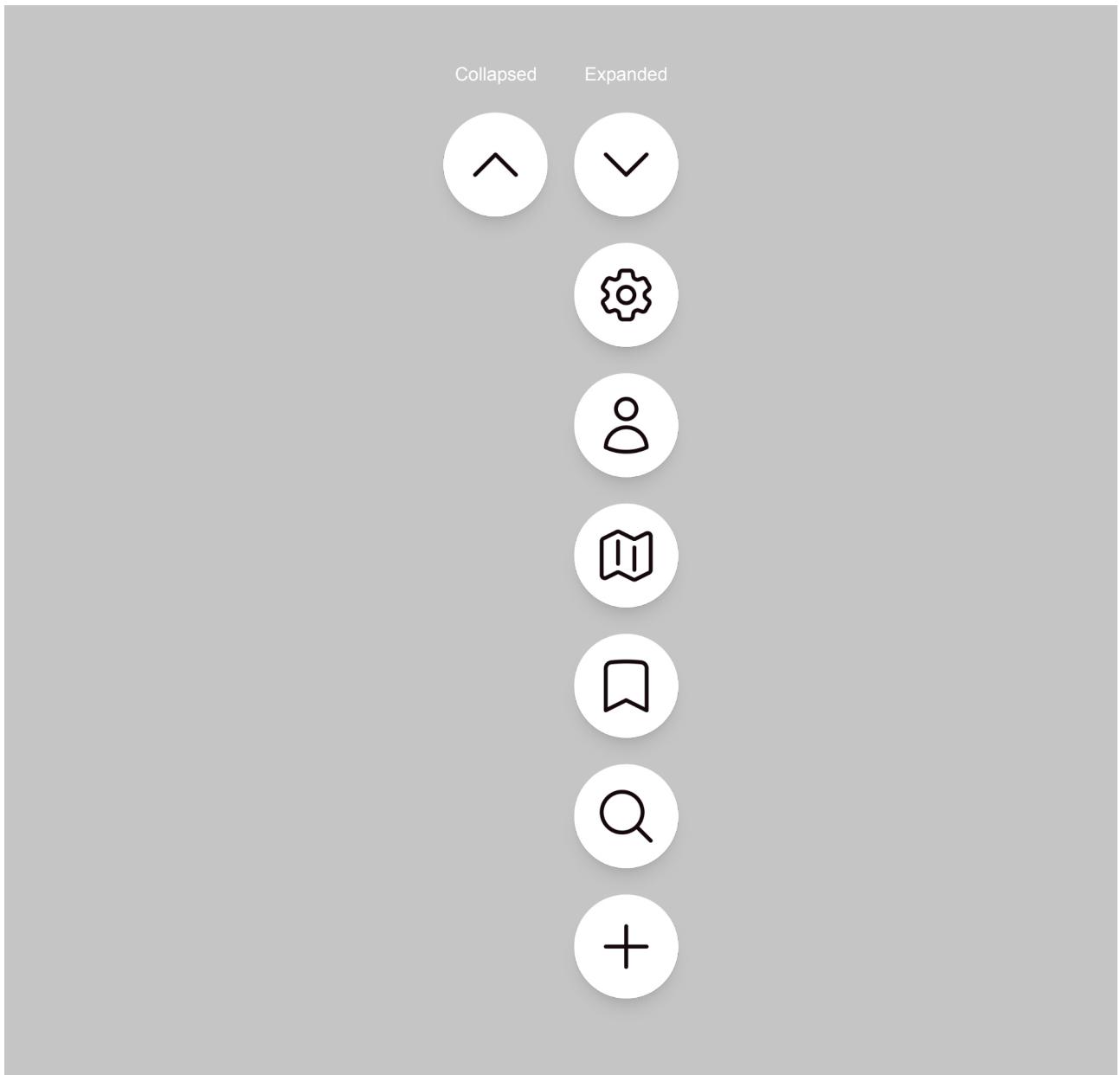
1. Circles should have a background of White and icon stroke of Shadow. Upon hover, a colour transition is to occur. The background will become Rust, and the icon stroke will become Offwhite.
2. Corners are rounded 100%.
3. Shadows are from TailwindCSS utility class of shadow-lg.
4. Stroke width of 3px for icons.

## Remarks

1. Media queries should be used to prevent the hover interaction from firing off on devices that utilise touch as it will lead to buggy visual states of the component.
2. Icons are obtained from [heroicons](#).

# Menu Component

---



## Style

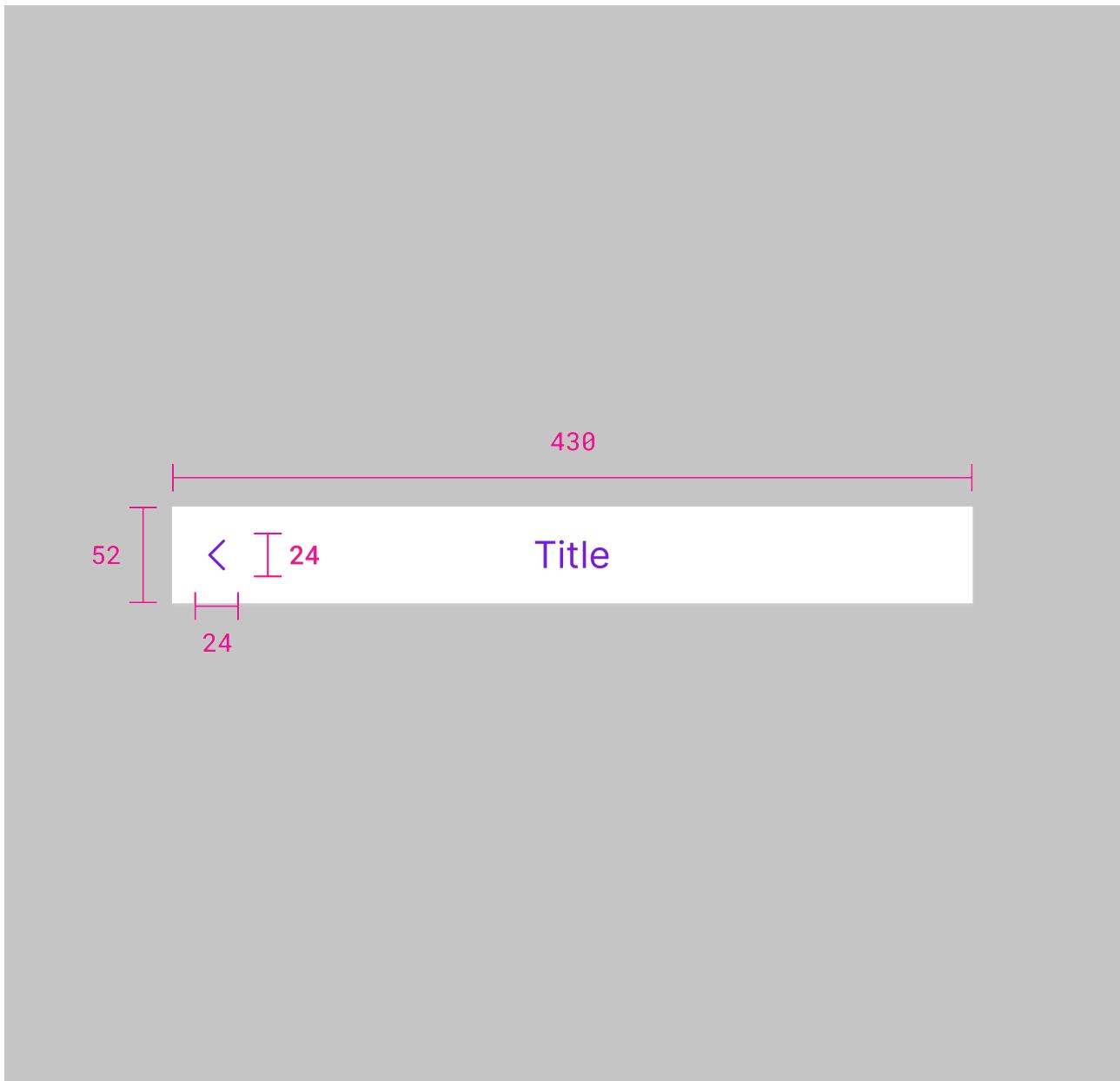
1. Items are arranged in a flexbox with flex style as column.
2. There is a y axis gap of 20px between elements.
3. In the expanded state, the chevron circle element is rotated 180° and is animated via the transition property.

## Remarks

1. When in the collapsed state, only one instance of the circle elements can have the shadow property. This is to prevent stacking of shadows that result in an over exaggerated shadow.

