



# Hi, I'm Yogev Raved

Building end-to-end user experiences for complex systems

# What makes me tick

Dragging my family  
to hiking trips



Thousands of wasted hours  
on city building games



Geeking out  
with Lego



# Every design has a story



Config-templates  
design revamp

A screenshot of the Spot UI interface. On the left, there's a search bar with 'Tags' dropdowns set to 'All (34)' and 'All (6)'. Below it is a table with columns for 'Config-template-id', 'Name', 'Author', and 'Last Editor'. The table lists several entries, each with a 'User Name' column containing 'User Name 2'. On the right, there's a code editor window titled 'Latest Version: v. 6' showing a JSON-like configuration file:

```
1 {  
2   "type": "Python",  
3   "sparkVersion":  
4   "image": "example",  
5   "mainApplication":  
6   "arguments": [  
7     "--ProcessPro",  
8     "f8b2c391-d5",  
9     "--ProcessPr",  
10    "notebook-k",  
11    "--ProcessP",  
12    "eager",  
13    "--config",  
14    "SAMPLE_XE",  
15    "sparkConf",  
16    "spark.lo",  
17    "spark.hi",  
18    "spark.d",  
19    "spark.s",  
20    "spark.",  
21    "spark.",  
22    "spark.",  
23    "spark.",  
24    "spark.",  
25    "spark.",  
26    "spark.",  
27    "spark.",  
28    "spa",  
29    "spa",  
30    "spa",  
31    "spa",  
32    "spa",  
33    "spa",  
34    "spa",  
35    "spa",  
36    "spa",  
37    "spa",  
38    "spa",  
39    "spa",  
40    "spa",  
41    "spa",  
42    "spa",  
43    "spa",  
44    "spa",  
45    "spa",  
46    "spa",  
47    "spa",  
48    "spa",  
49    "spa",  
50    "spa",  
51    "spa",  
52    "spa",  
53    "spa",  
54    "spa",  
55    "spa",  
56    "spa",  
57    "spa",  
58    "spa",  
59    "spa",  
60    "spa",  
61    "spa",  
62    "spa",  
63    "spa",  
64    "spa",  
65    "spa",  
66    "spa",  
67    "spa",  
68    "spa",  
69    "spa",  
70    "spa",  
71    "spa",  
72    "spa",  
73    "spa",  
74    "spa",  
75    "spa",  
76    "spa",  
77    "spa",  
78    "spa",  
79    "spa",  
80    "spa",  
81    "spa",  
82    "spa",  
83    "spa",  
84    "spa",  
85    "spa",  
86    "spa",  
87    "spa",  
88    "spa",  
89    "spa",  
90    "spa",  
91    "spa",  
92    "spa",  
93    "spa",  
94    "spa",  
95    "spa",  
96    "spa",  
97    "spa",  
98    "spa",  
99    "spa",  
100   "spa",  
101   "spa",  
102   "spa",  
103   "spa",  
104   "spa",  
105   "spa",  
106   "spa",  
107   "spa",  
108   "spa",  
109   "spa",  
110   "spa",  
111   "spa",  
112   "spa",  
113   "spa",  
114   "spa",  
115   "spa",  
116   "spa",  
117   "spa",  
118   "spa",  
119   "spa",  
120   "spa",  
121   "spa",  
122   "spa",  
123   "spa",  
124   "spa",  
125   "spa",  
126   "spa",  
127   "spa",  
128   "spa",  
129   "spa",  
130   "spa",  
131   "spa",  
132   "spa",  
133   "spa",  
134   "spa",  
135   "spa",  
136   "spa",  
137   "spa",  
138   "spa",  
139   "spa",  
140   "spa",  
141   "spa",  
142   "spa",  
143   "spa",  
144   "spa",  
145   "spa",  
146   "spa",  
147   "spa",  
148   "spa",  
149   "spa",  
150   "spa",  
151   "spa",  
152   "spa",  
153   "spa",  
154   "spa",  
155   "spa",  
156   "spa",  
157   "spa",  
158   "spa",  
159   "spa",  
160   "spa",  
161   "spa",  
162   "spa",  
163   "spa",  
164   "spa",  
165   "spa",  
166   "spa",  
167   "spa",  
168   "spa",  
169   "spa",  
170   "spa",  
171   "spa",  
172   "spa",  
173   "spa",  
174   "spa",  
175   "spa",  
176   "spa",  
177   "spa",  
178   "spa",  
179   "spa",  
180   "spa",  
181   "spa",  
182   "spa",  
183   "spa",  
184   "spa",  
185   "spa",  
186   "spa",  
187   "spa",  
188   "spa",  
189   "spa",  
190   "spa",  
191   "spa",  
192   "spa",  
193   "spa",  
194   "spa",  
195   "spa",  
196   "spa",  
197   "spa",  
198   "spa",  
199   "spa",  
200   "spa",  
201   "spa",  
202   "spa",  
203   "spa",  
204   "spa",  
205   "spa",  
206   "spa",  
207   "spa",  
208   "spa",  
209   "spa",  
210   "spa",  
211   "spa",  
212   "spa",  
213   "spa",  
214   "spa",  
215   "spa",  
216   "spa",  
217   "spa",  
218   "spa",  
219   "spa",  
220   "spa",  
221   "spa",  
222   "spa",  
223   "spa",  
224   "spa",  
225   "spa",  
226   "spa",  
227   "spa",  
228   "spa",  
229   "spa",  
230   "spa",  
231   "spa",  
232   "spa",  
233   "spa",  
234   "spa",  
235   "spa",  
236   "spa",  
237   "spa",  
238   "spa",  
239   "spa",  
240   "spa",  
241   "spa",  
242   "spa",  
243   "spa",  
244   "spa",  
245   "spa",  
246   "spa",  
247   "spa",  
248   "spa",  
249   "spa",  
250   "spa",  
251   "spa",  
252   "spa",  
253   "spa",  
254   "spa",  
255   "spa",  
256   "spa",  
257   "spa",  
258   "spa",  
259   "spa",  
260   "spa",  
261   "spa",  
262   "spa",  
263   "spa",  
264   "spa",  
265   "spa",  
266   "spa",  
267   "spa",  
268   "spa",  
269   "spa",  
270   "spa",  
271   "spa",  
272   "spa",  
273   "spa",  
274   "spa",  
275   "spa",  
276   "spa",  
277   "spa",  
278   "spa",  
279   "spa",  
280   "spa",  
281   "spa",  
282   "spa",  
283   "spa",  
284   "spa",  
285   "spa",  
286   "spa",  
287   "spa",  
288   "spa",  
289   "spa",  
290   "spa",  
291   "spa",  
292   "spa",  
293   "spa",  
294   "spa",  
295   "spa",  
296   "spa",  
297   "spa",  
298   "spa",  
299   "spa",  
300   "spa",  
301   "spa",  
302   "spa",  
303   "spa",  
304   "spa",  
305   "spa",  
306   "spa",  
307   "spa",  
308   "spa",  
309   "spa",  
310   "spa",  
311   "spa",  
312   "spa",  
313   "spa",  
314   "spa",  
315   "spa",  
316   "spa",  
317   "spa",  
318   "spa",  
319   "spa",  
320   "spa",  
321   "spa",  
322   "spa",  
323   "spa",  
324   "spa",  
325   "spa",  
326   "spa",  
327   "spa",  
328   "spa",  
329   "spa",  
330   "spa",  
331   "spa",  
332   "spa",  
333   "spa",  
334   "spa",  
335   "spa",  
336   "spa",  
337   "spa",  
338   "spa",  
339   "spa",  
340   "spa",  
341   "spa",  
342   "spa",  
343   "spa",  
344   "spa",  
345   "spa",  
346   "spa",  
347   "spa",  
348   "spa",  
349   "spa",  
350   "spa",  
351   "spa",  
352   "spa",  
353   "spa",  
354   "spa",  
355   "spa",  
356   "spa",  
357   "spa",  
358   "spa",  
359   "spa",  
360   "spa",  
361   "spa",  
362   "spa",  
363   "spa",  
364   "spa",  
365   "spa",  
366   "spa",  
367   "spa",  
368   "spa",  
369   "spa",  
370   "spa",  
371   "spa",  
372   "spa",  
373   "spa",  
374   "spa",  
375   "spa",  
376   "spa",  
377   "spa",  
378   "spa",  
379   "spa",  
380   "spa",  
381   "spa",  
382   "spa",  
383   "spa",  
384   "spa",  
385   "spa",  
386   "spa",  
387   "spa",  
388   "spa",  
389   "spa",  
390   "spa",  
391   "spa",  
392   "spa",  
393   "spa",  
394   "spa",  
395   "spa",  
396   "spa",  
397   "spa",  
398   "spa",  
399   "spa",  
400   "spa",  
401   "spa",  
402   "spa",  
403   "spa",  
404   "spa",  
405   "spa",  
406   "spa",  
407   "spa",  
408   "spa",  
409   "spa",  
410   "spa",  
411   "spa",  
412   "spa",  
413   "spa",  
414   "spa",  
415   "spa",  
416   "spa",  
417   "spa",  
418   "spa",  
419   "spa",  
420   "spa",  
421   "spa",  
422   "spa",  
423   "spa",  
424   "spa",  
425   "spa",  
426   "spa",  
427   "spa",  
428   "spa",  
429   "spa",  
430   "spa",  
431   "spa",  
432   "spa",  
433   "spa",  
434   "spa",  
435   "spa",  
436   "spa",  
437   "spa",  
438   "spa",  
439   "spa",  
440   "spa",  
441   "spa",  
442   "spa",  
443   "spa",  
444   "spa",  
445   "spa",  
446   "spa",  
447   "spa",  
448   "spa",  
449   "spa",  
450   "spa",  
451   "spa",  
452   "spa",  
453   "spa",  
454   "spa",  
455   "spa",  
456   "spa",  
457   "spa",  
458   "spa",  
459   "spa",  
460   "spa",  
461   "spa",  
462   "spa",  
463   "spa",  
464   "spa",  
465   "spa",  
466   "spa",  
467   "spa",  
468   "spa",  
469   "spa",  
470   "spa",  
471   "spa",  
472   "spa",  
473   "spa",  
474   "spa",  
475   "spa",  
476   "spa",  
477   "spa",  
478   "spa",  
479   "spa",  
480   "spa",  
481   "spa",  
482   "spa",  
483   "spa",  
484   "spa",  
485   "spa",  
486   "spa",  
487   "spa",  
488   "spa",  
489   "spa",  
490   "spa",  
491   "spa",  
492   "spa",  
493   "spa",  
494   "spa",  
495   "spa",  
496   "spa",  
497   "spa",  
498   "spa",  
499   "spa",  
500   "spa",  
501   "spa",  
502   "spa",  
503   "spa",  
504   "spa",  
505   "spa",  
506   "spa",  
507   "spa",  
508   "spa",  
509   "spa",  
510   "spa",  
511   "spa",  
512   "spa",  
513   "spa",  
514   "spa",  
515   "spa",  
516   "spa",  
517   "spa",  
518   "spa",  
519   "spa",  
520   "spa",  
521   "spa",  
522   "spa",  
523   "spa",  
524   "spa",  
525   "spa",  
526   "spa",  
527   "spa",  
528   "spa",  
529   "spa",  
530   "spa",  
531   "spa",  
532   "spa",  
533   "spa",  
534   "spa",  
535   "spa",  
536   "spa",  
537   "spa",  
538   "spa",  
539   "spa",  
540   "spa",  
541   "spa",  
542   "spa",  
543   "spa",  
544   "spa",  
545   "spa",  
546   "spa",  
547   "spa",  
548   "spa",  
549   "spa",  
550   "spa",  
551   "spa",  
552   "spa",  
553   "spa",  
554   "spa",  
555   "spa",  
556   "spa",  
557   "spa",  
558   "spa",  
559   "spa",  
560   "spa",  
561   "spa",  
562   "spa",  
563   "spa",  
564   "spa",  
565   "spa",  
566   "spa",  
567   "spa",  
568   "spa",  
569   "spa",  
570   "spa",  
571   "spa",  
572   "spa",  
573   "spa",  
574   "spa",  
575   "spa",  
576   "spa",  
577   "spa",  
578   "spa",  
579   "spa",  
580   "spa",  
581   "spa",  
582   "spa",  
583   "spa",  
584   "spa",  
585   "spa",  
586   "spa",  
587   "spa",  
588   "spa",  
589   "spa",  
590   "spa",  
591   "spa",  
592   "spa",  
593   "spa",  
594   "spa",  
595   "spa",  
596   "spa",  
597   "spa",  
598   "spa",  
599   "spa",  
600   "spa",  
601   "spa",  
602   "spa",  
603   "spa",  
604   "spa",  
605   "spa",  
606   "spa",  
607   "spa",  
608   "spa",  
609   "spa",  
610   "spa",  
611   "spa",  
612   "spa",  
613   "spa",  
614   "spa",  
615   "spa",  
616   "spa",  
617   "spa",  
618   "spa",  
619   "spa",  
620   "spa",  
621   "spa",  
622   "spa",  
623   "spa",  
624   "spa",  
625   "spa",  
626   "spa",  
627   "spa",  
628   "spa",  
629   "spa",  
630   "spa",  
631   "spa",  
632   "spa",  
633   "spa",  
634   "spa",  
635   "spa",  
636   "spa",  
637   "spa",  
638   "spa",  
639   "spa",  
640   "spa",  
641   "spa",  
642   "spa",  
643   "spa",  
644   "spa",  
645   "spa",  
646   "spa",  
647   "spa",  
648   "spa",  
649   "spa",  
650   "spa",  
651   "spa",  
652   "spa",  
653   "spa",  
654   "spa",  
655   "spa",  
656   "spa",  
657   "spa",  
658   "spa",  
659   "spa",  
660   "spa",  
661   "spa",  
662   "spa",  
663   "spa",  
664   "spa",  
665   "spa",  
666   "spa",  
667   "spa",  
668   "spa",  
669   "spa",  
670   "spa",  
671   "spa",  
672   "spa",  
673   "spa",  
674   "spa",  
675   "spa",  
676   "spa",  
677   "spa",  
678   "spa",  
679   "spa",  
680   "spa",  
681   "spa",  
682   "spa",  
683   "spa",  
684   "spa",  
685   "spa",  
686   "spa",  
687   "spa",  
688   "spa",  
689   "spa",  
690   "spa",  
691   "spa",  
692   "spa",  
693   "spa",  
694   "spa",  
695   "spa",  
696   "spa",  
697   "spa",  
698   "spa",  
699   "spa",  
700   "spa",  
701   "spa",  
702   "spa",  
703   "spa",  
704   "spa",  
705   "spa",  
706   "spa",  
707   "spa",  
708   "spa",  
709   "spa",  
710   "spa",  
711   "spa",  
712   "spa",  
713   "spa",  
714   "spa",  
715   "spa",  
716   "spa",  
717   "spa",  
718   "spa",  
719   "spa",  
720   "spa",  
721   "spa",  
722   "spa",  
723   "spa",  
724   "spa",  
725   "spa",  
726   "spa",  
727   "spa",  
728   "spa",  
729   "spa",  
730   "spa",  
731   "spa",  
732   "spa",  
733   "spa",  
734   "spa",  
735   "spa",  
736   "spa",  
737   "spa",  
738   "spa",  
739   "spa",  
740   "spa",  
741   "spa",  
742   "spa",  
743   "spa",  
744   "spa",  
745   "spa",  
746   "spa",  
747   "spa",  
748   "spa",  
749   "spa",  
750   "spa",  
751   "spa",  
752   "spa",  

