Appendix

Addition to the wireframes for FlyUX

Objectives

A prototype of FlyUX, created in Axure RP 9. The design is the product of research and ideation and sketching.

This prototype is created for desktop/laptop and is not responsive, but this document will also contain guidance with regards to behavior of UI elements on touch devices, such as laptops with touchscreen, iPhones, Android smartphones and tablets.

Prototype
https://lvumsa.axshare.com/forside.html

Location:
The design and payment options are targeted for European and some American credit and debit cards only.

Overview of elements with hover effects

This is a summary of the hover effects on the UI elements.
These effects have been added to meet the needs of users that are either colorblind or/and have other visual impairment.
Note: All hover effects do not work on touch devices.

Buttons:
Hover effects on buttons are color change, change of shape and larger text size, unless mentioned in the document.

Tabs:
Hover effects on tabs are color change, change of shape and larger text size .

Auto search (incremental search):
The hover over search results in list have a grey background.

Background color in boxes:
Background color changes when each search results is hovered.

Links:
Hover effect on links on a button or inside a box, have an increased font size.

Accessibility

General information about the design and Accessibility.
The current design is not compliant with Web Content Accessibility Guidelines (WCAG) 2.1.
if the stakeholders and the management of FlyUX decide that the design should comply to the Web Content Accessibility Guidelines (WCAG) 2.1, as required by law, some changes to the design of FlyUX will be necessary.

An example:
Hint text (placement text)
The hint text in forms does not comply with the rules of (WCAG) 2 with regards to e.g. navigation with tabs, vision impairment and not all text readers can read the hint text (placeholder text).
Furthermore, there are issues with cross over platform and the different interpretation of the code used coding hint text.

Current search field design:

From

Enter city, region or country

Form design after revision:

From
Enter city, region or country

The hint text (placement text) is placed outside the form element.
Both label and the hint text is placed above the field.

Form fields

Error message

Validation occurs after the cursor left the field. If the validation fails, an error message is shown under the field. Depending on the width and the entered data, add an icon like this beside the error message. If there is no room for the icon leave the icon out, but keep the color and background color tone.

E-mail

⚠ Error! Please, add @ in the e-mail address

Validation

In the case that the validation is successful, show micro animation. This only applies fields on the payment page, to validate card details.

Label

✅

Form fields with hint text

When a field is selected the field is highlighted and raised and when the user starts typing the hint text is hidden.
The hint text (placeholder text) in side the field must be easy to read, not too light a grey. Check the contrast.
When the cursor exits the field, the hint text is visible again. The label is placed over the field.

Label

Hint text - smaller font - less contrast

Links

Prototype
https://lvumsa.axshare.com/forside.html

Web Content Accessibility Guidelines (WCAG) 2.1
https://www.w3.org/TR/WCAG21/

US government accessibility standards
https://www.powermapper.com/blog/government-accessibility-standards/

Baymard - input types touch keyboard
https://baymard.com/labs/touch-keyboard-types

Baymard - credit card patterns
https://baymard.com/checkout-usability/credit-card-patterns

HTML 5 input types for touch devices

E-mail field
Disable auto-correct and disable auto-capitalization. Invoke special @ keyboard.

```
<input type="email" autocapitalize="off" autocorrect="off" autocomplete="email">
```

Phone field
Invoke special phone keyboard. (Note: iOS doesn't allow input of special characters such as parenthesis and dash with the phone keyboard. Thus, never require phone numbers to be formatted with such characters.)

```
<input type="tel" autocorrect="off" autocomplete="tel">
```

Name field
Disable auto-correct. (Note: while it is recommended to use a single name field, if you split it across multiple fields, be sure to assign the appropriate autocomplete values.)

```
<input type="text" autocorrect="off" autocomplete="name">
```

Address line fields
Disable auto-correct. (Note: be sure to update the autocomplete attribute accordingly if using multiple address line fields.)

```
<input type="text" autocorrect="off" autocomplete="address-line1">
```

City
Disable auto-correct.

```
<input type="text" autocorrect="off" autocomplete="address-level2">
```

State
Disable auto-correct.

```
<input type="text" autocorrect="off" autocomplete="address-level1">
```

ZIP code field
Set input pattern to numeric input. (Note: Argentina, Canada, Netherlands, and UK, may use letters in their postal code. To support these, dynamically change the input pattern

depending on selected country. Also, for the numeric keyboard to be invoked on all Android devices, the field type must be changed to type=number, however, this may cause issues with leading zeroes in some browser versions – therefore if changing to type=number, make absolutely sure to handle these exceptions.)

```
<input type="text" inputmode="numeric" pattern="[0-9]*" novalidate autocorrect="off" autocomplete="postal-code">
```

