

Selected projects 2015-2023

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)

Merge properties panel unification

From feedback from our users, we find out that properties panel for merge components is not clear to them. The biggest issue was that they had to learn new, different concept than for regular components.

On the other hand we would like to create experience that makes feel that Merge components are something natural and consistent with UXPin editor. We decided to unify properties panel and align Merge components to use same concept as UXPin's classic elements.

2023 | UXPin
role :design

Previous and unified properties panel

The image shows two side-by-side screenshots of the UXPin editor interface, illustrating the evolution of the Merge components properties panel. On the left, labeled 'old Merge components properties panel', the interface is divided into several sections: 'Interactions' (with a plus icon), 'W 89 H 37 X 1281 Y 181' (with a move icon), 'AUTO LAYOUT' (with a plus icon), and a large 'BUTTON' section containing a 'Label' input field ('Button'), 'Color' dropdown ('primary'), and various state checkboxes ('Disabled', 'Disable el...', 'Full width', 'Size', 'Variant'). Below these are 'Href', 'Start icon', 'End icon', 'Click' (with a 'Set Interactions' button), and 'Sx'. On the right, labeled 'unified Merge components properties panel', the interface is simplified. It retains the 'Interactions' and coordinate fields, but the 'LABEL' and 'COLOR' fields are now part of a single 'LABEL' section. The 'DISABLED', 'SIZE', 'VARIANT', 'HREF', 'START ICON', 'END ICON', and 'CLICK' fields are also grouped together. A 'Bring to front Box' button is present under the 'CLICK' section. At the bottom of both panels are standard UXPin controls: gear, user, AAA, checkmark, and question mark.

Merge components sections states

This section displays a grid of Merge component sections, each with its state variations and associated UI behaviors:

- collapsed**: null / undefined / "" (string) / false (boolean) property value. Shows a 'textfield control' section with a 'LABEL' input field ('Button').
 - click on plus: showing textarea and focus in field
 - click on clear: clean property value and hide textarea
- expanded**: with value / true (boolean) property. Shows a 'select control' section with a 'COLOR' dropdown ('primary').
 - click on plus: showing select; should be open and display available options
 - click on clear: clean property value and hide select
- boolean control**: BooleanPropertyHandler. Shows a 'boolean control' section with a 'DISABLED' switch ('True False').
 - click on plus: show switcher with selected true value
 - click on clear: set up false value and hide switcher
- string control**: JSONStringPropertyHandler. Shows a 'string control' section with an 'HREF' input field.
 - click on plus: show input and focus field
 - click on clear: clean up value and hide input
- function control**: FunctionPropertyHandler. Shows a 'function control' section with a 'CLICK' button.
 - click on plus: opens add new interaction form
 - click on interaction: opens edit interaction form
 - click on clear: remove interaction
- code control**: CodePropertyHandler. Shows a 'code control' section with an 'SX' input field.
 - click on plus: show code control with opened code editor
 - click on clear: clean up value and hide control
- color control**: ColorPropertyHandler. Shows a 'color control' section with a 'COLOR' picker (#FFC7C7).
 - click on plus: show color control with opened color picker
 - click on clear: clean up value and hide control
- PROPERTY HAS DESCRIPTION**: Shows a 'DISABLE ELEVATION' switch ('True False').
 - if property has description:
 - property heading has dotted underline
 - cursor when hovering over a heading is question mark (help)
 - description is displayed in a Tooltip component; default position of the tooltip should be top
- DEFAULT VALUE HAS BEEN OVERWRITTEN**: Shows a 'TITLE' input field ('Toggle element visibility').
 - after click in reset icon: value should be reset to default
 - control could be collapsed if there was no value or default value was false (for boolean)
- default is empty**: Shows an 'ERROR MESSAGE' input field.
- default has value**: Shows an 'ERROR MESSAGE' input field.

Information about component's library in properties panel

A screenshot of the UXPin editor interface showing the component library. The library is a large, empty list area with a dark header bar. At the bottom of the screen, the standard UXPin toolbar is visible, including icons for gear, user, AAA, checkmark, and question mark.