```



# Config-templates design revamp

---

## Role

Product Designer  
and Researcher

## Teammates

Product Manager,  
fullstack engineers

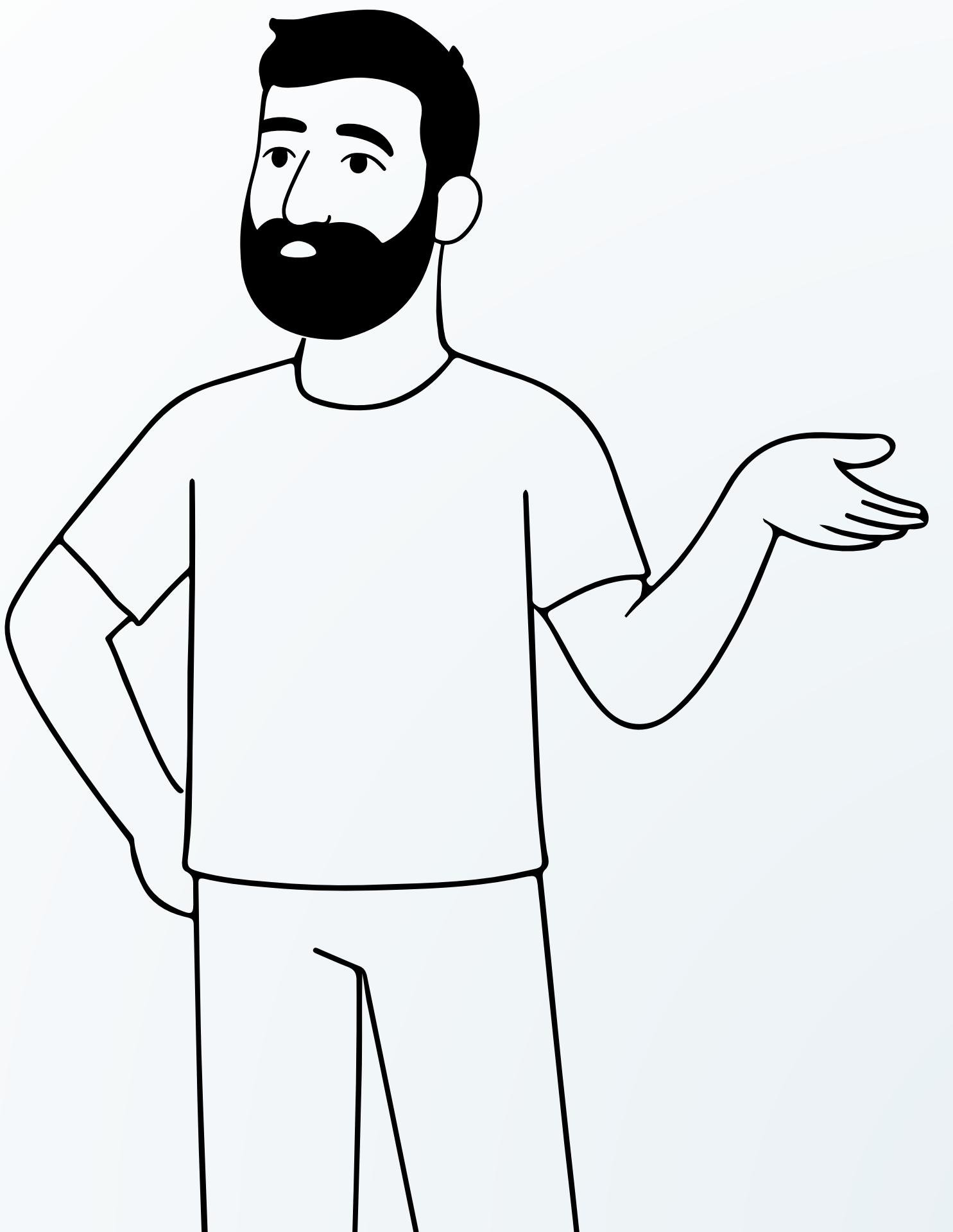
## Deliverables

UX research, wireframes,  
prototypes, user testing

What the #\*% is it?

# What are Config-templates?

- Data engineers use Apache Spark to run big-data applications.  
Each type of application needs its' own configuration (in a JSON format)
- The Config-templates page is a workspace within the Spot console for storing and managing them



[What are Config-templates?](#)

# How do users use them?

- Select a template and copy its' code
- Paste it in Apache Spark environment and run the app
- Come back to Spot's console to view the app's progress, stats and issues

# Motivation for the redesign

Improving usability issues of the current design:

- Template ID is visible only for the selected template
- Viewing few amount of templates - **only 3 on small screens!** makes it hard to browse between templates
- Adding more info to each template such as last editor, number of versions and tags will make each card much more overloaded

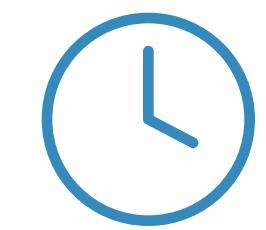
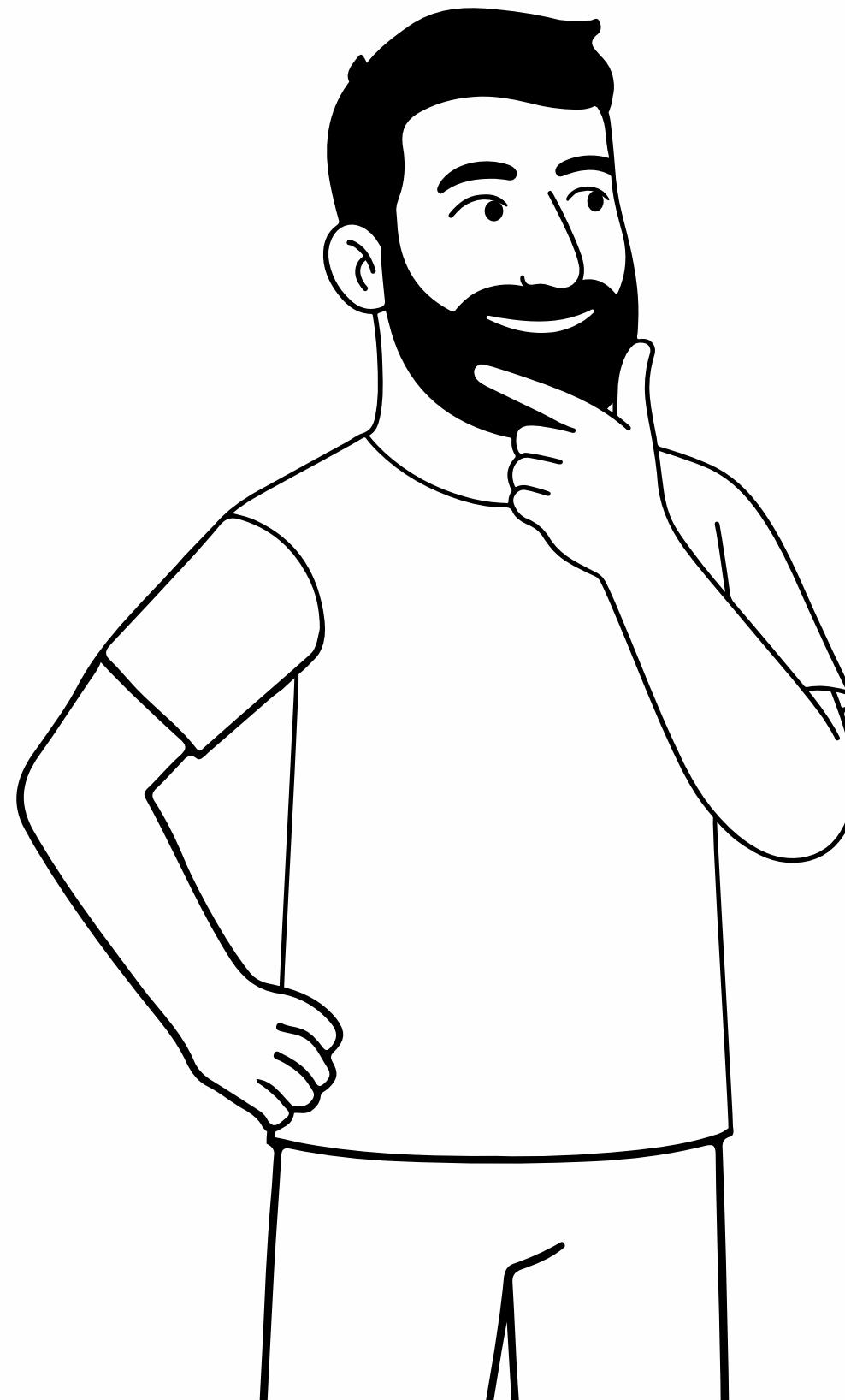
The screenshot shows the 'Configuration Templates' page in the Spot by NetApp interface. On the left, there's a sidebar with 'Ocean For Spark' and several navigation items: Applications, Jobs, Clusters, Workspaces, and Configuration Templates (which is currently selected). The main area displays a list of configuration templates with columns for Cluster Name, Author, and Template ID. The 'alex-ofas' template is highlighted. On the far right, a code editor window shows a JSON configuration file with 10 lines of code, starting with "type": "Python", "sparkVersion": "3.2.1", and so on.

Template ID	Cluster Name	Author
alex-ofas	alex-ofas-cluster-15	Alex Tarasov ...
asteroid	asteroid	Sigmar Karl
awendligner-notebooks	awendligner-tobys	Antoine Wen...
basic-notebook	clement-aws-dev-2	Clement Rez...
buggun		

```
1  {  
2   "type": "Python",  
3   "sparkVersion": "3.2.1",  
4   "sparkConf": {  
5     "spark.dynamicAllocation.enabled": "true",  
6     "spark.dynamicAllocation.maxExecutors": "10",  
7     "spark.dynamicAllocation.minExecutors": "0",  
8     "spark.dynamicAllocation.initialExecutors": "  
9   }  
10 }
```

The goal

# What do we want to achieve?



## Reduce time spent on troubleshooting

Increase user satisfaction



## Reduce support calls

Save time and money for the customers and for our organization



## More users per team

Increase product value



## Help selling the product to more clients

More revenue to our organization



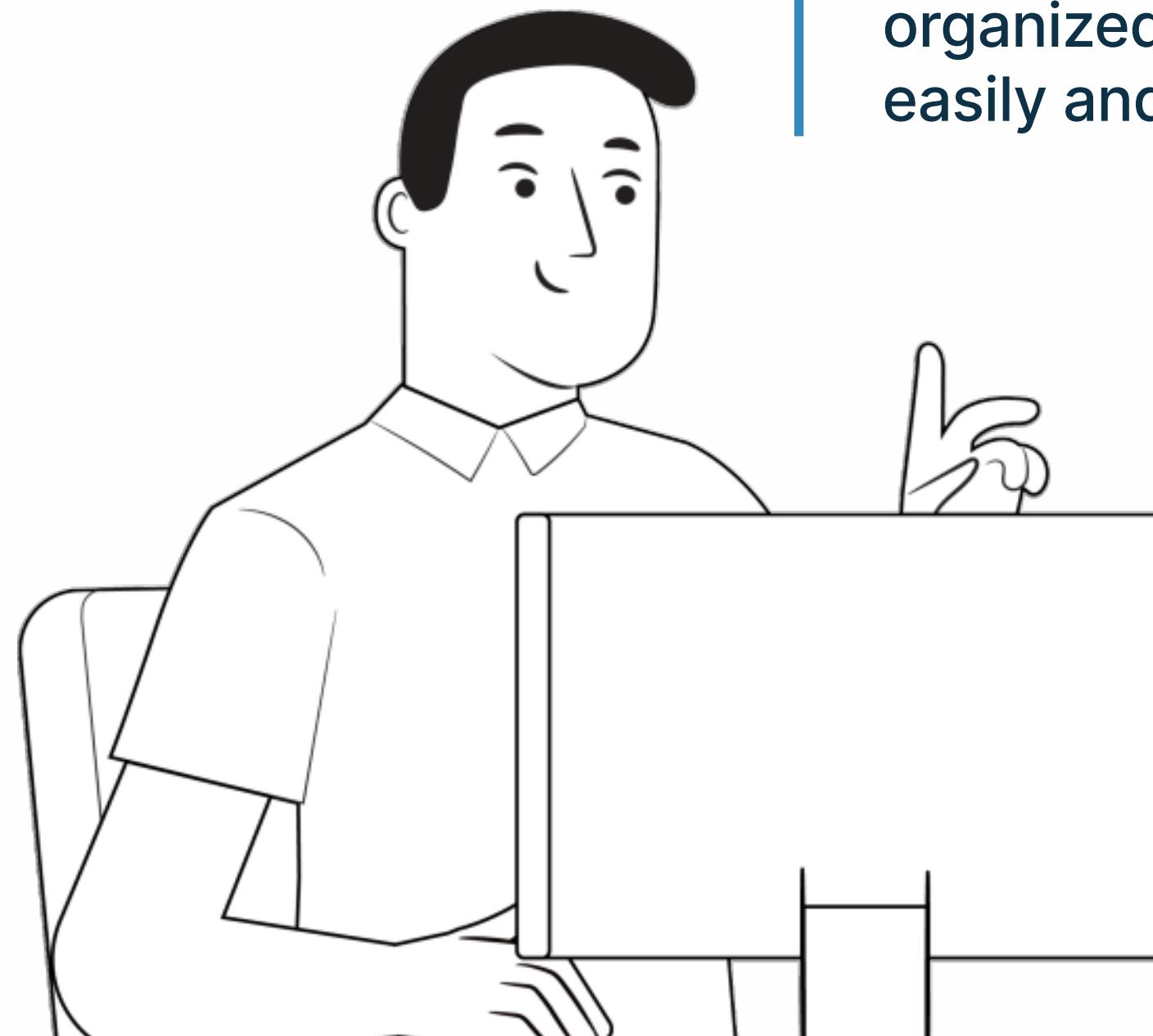
# Research

# Who are the users?

## Data engineer

Designs, builds, and optimizes data pipelines on Apache Spark, ensuring efficient data ingestion, transformation, and processing for analytics and machine learning applications.

“I need things to be very organized so I can find what I need easily and focus on my tasks”



## Data Platform Engineer (DevOps)

Develops and maintains the infrastructure for running Apache Spark applications, managing scalability, performance, and reliability to support seamless data operations.

“I need an option to view what changes were made in the templates over time and the reasons they were made”



# Reaching out to the users

## Why?

- To make sure our vision for the product is relevant
- Validate assumptions

## How?

- FullStory
- Surveys
- Interviews

Reaching out to the users

# Surveys

- Open-ended and closed questions
- Features rating and ideas ranking
- Gained qualitative and quantitative insights

Please rate how much you find useful these following features:

1 Useless    2 Somewhat useful    3 Useful    4 Extremely useful

	Comparing 2 different configuration templates	History, change log and revert options	Ability to add tags on configuration templates	Have starter templates when creating a new template	Be able to edit a configuration templates from the app page
Customer A	3	4	4	3	3
Customer B	3	4	4	3	3
Customer C	3	4	3	4	2
Customer D	4	3	2	3	3
Customer E	3	3	4	3	2
Customer F	4	4	4	4	4
Average	<b>3.3</b>	<b>3.6</b>	<b>3.5</b>	<b>3.3</b>	<b>2.8</b>

Reaching out to the users

# Customer & Stakeholder interviews

- Explored their daily workflow for creating and maintaining config-templates
- Examined integration with existing tools and processes
- Assessed work aspects both within and outside our product
- Analyzed current usage patterns and desired capabilities
- Studied user interactions and workflows
- Uncovered key pain points and frictions
- Identified short-term and long-term objectives



# User Journey Mapping

- Collaborative workshops with cross-functional teams
- Identified key interaction points and friction areas
- Prioritized problem spaces
- Created alignment on solution requirements



# Key Research Findings

## “As a data engineer I want to:

- Filter and sort templates based on various criteria so that I can focus on relevant items
- Organize and filter config-templates more effectively
- Group templates by different attributes so that I can better organize my view
- Perform bulk actions (like editing tags or deleting) on multiple templates at once so that I can manage them more efficiently
- Compare templates side by side so that I can easily identify differences for debugging

## Insights

1. Users strongly desired version control and change history
2. Template organization was a major pain point: Need for logical grouping and categorization and no way to track template purposes, and uses per application
3. Search and filtering capabilities were crucial for large template libraries



# Ideation

# Design solution

**So, we had to do some prioritization based on development capabilities**

- Add version history and changelog option
- Tagging system - This feature did not exist in the Spot console, and deciding on its behavior was challenging
- Advanced search and filtering
- Template comparison tools
- Applications running on config-templates list
- Presets for templates with common configurations
- Bulk actions for template management



The one page improvement task evolved into an enhanced design and development project...