# Sidebar Header

---



## Style

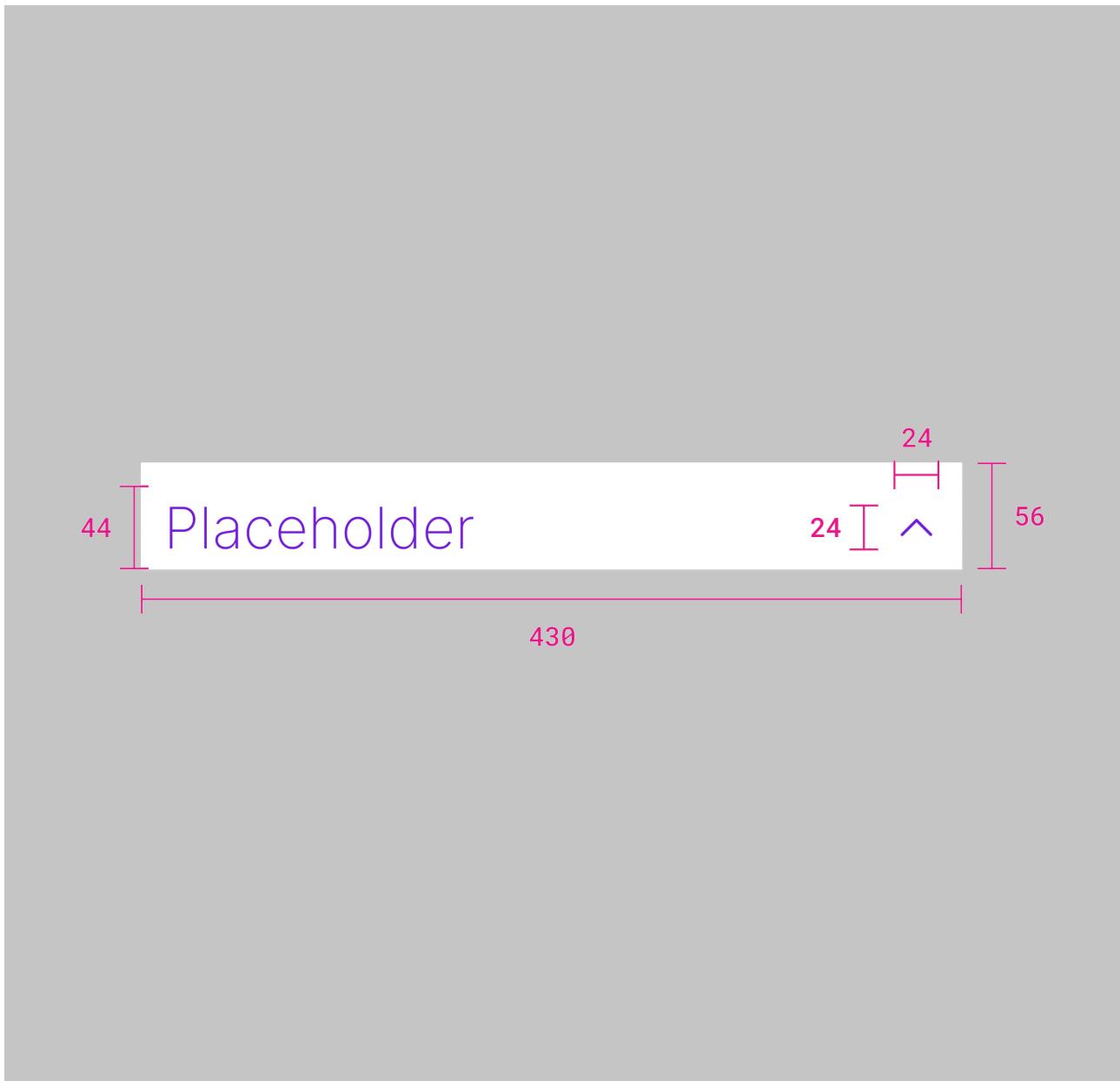
1. Colour used for text and icons is Violet.
2. The icon present in this element has a stroke width of 1.5px.
3. This instance should always have the TailwindCSS utility class shadow-base to produce a noticeable division between this element and the elements below.

## Remarks

1. Icons are obtained from [heroicons](#).

# Sidebar Section Headers

---



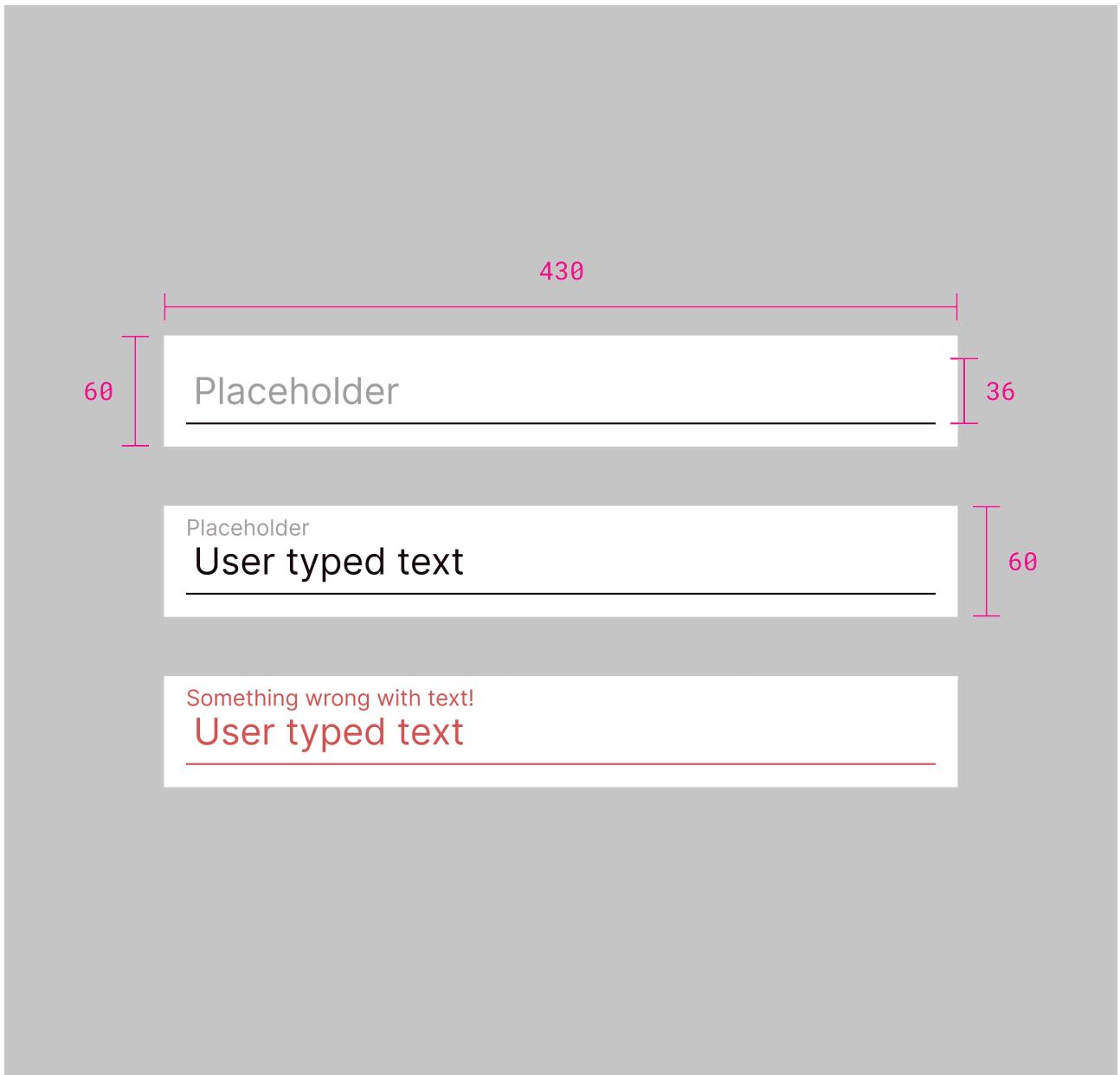
## Style

1. Violet to be used as the colour for text and icon.
2. Icon has a stroke width of 1.5px.
3. Bottom of element has no padding.
4. Chevron element is to rotate 180° upon toggling of state.

## Remarks

1. Icon is clickable to expand fields.
2. Chevron should point in the direction of user intent; pointing upwards to indicate to the user that clicking the icon will induce the fields to collapse upwards.
3. Icons are obtained from [heroicons](#).