UXPin Design System

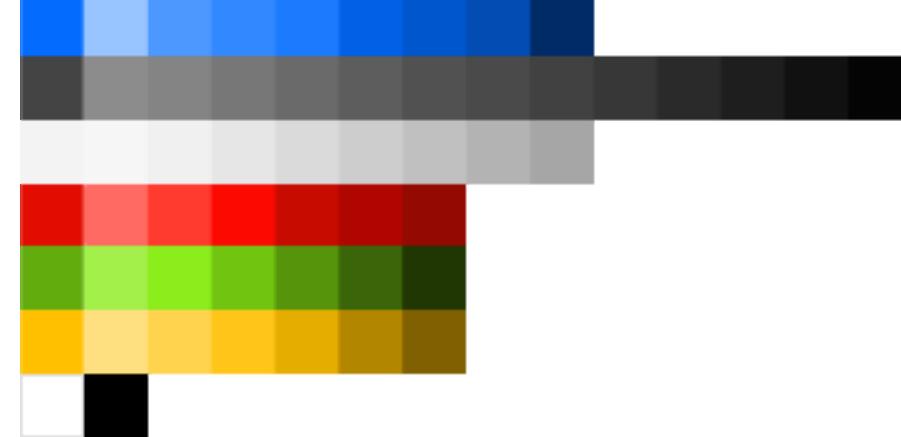
Basing on the actual design of the product I have prepared a design system. The goal was to clean and organize and simplify colors, typography, iconography, and interface elements. During this work, I was in close contact with the design and development team.

2017 - 2022 | 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

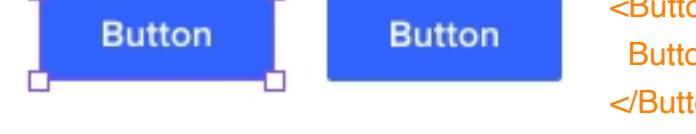
In UXPin was 291 used icons. A lot of those icons were duplicate and similar. During cleaning, I decrease the number of icons to 172.



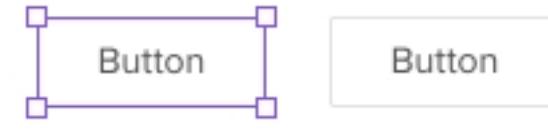
components

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 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

The screenshot shows the UXPin interface. On the left, the 'Design' tab is selected, displaying a library catalog with categories like General, Actions, Navigation, Layout, and Tabs. A tooltip on the 'Tabs' category indicates it uses the 'rc-1.2' tag. In the center, a modal window titled 'Edit library' is open, showing fields for 'Library name' (set to 'Material UI') and 'Library authorization token'. It also displays a dropdown for 'Default library version in new projects' set to 'Latest' and a 'Tag' dropdown set to 'rc-1.1'. Below these are sections for 'Who can access' (Me) and 'Who can edit' (Me). A sidebar on the right lists 'TAGS' (dev-1.1, dev-1.2, dev-1.3, dev-1.4, rc-1.1, rc-1.2, rc-1.3, local-20.02.21, local-30.03.21) and 'BRANCHES' (new-button, upgrade-react, poc-input2, release-1.1, release-1.2). At the bottom of the modal, there are 'Delete' and 'Save' buttons. On the far right, the 'version control' tab is active, showing a 'Manage Library Versions' dialog. This dialog shows the current tag as 'rc-1.1' and a list of available tags: dev-1.1, dev-1.2, dev-1.3, dev-1.4, rc-1.1, rc-1.2, rc-1.3, local-20.02.21, local-30.03.21. It also lists 'BRANCHES' (new-button, enums, upgrade-react) and includes a note about selecting a branch vs a tag. The 'Save' button is at the bottom right of the dialog.

setting library version on project level

This screenshot shows the 'Manage Library Versions' dialog from the previous image, centered over the main design canvas. The dialog title is 'Manage Library Versions' and it states 'The library in this project uses version from the tag rc-1.1.' It has two options: 'Always use the latest version.' (unchecked) and 'Select version:' (checked). Under 'Select version:', there are two tabs: 'TAGS' (which shows dev-1.1, dev-1.2, dev-1.3, dev-1.4, rc-1.1, rc-1.2, rc-1.3, local-20.02.21, local-30.03.21) and 'BRANCHES' (which shows new-button, enums, upgrade-react). Below the tabs, there is a note: 'Heads up: If you select a 'Branch', your prototype will always update whenever someone changes the 'Branch'. Want to avoid that? Choose 'Tag' instead.' At the bottom are 'Cancel' and 'Save' buttons.