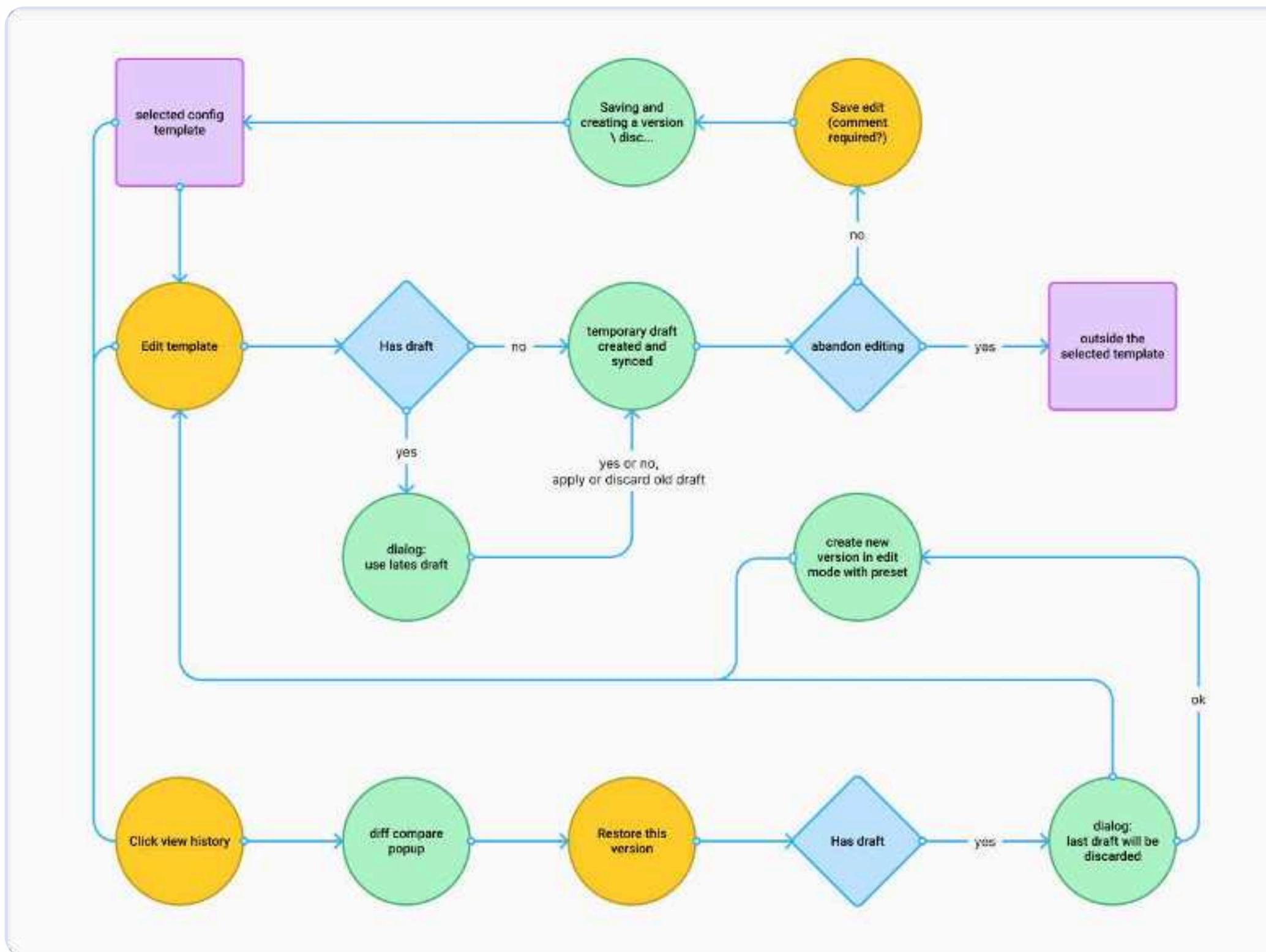
# Design process

## How should we approach each feature's design

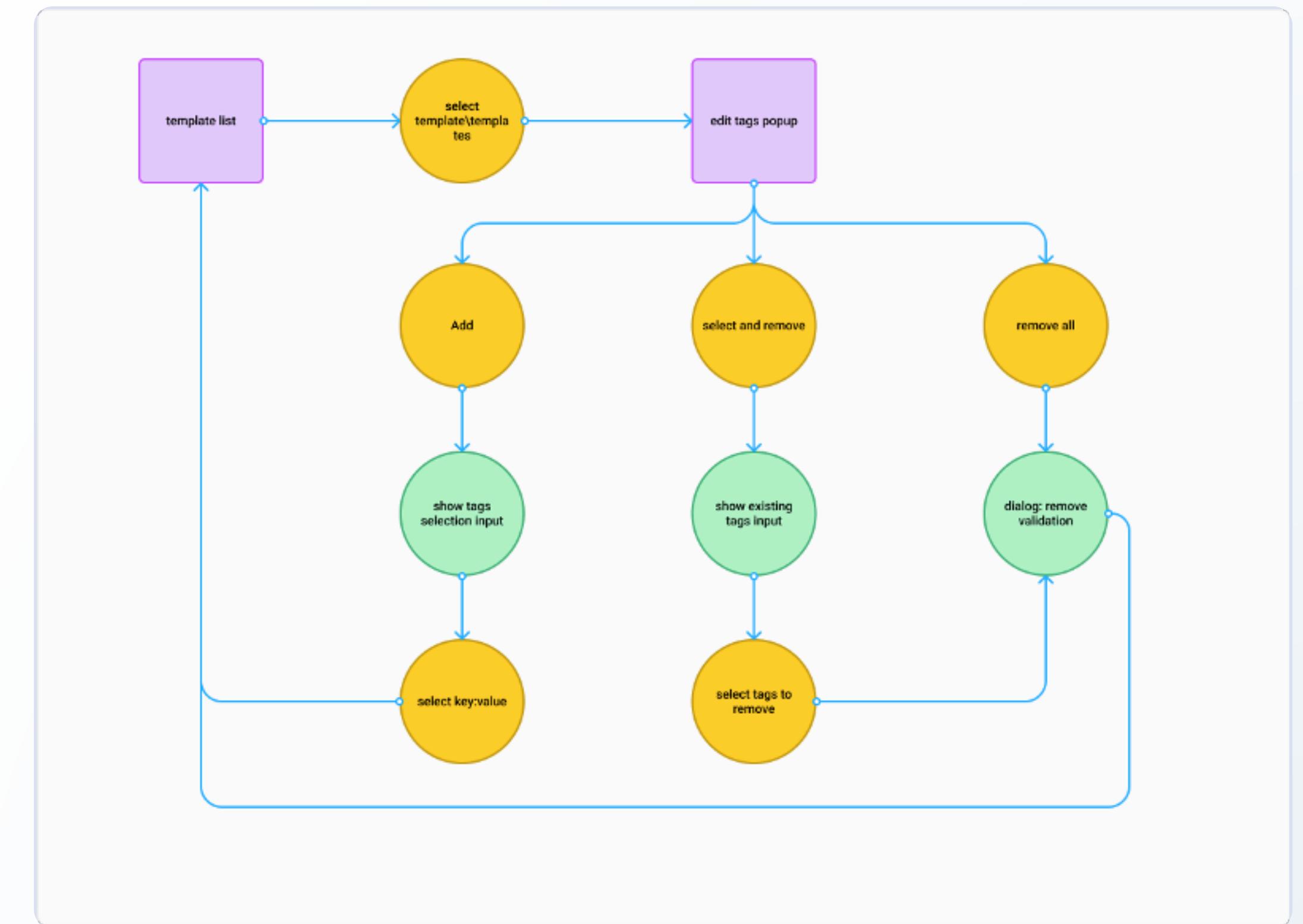
- Analyzed existing design system components
- Researched applicable design patterns
- Should we add a draft mechanism for unfinished editing
- Should commenting on a changelog be mandatory
- Mechanism of the tagging system
- Low fidelity wireframes on the page layout
- Original Cards layout vs a new table design
- Working with tags
- History view - compare and changelog
- Presets - custom and built in use cases

# User flows

## Editing

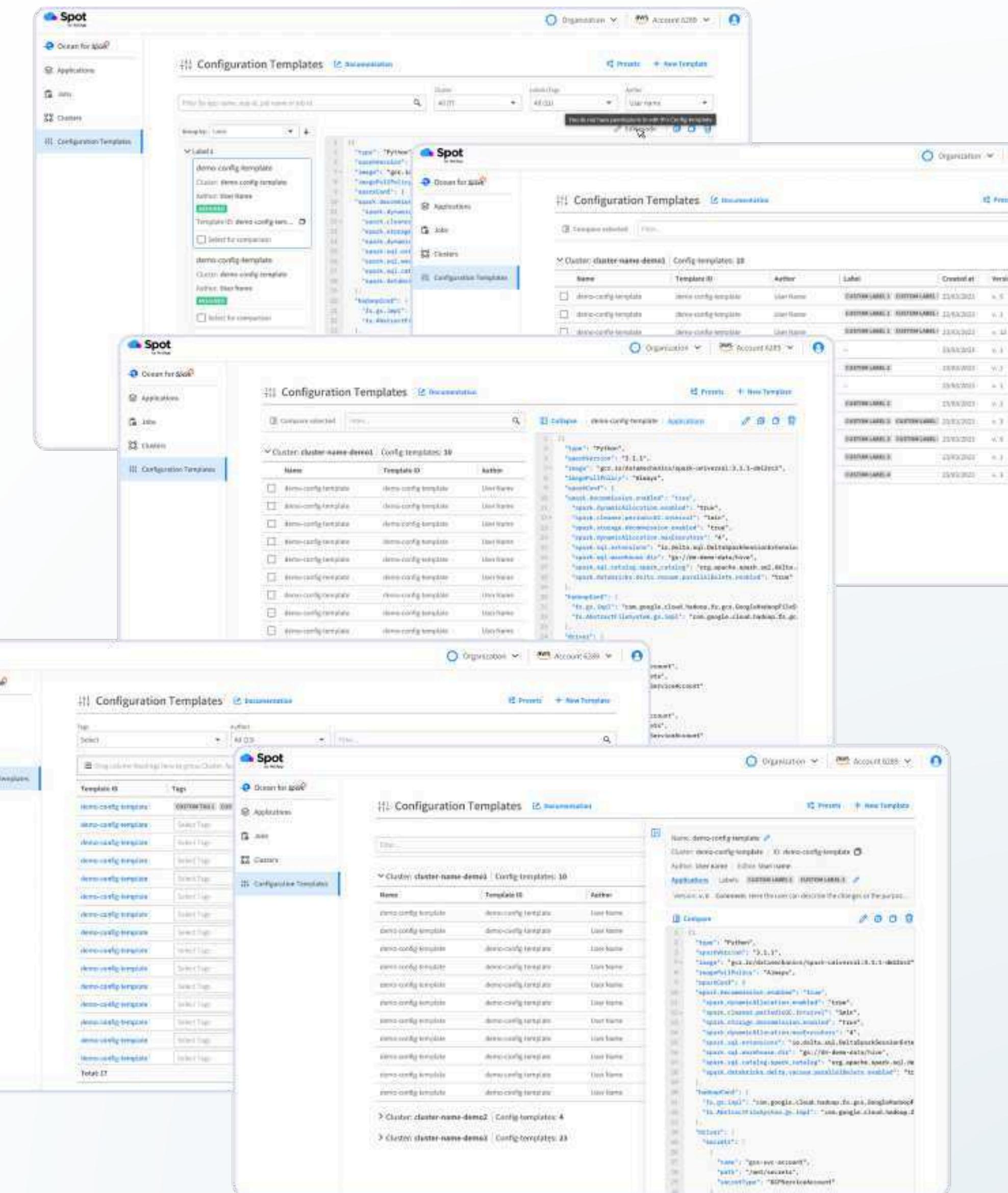


## Tagging Process



# Exploring layout directions

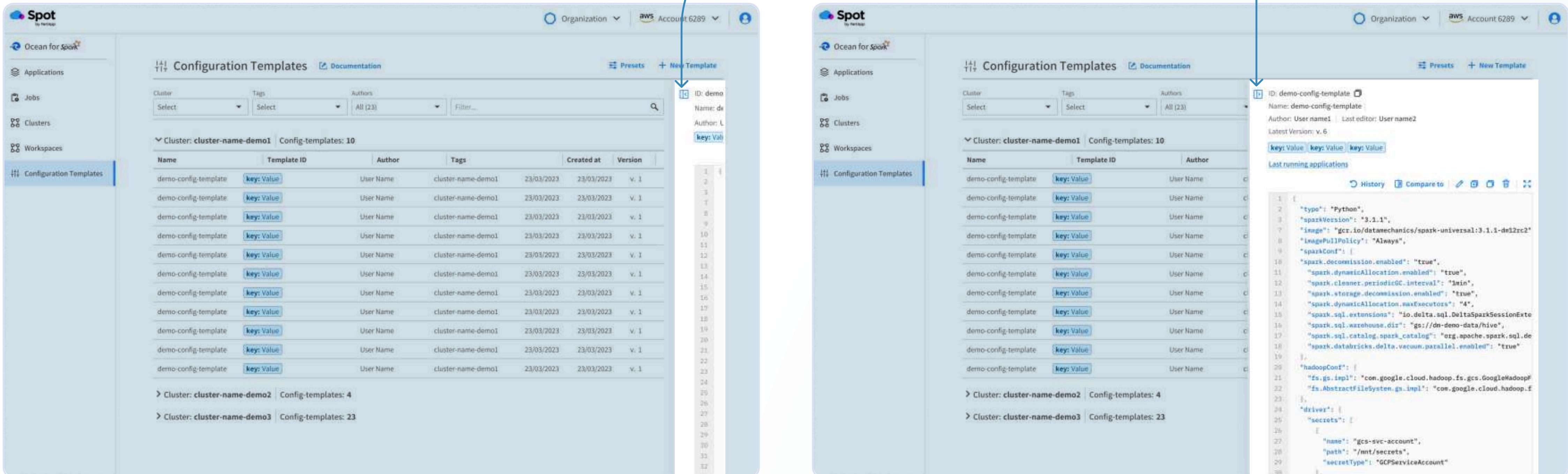
- The new features were added to the cards design.  
Several attempts and iterations of various layouts were made, such as wider cards to hold more data, an expanding panel and a table
- Several users' interviews and testing were conducted with internal users and with customers
- The testers were presented with the various layouts, and were asked to perform tasks such as filtering, sorting and performing bulk actions



# Exploring layout directions

# Expanded panel

Clicking the expand icon  
expand the panel over  
the config-templates list



By default: The template's content is collapsed

## User research

# Exploring layout directions

## Original cards

The screenshot shows a card-based user interface for managing configuration templates. At the top, there are filters for Tags, Cluster, and Authors. Below this is a search bar and a 'Group by' dropdown set to 'Default'. A sidebar on the left lists several templates with checkboxes:

- ID: be-basic-notebook-stg, Author: Samantha Smith, Cluster: demo-cluster config-template, Env: Staging, Team: Backend, Project: Flying Toothbrush
- ID: spark-connect-dev, Author: Woody Evered, Cluster: cluster-config-template, Env: Dev, Team: Frontend, Project: Hail Mary
- ID: be-basic-notebook-pr, Author: Allana Jerome, Cluster: Cluster-name-1, Env: Prod, Team: Backend, Project: Flying Toothbrush
- ID: HM-basic-notebook-1, Author: Woody Evered, Cluster: cluster-config-template, Env: Staging, Team: Infra, Project: Hail Mary
- ID: HM-basic-notebook-2

The main area displays the 'Latest Version: v.6' of a template, showing its JSON configuration. The configuration includes details like sparkVersion, image, arguments, and various sparkConf settings.

## Table layout

The screenshot shows the same configuration template data presented in a tabular format. The top navigation and sidebar are identical to the card layout. The main area is a table with columns: Template ID, Name, Author, and Last Editor. Each row represents a configuration template, with a checkbox to its left and a 'key: Value' button. The table shows the same five templates as the card layout. To the right of the table, the JSON configuration for the latest version is displayed in a monospaced font, showing the full JSON structure with line numbers.

# Exploring layout directions

## New features on cards layout

The image displays two side-by-side screenshots of a "Configuration Templates" interface, likely from a software application like AWS CloudFormation or a similar configuration management tool. Both screenshots show a list of templates with detailed configuration details.

