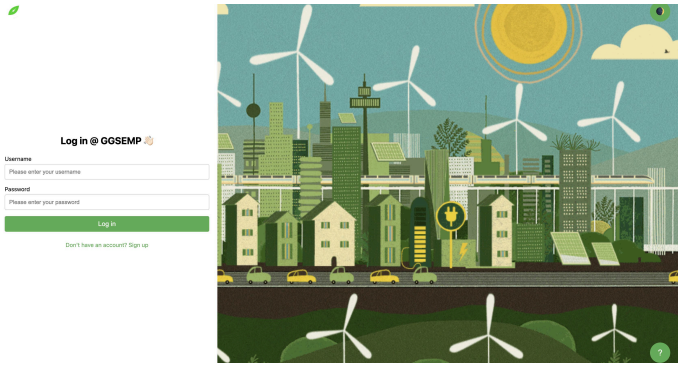
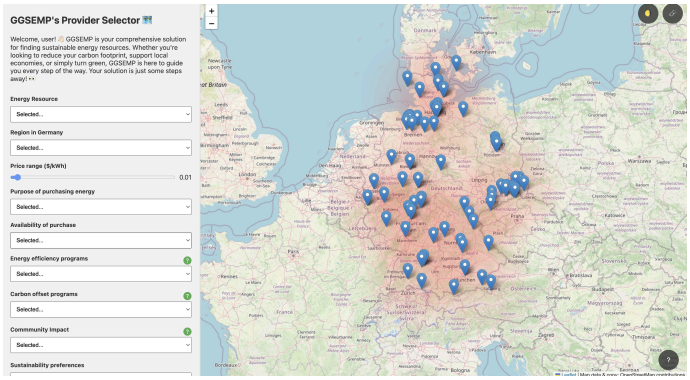
Bachelor Practical Course: Evaluation of User Interfaces

3rd Phase Documentation

Team 5 : Edera Ndoj + Ayca Yapar





Submitted to: Dr. Wolfgang Wörndl

Documentation outline of the third phase of the practical course:

A. Extended User Interface Prototype of GGSEMP

B. Controlled Experiments

C. Analysis of the Experiments

D. Conclusions

Documentation of the third phase of the practical course

A. Extended User Interface Prototype of GGSEMP

A.1 Bringing Figma concepts and design to life with HTML, CSS and JavaScript

After developing a partially functioning prototype on FIgma, it became evident that the user interface did not allow for the user-defined input or various choice of conditions, which were crucial aspects of the design. Our team engaged in brainstorming sessions to explore alternative solutions that would enable this functionality. This effort in alignment with the requirements for the third and final phase of the practical course. In order to finalise controlled experiments, it was essential for us to implement and focus on quantitative changes within the prototype - something that could not be effectively and efficiently achieved using Figma alone.

Given these limitations, we decided to transition from Figma to a more robust development approach using web technologies. Specifically, we opted to create a map interface leveraging HTML, CSS and JavaScript, which allowed us to incorporate dynamic features and data handling that were essential for the project's success.

A.2 Detailed explanation of the extended prototype

A.2.1 Project structure

The prototype is the finalised product of a project, structured into three distinct files:

- `index.html`: This file serves as the main entry point for the user, offering the option to log in or sign up, just like in the Figma prototype. It also includes the associated CSS for styling the page.
- `map.html`: This file contains the HTML structure for the map interface (including a heat map extension and pins locating the providers on the map) and also a filtering section for provider selection, built using the Leaflet library, and is accompanied by the necessary styling to ensure a smooth user experience.
- `app.js`: This JavaScript file is responsible for the map's functionality, including the display and interaction with the energy provider data for Germany. It processes the list of providers and allows users to filter them based on various criteria, such as: resource type, federal state, price range, reason of purchase, carbon offset program etc., as well as interact with the map to view provider locations.

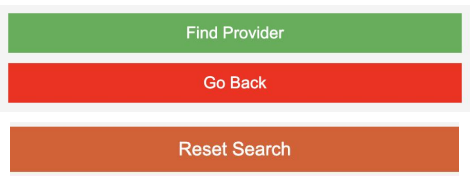
This transition to a full-fledged-web-based implementation has significantly enhanced the interactivity of the prototype, facilitating the completion of the experiments and extending the project in accordance with the course requirements.

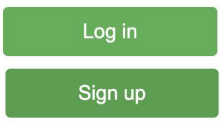
A.2.2 Design of the extended prototype

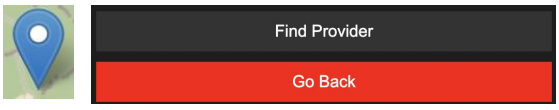
The design of the extended prototype was carefully crafted to fulfil user requirements and needs which our team gathered from the second phase qualitative user evaluation. We also tried to mimic the design and layout of the first prototype with the aim to offer the users a familiar and user-friendly environment. Listed you will find all elements used in the design of our extended prototype, including both `index.html` and `map.html`:

Logo: 

Buttons: 







Font Family: `system-ui, -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen, Ubuntu, Cantarell, 'Open Sans', 'Helvetica Neue', sans-serif;`