version conflicts

This screenshot shows the 'Design' tab with a warning message: 'Some Merge components used in this project are unavailable in the selected library version – rc-1.2.' The message lists missing elements: 'Default Button' and 'Destructive Button'. It also shows a preview of two buttons: 'Default Button' (blue) and 'Destructive Button' (red). At the bottom right of the preview area is a 'Proceed' button. The right side of the screen shows the 'version control' tab with its own configuration dialog, similar to the one in the first screenshot.

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" in a smaller font 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	
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

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

[Learn more & download](#)

[Start a free trial now!](#)

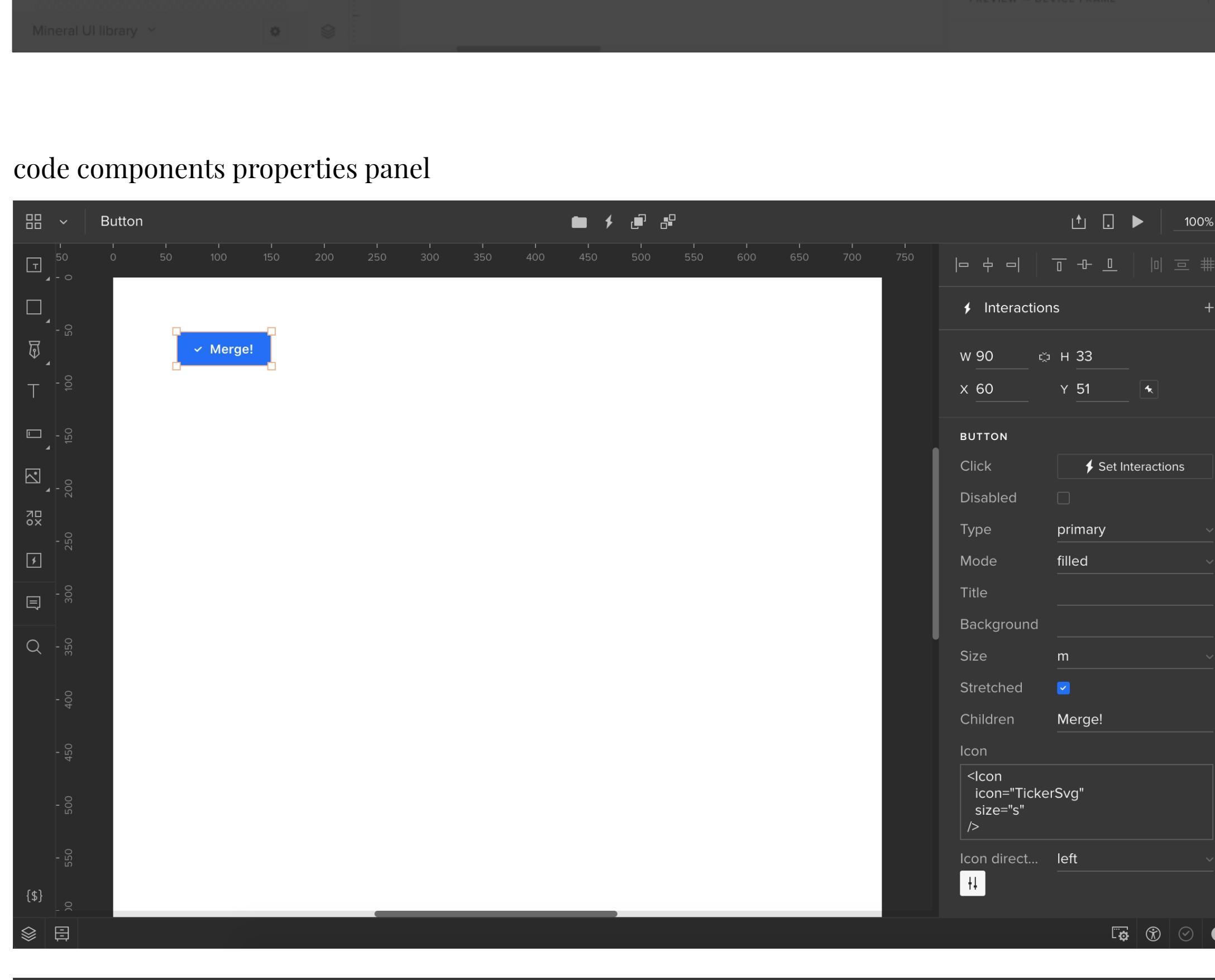
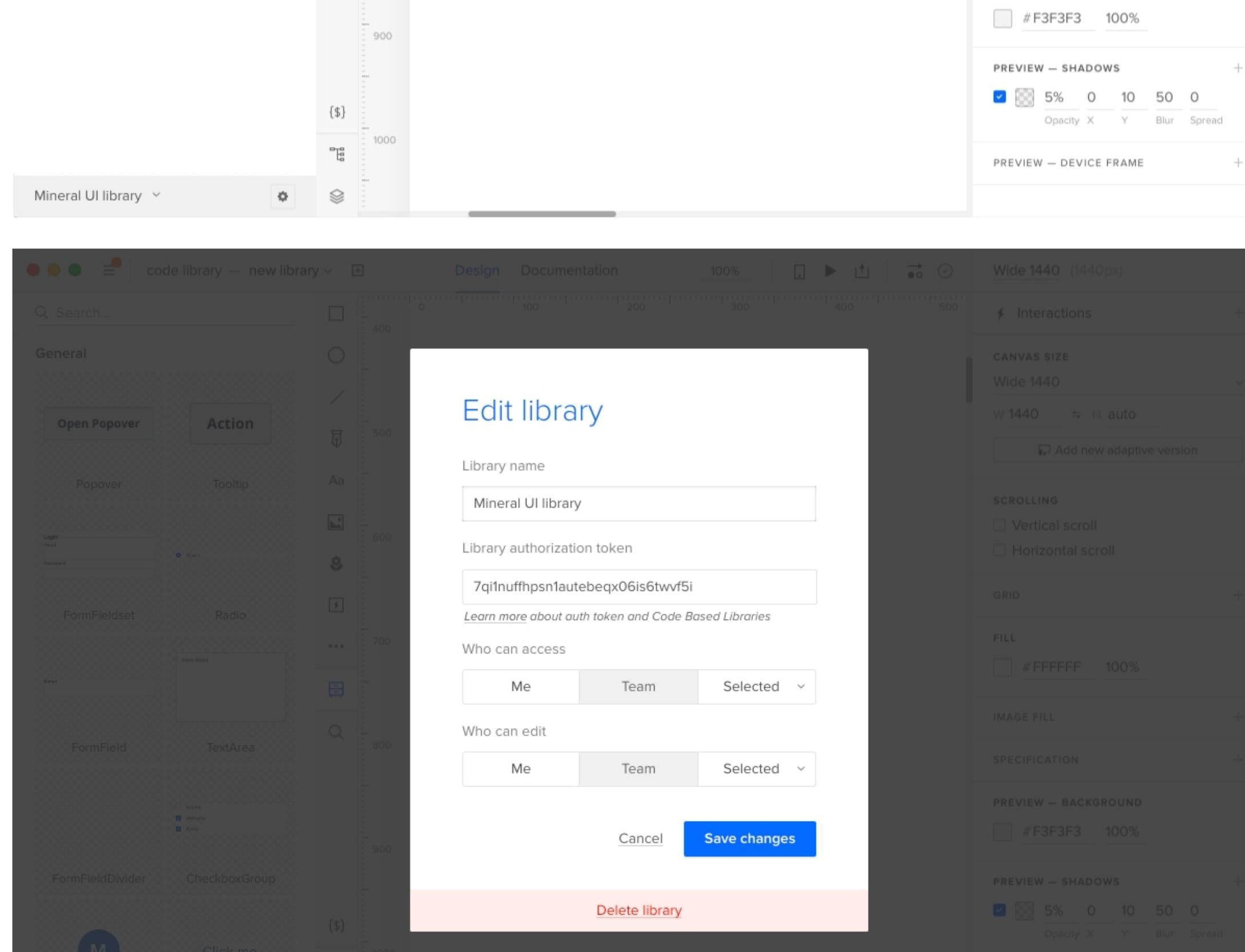
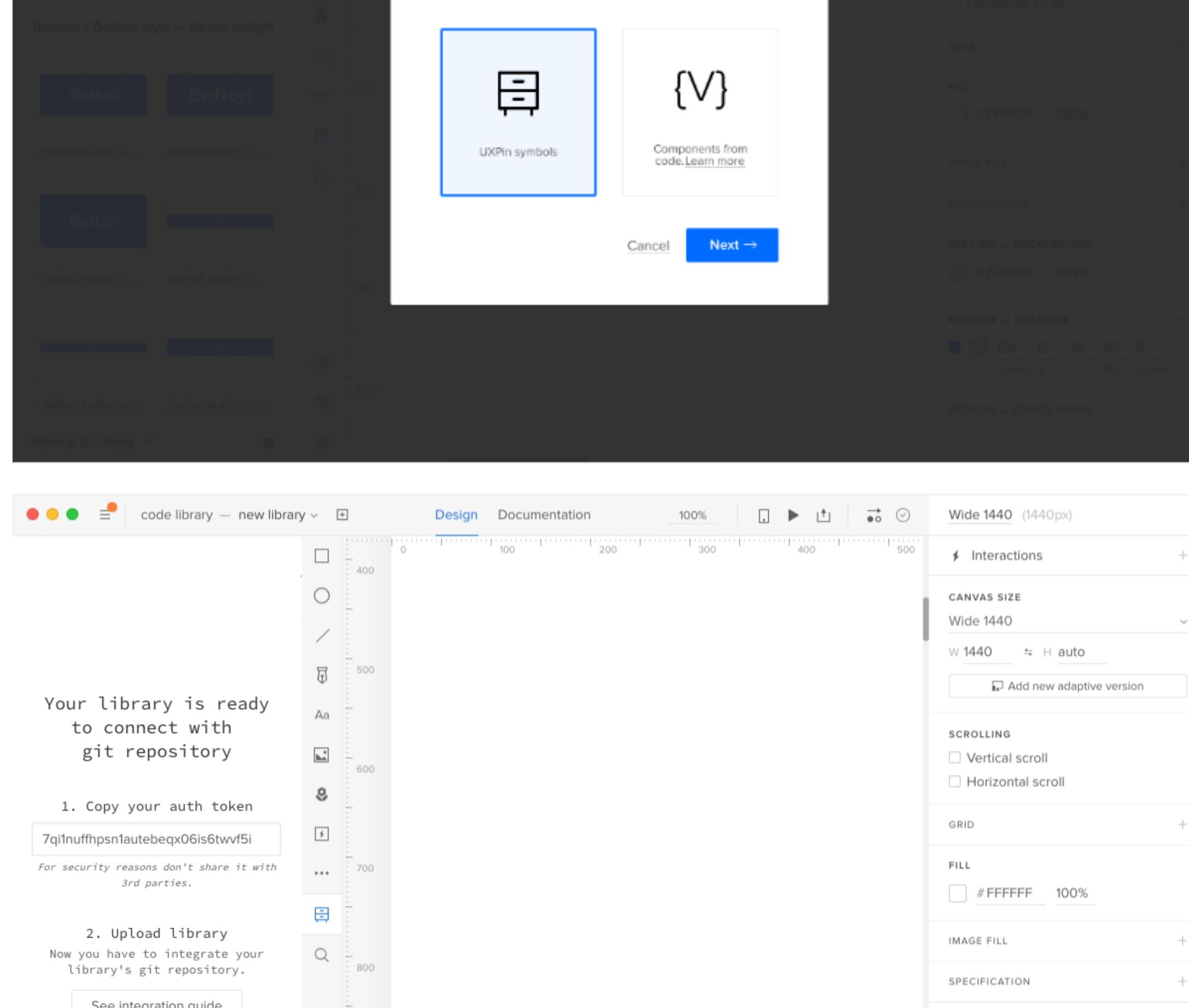
UXPin Merge