**Left Screenshot:** This screenshot shows a single template selected. A blue arrow points from the text "Selecting a template opens a modal with options" to the modal window that appears over the list. The modal contains fields for "Env", "Team", and "Project", all set to "Flying Toothbrush".

```
1 {
  "type": "Python",
  "sparkVersion": "3.2.1",
  "image": "public.ecr.aws/ocean-spark/spark:platform-3.2.1-latest;wget pc539riyvz3qvzyyplus8q9u7l
  "mainApplicationFile": "local:///opt/spark/work-dir/launch_ipykernel.py",
  "arguments": [
    "--RemoteProcessProxy.kernel-id",
    "9cad545f-0463-41ba-8b55-6567b092d0b8",
    "--RemoteProcessProxy.response-address",
    "bigdata-notebook-service-kernel-connection.spot-system.svc.cluster.local:50100",
    "--RemoteProcessProxy.spark-context-initialization-mode",
    "lazy",
    "${id}",
    "MIGEMABGC5qGS1b3DQEBAQIA4GNADCBiQKBg0DXGn+UeRTsAoqgcfljAvtEvY+p4DKGwIB1Pzb0SKEvV/6672QFn23p
  ],
  "sparkConf": {
    "spark.local.dir": "/tmp/spark-local-dir-shuffle-4B754b66",
    "spark.hadoop.fs.s3.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
    "spark.decommission.enabled": "true",
    "spark.sql.adaptive.enabled": "true",
    "spark.eventLog.rolling.enabled": "true",
    "spark.dynamicAllocation.enabled": "true",
    "spark.cleaner.periodicGC.interval": "1min",
    "spark.storage.decommission.enabled": "true",
    "spark.dynamicAllocation.maxExecutors": "10",
    "spark.dynamicAllocation.minExecutors": "0",
    "spark.kubernetes.allocation.batch.size": "10",
    "spark.kubernetes.driver.requestTimeout": "30000",
    "spark.dynamicAllocation.initialExecutors": "1",
  }
}
```

**Right Screenshot:** This screenshot shows two templates selected. A blue arrow points from the text "Selecting 2 items enables comparing" to the "Compare selected" button at the bottom right of the interface. The selected templates have their "Env", "Team", and "Project" fields set to "Hail Mary".

```
1 {
  "type": "Python",
  "sparkVersion": "3.2.1",
  "image": "public.ecr.aws/ocean-spark/spark:platform-3.2.1-latest;wget pc539riyvz3qvzyyplus8q9u7l
  "mainApplicationFile": "local:///opt/spark/work-dir/launch_ipykernel.py",
  "arguments": [
    "--RemoteProcessProxy.kernel-id",
    "9cad545f-0463-41ba-8b55-6567b092d0b8",
    "--RemoteProcessProxy.response-address",
    "bigdata-notebook-service-kernel-connection.spot-system.svc.cluster.local:50100",
    "--RemoteProcessProxy.spark-context-initialization-mode",
    "lazy",
    "${id}",
    "MIGEMABGC5qGS1b3DQEBAQIA4GNADCBiQKBg0DXGn+UeRTsAoqgcfljAvtEvY+p4DKGwIB1Pzb0SKEvV/6672QFn23p
  ],
  "sparkConf": {
    "spark.local.dir": "/tmp/spark-local-dir-shuffle-4B754b66",
    "spark.hadoop.fs.s3.impl": "org.apache.hadoop.fs.s3a.S3AFileSystem",
    "spark.decommission.enabled": "true",
    "spark.sql.adaptive.enabled": "true",
    "spark.eventLog.rolling.enabled": "t
  },
  "spark.dynamicAllocation.enabled": "true",
  "spark.cleaner.periodicGC.interval": "1min",
  "spark.storage.decommission.enabled": "true",
  "spark.dynamicAllocation.maxExecutor"
}
```

**Annotations:**

- A blue arrow points from the text "Selecting a template opens a modal with options" to the modal window in the left screenshot.
- A blue arrow points from the text "Selecting 2 items enables comparing" to the "Compare selected" button in the right screenshot.

## Cards design

# User testing insights

## Cards design - Actions popup

### Issue:

Placement of bulk actions buttons outside this area could be confusing

### Solution:

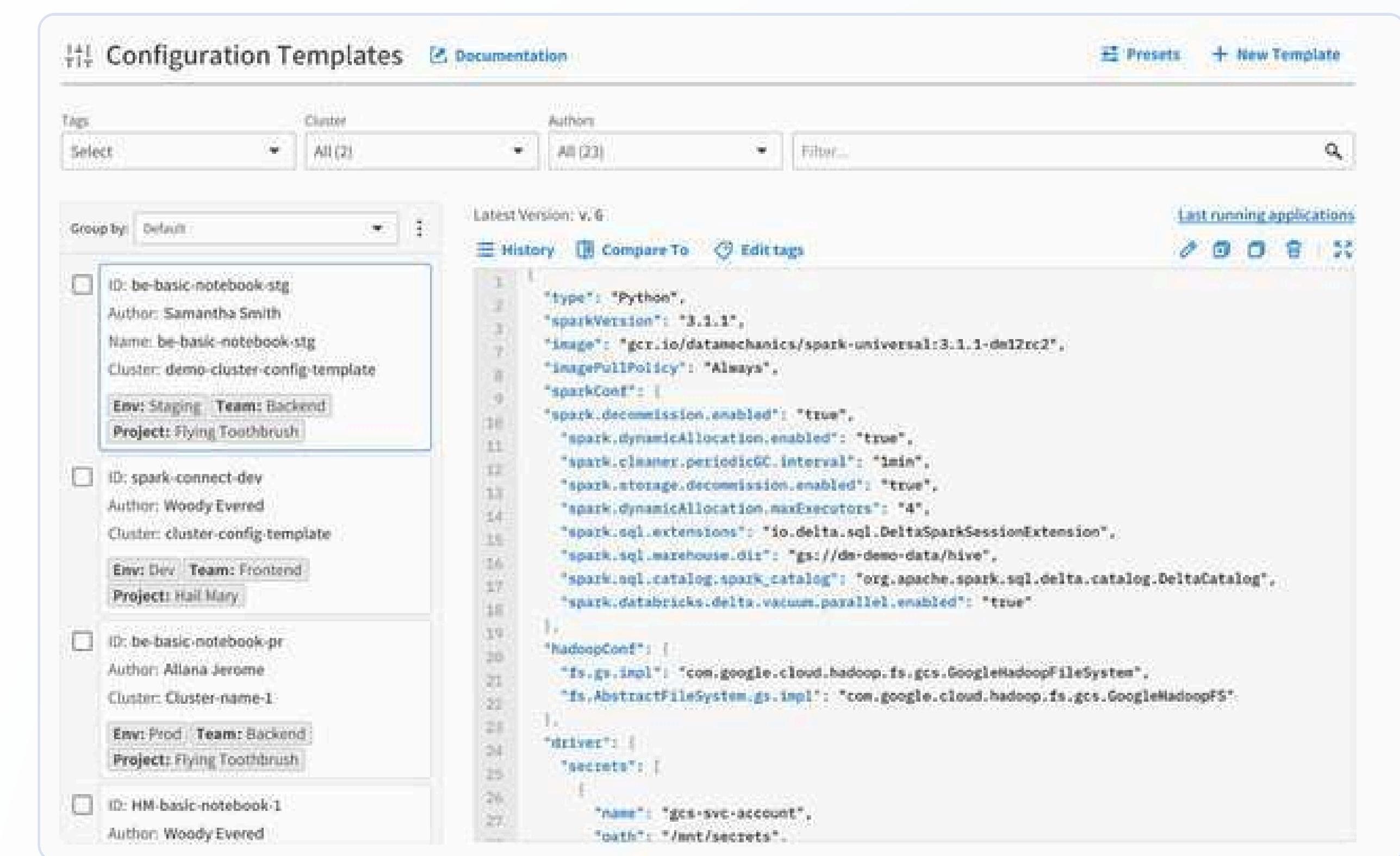
Actions popup - Triggered by checking an item

#### Pros:

Saves space on the page

#### Cons:

No indication for the actions without checking an item



## Cards design

# User testing insights

### Strength:

- Intuitive grouping by attribute - Tags, Author, Cluster
- Template view is wide from the start

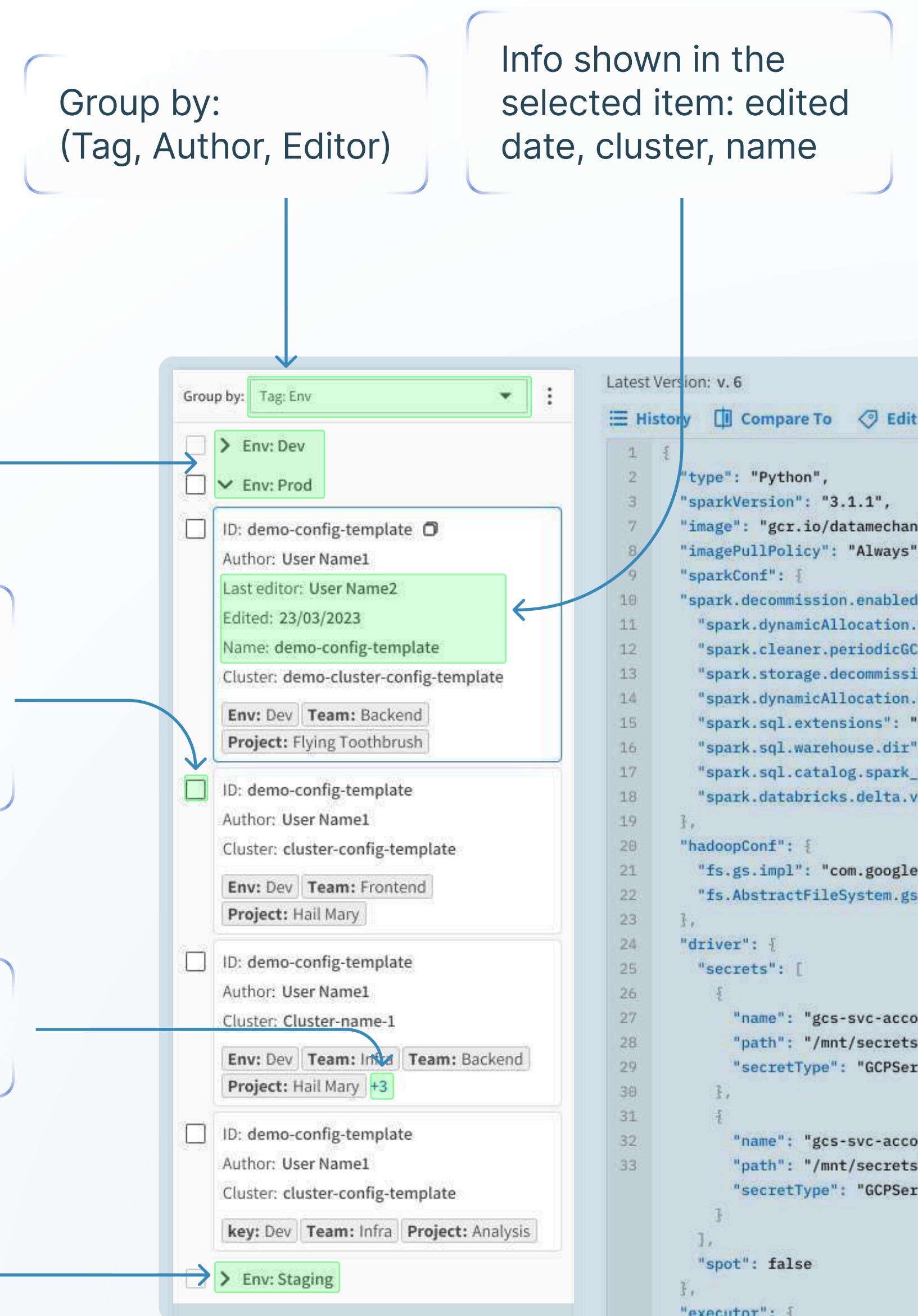
### Weaknesses:

- Sorting is limited
- Only a few templates are on screen
- Added more info to each template makes the cards much more overloaded and harder to browse

Example: Grouped by Tag "Env", collapsed and expanded

Checking an item triggers actions popup (compare, edit tags, delete), that can be performed in bulk

"Show more" icon for high amount tags



## Table design

# User testing insights

Configuration Templates [Documentation](#)

Filter... Search icon

Show Tags Select 2 templates for comparison Compare selected Edit tags

Drag column headings here to group Clusters or Authors

Template ID	Name	Author	Last Editor	Cluster	Created	Last
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Config-template-id	Config-template-name	User Name 1	User Name 2	cluster-name-demo1	23/03/2023	23/0
				<a href="#">key: Value</a> <a href="#">key: Value</a> <a href="#">key: Value</a>		
Total: 23						

Presets [New Template](#)

Latest Version: v. 6

Last running applications

History [Compare To](#) [Edit tags](#)

Presets [New Template](#)

```
1, {
2,   "type": "Python",
3,   "sparkVersion": "3.1.1",
4,   "image": "gcr.io/datamechanics/spark-universal:3.1.1-dm12rc2",
5,   "imagePullPolicy": "Always",
6,   "sparkConf": [
7,     "spark.decommission.enabled": "true",
8,     "spark.dynamicAllocation.enabled": "true",
9,     "spark.cleaner.periodicGC.interval": "1min",
10,    "spark.storage.decommission.enabled": "true",
11,    "spark.dynamicAllocation.maxExecutors": "4",
12,    "spark.sql.extensions": "io.delta.sql.DeltaSparkSessionExtension",
13,    "spark.sql.warehouse.dir": "gs://dm-demo-data/hive",
14,    "spark.sql.catalog.spark_catalog": "org.apache.spark.sql.delta.catalog.DeltaCatalog",
15,    "spark.databricks.delta.vacuum.parallel.enabled": "true"
16,  ],
17,  "hadoopConf": [
18,    "fs.gs.impl": "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFileSystem",
19,    "fs.AbstractFileSystem.gs.impl": "com.google.cloud.hadoop.fs.gcs.GoogleHadoopFileAbstractFileSystem"
20  ],
21  "driver": [
22    "secrets": [
23      {
24        "name": "gcs-svc-account",
25        "path": "/mnt/secrets",
26        "secretType": "GCPServiceAccount"
27      },
28      {
29        "name": "gcs-svc-account",
30        "path": "/mnt/secrets",
31        "secretType": "GCPServiceAccount"
32      }
33    ],
34    "spot": false
35  ],
36  "executor": [
37    "secrets": [
38      {
39        "name": "gcs-svc-account",
40        "path": "/mnt/secrets",
41        "secretType": "GCPServiceAccount"
42      }
43    ]
44  ]
```

# Table design

# User testing insights

## Group by column header - Tag, Author, Editor

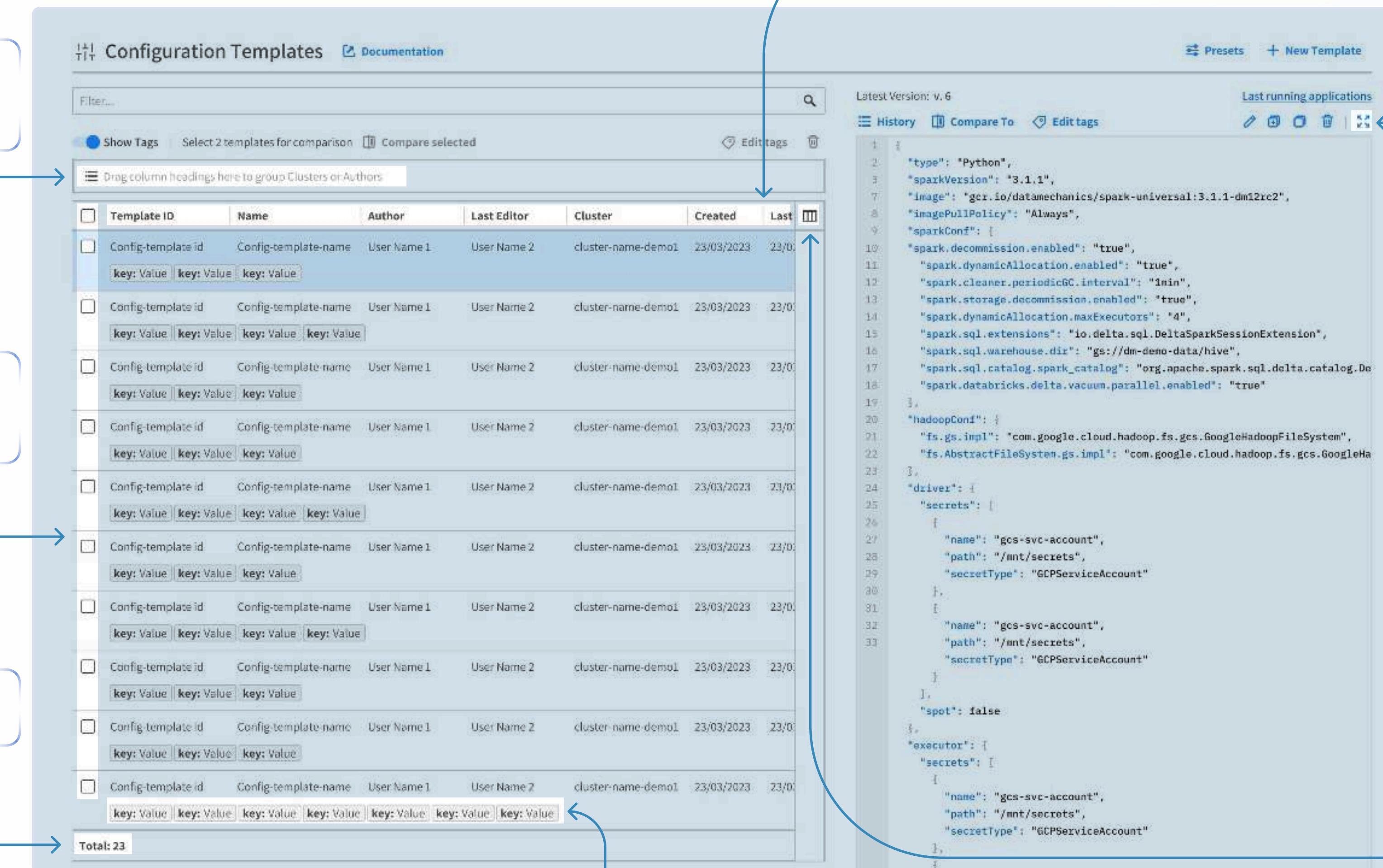
# Bulk selection for actions - compare, edit tags, delete