# Floating Label Input Field



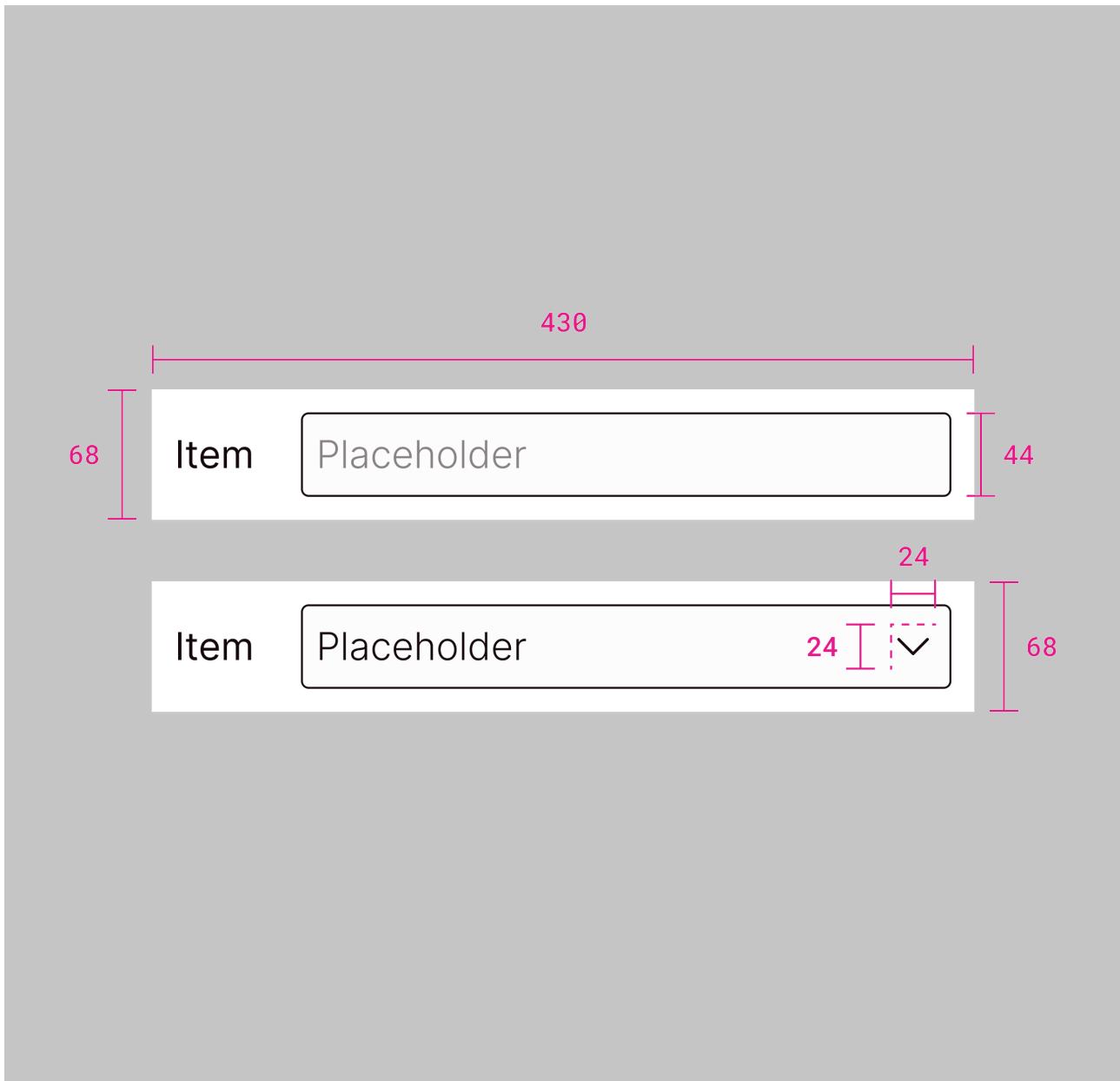
## Style

1. This element should only be used against White or Offwhite backgrounds.
2. Colour used for boarder and text is Shadow. No shadow for element.
3. Error state colour of Rust.
4. Default label has an opacity of 40%.
5. The label should animate to the top left upon user input.

## Remarks

1. Error checking should be done on every keystroke.
2. Element should transit to error state once error in input has been detected.
3. Label is replaced with error text.
4. This element style can be utilised for both text inputs and text areas.

## Short Text Field (Input and Select)



### Style

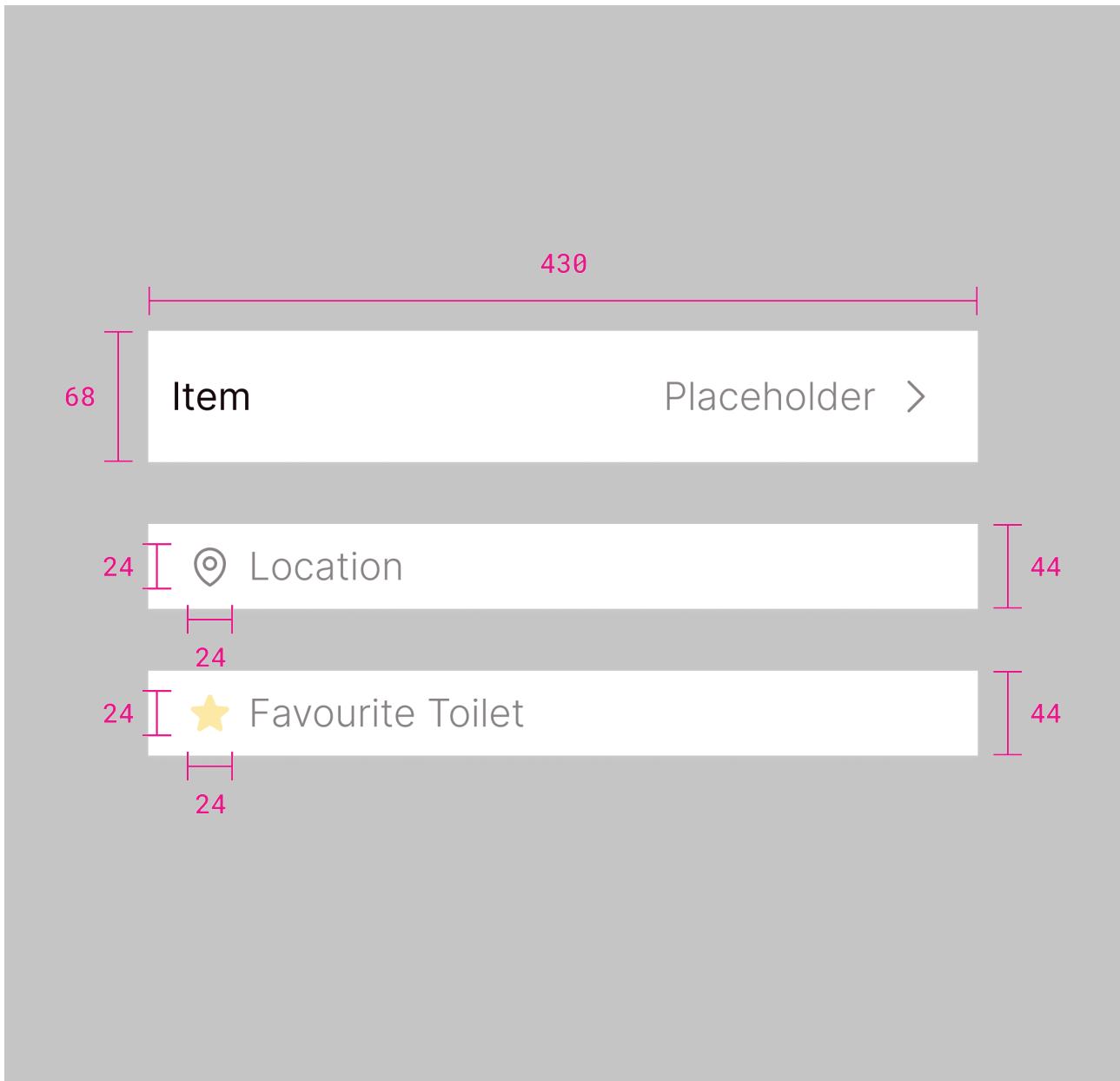
1. This element should only be used against White or Offwhite backgrounds.
2. Colour used for boarder and text is Shadow.
3. Placeholder for input has an opacity of 40%.
4. The element itself should not have any shadow.

### Remarks

1. These fields should be used only for short entries.
2. Icons are obtained from [heroicons](#).

# Clickable Elements

---



## Style

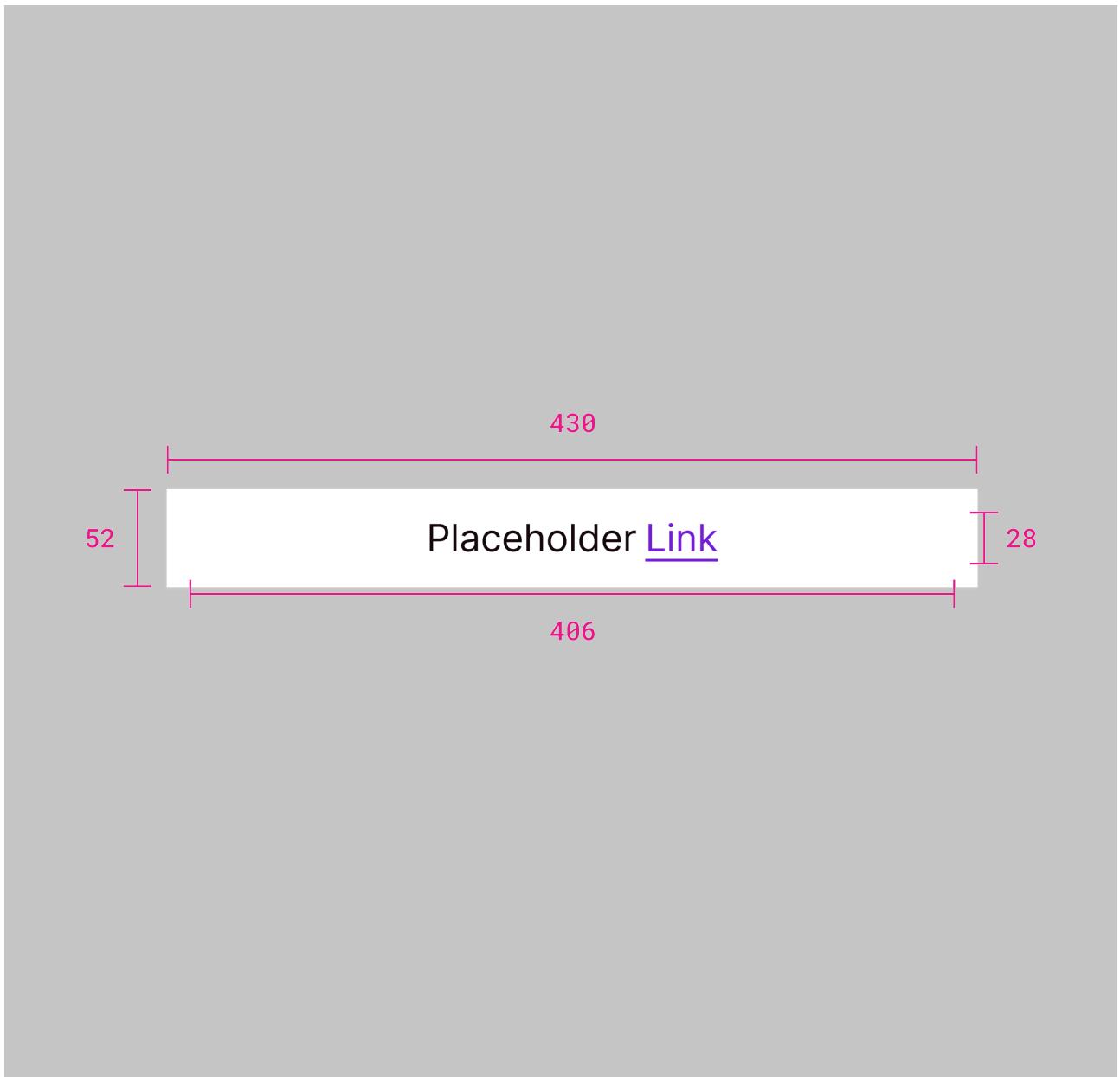
1. Text and icons to use the colour Shadow, with the exception of the star icon which uses TailwindCSS utility class colour yellow-300.
2. Grey colour elements are Shadow in colour with 50% opacity.
3. The star icon is also at 50% opacity.
4. These elements do not have the shadow style.

