

Selected projects 2015-2021

Bartosz Dębicki

mobile :+48 508 343 705
email :bartosz.debicki@zoin.pl

linkedin :[/in/bartoszdebicki](https://in/bartoszdebicki)
github :[/bdebicki](https://github.com/bdebicki)
behance :[/earldebicki](https://www.behance.net/earldebicki)

UXPin Design System

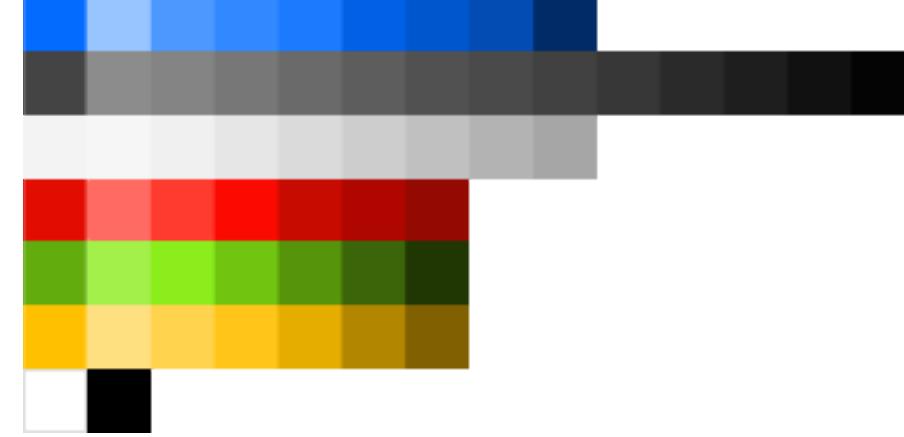
Basing on the actual design of the product I have prepared a design system. The goal was to clean and organize and simplify colors, typography, iconography, and interface elements. During this work, I was in close contact with the design and development team.

Ongoing since 2017 | UXPin
role :leader / code + design + evangelisation

color palette

Based on colors used in the UXPin application I've prepared palate of colors. ~5500 colors defined in code as hex has been limited to 55 defined colors. Each color has its own base and number of lighten / darken accents.

Except for colors itself, I have prepared the naming pattern.



typography scale

Similar to colors I have check all text styles used in the product and prepare a close list of text styles.

In addition, we decide to use only 2 font families - Proxima Nova and Source Code Pro. Each text or heading can behave in one of the available font families.

heading 1

heading 2

heading 3

heading 4

heading 5

heading 6

heading 7

heading 8

HEADING 9

HEADING 10

text XS: The quick brown fox jumps over the lazy dog.

text S: The quick brown fox jumps over the lazy dog.

text M: The quick brown fox jumps over the lazy dog.

text L: The quick brown fox jumps over the lazy dog.

icons

In UXPin was 291 used icons. A lot of those icons were duplicate and similar. During cleaning, I decrease the number of icons to 172.



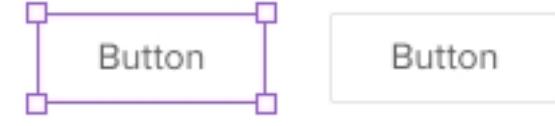
components

The fourth part of the design system is components. Similar to previous areas I have check each version of each component and normalize it to one with few versions. Currently, in the UXPin Design System library, we designed and developed 22 components.

Components were recreated as reusable UXPin symbols and coded as a React component. Each of the components has detailed documentation with use cases and examples



```
<Button>  
  Button  
</Button>
```



```
<Button  
  appearance="neutral"  
  weight="light">  
  Button  
</Button>
```

Merge Library Manager

Merge Library manager is an evolution of the Merge integration process. The best description for the feature is that Merge Library Manager is the User Interface layer for Merge integration. Because classic integration is based on a GIT repository, it forced users to be code-know. New integration process based on the NPM package. This approach doesn't force the user to have access to the git repository and does not force him to prepare any configuration from the code side. This approach opens Merge integration to non-coding users.

I am responsible for the design and front-end development.

2021 - present | UXPin
role :code + design

merge library manager for nom and git integrated libraries

The screenshot shows the Material UI component editor interface. On the left is a sidebar with categories like General, Actions, Navigation, Layout, and Card. The 'Actions' section has 'Button' selected. The main area shows a preview of a blue 'SIGN UP' button and its properties. Properties include 'Color' (blue), 'Property name' (Color), 'Display name' (Color), 'Description' (empty), 'Property type' (one of), 'Property control' (select), 'Options' (default, inherit, primary, secondary), 'Default value' (primary), 'Children' (empty), 'Disabled' (disabled focus ripple, disabled ripple), and 'Full width' (empty). Top right buttons: 'Discard changes' and 'Save changes'.

merge tab in dashboard

The screenshot shows the Merge Libraries dashboard. It features two main sections: 'Components from code' (with icons for Material UI and Storybook) and 'Components from Storybook' (with a large 'S' icon). Below each section is a 'Create library with components from [source]' button. A note at the bottom says 'Sync your UI components from your Git repository, NPM package or Storybook. Design with the same components that your devs are building with and achieve the ultimate consistency.' The top navigation bar includes 'Projects', 'Design Systems', 'Merge', and 'Manage your team'.

adding new component flow

The screenshot shows the 'Add new components' dialog. It has a text input for 'List import for components you would like to use in your library' containing code like 'import { Button, Input } from '@your-library';'. A 'select category' dropdown is open. A 'Cancel' button and an 'Add components' button are at the bottom. Below the dialog are sections for 'Size' (Size: 11.5) and 'Variant' (Variant: latest).

Great work!

You have added 3 new components to your library.
Select one from the list below and fill it with properties

Button
Checkbox
Input

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

IconButton

Link

Navigation

AppBar

ButtonNavigation

ButtonNavigationAction

Drawer

Tab

Tabs

TabsContainer

Toolbar

TabContent

Layout

Grid

GridList

GridListTile

GridListTitleBar

Card

Card

Cardctions

CardContent

CardHeader

CardMedia

Form

Checkbox

FilledInput

FormControl

FormControlLabel

FormGroup

FormHelperText

FormLabel

Input

InputList

InputLabel

MenuItem

+ Add new component

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

IconButton

Link

Navigation

AppBar

ButtonNavigation

ButtonNavigationAction

Drawer

Tab

Tabs

TabsContainer

Toolbar

TabContent

Layout

Grid

GridList

GridListTile

GridListTitleBar

Card

Card

Cardctions

CardContent

CardHeader

+ Add new component

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

IconButton

Link

Navigation

AppBar

ButtonNavigation

ButtonNavigationAction

Drawer

Tab

Tabs

TabsContainer

Toolbar

TabContent

Layout

Grid

GridList

GridListTile

GridListTitleBar

Card

Card

Cardctions

CardContent

CardHeader

+ Add new component

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

IconButton

Link

Navigation

AppBar

ButtonNavigation

ButtonNavigationAction

Drawer

Tab

Tabs

TabsContainer

Toolbar

TabContent

Layout

Grid

GridList

GridListTile

GridListTitleBar

Card

Card

Cardctions

CardContent

CardHeader

+ Add new component

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

IconButton

Link

Navigation

AppBar

ButtonNavigation

ButtonNavigationAction

Drawer

Tab

Tabs

TabsContainer

Toolbar

TabContent

Layout

Grid

GridList

GridListTile

GridListTitleBar

Card

Card

Cardctions

CardContent

CardHeader

+ Add new component

Material UI

Discard changes Save changes

All categories

General

Icon

Typography

Actions

Button

Fab

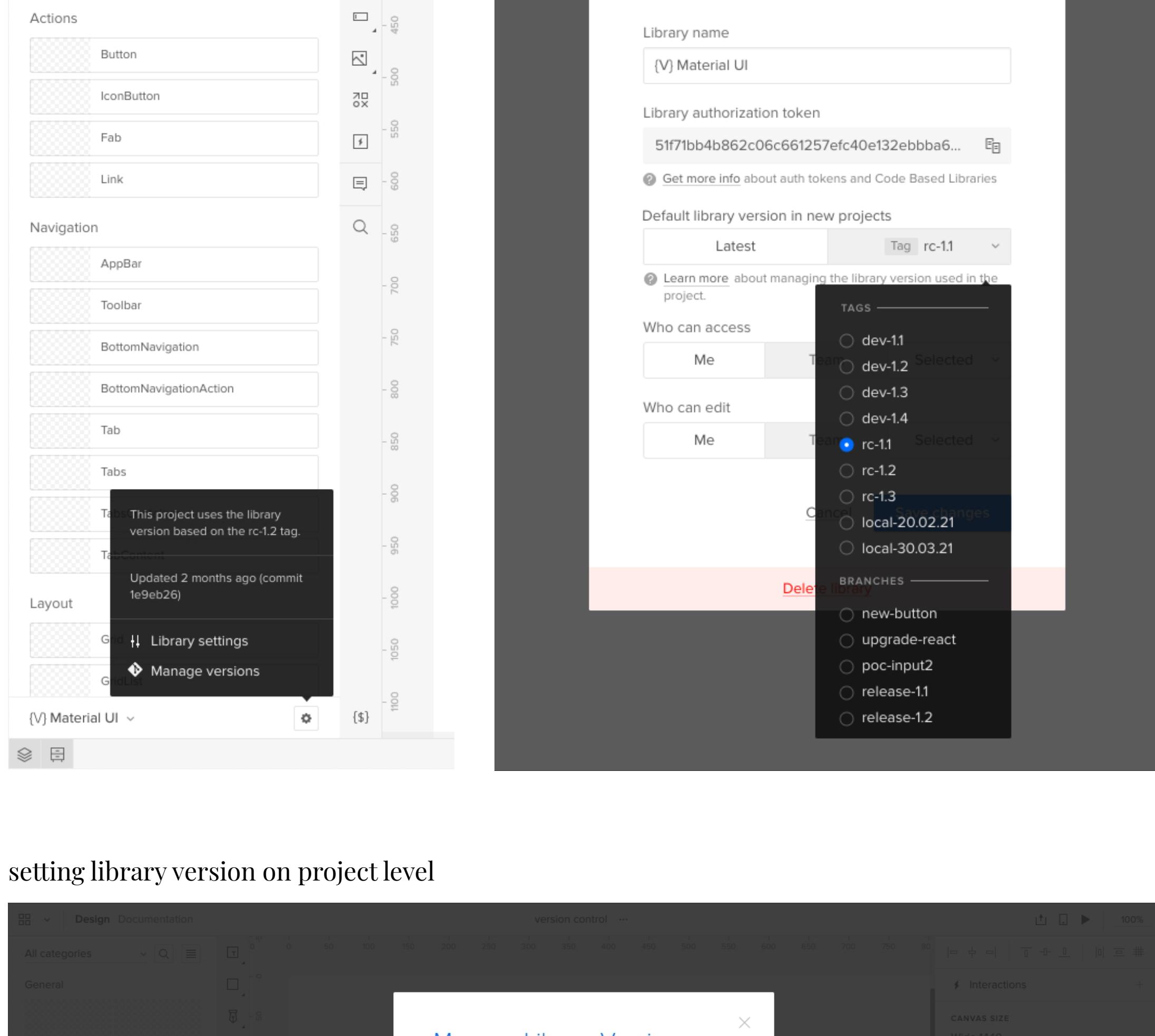
Merge Libraries Version Control

Based on feedback from our users, how they are implementing Merge to their organisation we realise there's need to handle multiple versions of the Merge library. Some cases was working on redesign, some was refactoring. Different projects might need different versions on components for testing purposes.