## Total amount of templates

## Easy sorting, filtering and grouping

Expand the preview to full screen

## Custom columns to arrange and to filter data



# User testing insights

## Strength:

- Sortable by multiple parameters (ID, name, cluster, author, date)
- No need to hide config-template info
- Easier template browsing

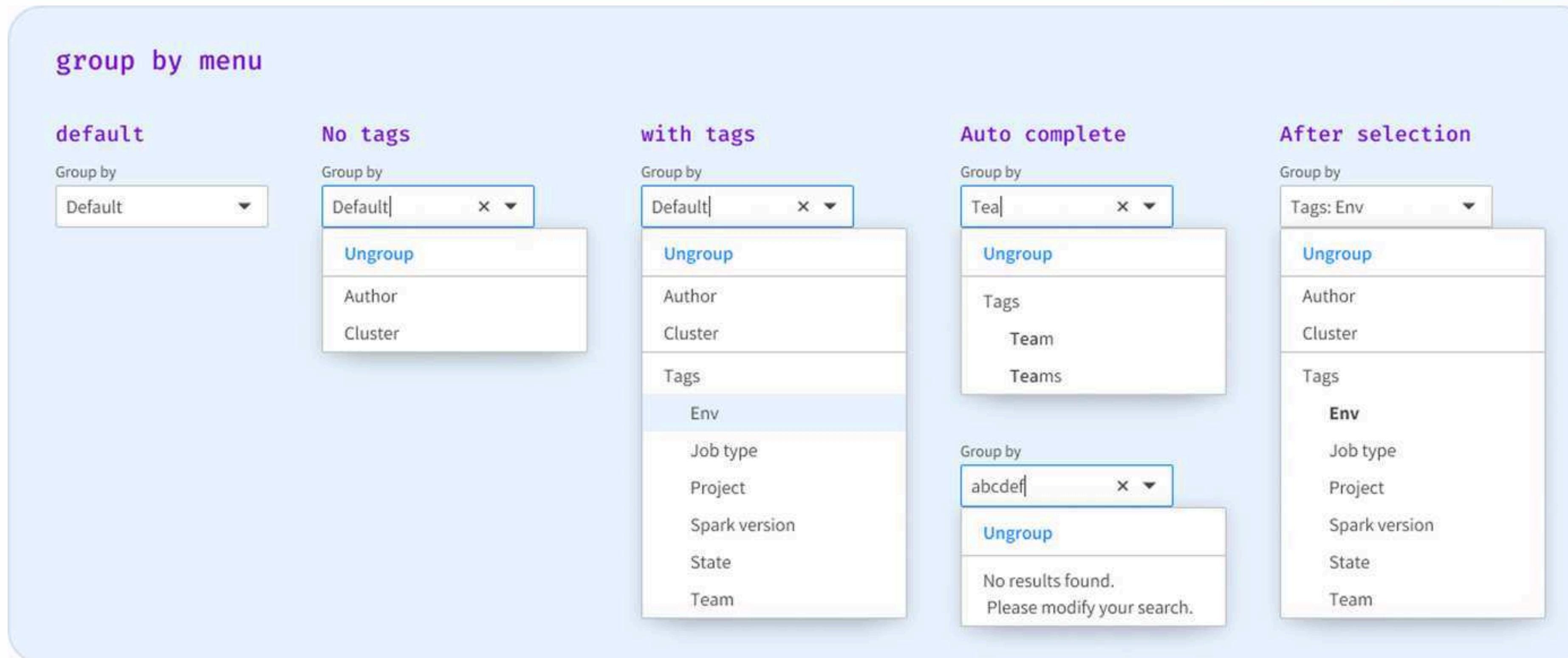
## Weaknesses:

- Less initial space for templates (solved with full-screen option)
- Grouping limited to column headers, missing tag-based grouping (design system)

User testing insights

# Grouping solution

Instead of the design system's grouping option I switched to a simple “Group by” selection





# **Final design & New UI**

# Main page

The screenshot shows the Spot by NetApp interface. The left sidebar includes links for Ocean for Spark, Application, Jobs, Clusters, Workspaces, and Configuration Templates. The main area is titled "Configuration Templates" and displays a table of templates. The table columns are Template ID, Name, Author, and Last Editor. A "Show Tags" checkbox is checked, and dropdown menus show Tags (All (34)), Author (All (6)), and Cluster (All (3)). A search bar and a "Filter" button are also present. On the right, a preview pane shows the latest version (v. 6) of a template, with a code editor containing the following JSON configuration:

```
1  {
2    "type": "Python",
3    "sparkVersion": "3.4.0",
4    "image": "example.registry.com/spark:3.4.0-base",
5    "mainApplicationFile": "local:///opt/spark/scripts/main.p
"arguments": [
6      "--ProcessProxy.kernel-id",
7      "f8b2c391-d5e7-4a12-9f3d-123456789abc",
8      "--ProcessProxy.response-address",
9      "notebook-kernel-service.example-namespace.svc.cluster.
10     --ProcessProxy.spark-init-mode",
11     "eager",
12     "--config",
13     "SAMPLE_KEY_123456789"
14   ],
15   "sparkConf": {
16     "spark.local.dir": "/tmp/spark-local-dir-cache-abc123",
17     "spark.hadoop.fs.s3.impl": "org.apache.hadoop.fs.s3a.S3
18     "spark.decommission.enabled": "true",
19     "spark.sql.adaptive.enabled": "true",
20     "spark.eventLog.rolling.enabled": "true",
21     "spark.dynamicAllocation.enabled": "true",
22     "spark.cleaner.periodicGC.interval": "2min",
23     "spark.storage.decommission.enabled": "true",
24     "spark.dynamicAllocation.maxExecutors": "8",
25     "spark.dynamicAllocation.minExecutors": "1",
26     "spark.kubernetes.allocation.batch.size": "5",
27   }
}
```

# Main page

The screenshot shows the Configuration Templates page in the Spot by NetApp interface. The left sidebar includes links for Ocean for Spark, Application, Jobs, Clusters, Workspaces, and Configuration Templates. The main area displays a list of templates with columns for Template ID, Name, Author, and Last Editor. A modal window on the right provides a detailed view of a selected template's configuration, showing its JSON structure. The JSON content includes:

```
1 {
2   "type": "Python",
3   "sparkVersion": "3.4.0",
4   "image": "example.registry.com/spark:3.4.0-base",
5   "mainApplicationFile": "local:///opt/spark/scripts/main.p
"arguments": [
6     "--ProcessProxy.kernel-id",
7     "f8b2c391-d5e7-4a12-9f3d-123456789abc",
8     "--ProcessProxy.response-address",
9     "notebook-kernel-service.example-namespace.svc.cluster.
10    --ProcessProxy.spark-init-mode",
11    "eager",
12    "--config",
13    "SAMPLE_KEY_123456789"
14  ],
15  "sparkConf": {
16    "spark.local.dir": "/tmp/spark-local-dir-cache-ab
17    "spark.hadoop.fs.s3.impl": "org.apache.hadoop.fs.
18    "spark.decommission.enabled": "true",
19    "spark.sql.adaptive.enabled": "true",
20    "spark.eventLog.rolling.enabled": "true",
21    "spark.dynamicAllocation.enabled": "true",
22    "spark.cleaner.periodicGC.interval": "2min",
23    "spark.storage.decommission.enabled": "true",
24    "spark.dynamicAllocation.maxExecutors": "8",
25    "spark.dynamicAllocation.minExecutors": "1",
26    "spark.kubernetes.allocation.batch.size": "5",
27  }
28}
```

Annotations on the page include:

- Option to hide the tags row
- Version No.
- Selected item actions
- Template run apps list
- Expand to full screen
- When no tags, the user is suggested to add some
- Infinite scroll

# Empty state

Configuration Templates Documentation ↗

Presets + New Template



Configuration templates are stored fragments of Spark application configuration, for when you need to share a large default configuration between multiple applications, or to avoid storing configurations on your side. [Learn more](#)

You can create template from a Preset or start from scratch.

Presets + New Template

# Panel update on selection

Enabled when two items are selected

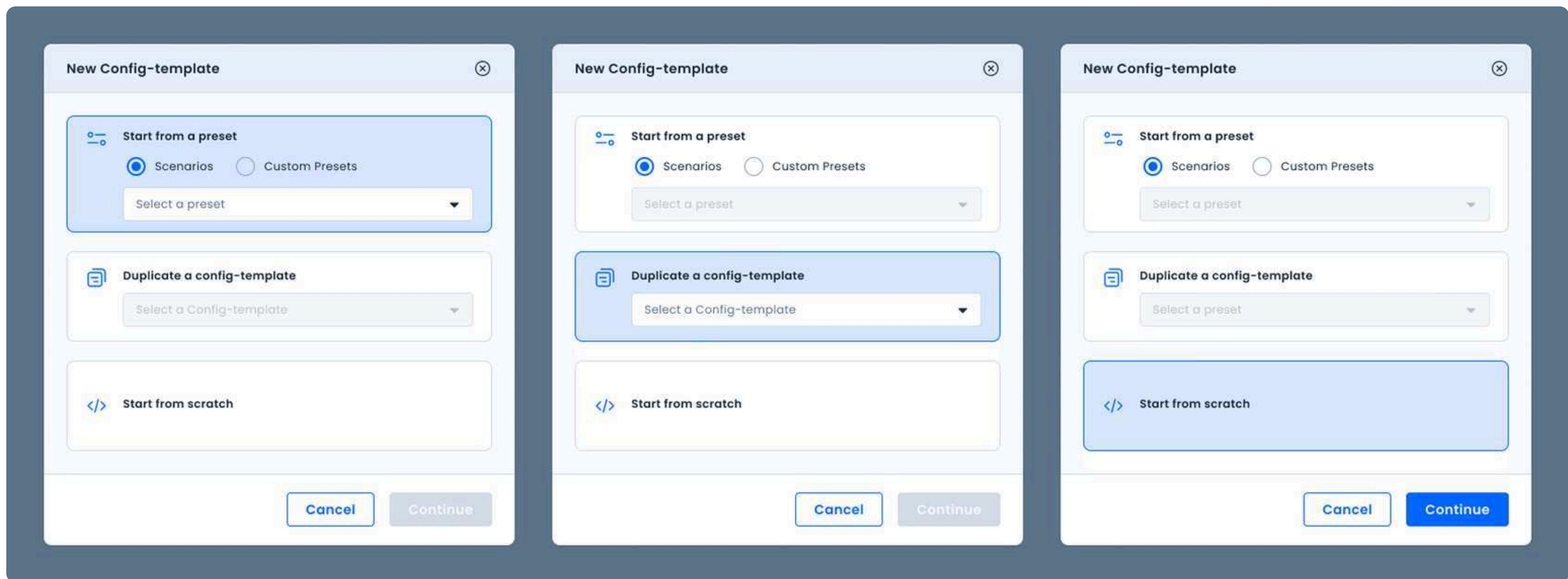
The diagram illustrates a user interface for managing configuration templates. It consists of three panels:

- Left Panel:** A list of configuration templates. One template is selected (indicated by a checked checkbox). The selected template's configuration details are displayed in a modal window.
- Middle Panel:** The configuration details for the selected template. It includes fields for Name, Author, Last Editor, Cluster, and a JSON representation of the configuration. The JSON content is as follows:

```
1: {  
2:   "type": "Python",  
3:   "sparkVersion": "3.4.0",  
4:   "image": "example.registry.com/spark:3.4.0-base",  
5:   "mainApplicationFile": "local:///opt/spark/scripts/main.py",  
6:   "arguments": [  
7:     "-ProcessProxy.kernel-id",  
8:     "j8b2c391-d5e7-4a12-9f3d-123456789abc",  
9:     "-ProcessProxy.response-address",  
10:    "notebook-kernel-service.example-namespace.svc.cluster.",  
11:    "-ProcessProxy.spark-init-node",  
12:    "eager",  
13:    "-config",  
14:    "SAMPLE_KEY_123456789"  
15:  },  
16:  "sparkConf": {  
17:    "spark.local.dir": "/tmp/spark-local-dir-cache-abc123",  
18:    "spark.hadoop.fs.s3a.inal": "org.apache.hadoop.fs.s3a.S3",  
19:    "spark.decommission.enabled": "true",  
20:    "spark.sql.adaptive.enabled": "true",  
21:    "spark.eventlog.relicensing.enabled": "true",  
22:    "spark.dynamicAllocation.enabled": "true",  
23:    "spark.yarn.executor.periodicGC.interval": "2min",  
24:    "spark.storage.decommission.enabled": "true",  
25:    "spark.dynamicAllocation.maxExecutors": "8",  
26:    "spark.dynamicAllocation.minExecutors": "1",  
27:    "spark.kubernetes.allocation.batch.size": "5",  
28:    "spark.kubernetes.driver.requests.timeout": "25000",  
29:    "spark.dynamicAllocation.initialExecutors": "2",  
30:    "spark.sql.execution.arrow.spark.enabled": "true",  
31:    "spark.kubernetes.driver.connectionTimeout": "25000",  
32:    "spark.sql.execution.arrow.pyspark.enabled": "true",  
33:    "spark.storage.decommission.rddBlocks.enabled": "true",  
34:    "spark.dynamicAllocation.executorAllocationRatio": "0.5",  
35:    "spark.dynamicAllocation.shuffleTracking.enabled": "true",  
36:    "spark.storage.decommission.shuffleBlocks.enabled": "true",  
37:    "spark.kubernetes.allocation.driver.readinessTimeout": "10s",  
38:    "spark.dynamicAllocation.sustainedSchedulerBacklogTime": "10s",  
39:    "spark.hadoop.mapreduce.fileoutputcommitter.algorithm.v  
40:  }  
};
```

- Right Panel:** The configuration details for two selected templates. The UI shows a message "2 Templates selected" and provides options to "Compare selected" or "Edit tags".

# New template



# Presets

Configuration Templates / Presets

## Presets

+ New Preset

Search a preset

New Template

Presets are reusable code snippets to base your new config-templates on. You can select a scenario to start from or create your own.

+ New Custom Preset

Scenarios

- Basic Spark configuration
- Spark Connect
- Spark Streaming
- DBT Integration
- Notebooks

```
1 {  
2   "type": "Python",  
3   "sparkVersion": "3.1.1"  
4 }  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20
```

Edit

Presets are reusable code snippets to base your new config-templates on. You can select a scenario to start from or create your own.