## Remarks

1. These elements all induce an action.
2. The first element will induce a change in the Sidebar Menu.
3. The second and third element will induce a change in the main screen containing the map.

# Hyperlinks

---



## Style

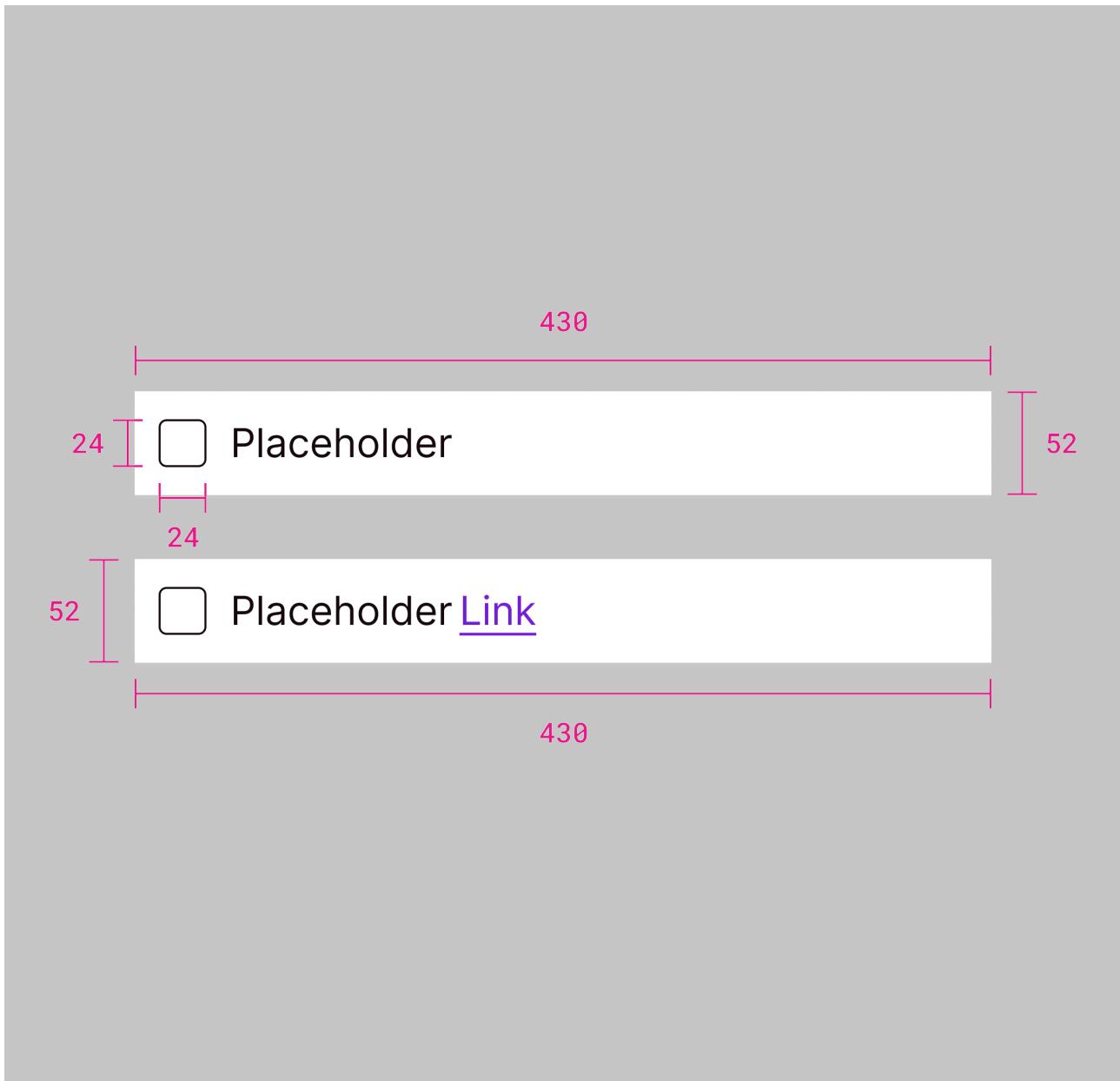
1. This element does not have any background colour or shadow property, but should likewise be only used on bright backgrounds to maintain contrast.
2. Text colour is Shadow.
3. Hyperlink has the colour value of Violet, and an underlined text decoration style.

## Remarks

1. Link should open in a “\_blank” tab, unless the purpose of the link was solely to redirect users (e.g. Signing up).

# Checkboxes

---



## Style

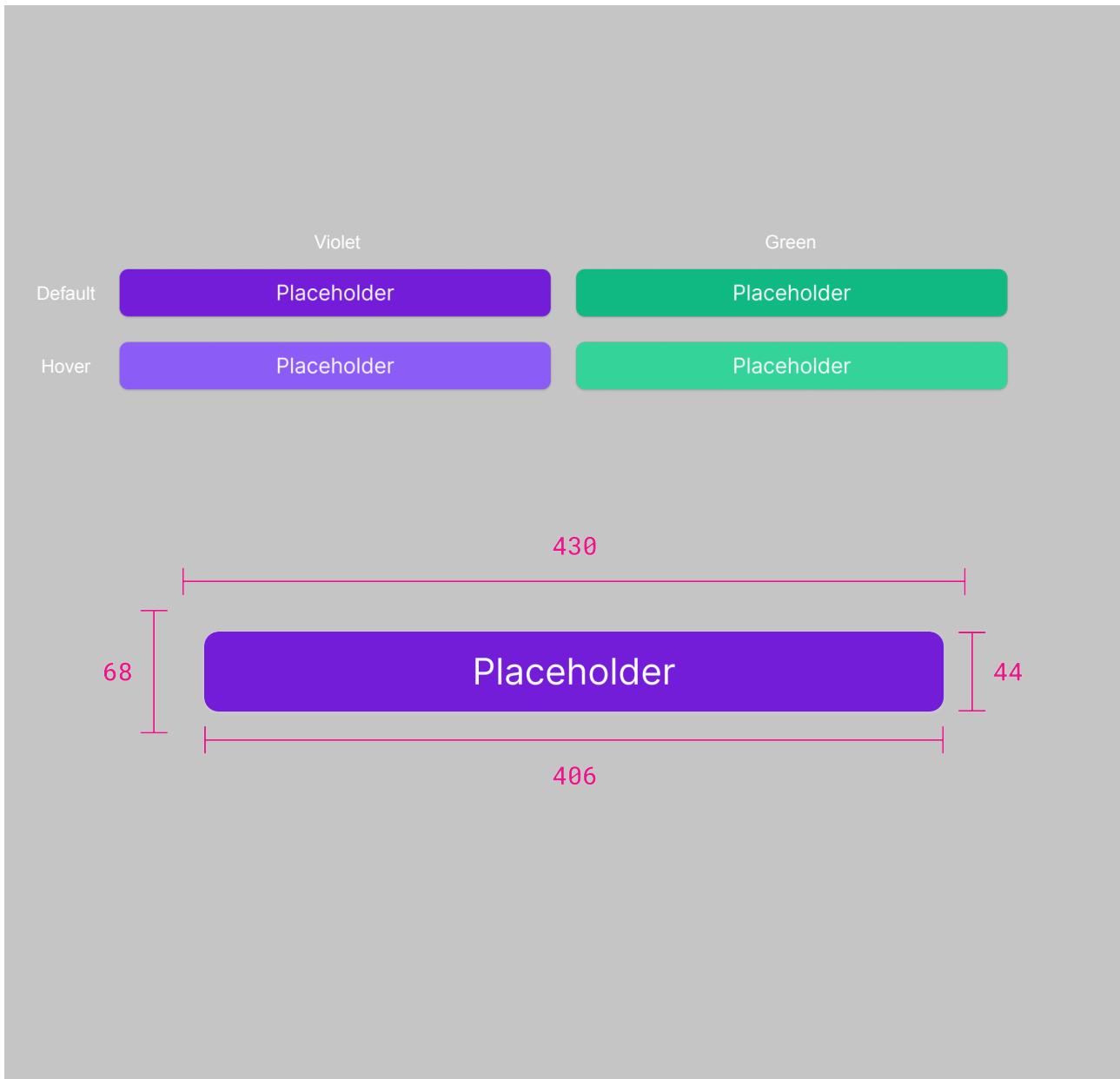
1. Background of these elements are White.
2. Text colour and checkbox outline is Shadow.
3. Link colour is Violet, with underline text decoration.
4. These elements do not have the shadow style.

## Remarks

1. 'Checked' versions of this element will be updated in future versions of this document.
2. Link should open up to “\_blank” by default.