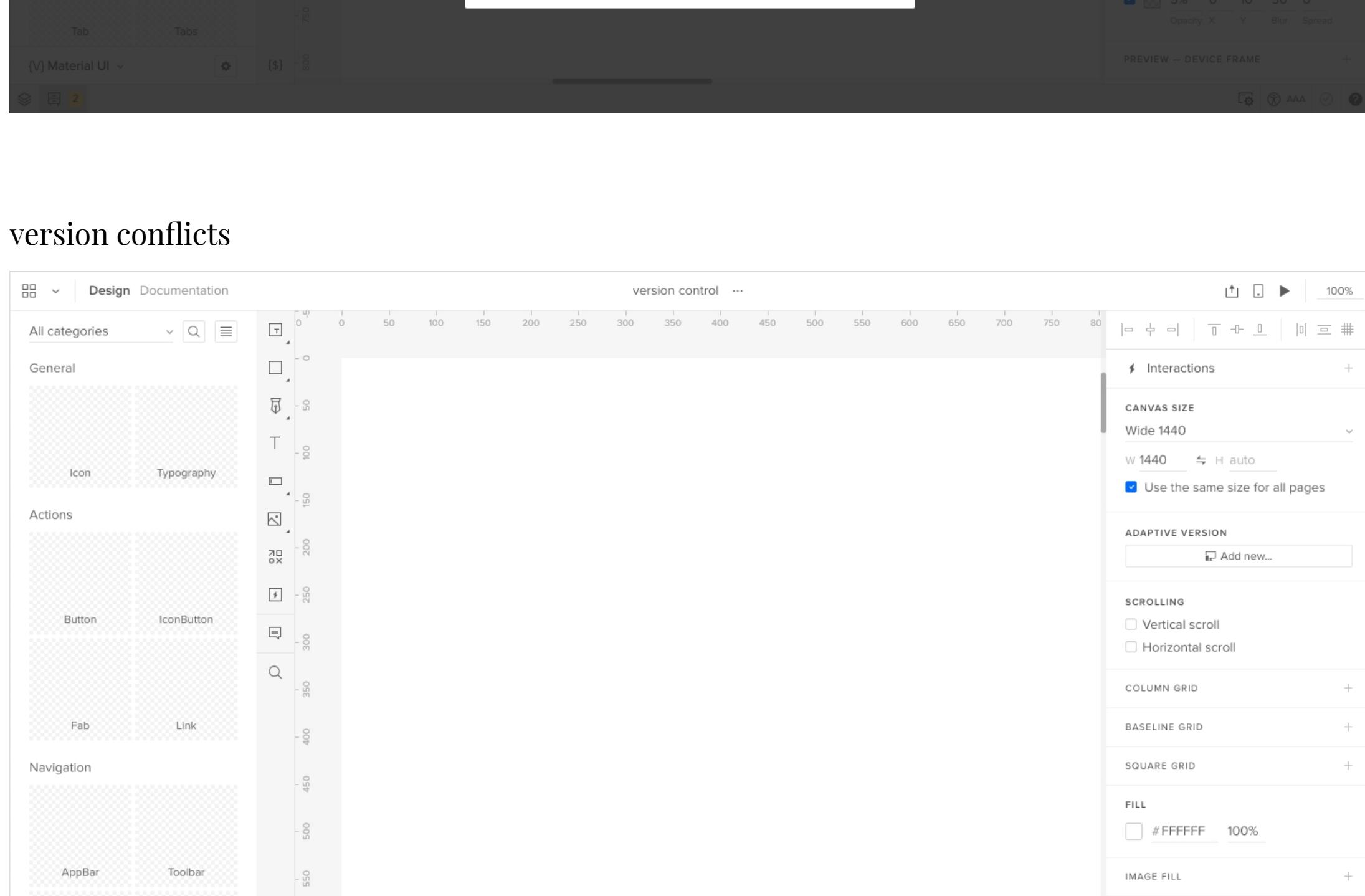
We cede to based on Git branches and tags.

2021 | UXPin
role :code + design

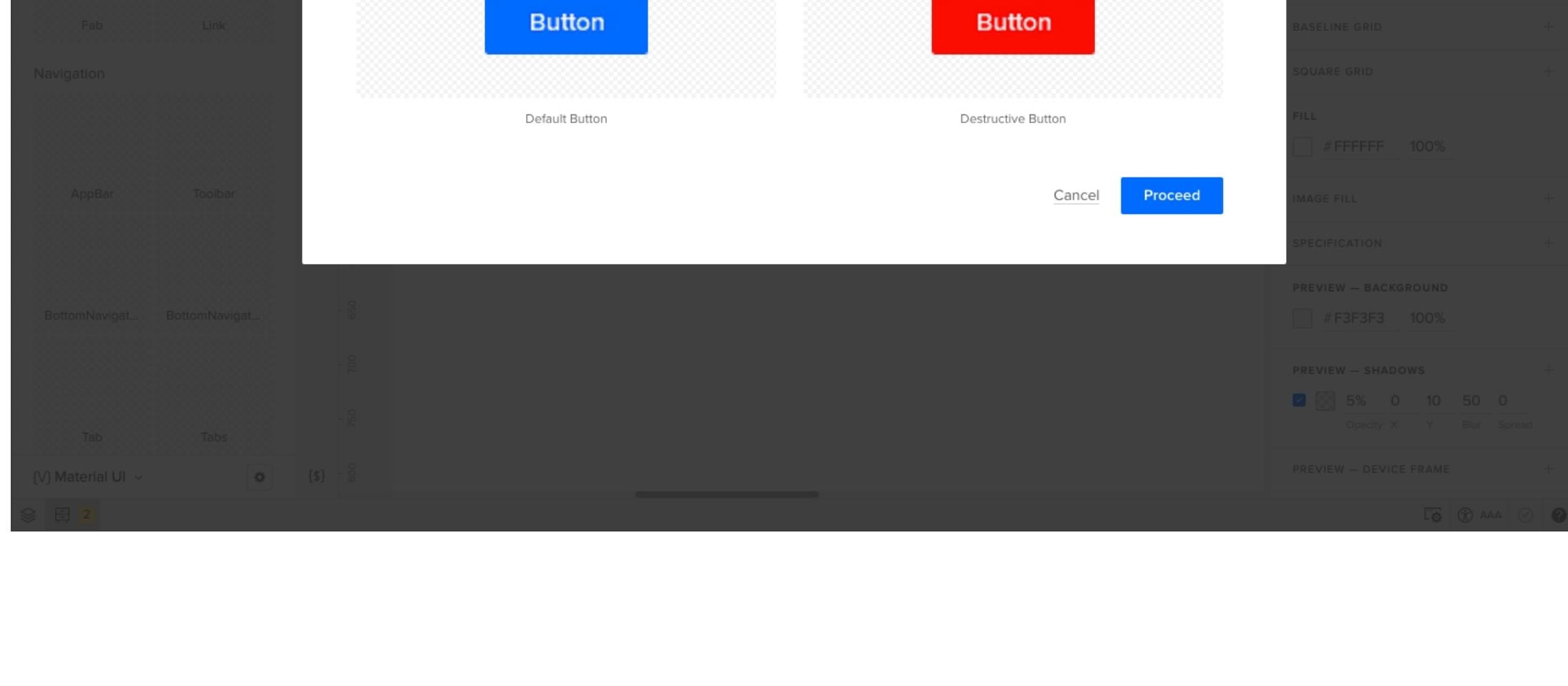
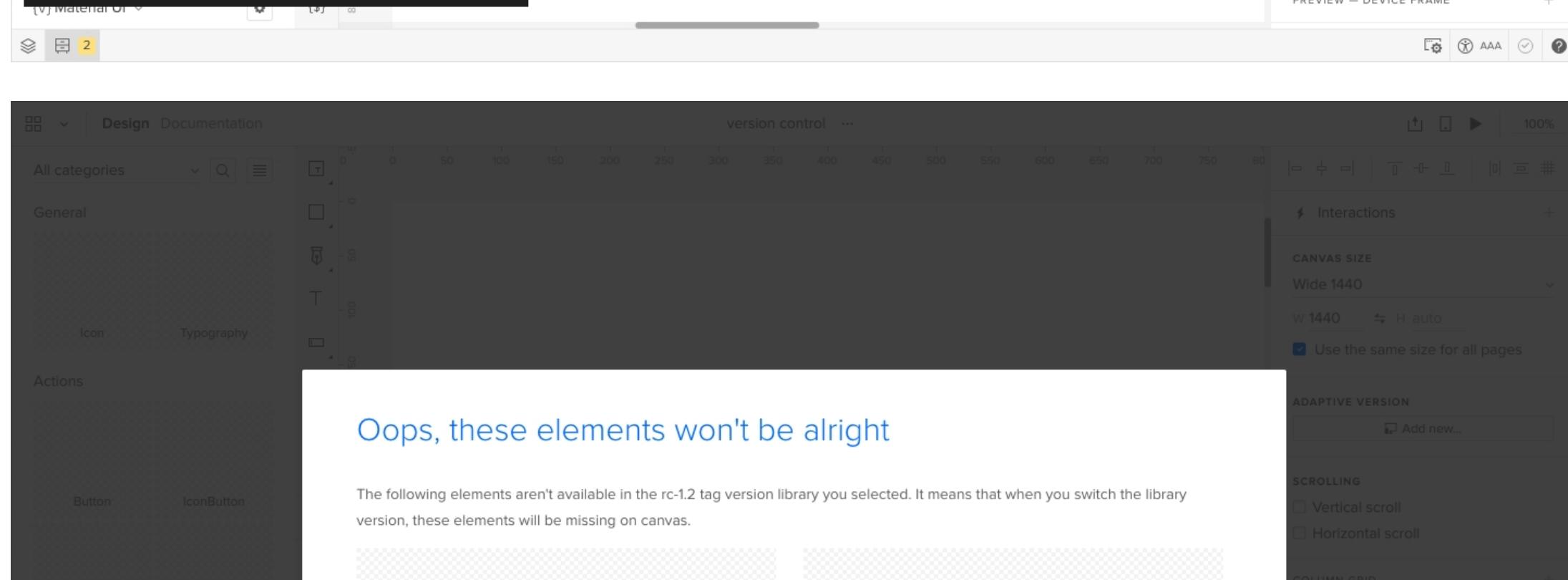
seeing default version of the library for all projects