+ New Custom Preset

Scenarios

- Basic Spark configuration
- Spark Connect
- Spark Streaming
- DBT Integration
- Notebooks

```
1 {  
2   "type": "Python",  
3   "sparkVersion": "3.1.1"  
4 }  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20
```

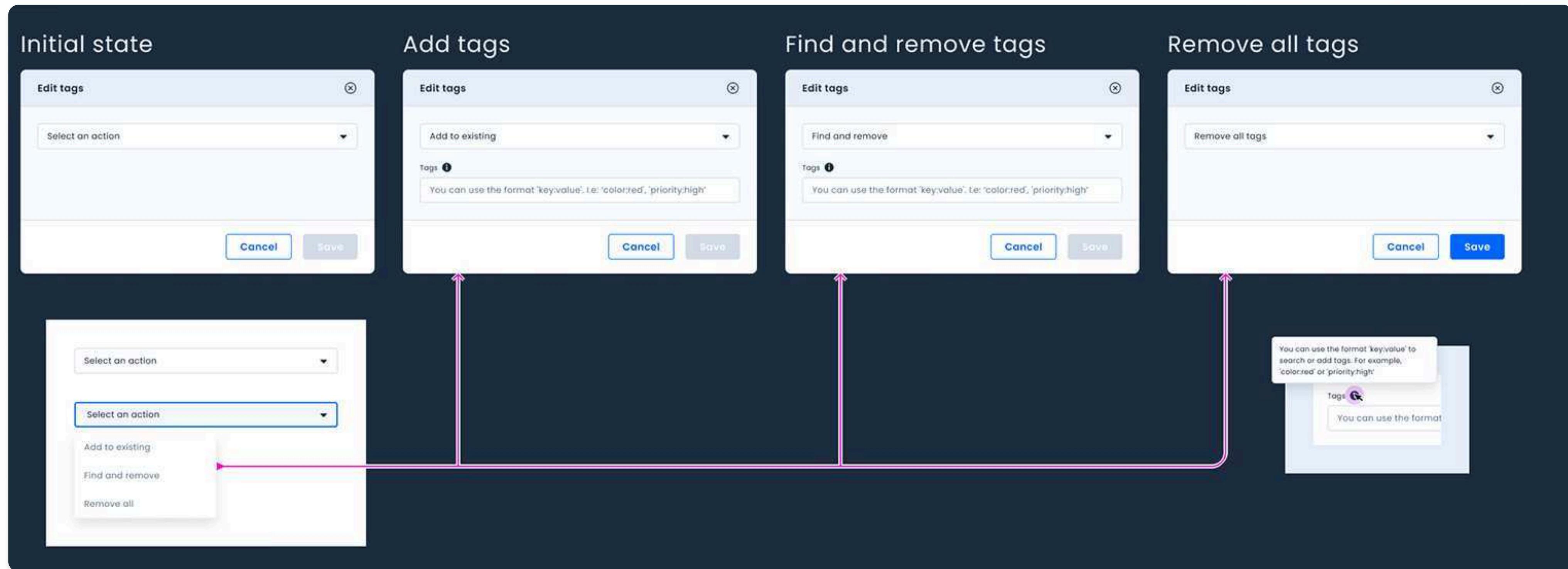
Edit

Create a template from this preset

User created preset specific for their needs

Common use-cases to aid selecting the right starter template

# Tagging system



# </> Ready for dev :)

**Main page**

- Config-templates page
- table
- Toggle tags
- Groups
- Templates selection
- template panel
- filter

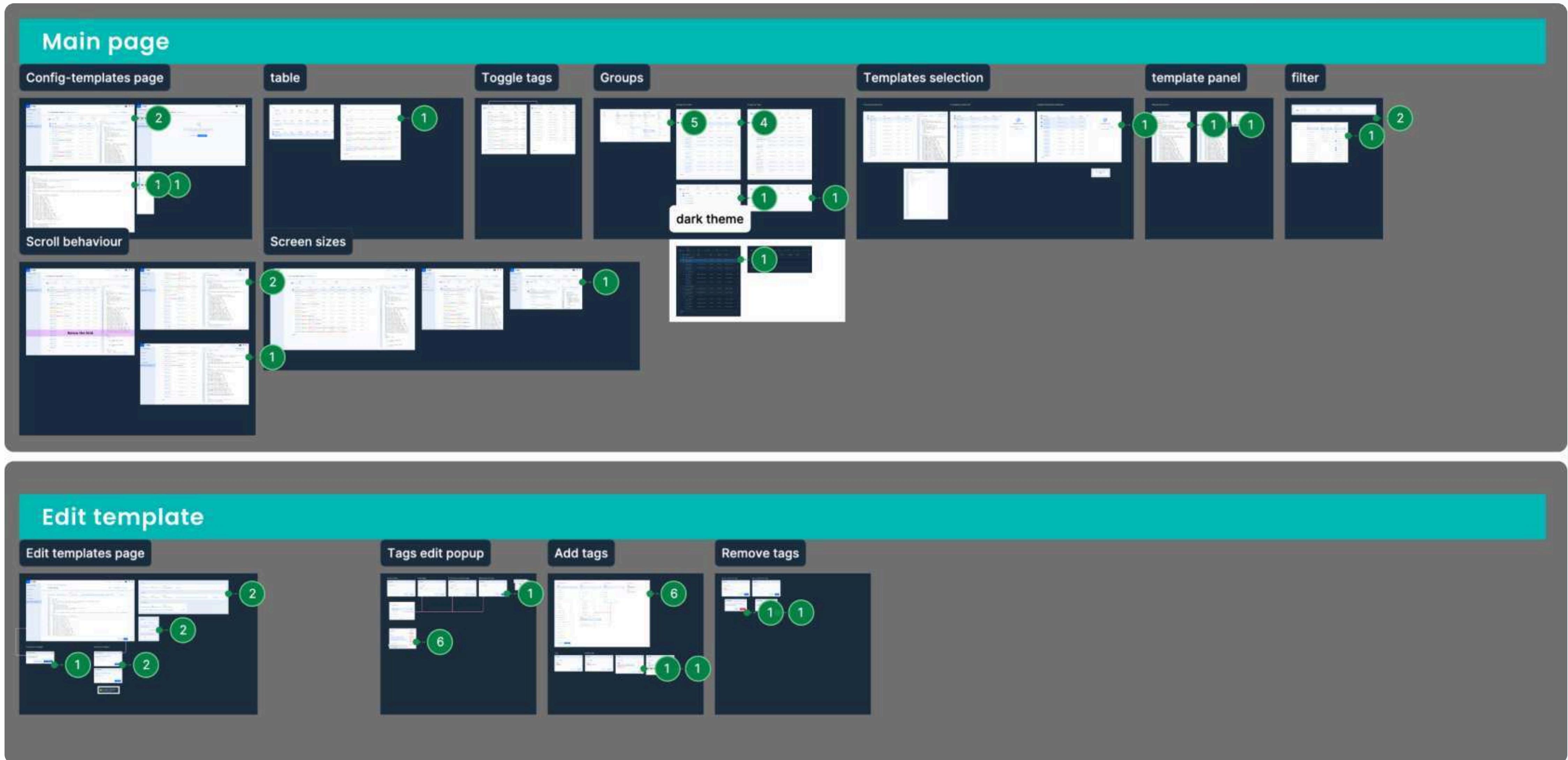
Scroll behaviour

Screen sizes

dark theme

**Edit template**

- Edit templates page
- Tags edit popup
- Add tags
- Remove tags



# So, how did we do?



## Time saving

Reduced troubleshooting and config-template creation time

Monitored via Pendo



## User satisfaction

More users per team will start working with the config-template

Monitored via Solution Architects



## Efficiency

Reduced the amount of unused config-template per customer

Monitored via BI tools



## Selling point

Helping tool for the sales team to recruit new clients over competitors

Monitored via sales reports



# Marketing campaigns monitoring tool

---

## Role

Product Designer

## Teammates

Product Managers,  
fullstack developers

## Deliverables

User research, wireframes,  
prototypes, UI design and design  
system creation

The goal

# Some context

- A web application designed to simplify marketing campaign management with a user-friendly interface
- It streamlines workflows and enhances collaboration, making campaign monitoring more efficient
- It had started as an inner product and expanded to be used with more than 30 partners

The goal

# What do we want to achieve?



## Real-time performance tracking

Support crucial decision making



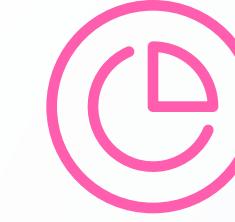
## Easy campaign selection

Better organization saves time



## Clear communication channels

Reduces frictions among business partners



## Transparent reporting mechanisms

Financial teams can monitor more closely



# Research

# Who are the users?

## Advertising and Marketing Managers

Responsible for driving online sales through digital marketing campaigns, optimizing customer retention, analyzing data, and collaborating with cross-functional teams.

“I need seamless tools to track performance, personalize campaigns, and scale efficiently while optimizing ad spend”

## Marketing affiliate manager

A strategic affiliate manager focused on partnerships, conversions, and commission optimization. Seeks reliable tracking, high-performing affiliates, and scalable growth.

“I need accurate tracking, strong relationships, and the right incentives to drive consistent, high-quality traffic”

Reaching out to the users

# Interview and surveys

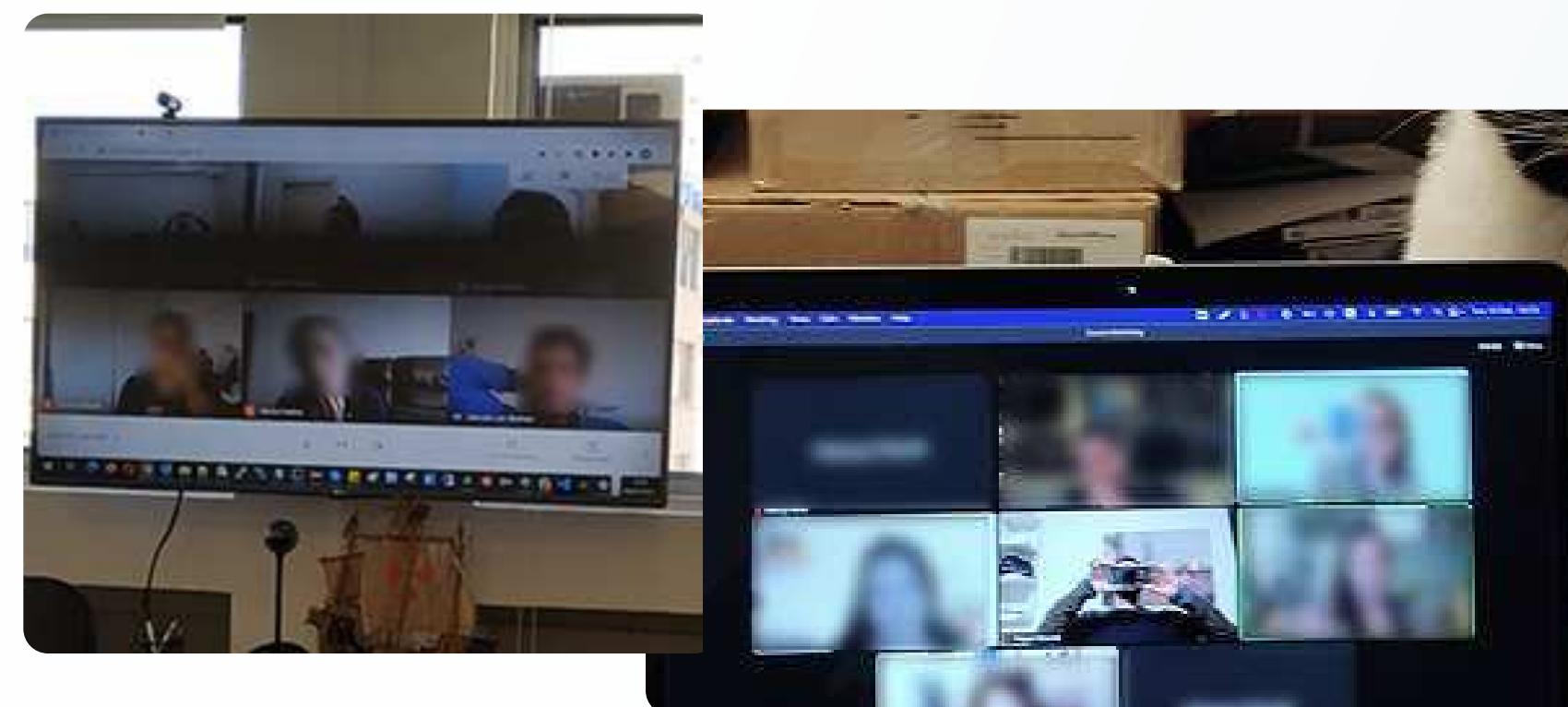
## Surveys

We composed a series of questions, sent it to partners and shared it within marketing and affiliates communities.

This survey is designed to gather insights about their experience with marketing campaign tools. We aimed to understand their needs, preferred tools, and challenges when working with partners.

## Interviews

We interviewed several affiliate managers, to gather insights on their experiences, challenges, and needs when working with marketing campaign management tools. The goal was to identify key pain points, preferred features, and opportunities for improvement to enhance usability and efficiency.



# Key Research Findings

**“As an marketing manager,  
I want to:**

- Improve affiliate engagement
- Ensure transparent reporting and analytics
- Track and compare performance to monitor spending
- Verify traffic quality

**“As an affiliate manager,  
I want to:**

- Optimize campaigns with real-time data
- Ensure fair and accurate payments
- Enable direct contact with advertisers for faster issue resolution
- Integrate multiple tracking tools into one dashboard

# Key Research Findings

## Insights

1. Most tools in the market fit less to medium\small networks
2. Simplify multi-channel campaign tracking
3. Streamline affiliate onboarding
4. Increase visibility into campaign metrics
5. Improve communication workflows



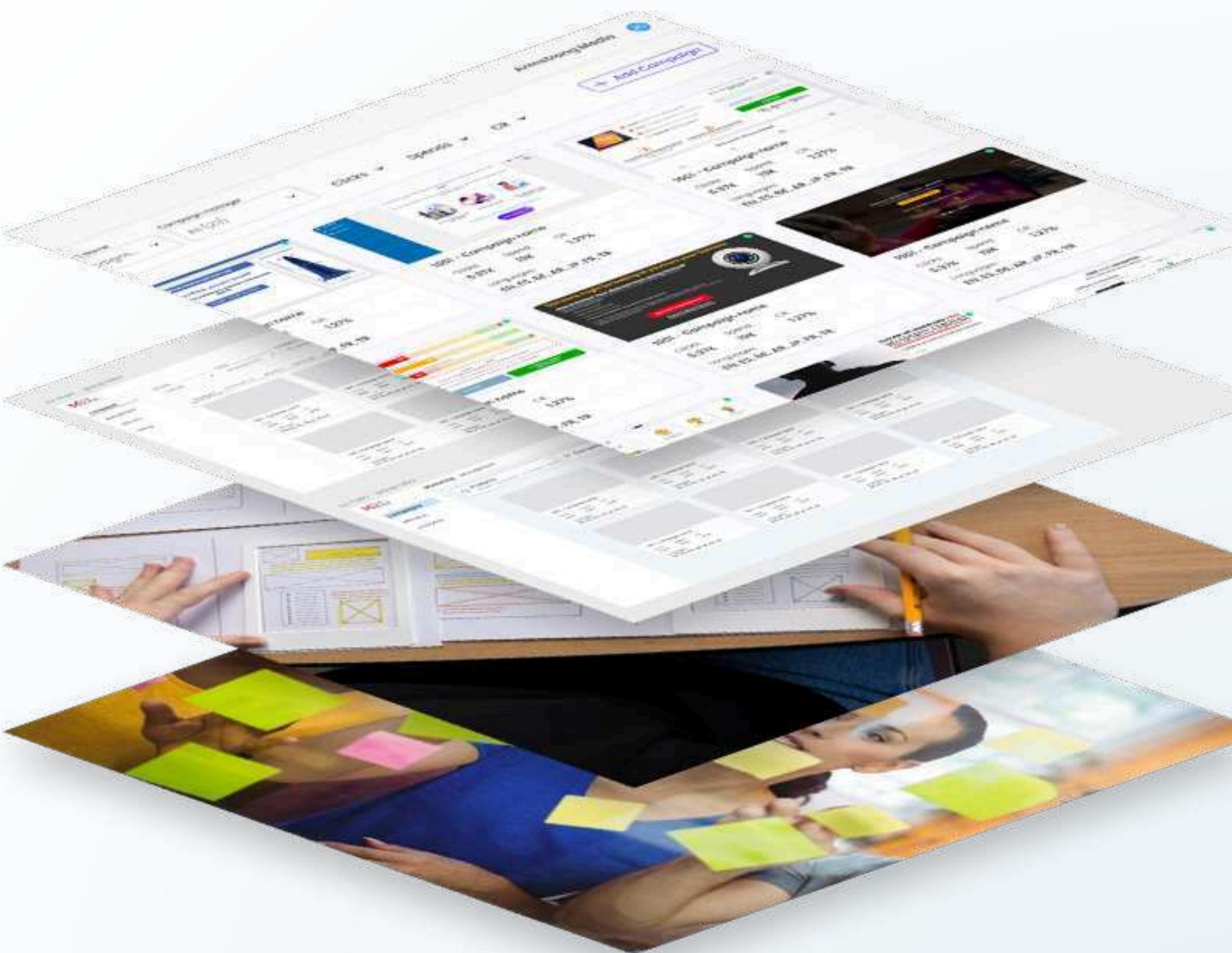
# Ideation & exploration

# Design solution

- UI including campaigns list and their data
- Performance dashboard with real-time stats
- Responsive design for all devices
- Embedding them within the emerging financial platform we had worked on