# Buttons

---



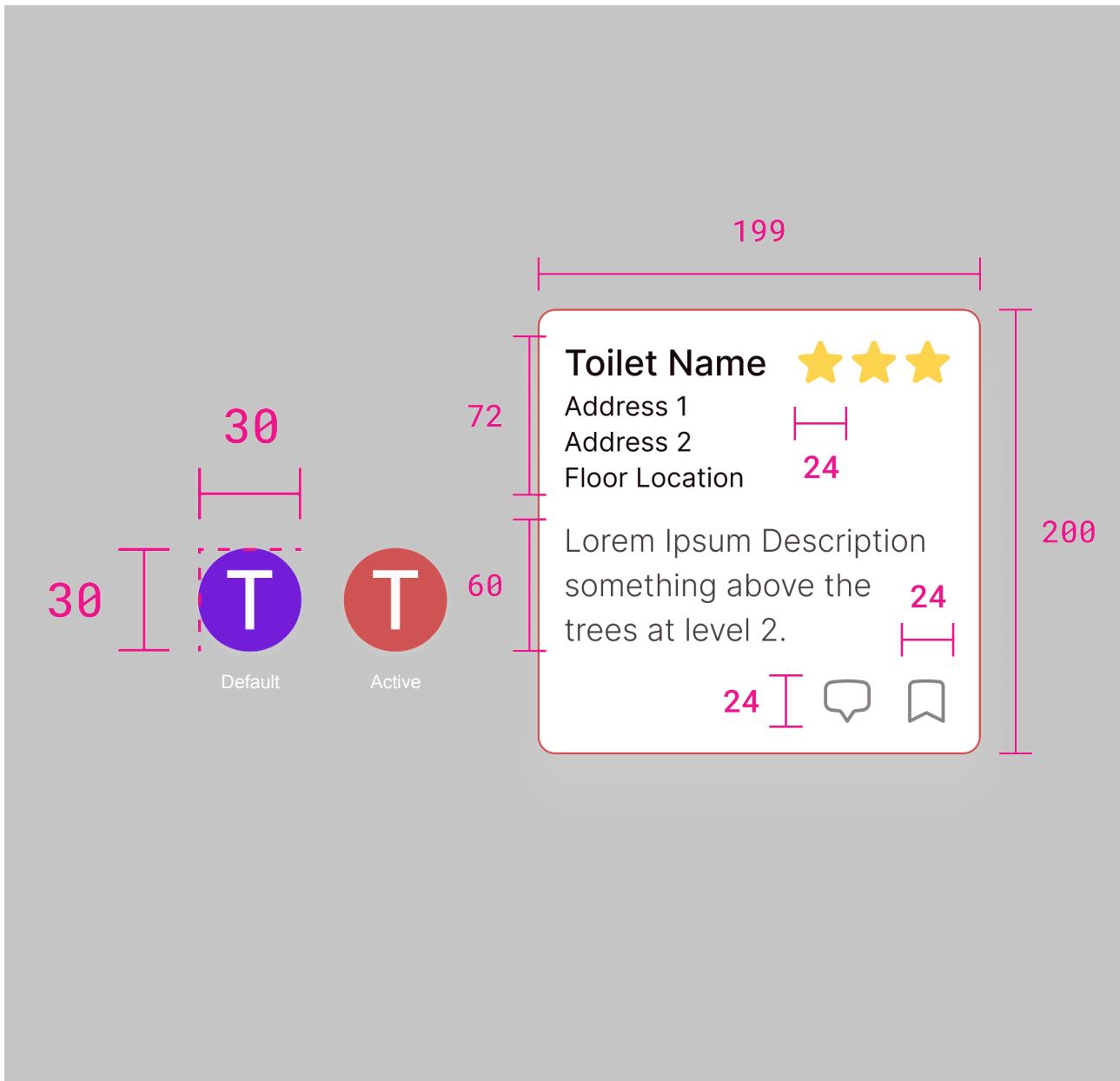
## Style

1. Colours used for default state are Violet and Green-500.
2. On hover, Violet transitions to TailwindCSS utility class Purple-500. Green-500 from TailwindCSS becomes Green-400.

## Remarks

1. The Green button is meant for interactions to update/submit information without changing the routing.
2. The Violet is intended for use where the user will be redirected to another page/component instance.

# Toilet Overlay



## Style

1. Icons on the left signify toilet locations on the map, being tagged to the targets' geolocation. They are Violet and Rust in colour (from left to right).
2. The card on the right is a floating div with information of the selected toilet.
3. Outline of the card is 1px Rust stroke.

## Remarks

1. Upon clicking the Violet toilet marker, its state is toggled to active, where the colour transitions to Rust.
2. Once in an active state, the floating div will appear.
3. The floating div should scale appropriately with its contents.
4. State and floating div will revert to the default state upon clicking away.

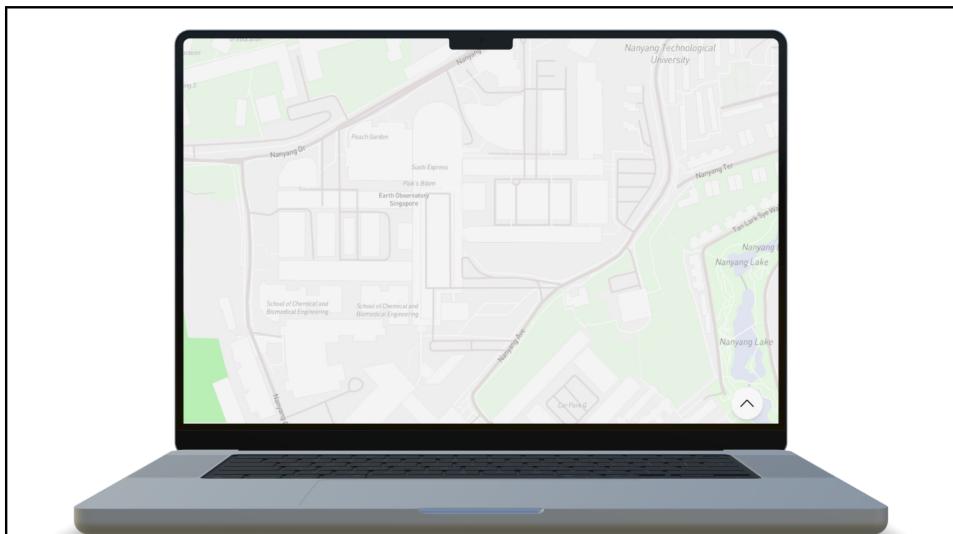
# Mockup Screens

**Design Mockups of the different screens within NaviLoo for mobile and desktop platforms. Refer to Figma design file for more detailed mockups.** Morbi pellentesque, mauris interdum porta tincidunt, neque orci molestie mauris, vitae iaculis dolor felis at nunc. Maecenas eu diam a leo porta interdum. In non massa quis odio feugiat sagittis. Quisque ac lorem. Maecenas ut sem sed ipsum suscipit malesuada. Nulla quis dui quis ante fermentum interdum. Proin eget est a augue vehicula mattis. Pellentesque sed nisi at nisi scelerisque iaculis. Phasellus orci. Nulla adipiscing. Suspendisse et leo vel elit ullamcorper gravida. Proin tempor, ligula ut tincidunt tempor, mauris nibh feugiat odio, nec tincidunt erat orci dictum purus. Etiam luctus odio. Aliquam adipiscing urna a felis. Quisque

# Opening Page



Mobile

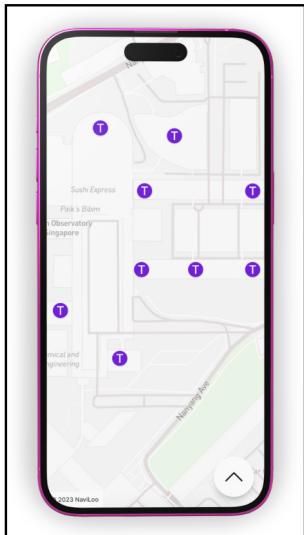


Desktop

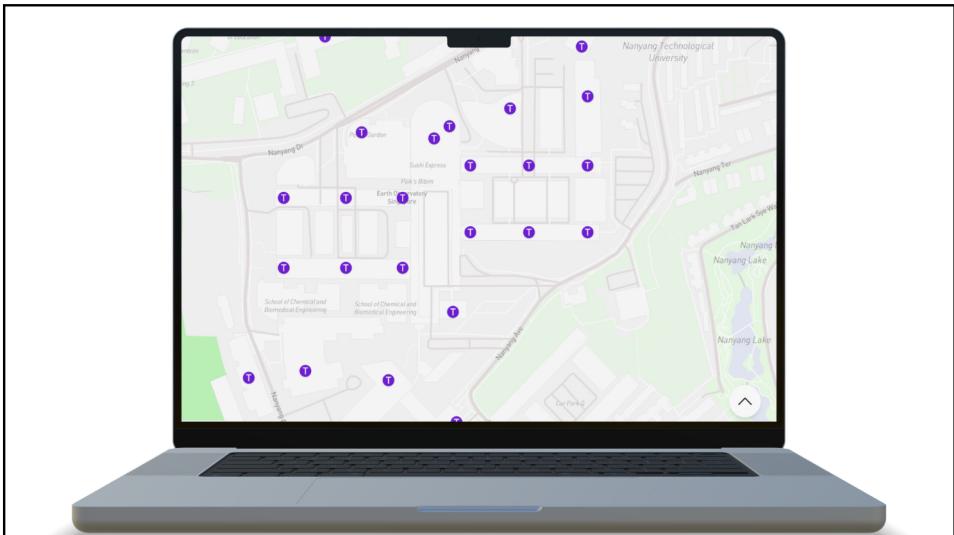
## Remarks

1. Upon loading into NaviLoo, the user should have minimal distractions. In this case, when there are no toilet locations present, only the 'expand menu' button should be visible at the bottom right of the screen.
2. Map attributions should appear on the bottom left of the screen, taking into account respective attributions required.
3. Refer to page 20 for a screen that has been populated with toilet location marks.