setting library version on project level



version conflicts



Adele Design System Page

The project was related to creating a dedicated page for each of the design systems from the list. The goal was clear.- increase SEO on Adele to provide traffic to UXPin.

2020 | UXPin
role :design

The page features a large, bold title "The Github Primer" with "Github" above "Primer". Below the title, it says "design system is called". At the bottom left, there are three sections: "Technology", "Design", and "Documentation & guidelines". Each section has a table with various design system components and their status (checkmark or X). At the bottom right, there's a call to action for UXPin.

Technology

Code Depth	HTML/CSS
Components	✓
JS Library/Framework	X
TypeScript	X
Web Components	X
Tests	AVA
Linter	StyleLint
CSS	Sass
CSS In JS	X
Design Tokens	Scss
Bundle Manager	Primer Module Build Scriptno
Distribution	npm

Design

UI Kit	
Brand Guidelines	X
Color Naming	natural (e.g. gray-300)
Contrast Analysis	X
Typography	✓ Go to Typography
Icons	Octicons (SVG)
Space / Grid	✓ Go to Space / Grid
Illustration	X
Data Visualization	X
Animation	X
Voice & Tone	X

Documentation & guidelines

Accessibility Guidelines	X
Design Principles	X
Documentation Website	X
Code Documentation	Markdown
Storybook	✓ Go to Storybook

Evangelize Design System with a Free Template!

UXPin: Prototype and Manage Your Design System

40+ Slides ready for action. Perfect for team and stakeholder presentation. Available as a powerpoint and keynote deck.

Join thousands of companies (including PayPal, Sapient and HBO) and optimize your design process with UXPin.

[Learn more & download](#)

[Start a free trial now!](#)

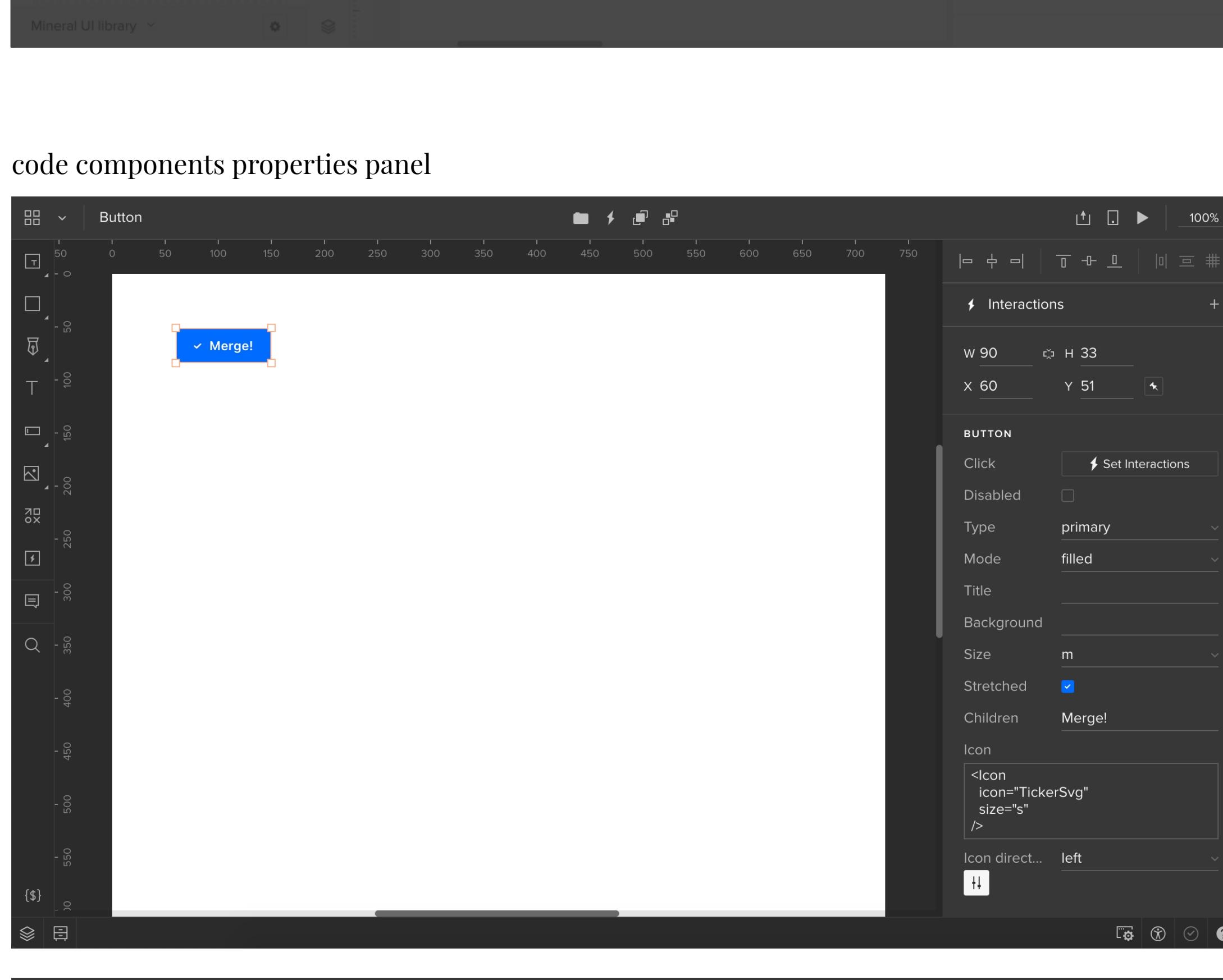
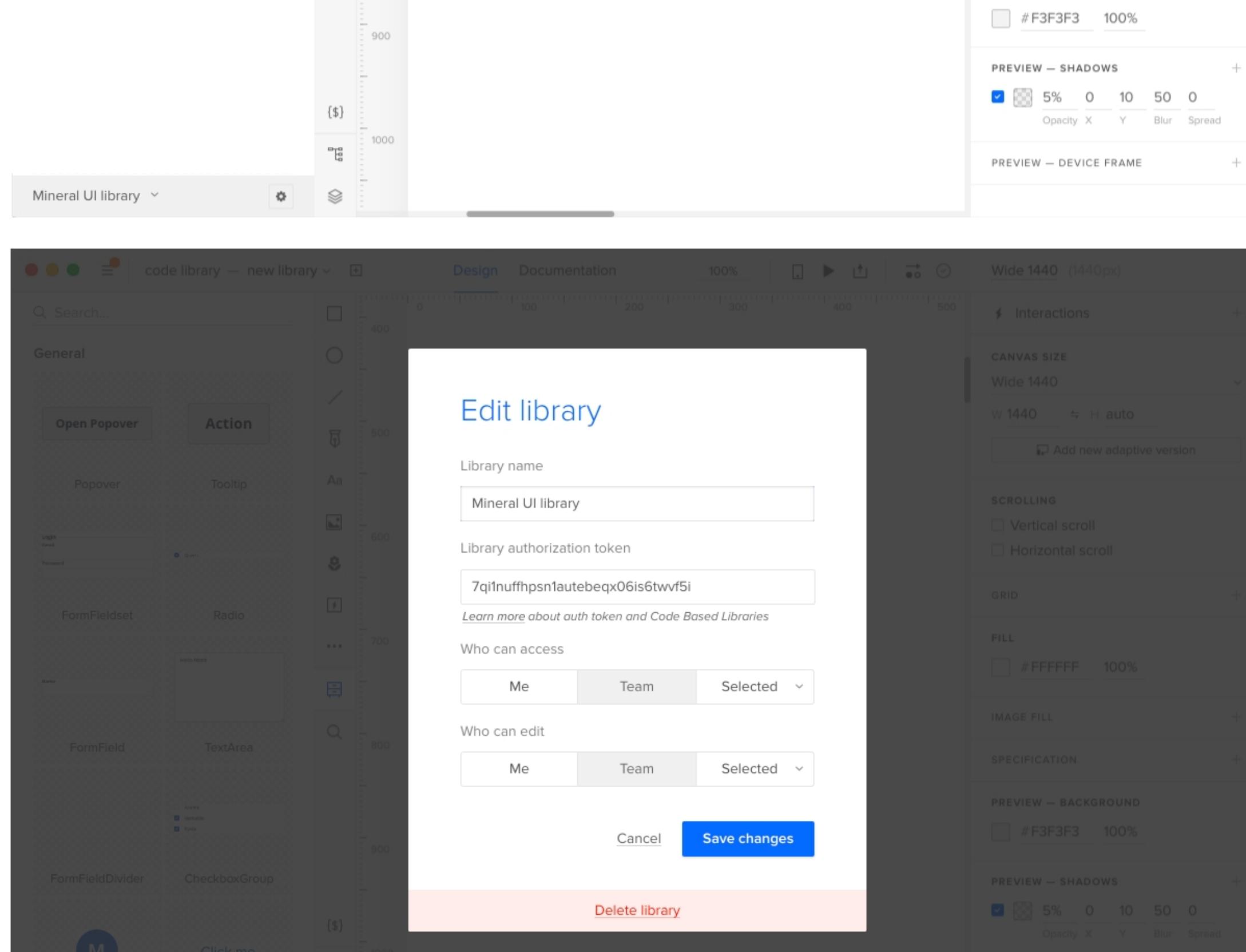
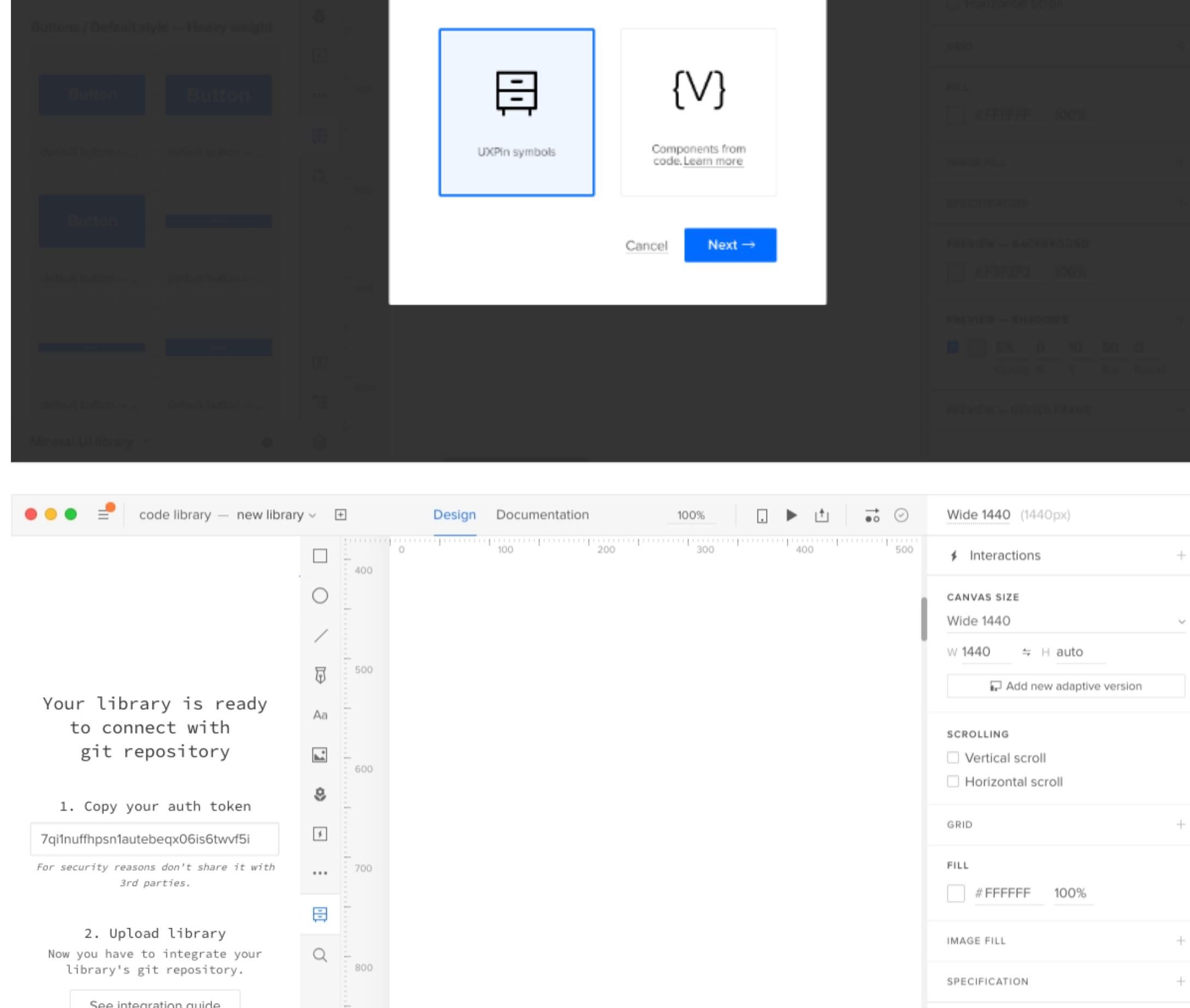
UXPin Merge