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



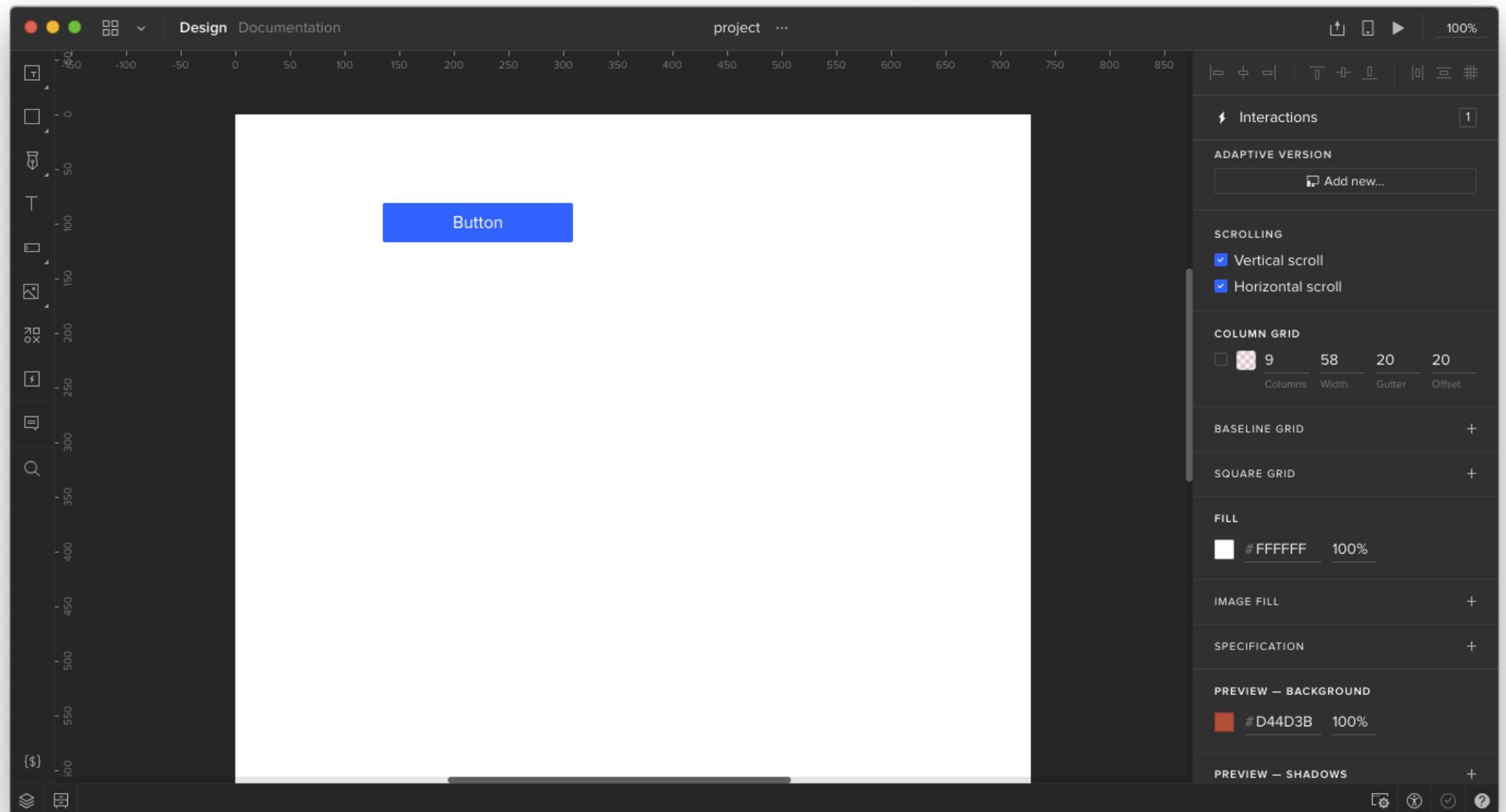
UXPin Desktop App

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

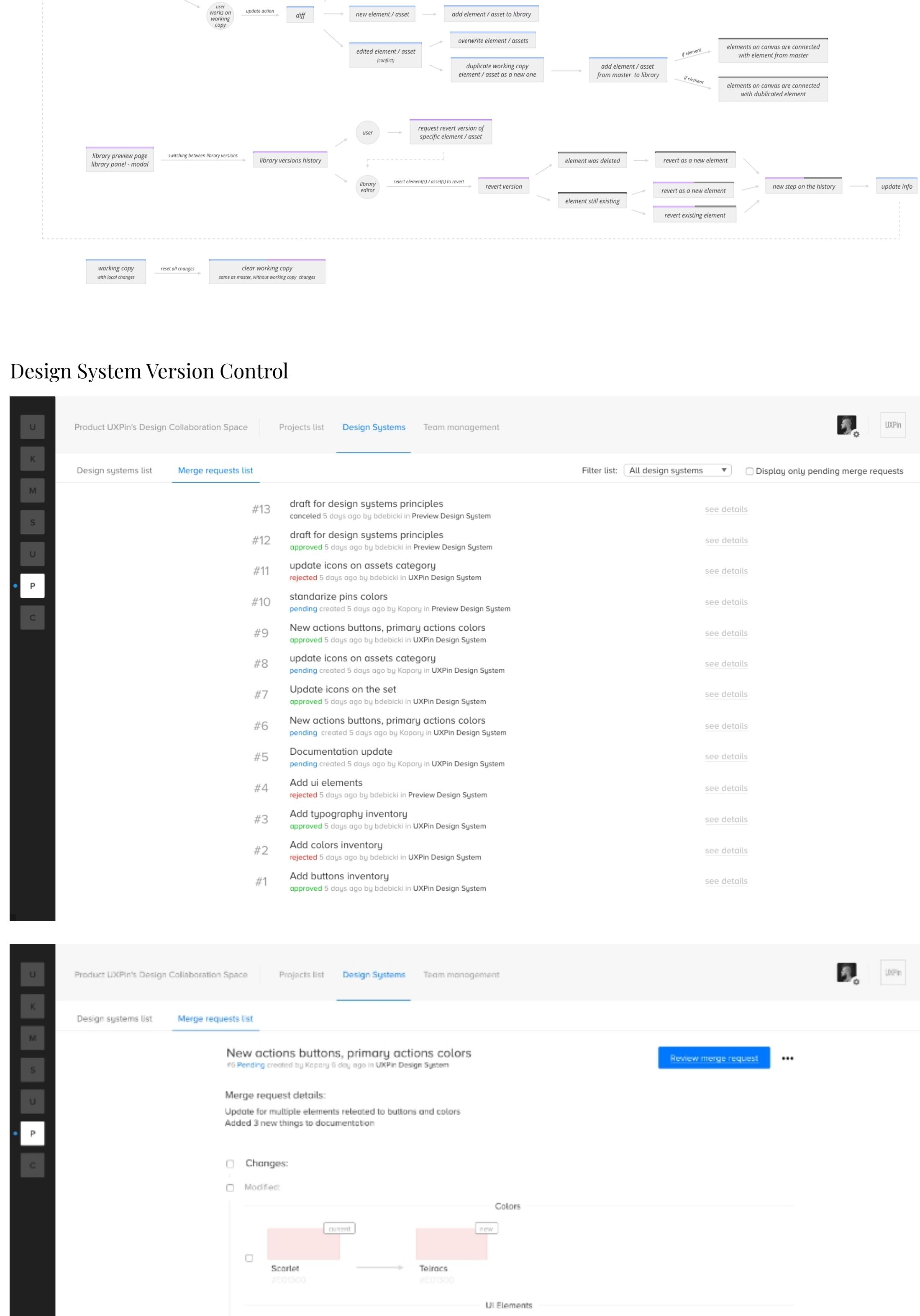
role :product lead / ux design + code



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. A large bracket labeled **Backend actions** covers the area under **E** and **D**. To the right, a bracket labeled **User actions** covers the area under the *action* arrow.

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

Typography

Typeface

Playfair Display

Regular *Italic* **Bold**

The quick brown fox jumps over the lazy dog.

New actions buttons, primary actions colors

v.2 changes by Kapary

draft for design systems principles

v.1 changes by Kapary

The screenshot shows a merge request in UXPin's Design System. The title is "New actions buttons, primary actions colors" (Accepted by bdebicki 1 day ago / created by Kapary 6 day ago on UXPin Design System). The merge request details mention updating multiple elements related to buttons and colors, adding 3 new things to documentation. The review details show 3 accepted changes from 5, with a note about picking only 3 changes because others weren't related to buttons. The approved changes section shows UI Elements, specifically Action buttons, with a comparison between current and new versions. The Buttons category description has been changed, and the Action button element description has also been changed. The right side of the interface shows properties like width, height, and scroll settings, along with a Set Grid and adaptive version buttons.