Credit card number field
Numeric keyboard. (Note: for the numeric keyboard to be invoked on all Android devices, the field type must be changed to type=number, however, this may cause issues with leading zeroes in some browser versions – therefore if changing to type=number, make absolutely sure to handle these exceptions.)

```
<input type="text" inputmode="numeric" pattern="[0-9]*" novalidate autocorrect="off" autocomplete="cc-number">
```

Credit card security code field
Numeric keyboard. (Note: for the numeric keyboard to be invoked on all Android devices, the field type must be changed to type=number, however, this may cause issues with leading zeroes in some browser versions – therefore if changing to type=number, make absolutely sure to handle these exceptions.)

```
<input type="text" inputmode="numeric" pattern="[0-9]*" novalidate autocorrect="off" autocomplete="cc-csc">
```

Quantity field
Semantic numeric keyboard.

```
<input type="number">
```

Date field
Date picker keyboard. (Note: you may want to implement an actual calendar date picker.)

```
<input type="date">
```

Submit button
For the 'Go' button on keyboard to work, avoid using type=button.
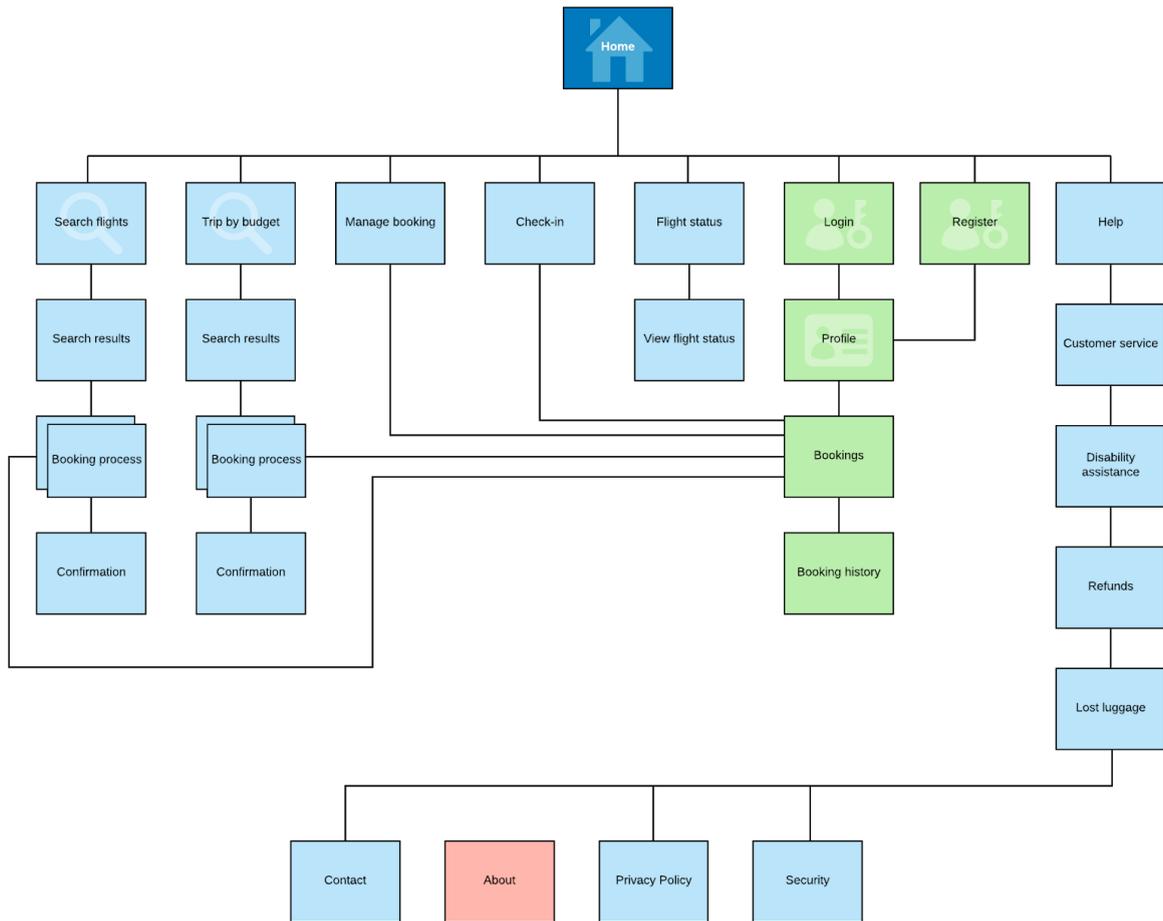```
<input type="submit">
```

Reference:
https://baymard.com/labs/touch-keyboard-types