UXPin Merge is a technology that bridges the gap between design and technology. Technology gives the possibility to import real production react components into UXPin and use them in designs. This approach gives developers and designers the possibility to work on the same elements and use a single source of truth - the git repository. Check out more about {V} from one of UXPin's webinars.

When I was a Product Lead in the Merge project except scoping, creating product roadmap, and supporting early access users and I was responsible for a project user experience and the design.

2021 | UXPin
role :product lead / ux design

creating new library flow



code components properties panel



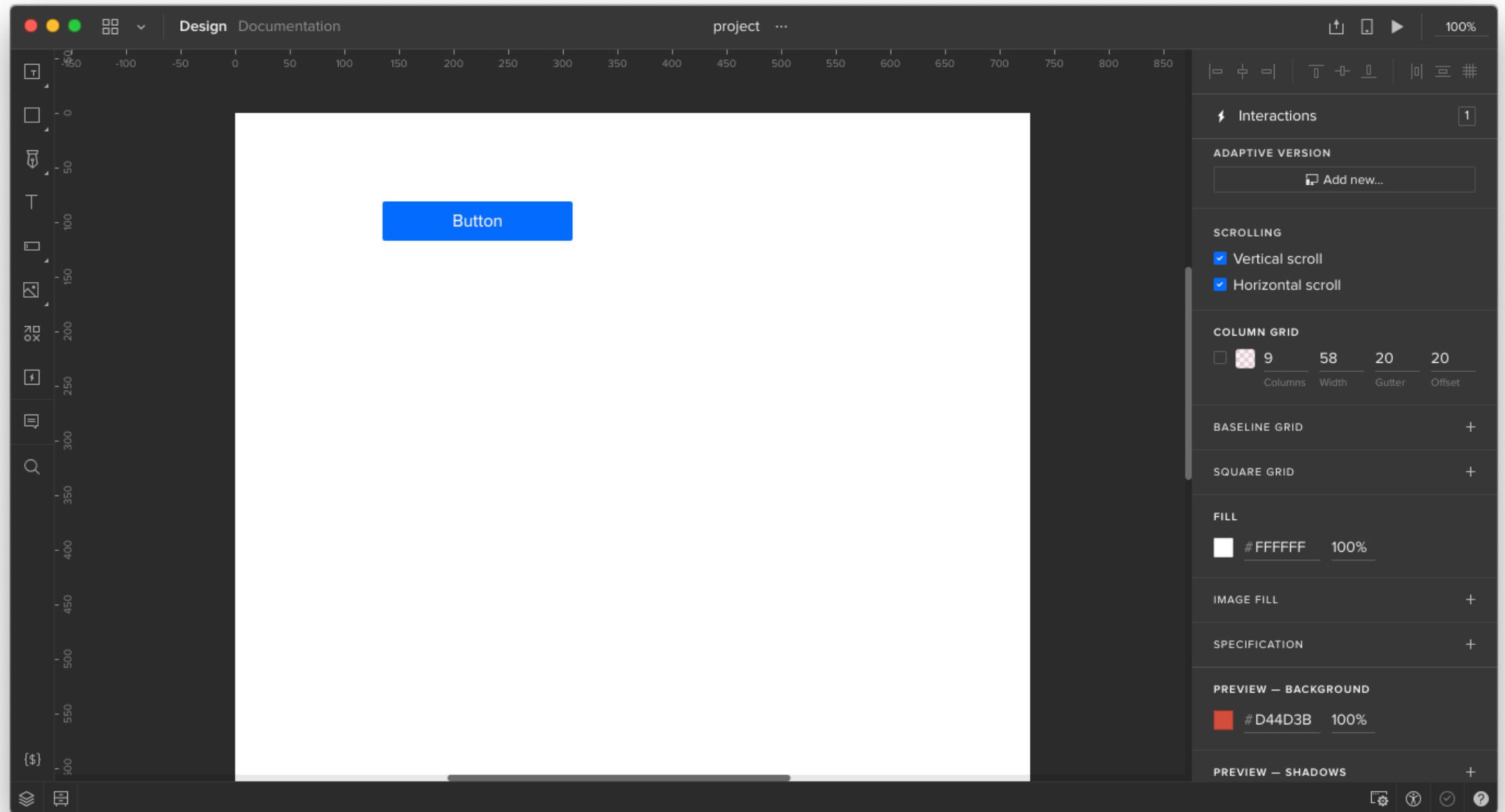
UXPin Desktop App

The project started after one of UXPin's internal hackathon. After POC done on the hackathon we decided to invest some additional time and release the desktop client. During this project, I was leading a team that builds the application. We used the electron.js library.

I was responsible for scoping, user experience, and coding.