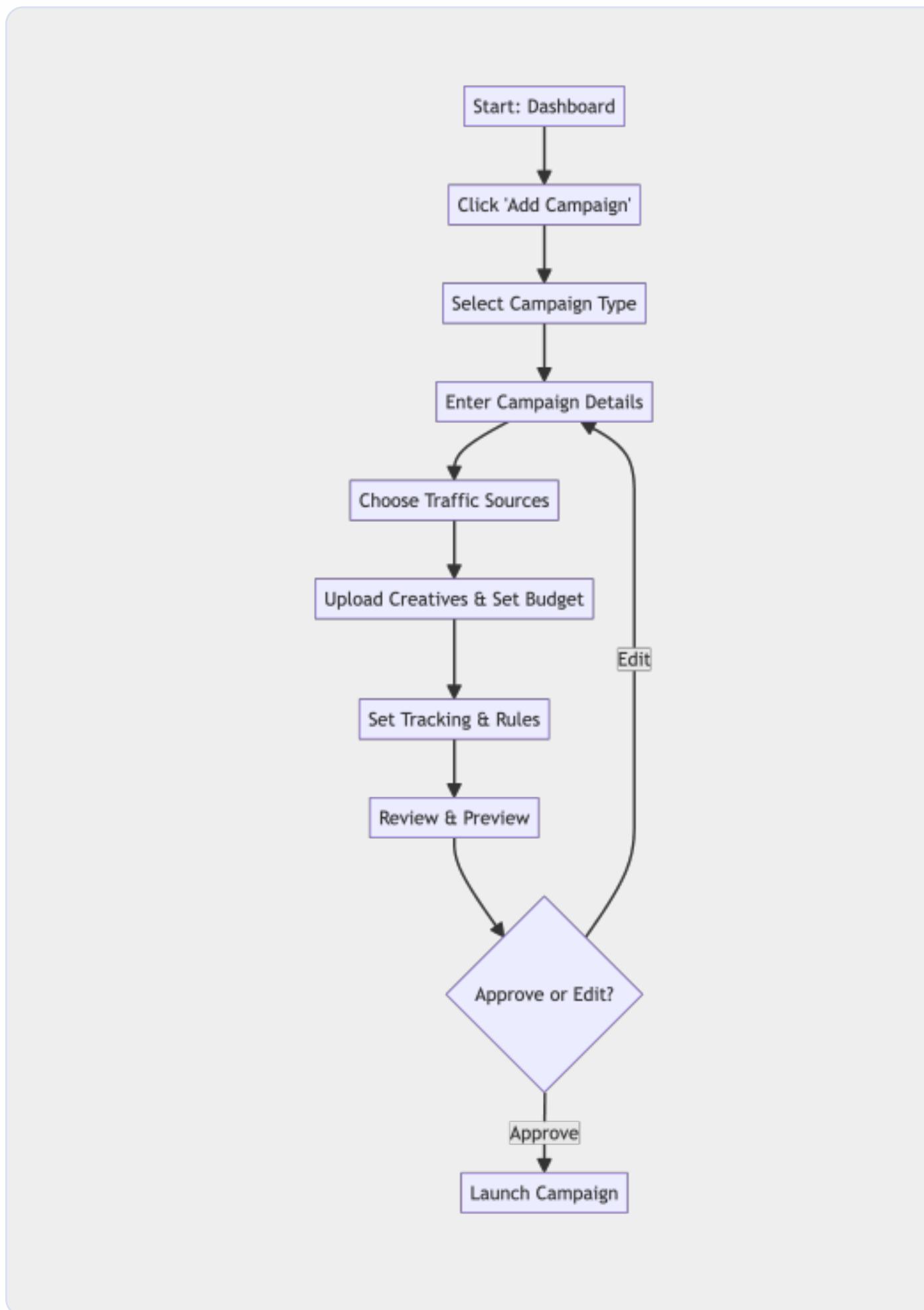
# Design process

- Information Architecture
- Designed an intuitive dashboard layout
- Developed clear navigation for campaign selection and management
- Wireframing & Prototyping
- Created low-fidelity wireframes
- Developed interactive prototypes for user testing
- Iterative design based on user feedback
- Design system and UI
- Developed a design system from scratch and implemented it over the wireframes

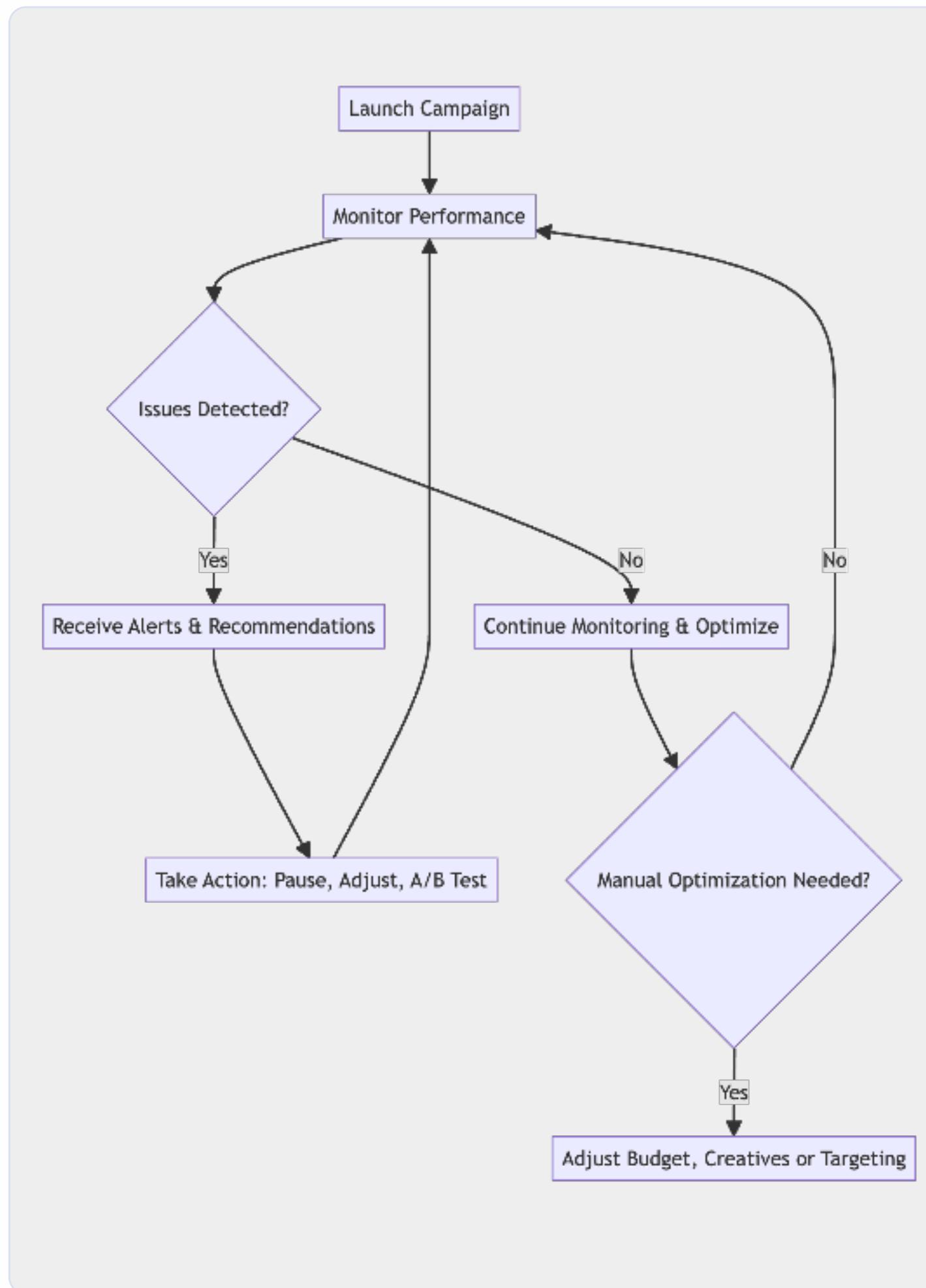


# User flows

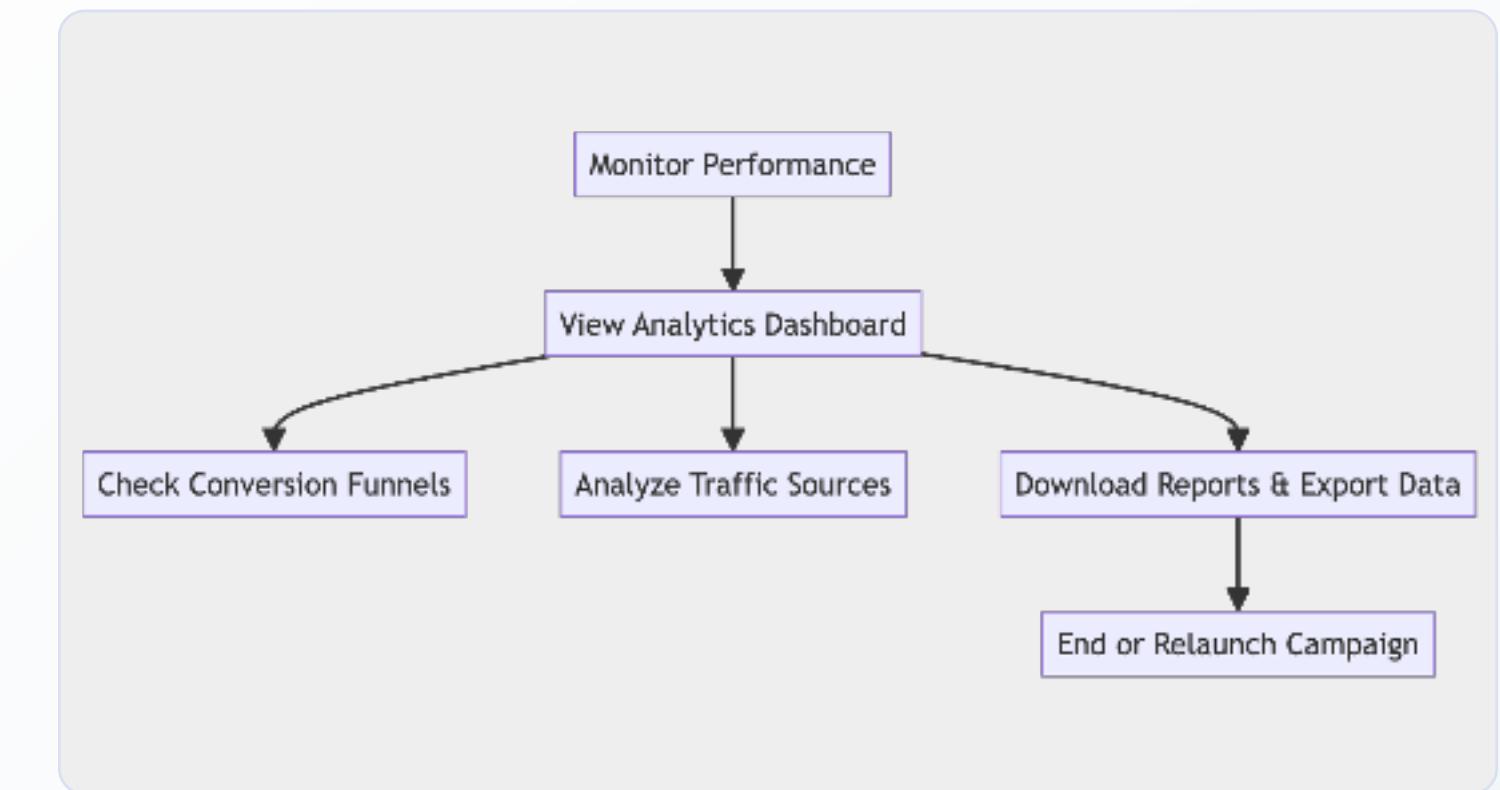
## Create campaign



## Edit



## Monitoring - main



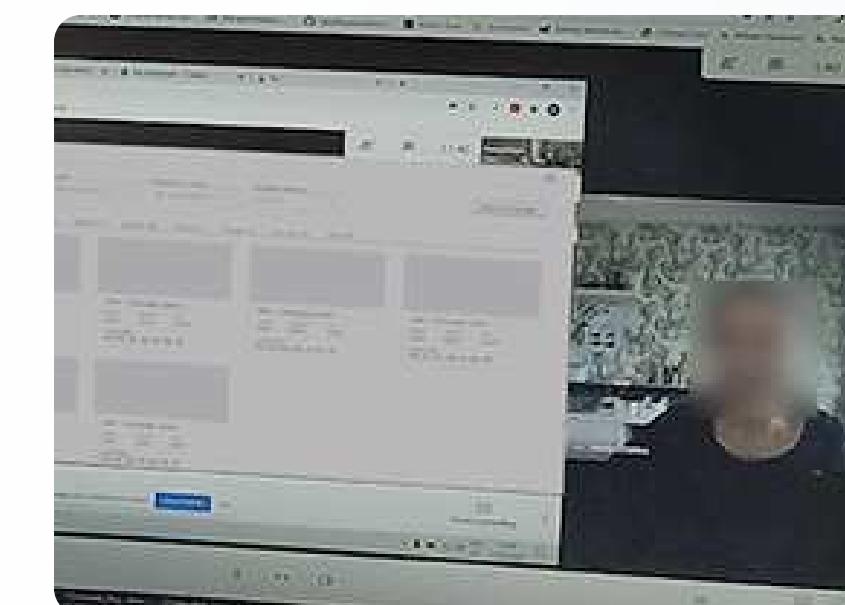
# User testing for grouping

## Expanded sections

The screenshot shows a user interface for 'CLIX SCALE'. On the left, there's a sidebar with 'Campaigns' selected, followed by 'Management', 'Affiliates', and 'Billing'. The main area has four tabs at the top: 'Group by product' (selected), 'Product', 'Campaign ID \ Name', and 'Campaign manager'. Below these are buttons for 'Add Campaign' and 'Add Product'. The main content area displays two sections: 'Games: 12' and 'Security: 6'. The 'Games' section is expanded, showing 12 items, each with a 'Campaign name' card. The 'Security' section is collapsed, showing a summary of 6 items. Each 'Campaign name' card contains fields for Clicks, Spend, CR, and Languages.

## Tabs division

The screenshot shows a user interface for 'CLIX SCALE' with a similar structure to the first one. It features a sidebar with 'Campaigns' selected, followed by 'Management', 'Affiliates', and 'Billing'. The top navigation bar includes 'Group by product' (selected), 'Product', 'Campaign ID \ Name', and 'Campaign manager', along with 'Add Campaign' and 'Add Product' buttons. The main content area is divided into several tabs: 'Security: 6', 'Downloads: 14', 'eBooks: 2', 'Games: 30', 'IQ Test: 7', 'Shops: 22', 'Sweeps: 31', and 'VOD: 28'. Each tab has a summary card with Clicks, Spend, CR, and Languages details. Below these tabs are four cards labeled '1001 - Campaign name' for each category, showing detailed metrics for each campaign.



# Some exploration

The collage consists of six screenshots arranged in two rows of three:

- Top Left:** A screenshot of a 'Gadgets' management interface. It shows a list of items under 'All products' with columns for Name, Type, Status, and Actions. Most items are listed as 'active'. One item has a status of 'inactive'.
- Top Middle:** A screenshot of an 'Edit account' dialog box. It shows fields for 'Edit account name' (www.website.com), 'Cap - Credit Card A' (USD 10,000 to USD 100,000), and 'Cap - Credit Card B' (USD 10,000 to USD 100,000). There are also 'Comments' sections for both cards.
- Top Right:** A screenshot of a dashboard titled '● Active'. It displays four main metrics: 'Credit Card A | 24.5% | 50(70)k' (90Tr), 'Credit Card B | 51.2% | 50(20)k' (40Tr), 'Volume | 8k' (10K), and a note 'Chase after silly colored fish toys around the house Dont wait for the storm to pass'.
- Bottom Left:** A screenshot of a 'Marketing' dashboard for '1001 - Campaign name stats'. It includes a bar chart for 'Clicks 12M' (50K/93K), a line graph for 'Impressions 12M' (22.3K to 14.5K), a bar chart for 'Runnel data' (Impressions 2.4M, Clicks 94K, Conversions 98K), and a stacked bar chart for 'Uv's By direct source'.
- Bottom Middle:** A screenshot of a 'Management' interface showing a grid of active bank accounts. Each account card includes details like 'Bank Name', 'Domain', 'Vista CB', 'Mastercard CB', 'MC CB', and 'Volume'.
- Bottom Right:** A screenshot of a 'Management' interface showing a grid of active bank accounts. This view appears to be a more detailed or specific subset of the data shown in the bottom middle panel.