# Reference credit/debit card digit pattern

| Card type | Length | IIN Range (begins with) | Spacing pattern |
|-----------|--------|-------------------------|-----------------|
| Visa (incl. VPay) | 13-19 | 4 | #### #### #### #### (4-4-4-4 |
| Visa Electron | 16 | 4026, 417500, 4405, 4508, 4844, 4913, 4917 | #### #### #### #### (4-4-4-4) |
| MasterCard | 16 | 51-55, 222100-272099 | #### #### #### #### (4-4-4-4) |
| American Express | 15 | 34, 37 | #### ###### ##### (4-6-5) |
| Diners Club International | 14 | 300-305, 309, 36, 38-39 | #### ###### #### (4-6-4) |

Reference:
https://baymard.com/checkout-usability/credit-card-patterns

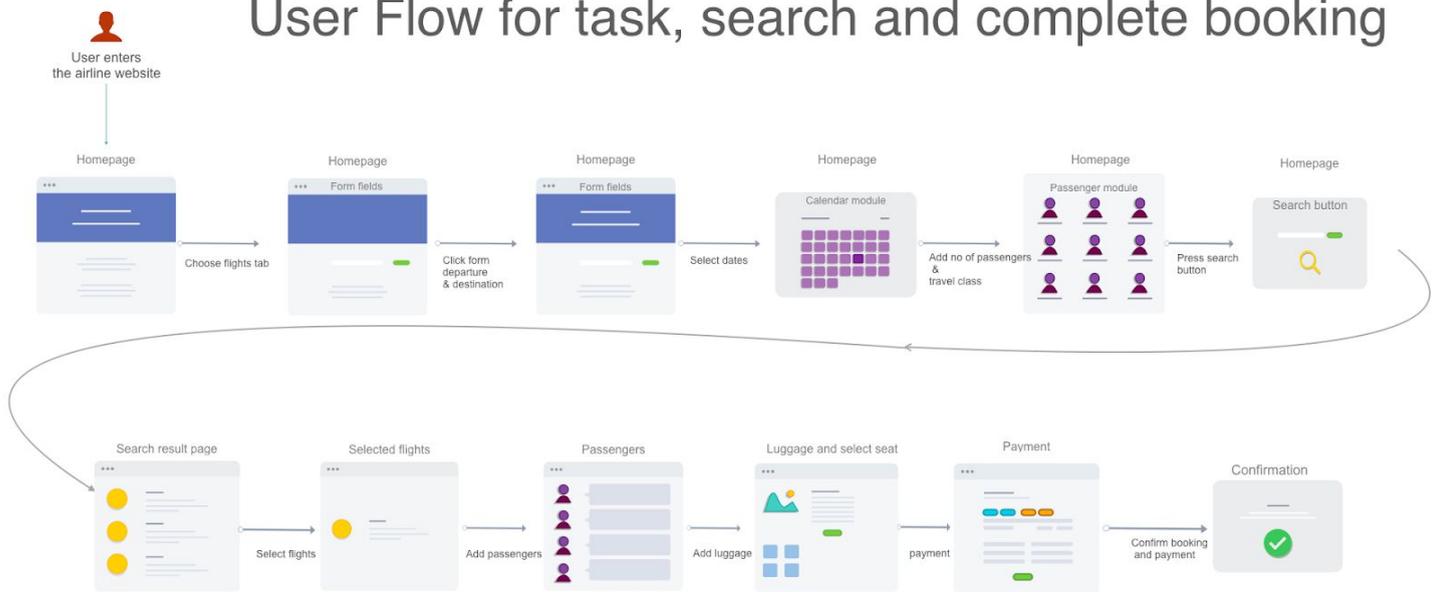# Sitemap

User flow
- task search flight and complete booking



User Flow for task, search and complete booking

User enters the airline website

**Homepage** — Choose flights tab
**Homepage** (Form fields) — Click form departure & destination
**Homepage** (Form fields) — Select dates
**Homepage** (Calendar module) — Add no of passengers & travel class
**Homepage** (Passenger module) — Press search button
**Homepage** (Search button)

Search result page — Select flights
Selected flights — Add passengers
Passengers — Add luggage
Luggage and select seat — payment
Payment — Confirm booking and payment
Confirmation

**Tasks:** Search for a flight, complete booking and payment