# Opening Page (Populated)



Mobile

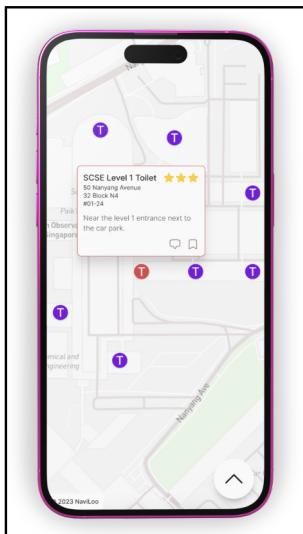


Desktop

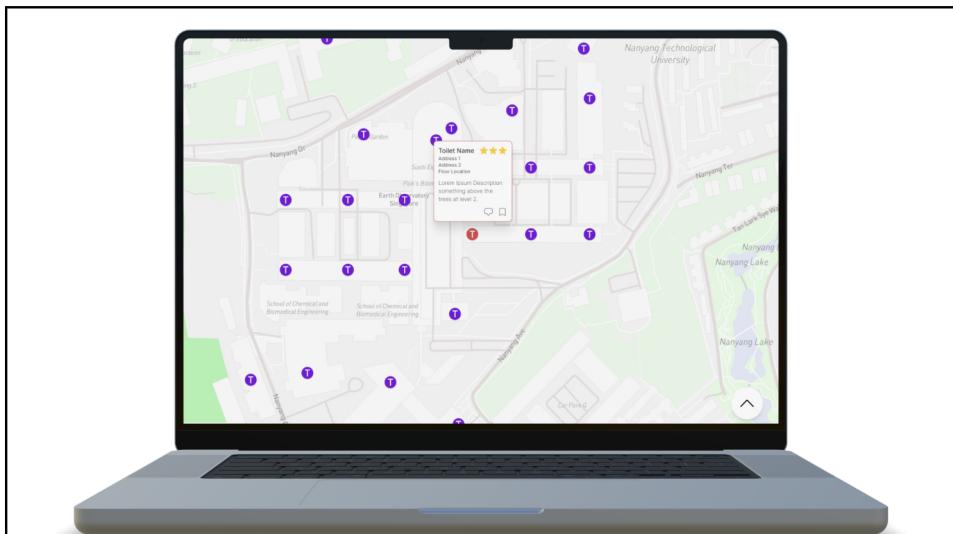
## Remarks

1. Ideally, the user should be greeted with a screen that has already been populated with toilet locations.
2. Each "T" marker indicates a toilet location stored.
3. The toilet markers are clickable elements that do not respond to hover effects.
4. Clicking the toilet marker will toggle its state, as seen on page 21.

# Opening Page (Selected)



Mobile

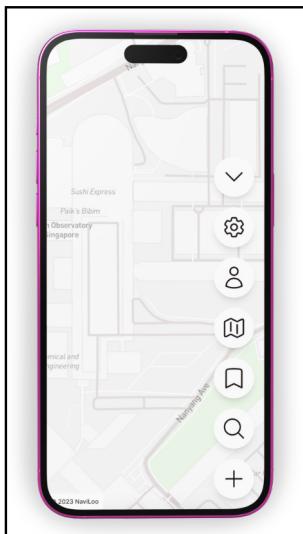


Desktop

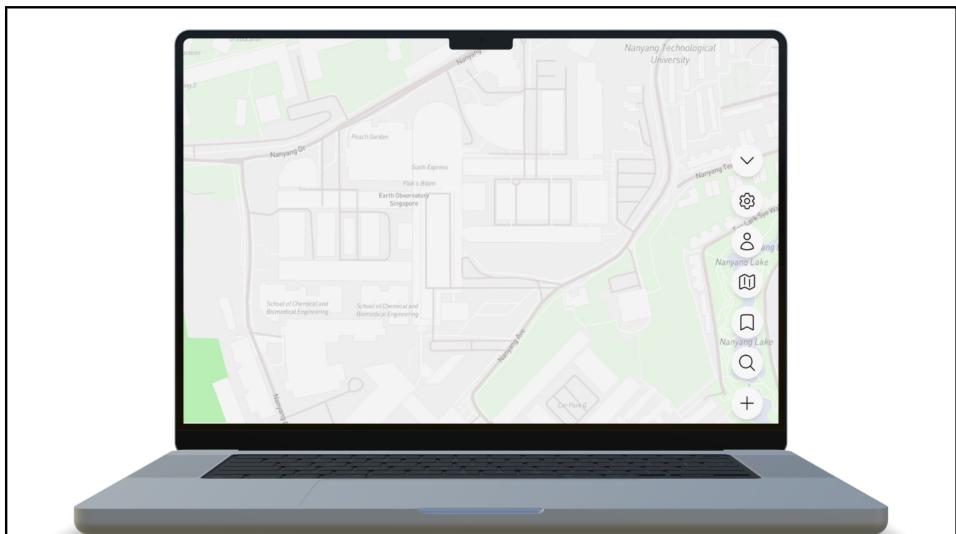
## Remarks

1. Upon toggling the marker to an “active” state, the icon will change colour from Violet to Rust.
2. A floating div will appear above or around the activated marker. This will follow the position of the marker should the user move the map.
3. Only one marker can be “active” at a time, thus upon clicking away, the marker state should reset back to default.

# Menu Expanded



Mobile



Desktop

## Remarks

1. Upon clicking the “expand menu” circle on the bottom right hand of the screen as seen on the opening page, the menu will toggle state to be expanded.
2. Once expanded, the chevron will be rotated 180° to point downwards.
3. After each user interaction with the menu, the menu should collapse after a predefined time (“delay”).

# Settings



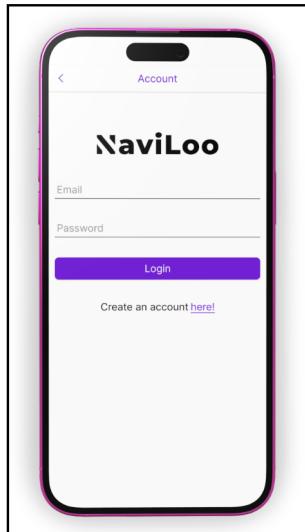
Mobile

Desktop

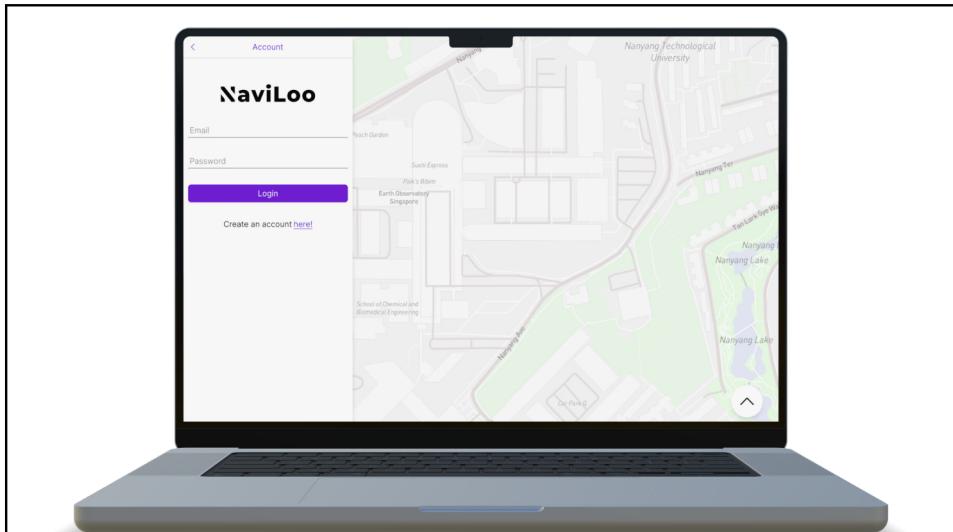
## Remarks

1. The settings page will allow users to change certain system settings found on NaviLoo.
2. On mobile devices, the navigation menu does not expand all the way to the bottom of the screen unlike laptop/desktop devices. This can be controlled using breakpoints.

# Account Login



Mobile

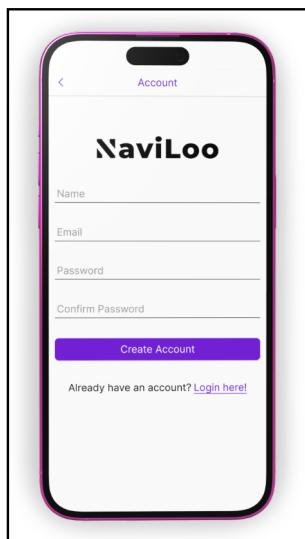


Desktop

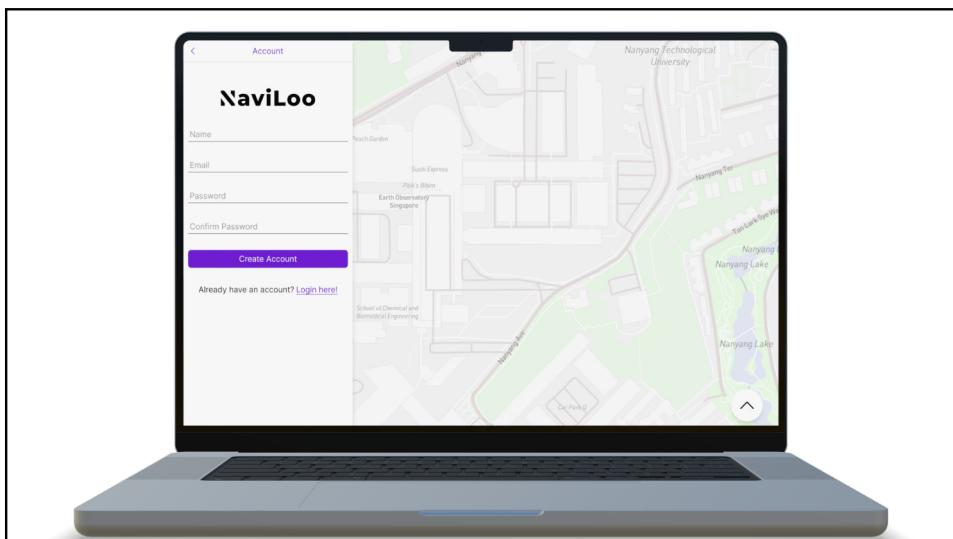
## Remarks

1. Login page for users.
2. Users are also able to create an account by clicking the hyper link below the "Login" button.