2018 | UXPin

role :product lead / ux design + code

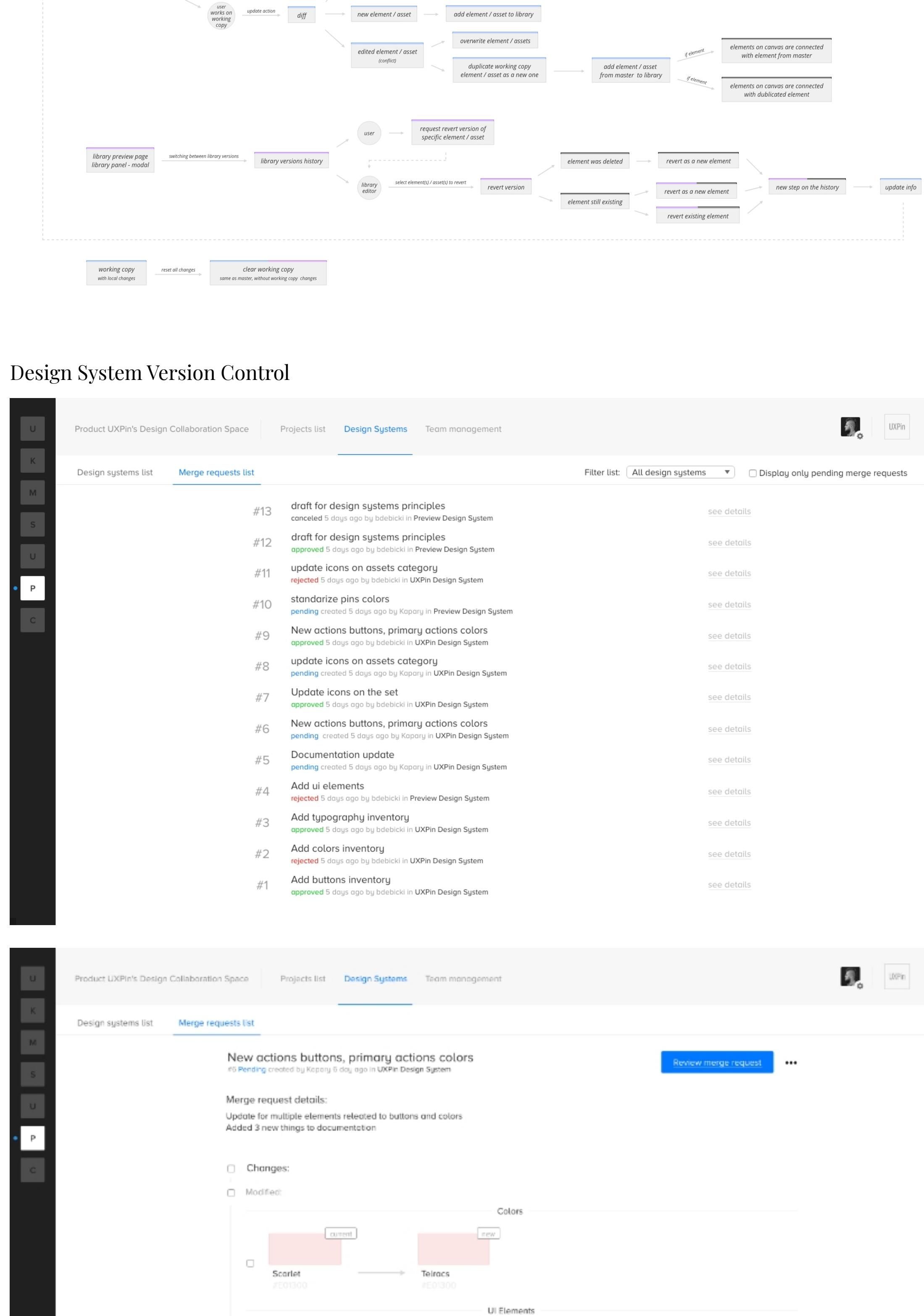


Right after the release of Design Systems and Design Libraries features in UXPin next step supposed to be version control of those areas. The project was very technical centered and complex because a lot of actions happened

The diagram illustrates the flow between the Editor and Dashboard areas. It shows three main components: **E** (Editor area), **D** (Dashboard / New view area), and **B** (Backend actions). The flow starts with **E** and **D** having their own internal actions. There is a double-headed arrow between **E** and **D**, indicating bidirectional communication or dependency. Below this, a horizontal arrow labeled *action* points from **E** to **D**, representing a specific action from the Editor to the Dashboard.

```
graph LR; E[E] -- "actions on Editor area" --> D[D]; D -- "actions on Dashboard / New view area" --> E; E <--> D; B[B] --- EA[Backend actions]; EA -- "User actions" --> D;
```

The diagram illustrates the relationship between a master version and a working copy. It consists of two main components: a 'master version' box on the left and a 'working copy' box on the right. An arrow points from the master version to the working copy, labeled with the text 'change some element / asset' above the arrow and 'add new element / asset' below it. Below this, another diagram shows a 'working copy' box on the left and an 'update info' box on the right, connected by an arrow labeled 'change some element / asset'.



- M
- S

New activity
#6 Approved by [REDACTED]

U

P

C

Details:
Update for multiple elements related to buttons and colors
Added 3 new things to documentation

Review details:
Approved 3 from 5 changes by bdebicki 2 days ago with a comment:
Good job. I picked only 3 changes because other wasn't related to buttons.

Approved changes:

Modified:

UI Elements

current new

Action button Action button

Buttons category description has been changed

current new

Material buttons trigger an ink reaction on press.
They may display **text or imagery**. Flat buttons
and raised buttons are the most commonly used
types.

Material buttons trigger an ink reaction on press.
They may display **text, imagery, or both**. Flat
buttons and raised buttons are the most
commonly used types.

display full description

UXPin Design System

Versions history

Add buttons inventory v5 current changes by Kopyny

Add typography inventory v4 changes by Kopyny

approved by bdebicki 6 days × merge request details × revert changes

Add typography general styles and headline styles

Prepare first documentation v3 changes by Kopyny

New actions buttons, primary actions colors v2 changes by Kopyny

draft for design systems principles v1 changes by Kopyny

Colors

Blue Ribbon #006cff Scarlet #e01300

Typography

Typeface

Playfair Display

Regular *Italic* **Bold**

The quick brown fox jumps over the lazy dog.

Lato

Light *Italic* Semibold **Bold**

The quick brown fox jumps over the lazy dog.

Text styles

Heading 1

Playfair Display Bold • 40px / 48px

Heading 2

Playfair Display Regular • 36px / 42px

Assets

Icons

Library Version Control

The screenshot shows the UXPin Design interface. At the top, there's a navigation bar with 'Design' selected. On the left, a sidebar lists various assets like 'Default library', 'Colors', 'All assets', 'ICONS', and 'Calculators'. The main canvas area displays a dark gray header with the word 'Playfair' in white. Below it, the text 'The quick brown fox jumps over the lazy dog.' is displayed in a large, bold, serif font. In the top right corner of the canvas, there are several UI elements: a 'New actions buttons, primary actions colors' card (v.2), a 'draft for design systems principles' card (v.1), and a 'UI Elements' card. The bottom right of the screen shows a 'Properties' panel with various styling options.

Pages: master +

Design Documentation Preview 100%

New actions buttons, primary actions colors
#6 Accepted by bdebicki 1 day ago / created by Kapary 6 day ago on UXPin Design System

Merge request details:
Update for multiple elements related to buttons and colors
Added 3 new things to documentation

Review details:
Accepted 3 from 5 changes by bdebicki 2 days ago with comment:
Good job. I picked only 3 changes because other wasn't related to buttons.

Approved changes:
Modified:

UI Elements

was it raining at open'er festival?

If you are a fan of the Open'er Festival or live in the Tricity Poland you probably know jokes about the weather during the festival. I decided to analyze historical data and refute or confirm the myth that it always rains during the Open'er. The project is still in development.

2017 | private project

role :design + code



was it raining at open'er festival? [2017](#) [2016](#) [2015](#) [2014](#) [2013](#) [2012](#)

2017
OPEN'ER FESTIVAL POWERED BY ORANGE
28th June - 1st July
Babie Doły Airport, Gdynia
RADIOHEAD / FOO FIGHTERS
THE WEEKEND / THE XX / LORDE
[see full lineup +](#)

RAIN **NO**

yep. this site use cookies. click [here](#) for more. made with ❤ in Gdynia 2017 — [about](#) [cookies](#)

was it raining at open'er festival? [2017](#) [2016](#) [2015](#) [2014](#) [2013](#) [2012](#) [2011](#) [2010](#) [2009](#) [2008](#) [2007](#) [2006](#) [2005](#) [2004](#) [2003](#) [2002](#)

2016
OPEN'ER FESTIVAL POWERED BY ORANGE
29th June - 2nd July
Babie Doły Airport, Gdynia
FLORENCE + THE MACHINE
RED HOT CHILI PEPPERS
LCD SOUNDSYSTEM / SIGUR RÓS
PHARRELL WILLIAMS / KYGO
[see full lineup +](#)

RAIN **YES**

yep. this site use cookies. click [here](#) for more. made with ❤ in Gdynia 2017 — [about](#)

was it raining at open'er festival? [2017](#) [2016](#) [2015](#) [2014](#) [2013](#) [2012](#) [2011](#) [2010](#) [2009](#) [2008](#) [2007](#) [2006](#) [2005](#) [2004](#) [2003](#) [2002](#)

LINEUP 2016

FLORENCE + THE MACHINE
RED HOT CHILI PEPPERS
LCD SOUNDSYSTEM / SIGUR RÓS
PHARRELL WILLIAMS / KYGO

AT THE DRIVE-IN | BASTILLE | BEIRUT | CARIBOU
CHVRCHES | DAWID PODSIADEŁO | FOALS | GRIMES
KURT VILLE & THE VIOLATORS | M83 | MAC DEMARCO
PAUL KALKBRENNER | PJ HARVEY | SAVAGES
SKEPTA | TAME IMPALA | THE 1975
THE LAST SHADOW PUPPETS | WIZ KHALIFA

CATZ 'N DOGS | DJ TENNIS | FLIRTINI | JACEK SIENKIEWICZ | KAMP! | KORTEZ
LUNICE | ŁONA | WEBBER | MAŁPA | MARIA PESZEK | NOTHING BUT THIEVES
PTAKI | RASMENTALISM | REBEKA | RYSY | SECTION BOYZ
WINCE STAPLES | XANAXX | ZBIGNIEW WODESKI with MITCH & MITCH ORCHESTRA
AN ON BAST | BAASCH | CHINO | COLDAIR | CZELUŚC LIUTRO X KOSA | GROBEL | HANA
HEROES GET REMEMBERED | JAAJAI | JÓGA | KUBA SOJKA | KROKI | LXMP | NAGROBKI
OLIVIER HEIM | OTSCHODZI | PIOTR BEJMAR | PIOTR ZIOŁA | SHY ALBATROSS | SONAR | SOOM T
SPOKEN LOVE | STARA RZEKA | SUUMO | T'NIEN LAI | TERRIFIC SUNDAY | V/O | WE DRAW A | ZAMILSKA

RAIN

29th June — yes
30th June — yes
1st July — no
2nd July — yes

yep. this site use cookies. click [here](#) for more. made with ❤ in Gdynia 2017 — [about](#)

Spoiler alert: for 14 of 18 editions of the festival rain was noticed]:->

Simplify editor

In 2017 in UXPin we did a huge usability test with our users about the condition of our application. Based on the research we decided to prepare a package of improvements in working with elements on the canvas and small redesign of UI.