**Final design**

# I deliver!

# Campaigns list

CLIX  
SCALE

Marketing Management

Armstrong Media

JC

Campaigns

Analytics

Affiliates

Group by  Product  Campaign ID \ Name  Campaign manager  Clicks  CR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

1001 - Campaign name

Clicks: 6.97K Spend: 19K CR: 1.27%

Languages:  
EN, ES, DE, AR, JP, FR, TR

# I deliver!

# Campaigns list

## Group by State, Product and Campaign manager

# Filtering and sorting mechanism

# Campaign actions

The image displays three screenshots of a software interface for managing campaigns. The central screenshot shows a tooltip for a campaign named "1001 - Campaign name". The tooltip contains the following information:

Clicks:	Spend:	CR:
6.97K	19K	1.27%

Below this, it lists the supported languages: EN, ES, DE, AR, JP, FR, TR. Two buttons are visible above the tooltip: "Stats" and "Edit". A black arrow points from the "Edit" button towards the "Edit" button in the tooltip. A pink arrow points from the bottom-left towards the tooltip, indicating its purpose.

Actions are exposed while hovering: Links to stats page and edit campaign

Upper head title

# Empty state

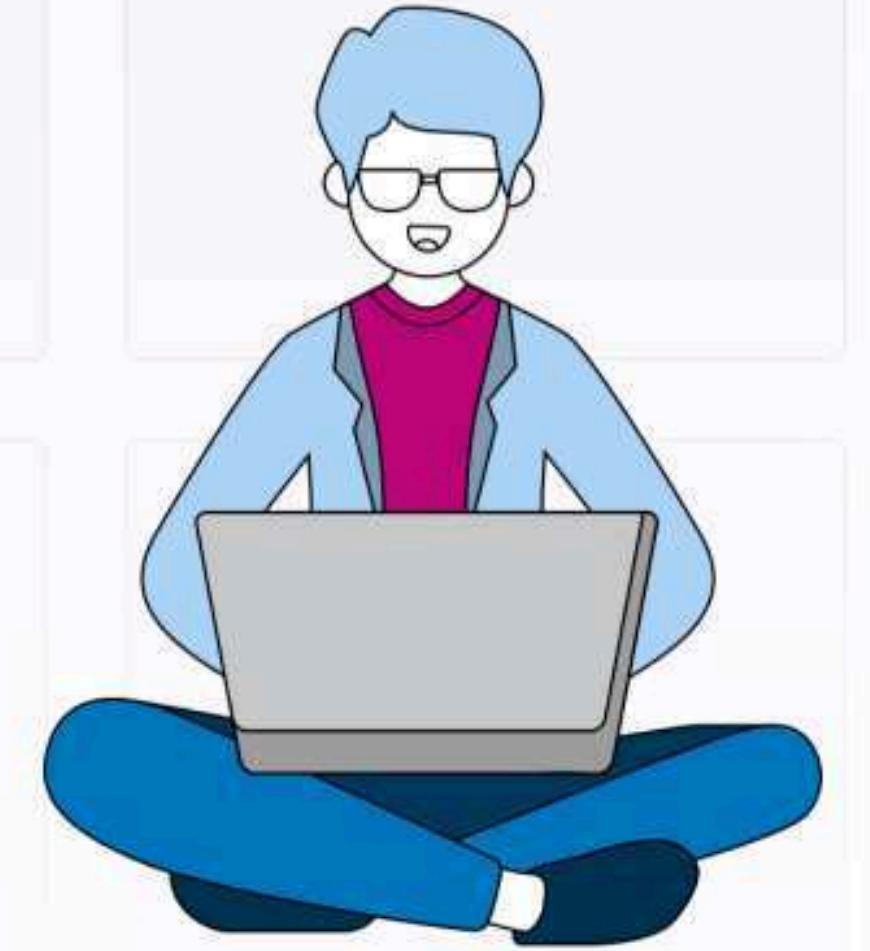
Marketing Management

CLIX SCALE Armstrong Media JC

Campaigns Analytics Affiliates

Group by Product Campaign ID \ Name Campaign manager Clicks Spends CR + Add Campaign

+ Add Campaign



It's time to create your first campaign!

+ Add Campaign

This screenshot shows the 'Marketing Management' section of the Clix Scale platform. The top navigation bar includes the Clix Scale logo, Marketing Management tabs, and account information for Armstrong Media. On the left, there are three main menu items: 'Campaigns' (selected), 'Analytics', and 'Affiliates'. The main content area features a search bar with dropdowns for 'Group by', 'Product', 'Campaign ID \ Name', 'Campaign manager', and metrics 'Clicks', 'Spends', and 'CR'. A prominent purple button '+ Add Campaign' is located at the top right of the search bar. Below the search bar, there are four large, empty rectangular boxes representing campaign slots. In the center of these boxes is a cartoon illustration of a person with blue hair and glasses, sitting cross-legged on the floor and working on a laptop. To the right of the illustration, the text 'It's time to create your first campaign!' is displayed, followed by another '+ Add Campaign' button.

# Edit campaign

CLIX SCALE Marketing Management Armstrong Media JC

Campaigns < Edit campaign: 1001 - Campaign name Active Delete Save

Analytics Affiliates

Product: Security

Campaign name: Campaign name

Campaign number: 1001

Languages: EN, ES, DE, AR, JP, FR, TR

Pre-lander  Landing page

url: <https://www.domain.com>

Direct URL  Redirect to page

Redirect from: <https://www>

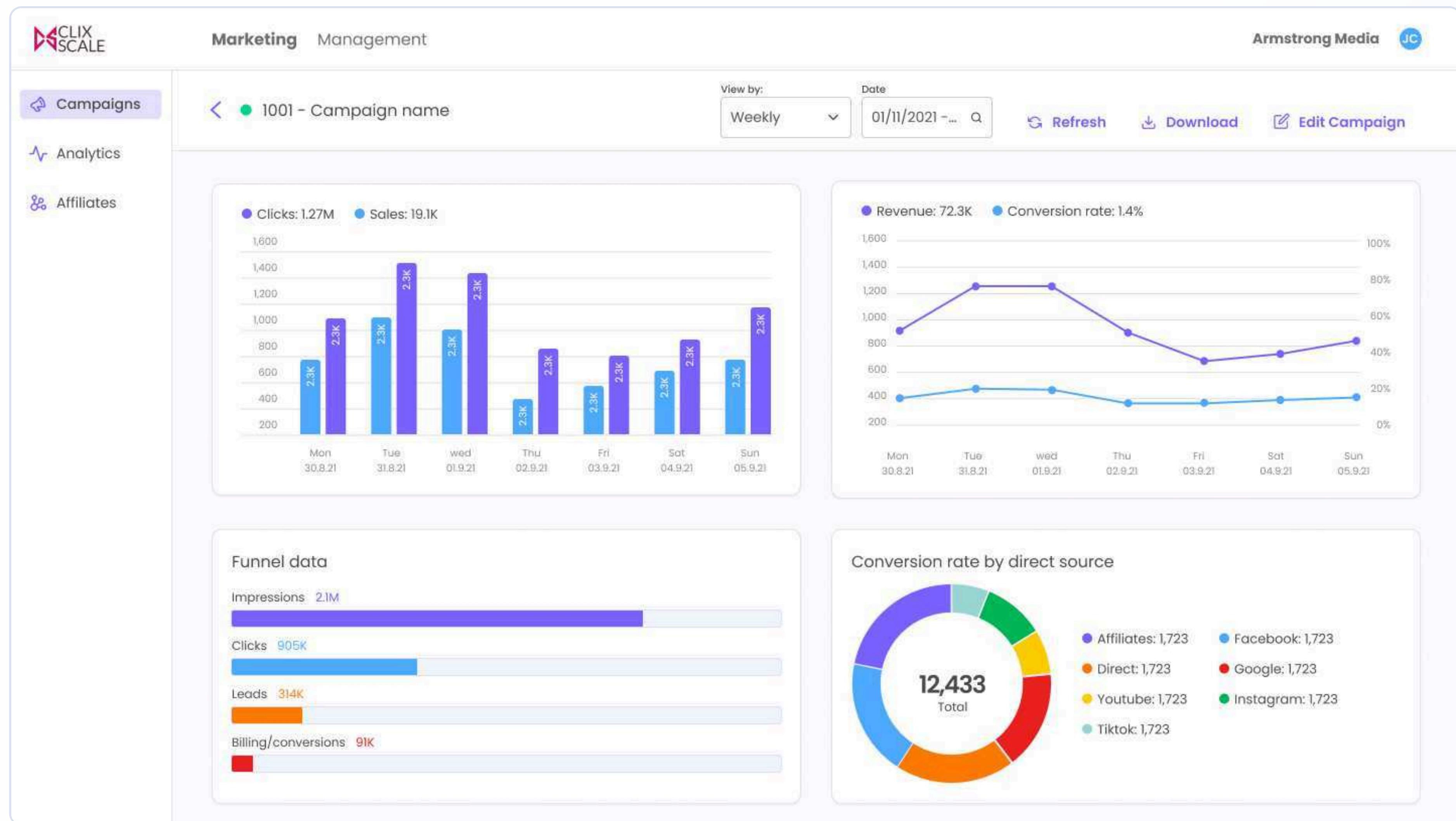
Comments

Award Winning Online Protection  
For a limited time only - Get it **FREE** now

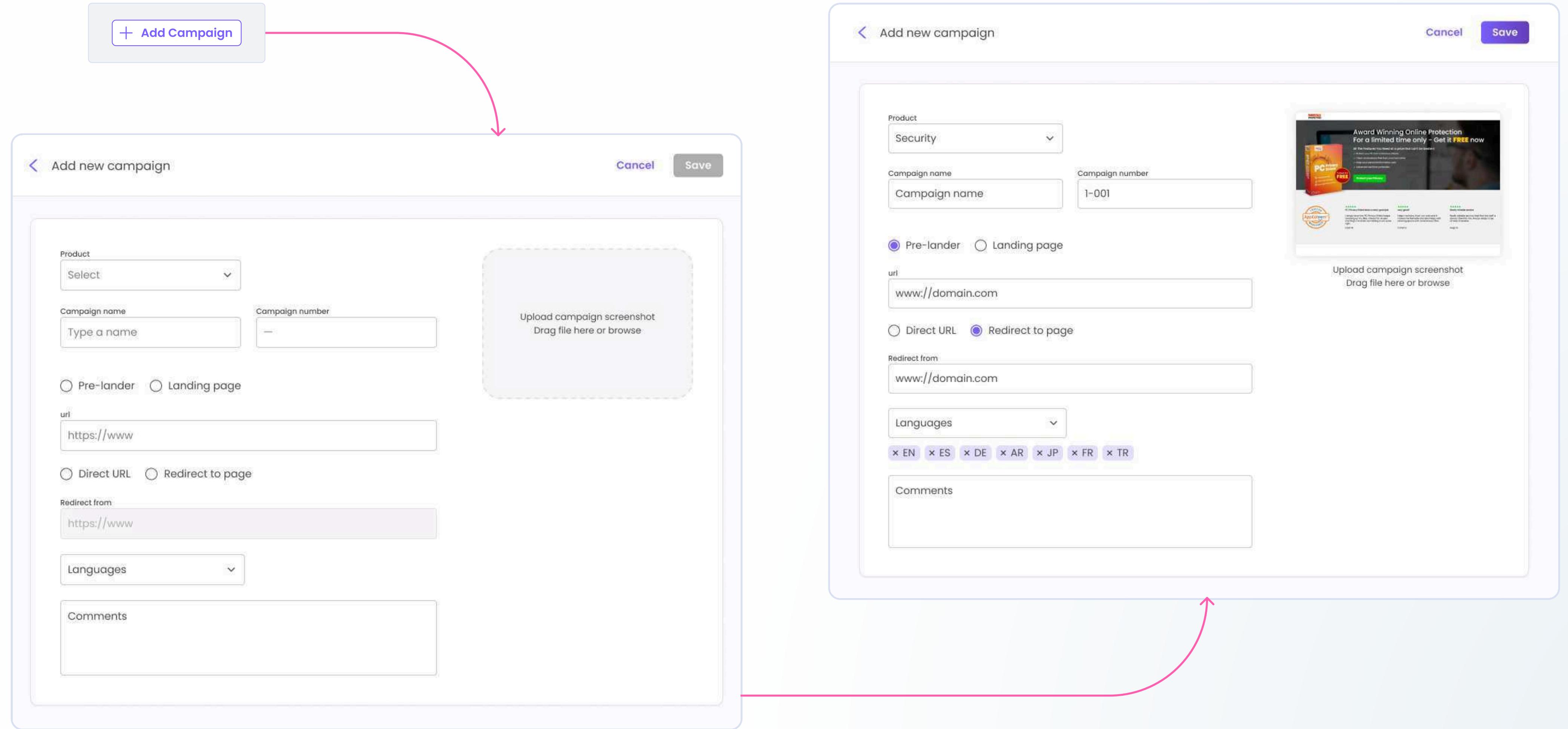
Upload campaign screenshot  
Drag file here or browse



# Campaign stats



# New campaign flow



## Additional screens

# Financial management

## Additional screens

# Financial management

The screenshot shows a financial management application interface. At the top, there's a navigation bar with the CLIX SCALE logo, Marketing, Management, and Armstrong Media/JC. On the left, there are sidebar tabs for Campaigns, Analytics, and Affiliates. The main area displays a grid of account cards. One card is highlighted and has a modal overlay titled "Edit: Account Name". The modal contains fields for "Change connector" (Select connector dropdown) and "Replace domain" (Select domain input). It also features two "Cap" sliders: "Cap - Credit Card A" set to USD 100,000 and "Cap - Credit Card B" set to USD 100,000. Below these sliders is a "Comments" text area. At the bottom of the modal are "Cancel" and "Save" buttons. The background grid shows various account details like Bank name, Connector name, and performance metrics (Credit Card A/B, Volume, 90Tr, 40Tr, 10K) across different categories (Active, Paused).

Marketing Management Armstrong Media JC

Campaigns Analytics Affiliates

Group by Banks Connectors

Default Select All (23)

Threat Status

Edit: Account Name

Active

Change connector Replace domain

Select connector Select domain

Cap - Credit Card A USD 100,000

Cap - Credit Card B USD 100,000

Comments

Cancel Save

• Active

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

• Paused

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

• Active

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

90Tr 40Tr 10K

Dont wait for

• Active

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

90Tr 40Tr 10K

Dont wait for

• Active

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

90Tr 40Tr 10K

Dont wait for

• Active

Bank: Bank name  
Connector: Connector name  
www.account-domain.com

Credit Card A | 24.5% | 50(70)k  
Credit Card B | 51.2% | 50(20)k  
Volume | 8k

Chase after silly colored fish toys around the house  
the storm to pass

90Tr 40Tr 10K

Dont wait for

## Additional screens

# Financial management - compact view

# Design system

The image displays a collection of user interface components and modal dialogs, illustrating a design system's visual language and interaction patterns.

- Primary Buttons:** A grid of buttons showing various states:
  - Style=main: State=default
  - Style=main: State:hover
  - Left icon: + Button
  - Right icon: Button ↴
  - 2 icons: + Button ↴
  - Only icon: +
- Secondary Buttons:** A grid of buttons showing various states:
  - Style=outline: State=default
  - Style=outline: State:hover
  - Left icon: + Button
  - Right icon: Button ↴
  - 2 icons: + Button ↴
  - Only icon: +
- Inputs:** A section showing different input types and states:
  - Empty: Filter by name or ID...
  - Hover: Filter by name or ID...
  - Focus: I
  - Filled: Typed text
  - Filled: Typed text  
Error message
- Selectors:** A section showing chip pill and checkbox states:
  - Chip pill: Hover on remove (with a cursor icon)
  - Checkboxes: Empty (unchecked), Filled (checked), Disabled (disabled checked)
  - Radio buttons: Empty (unchecked), Filled (checked), Disabled (disabled checked)
  - Switches: Empty (off), Filled (on), Disabled (disabled on)
- Delete modal:** A modal dialog titled "Delete modal" with the message "You are about to delete campaign:" followed by "ID: 1023" and "Campaign: Campaign name". It includes a "To confirm, type 'Delete me'" input field and "Cancel Action" and "Delete" buttons.
- Sidebar:** A sidebar menu with sections for "Campaigns", "Analytics", and "Affiliates".

**Thank You :)**