# Account Creation



Mobile

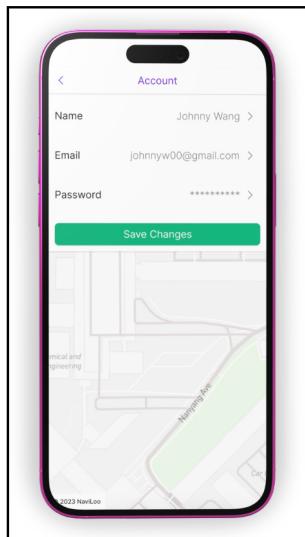


Desktop

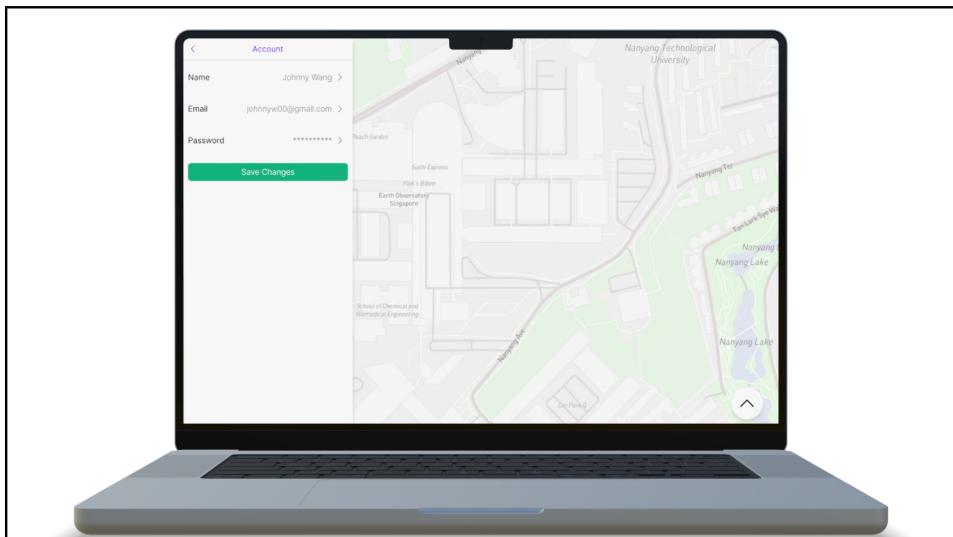
## Remarks

1. Once the user has created an account, the user should be logged into NaviLoo, unless OTP or email verification has been implemented.

# Account Information



Mobile



Desktop

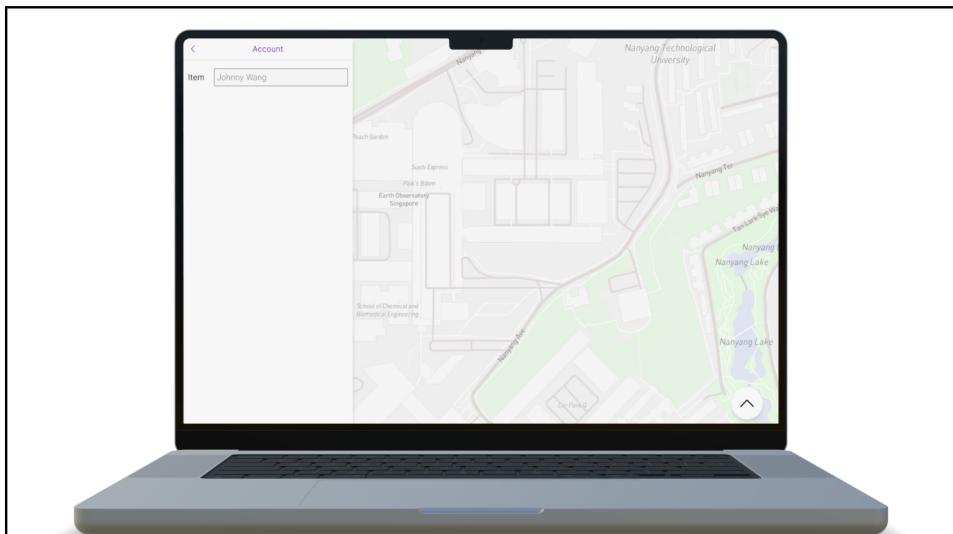
## Remarks

1. As users will be using their emails to login, the name is purely for display purposes.
2. The “Save Changes” button should update local states and server records of the user.

# Editing Account Information



Mobile

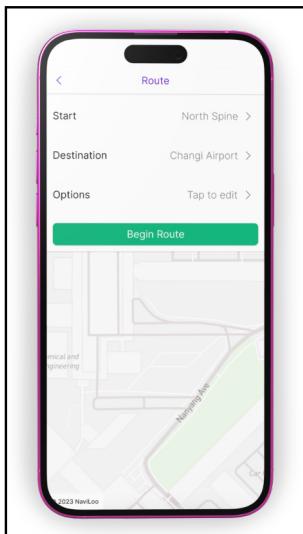


Desktop

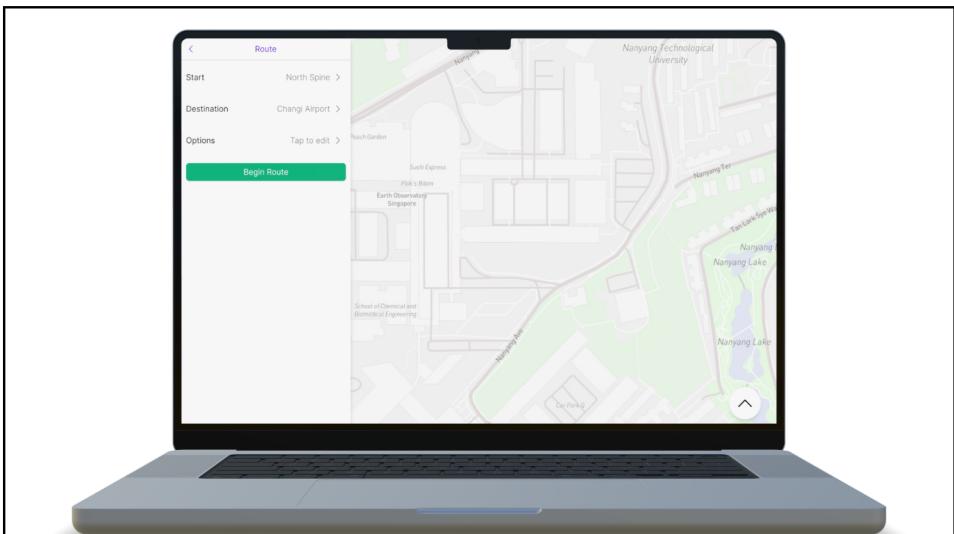
## Remarks

1. In this screen, the name and email can be updated using one field found on the page. The “Item” will have to be updated according to the field that is being updated.
2. Two input fields are required for users to change their passwords as they need to ensure that they have entered the correct password.

# Route



Mobile

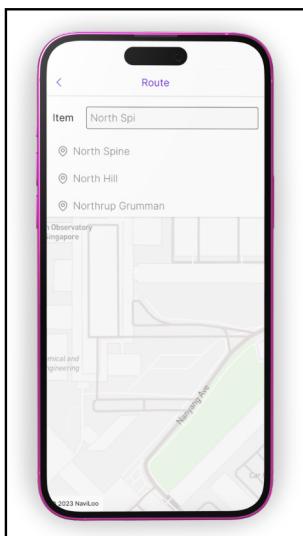


Desktop

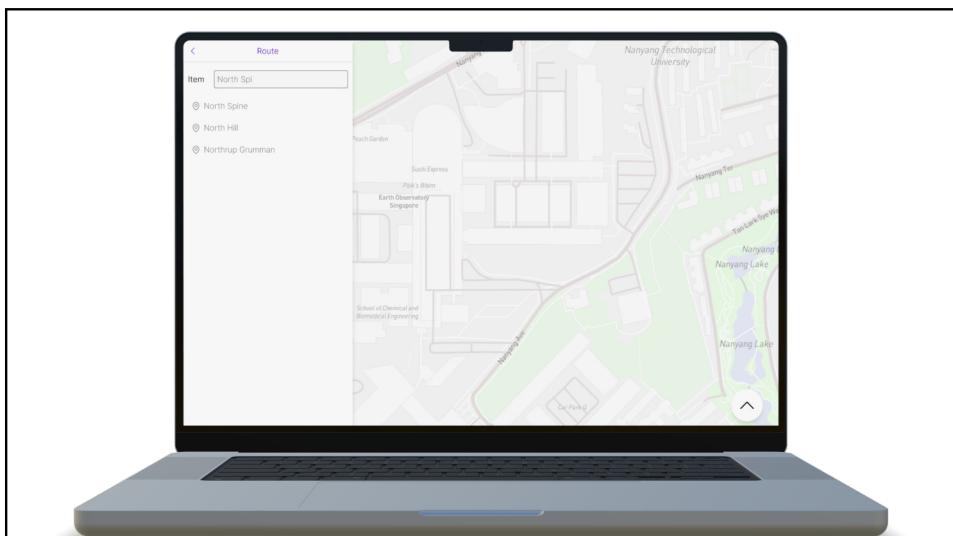
## Remarks

1. All three clickable fields will lead to a separate sidebar page to input the locations.
2. Upon clicking the “Begin Route” button, the menu will collapse and NaviLoo will return to the home page with the route being displayed.