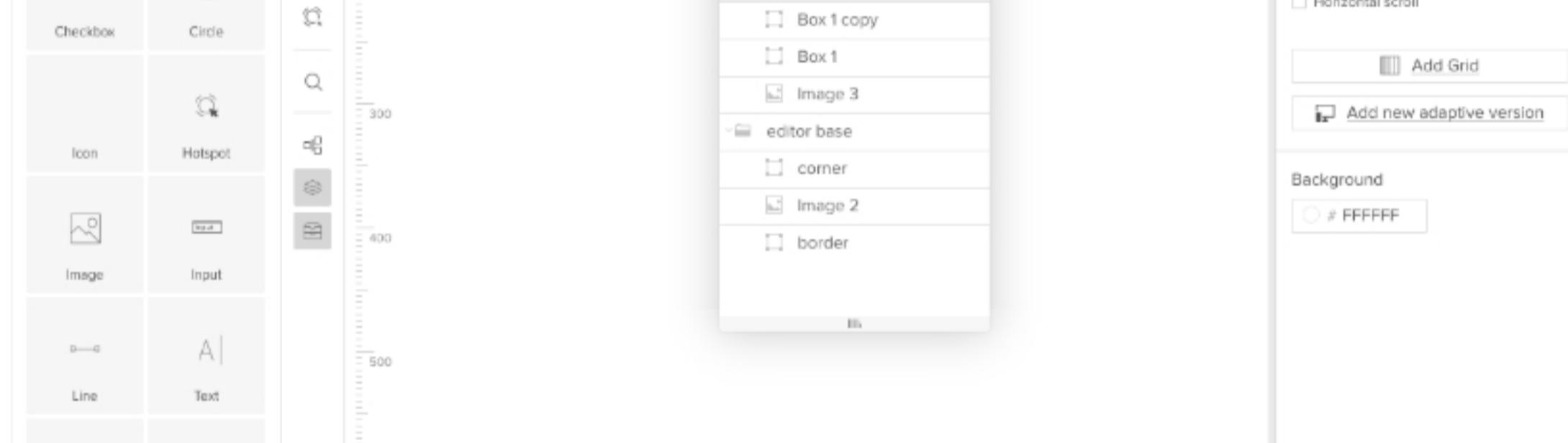
UI changes contain redesign of the top bar and release toolbar with quick access to the most popular elements and floating layers panel. Interactions flow was also simplified and enriched by predefined actions (based on the most popular ones).