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



rainning at open'er fe

2017 2016 2015 2014 2013 2012

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 2011 2010 2009 2008 2007 2006 2005 2004 2003 2002

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

rainning at open'er fe

2016 2015 2014 2013 2012 2011 2010 2009 2008 2007 2006 2005 2004 2003 2002

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

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

rainning at open'er fe

2017 2016 2015 2014 2013 2012 2011 2010 2009 2008 2007 2006 2005 2004 2003 2002

was it raining at open'er festival? 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.

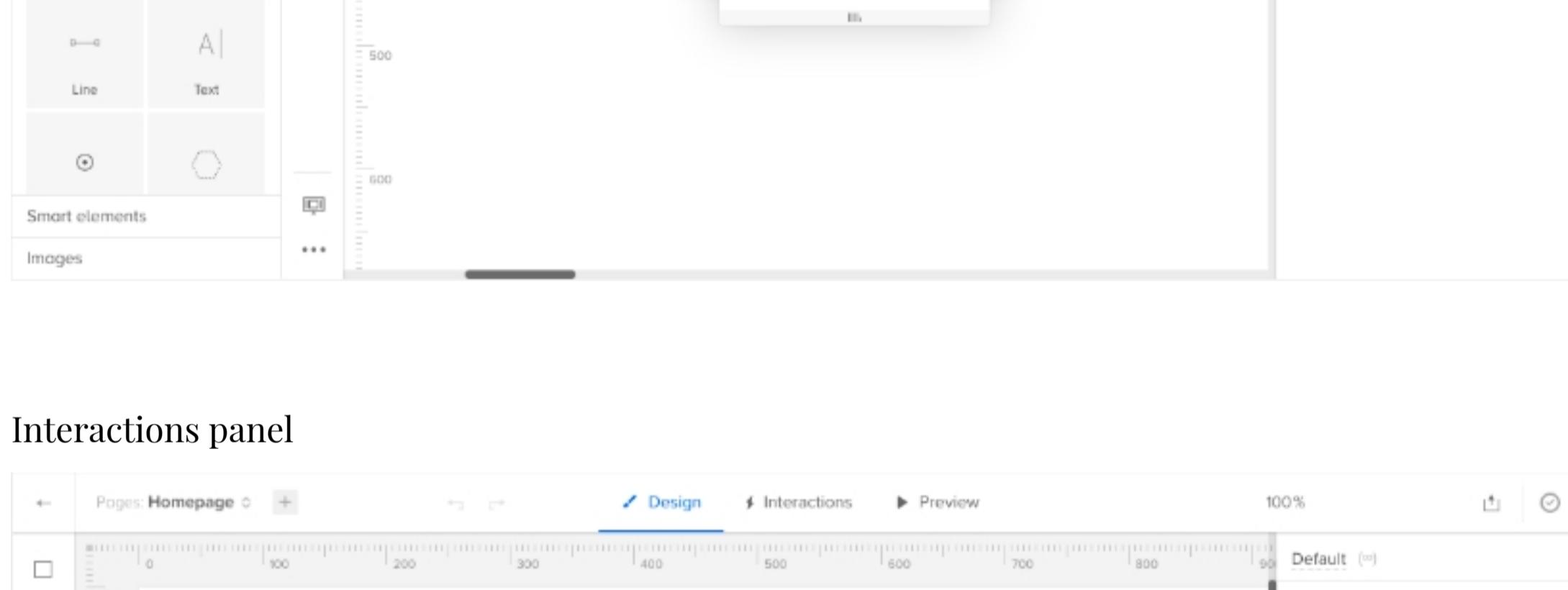
UI changes contain redesign of the top bar and release toolbar with quick access to the most popular elements and floating layers panel. Interactions flow was also simplified and enriched by predefined actions (based on the most popular ones).

2017 | UXPin
role :product lead / ux design

Top bar and toolbar



Left panels and floating layers



Interactions panel



Final redesign effect