# Set Start/Destination Location



Mobile

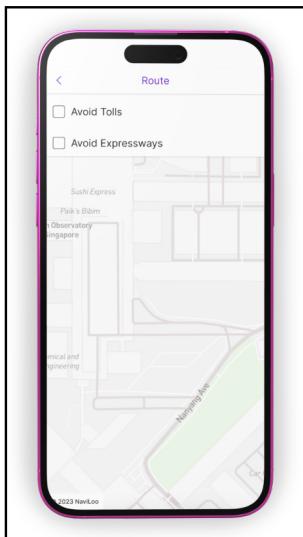


Desktop

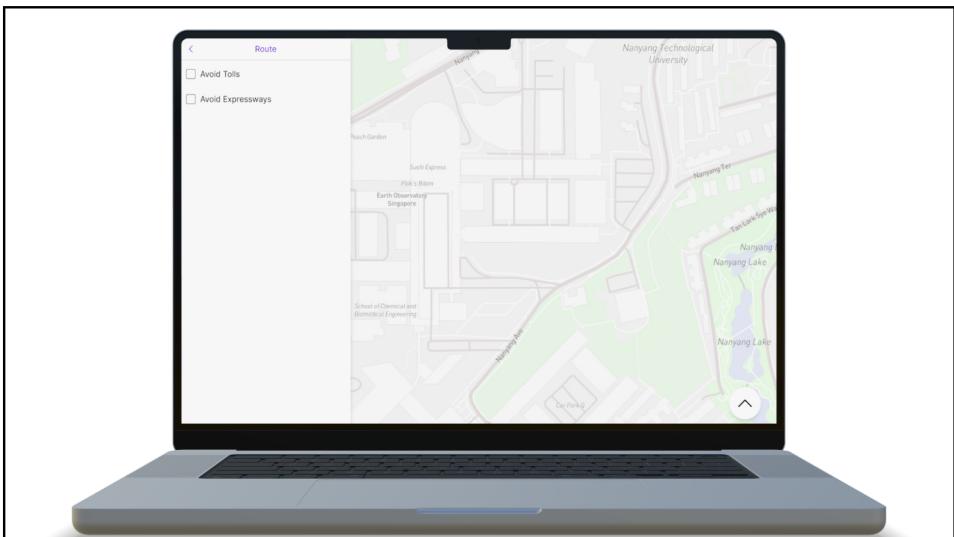
## Remarks

1. This screen is shared when the user is selecting the “Start” and “Destination” locations from the menu seen in page 28.
2. “Item” should be updated to the location being set (“Start”/“Destination”).

# Route Options



Mobile

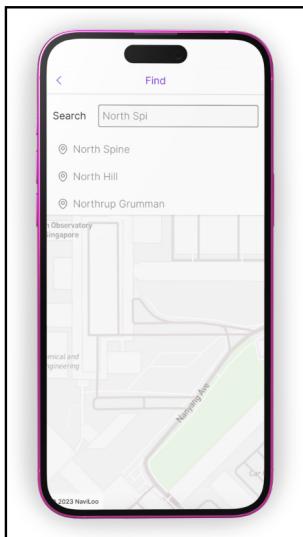


Desktop

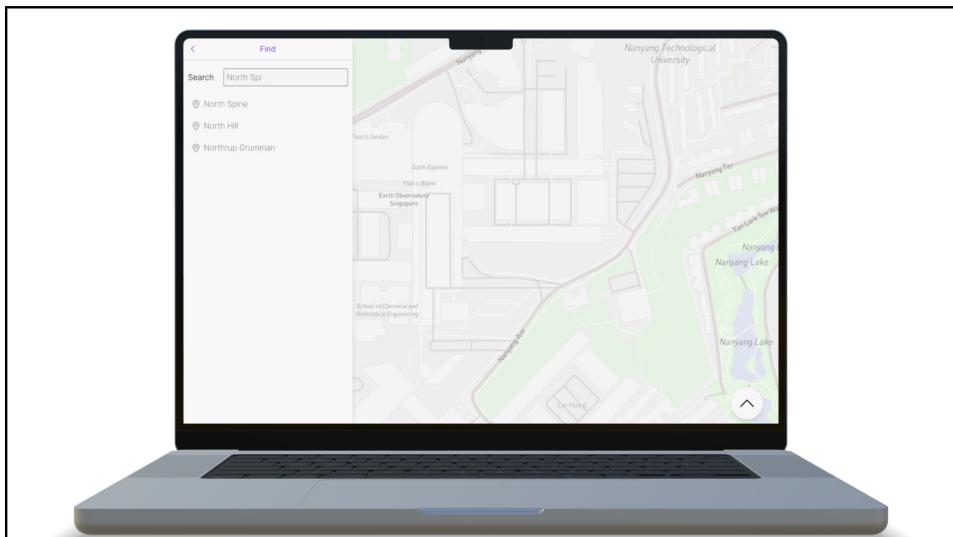
## Remarks

1. The route options in this screen are dependant on the options available from the map routing API service of choice.

# Find



Mobile

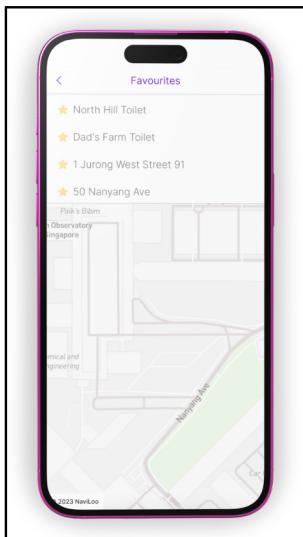


Desktop

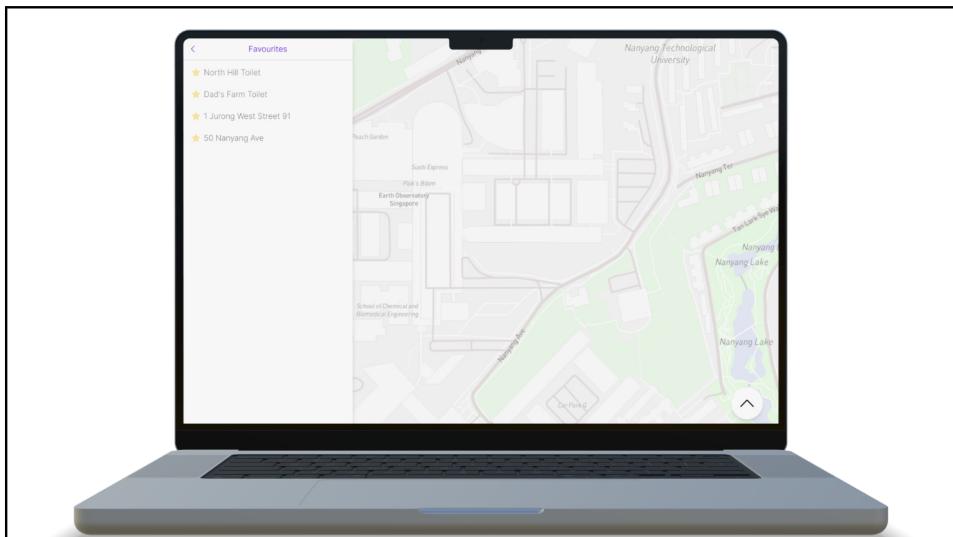
## Remarks

1. Searching for locations in this menu will change the viewport of the map layer to the address specified. It does not have to “snap” to any known toilet locations.

# Favourites



Mobile

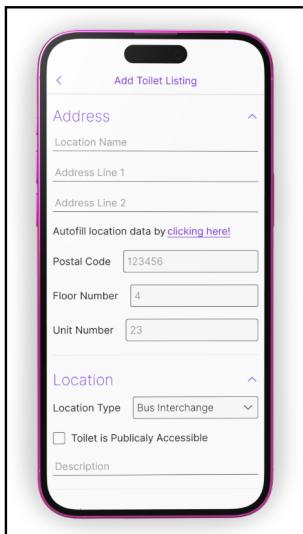


Desktop

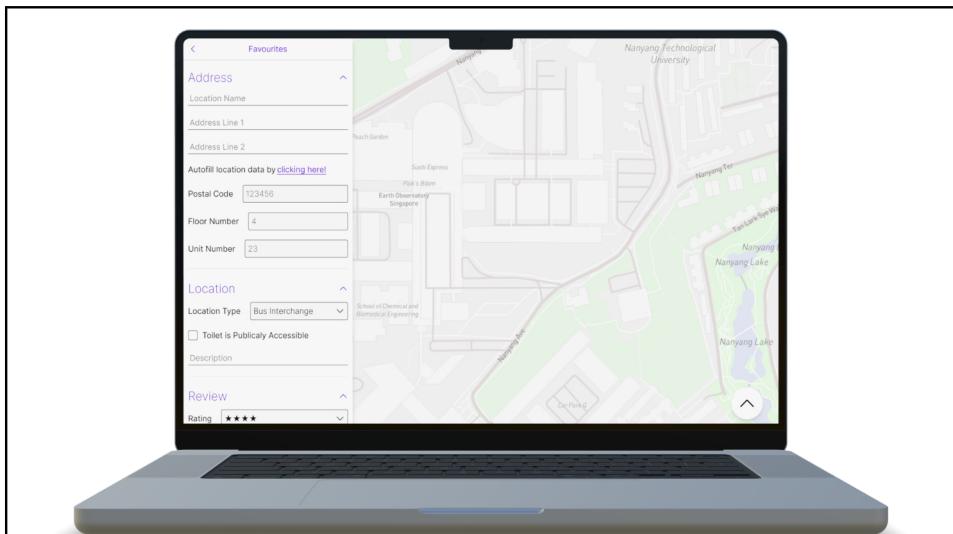
## Remarks

- Upon clicking on any locations on the list, NaviLoo should return back to the home screen. The viewport of the map should also be set on the location of interest.

# Add Toilet Listing



Mobile



Desktop

## Remarks

1. Toilets can be added on both mobile and desktop clients.
2. Autofill location is dependant on device support and API support.