2017 | UXPin
role :product lead / ux design

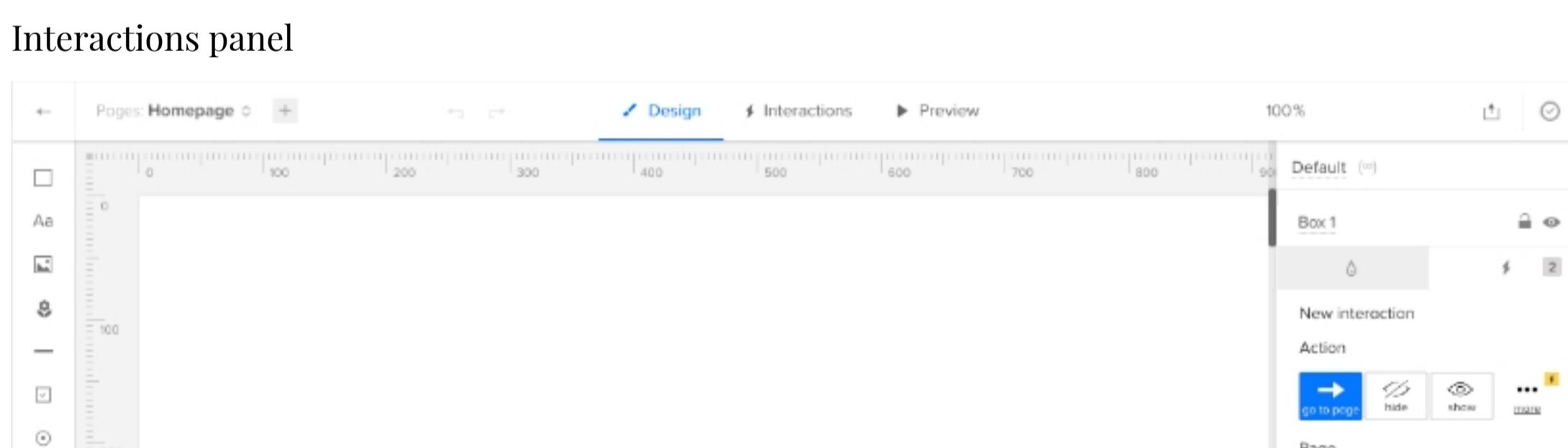
Top bar and toolbar



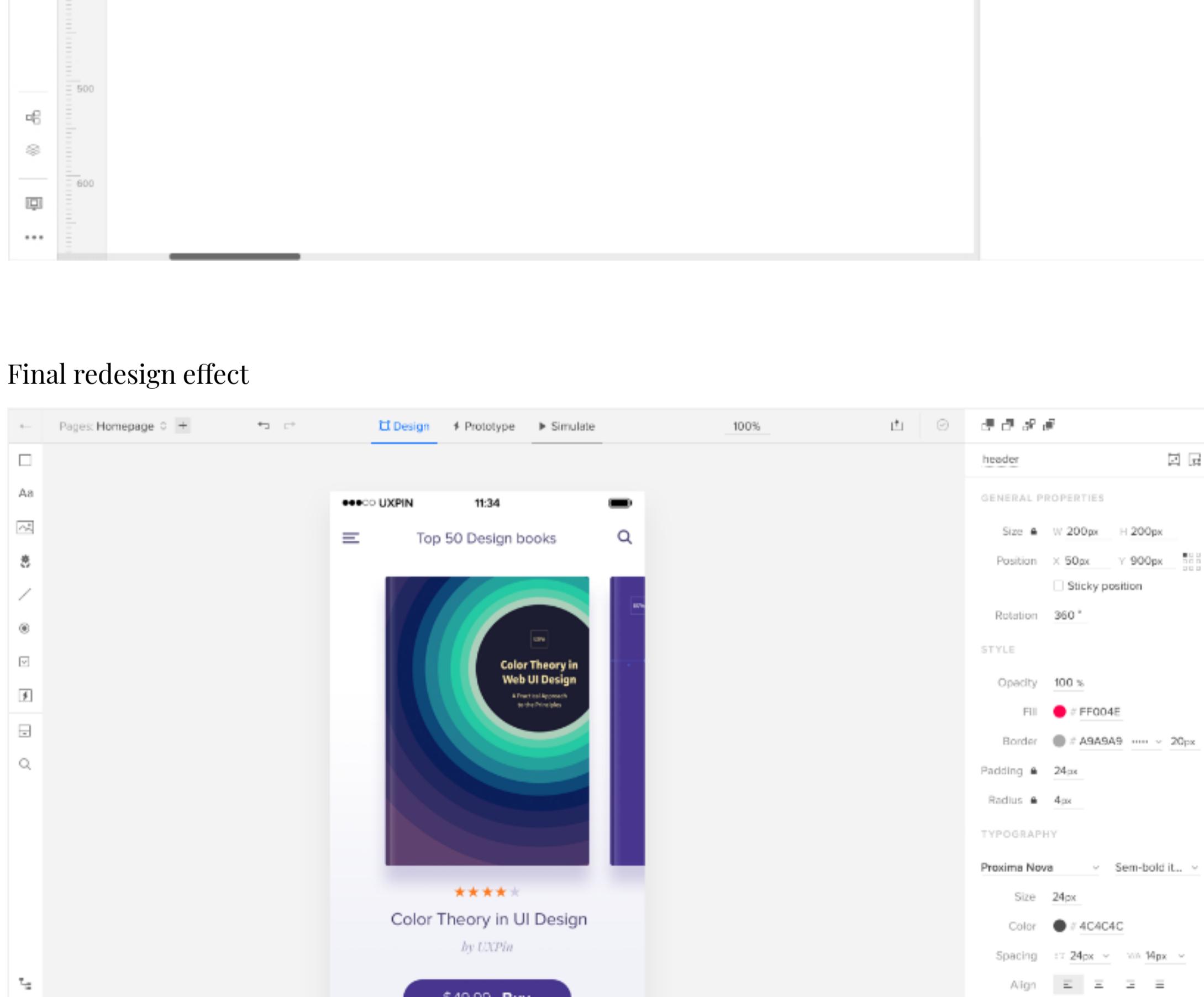
Left panels and floating layers



Interactions panel



Final redesign effect

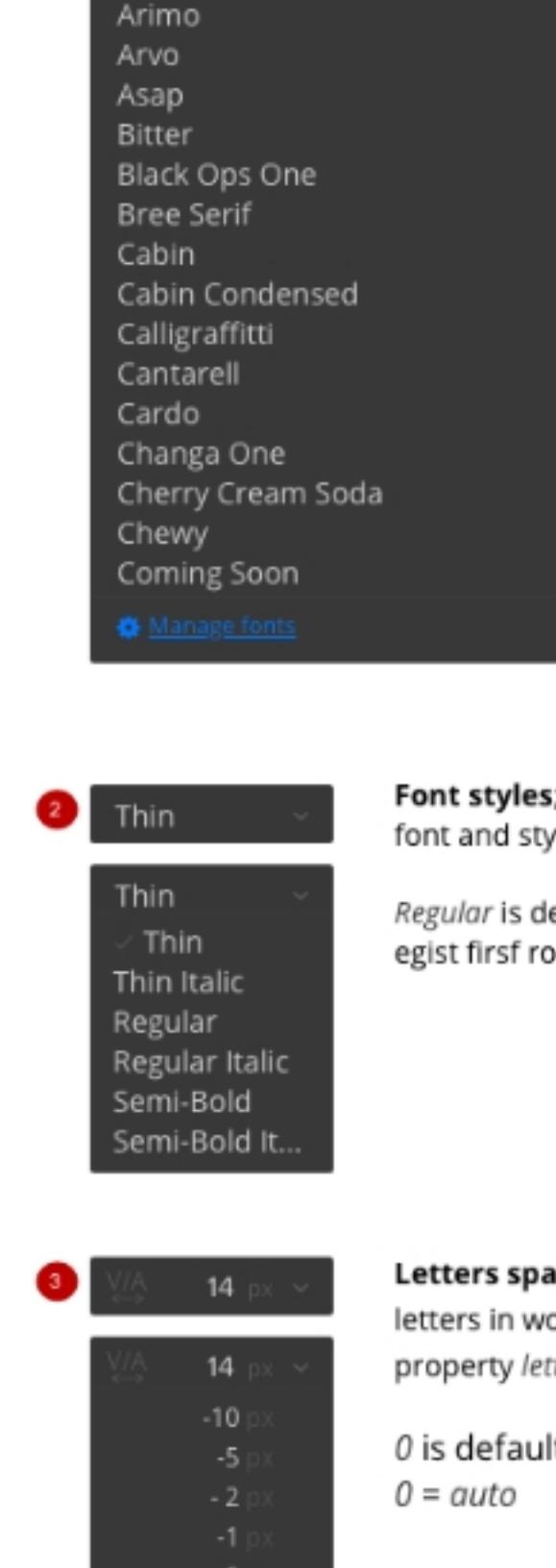
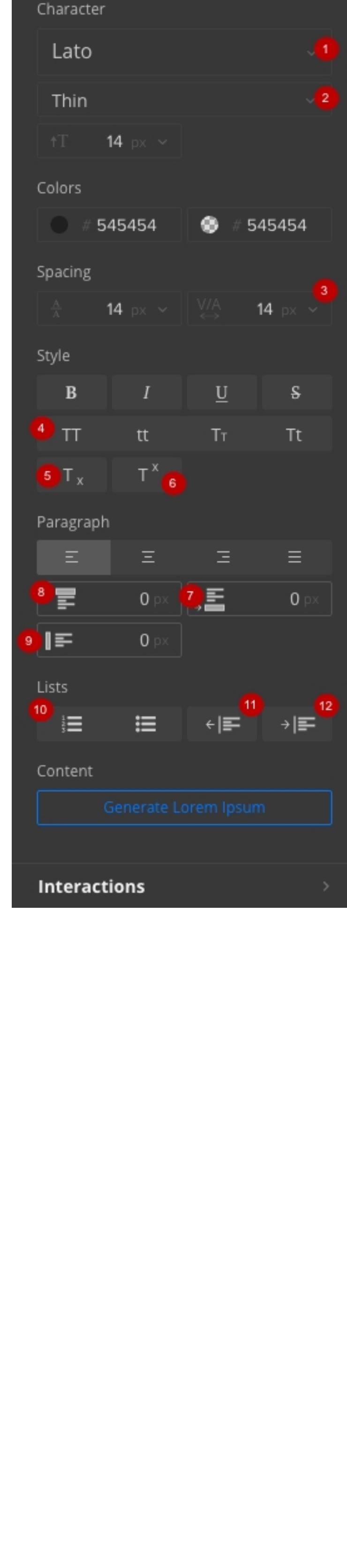
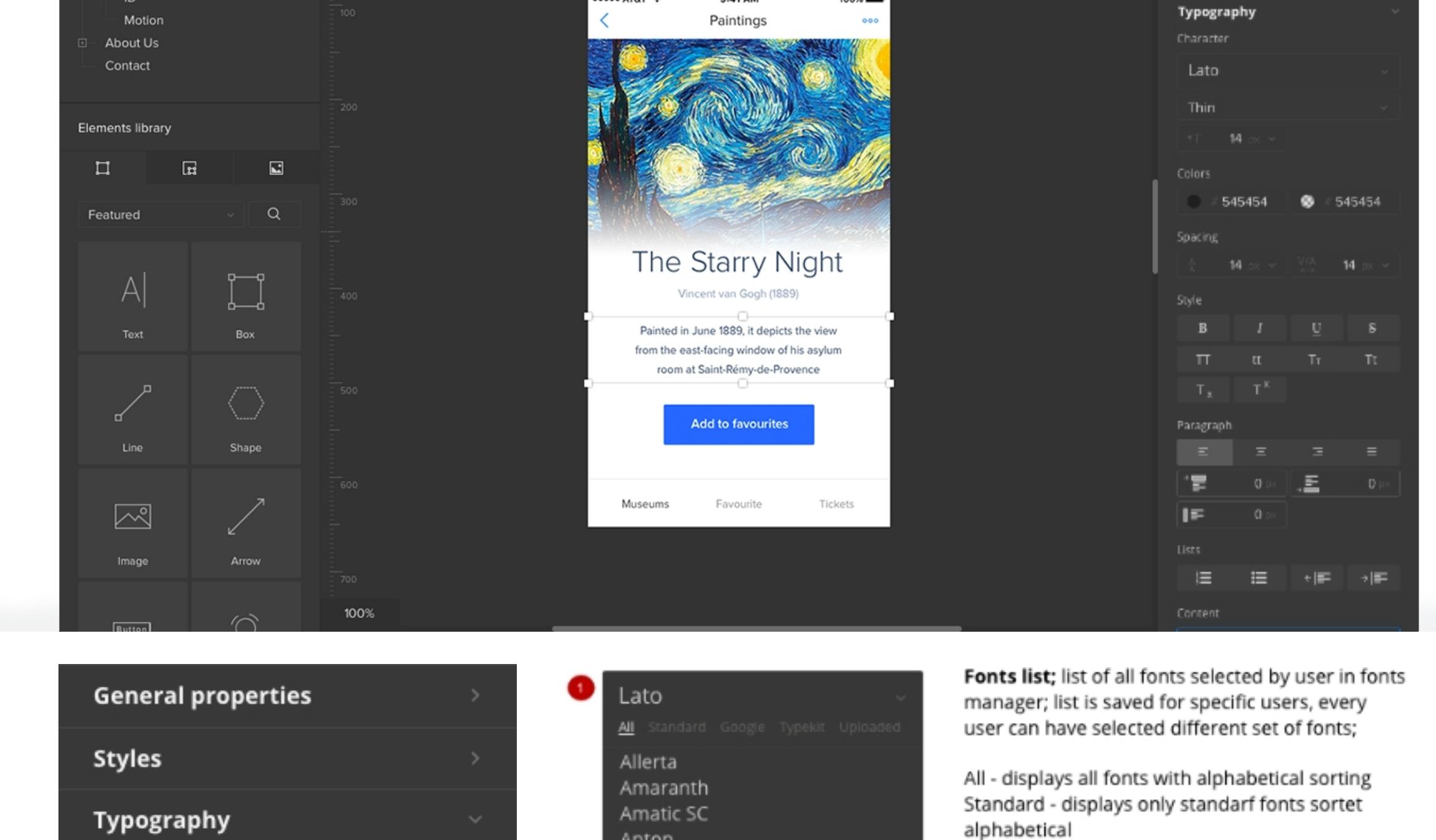


Advanced typography

The project concentrate on improve UXPin's text element. Whole work has been split into 4 releases. Text element property improvements (letter spacing, text transformations, paragraph spacing, and lists), integration with Google fonts, Typescript integration, and upload own fonts.

2016 | UXPin
role :product lead / ux design

Text element properties



Fonst list; list of all fonts selected by user in fonts manager; list is saved for specific users, every user can have selected different set of fonts;

All - displays all fonts with alphabetical sorting
Standard - displays only standard fonts sorted alphabetical
Google - displays only fonts from google fonts
Typekit - displays only fonts from typekit
Uploaded - displays fonts uploaded by user

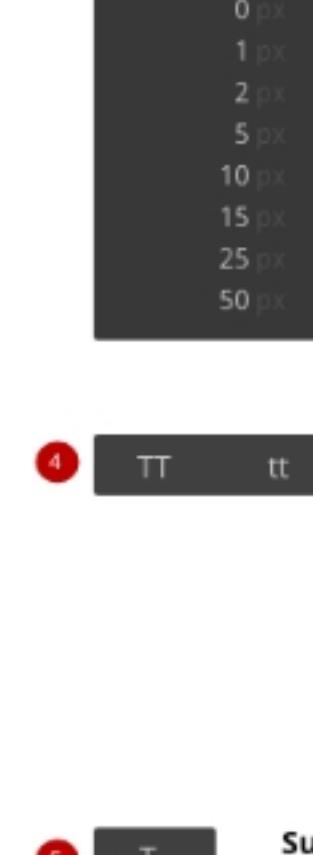
Selected category in list should be saved

By click in input with name of font you can type font name and search it on list.
(ex. <http://recordit.co/Zsi1ULnHKa>)



Font styles; list based on styles of font and styles added by user

Regular is default; if Regular doesn't exist first from top is default



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto



Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto

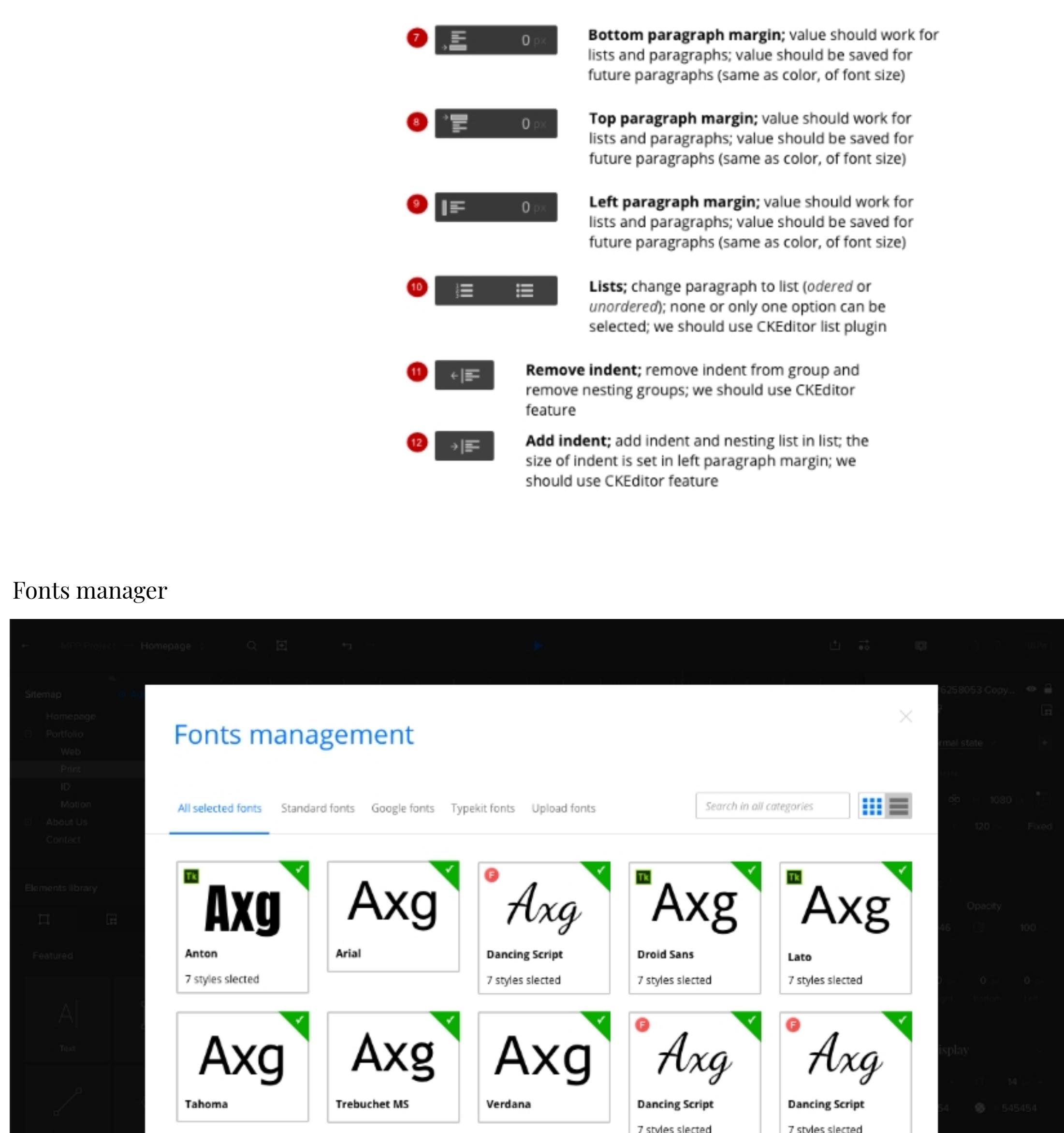


Letters spacing; space between letters in words; based on css property *letter-spacing*

0 is default

0 = auto

Fonst manager

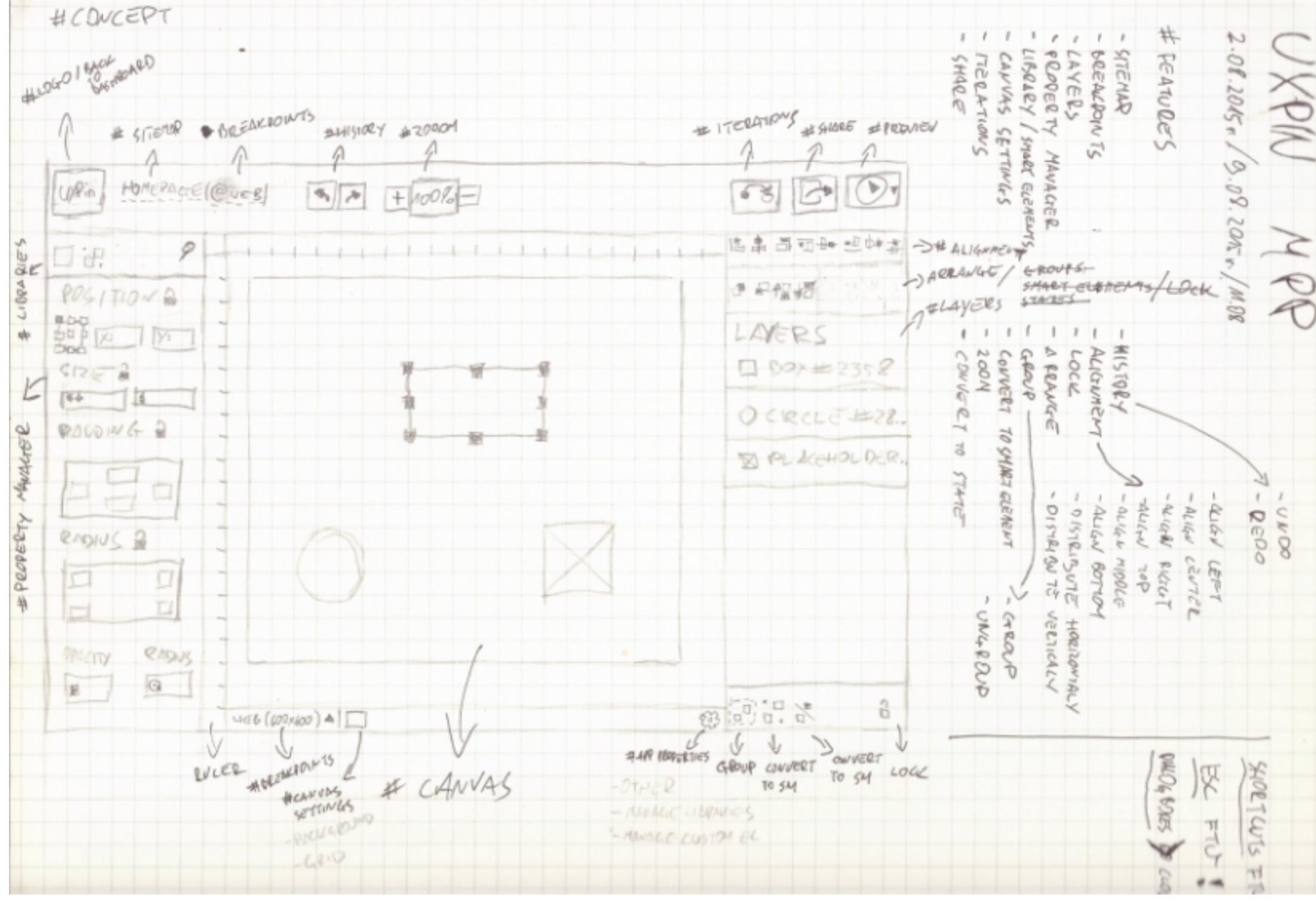


UXPin 3.0

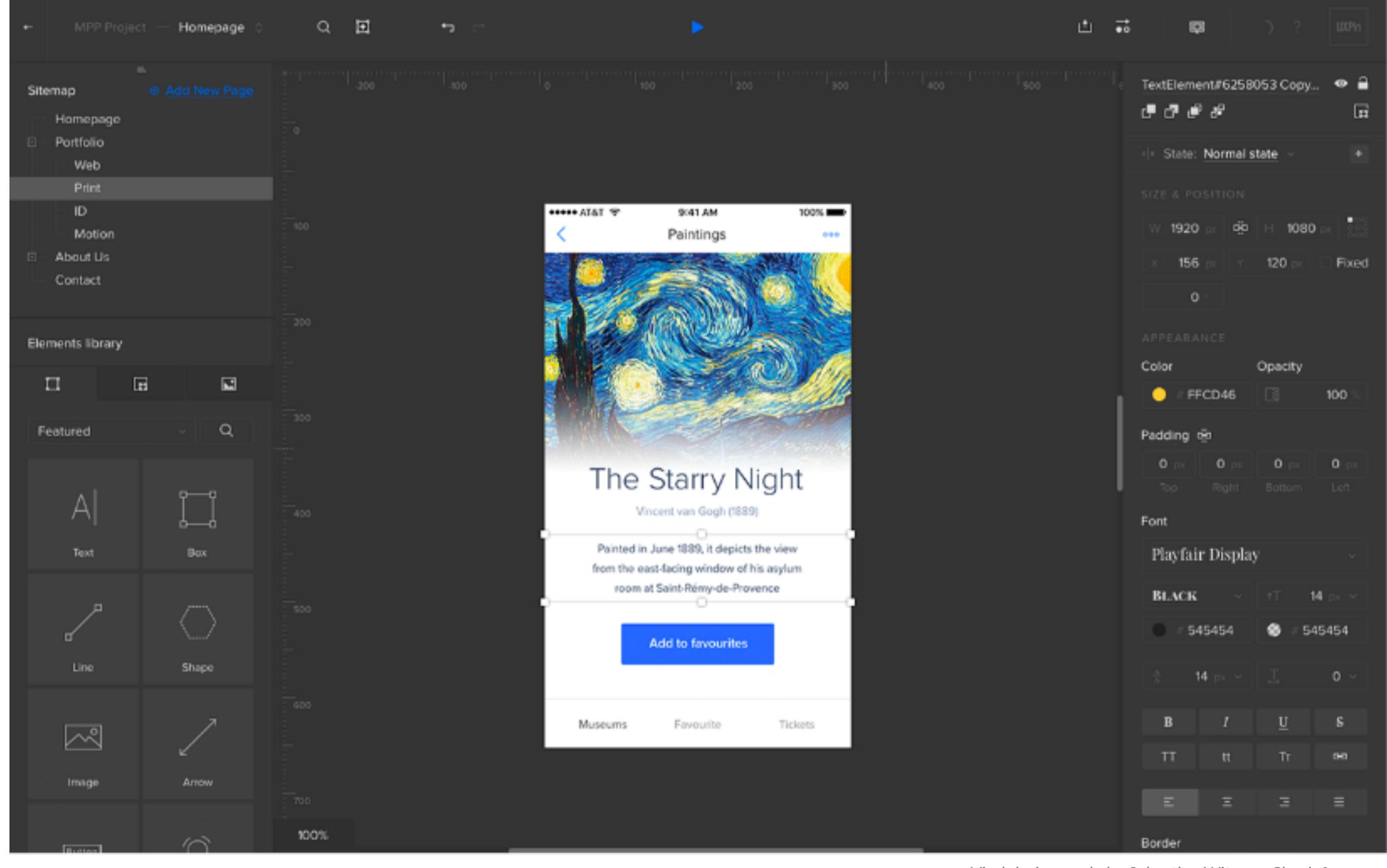
The project started by 1 email sent by CEO to some folks from the product team - "MPD - Minimum Pride Product". Then it happened. We started working on a new interface for the UXPin's editor. The time was perfect because we were rewriting UXPin's heart - the engine. Collectively we designed the new editor. I was a product lead who was leading a team in the development. Besides that, I was analyzing beta User's feedback and creating scope for future sprints.

2015 | UXPin
role :product lead / ux design + code

To one of corners for the MPP Editor.



Final design for UXPin 3.0



Visal design made by Sebastian Witman. [Check for more.](#)