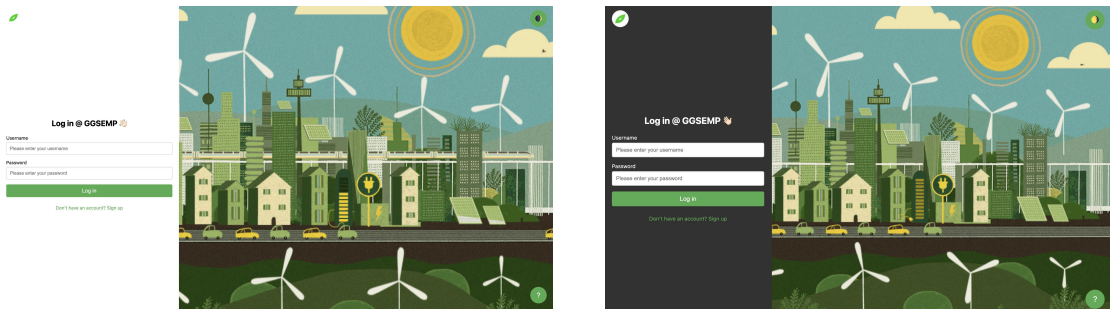
Color Palette:

#fff #333 #121212 #e0e0e0 #f4f4f4 #ddd #4CAF50 #45a049 #555

#2c2c2c #FF0000

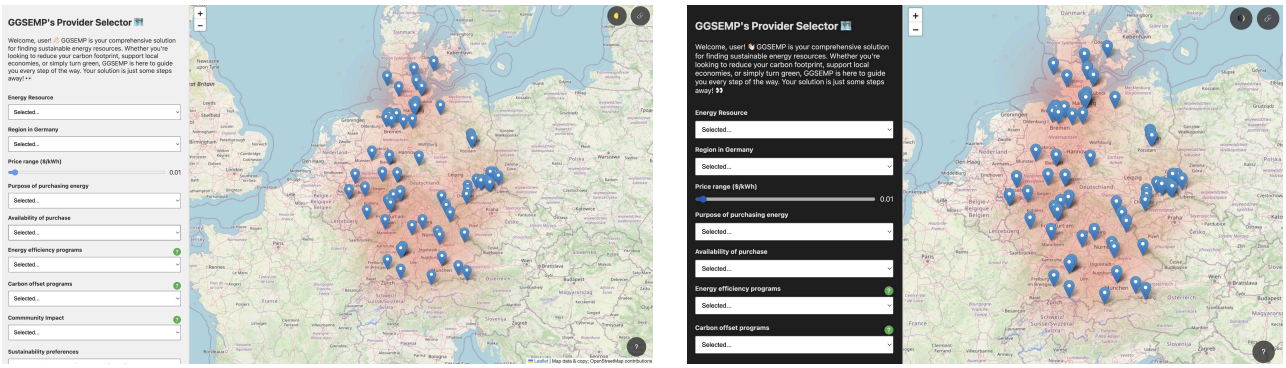
Emojis and Icons: 👋🚀👀🌿🌕🌑💨💧💸🏚️🏙️🔳🧪🎁🔴🟠🟢🔗🗺️🌝🌋

Design of the Log in/ Sign up Page:



The Log in/Sign up page contains of one GIF as a background (also used on the Figma prototype) and the required fields to log in or sign up into our platform. Other than that, the logo is placed on the top left, while on the top right there is our new “Dark/Light Mode” feature. On the bottom right, there is a helper button.

Design of the Map Page:



The Map Page is divided into 2 important parts: The Provider Selector Section and the Map Section. The Provider Selector is a bundle of questions/criteria selections that help the user find the desired provider in the map. On the right side of the selector, the map of Germany is displayed with the energy providers placed in different federal states. This page also has a “Dark/Light Mode” button and a helper button (displaying the price range of all energy resources). Additionally, there is a linking button that will lead the user to the Figma prototype (if needed to see the provider catalogue, support team etc).

A.3 Interactive and multiple conditions of the map interface

The map interface enables users to filter and interact with energy providers based on various criteria. It dynamically adjusts the displayed data on the map according to user input, enhancing usability and functionality compared to the Figma prototype:

1. **Filter-based map updates:** The interface includes filters such as energy resource type, region, price range, purpose of purchase, availability and additional factors. Filters dynamically adjust the map’s displayed provider pins, ensuring only relevant providers appear based on the selected criteria.

2. **Range sliders for price input:** A price range slider enabled users to specify the maximum energy price they are willing to pay. It is dynamically displayed as users adjust it, providing immediate feedback.

3. **Tooltips for guidance:** Contextual help is available for certain filters (e.g carbon offset programs, energy efficiency), using tooltips that appear when hovering over helper icons (?).

4. **Selection of multiple conditions:** Filter support selection of specific combinations between energy types, region or sustainability measures etc. This allows users to tailor their search to match their unique preferences and requirements in fewer steps in comparison to the Figma prototype.

5. **Dynamic and interactive design:** The interface utilises real-time updates, ensuring changes to any filter are immediately reflected on the map. Additional visual elements (e.g color-coded labels, hover effects) provide feedback for user actions.

6. **Responsive design:** The layout is designed to adapt to various screen sizes, ensuring usability across devices. The map and the form container are flexibly positioned to maximise space. Also, the interface alerts the user if e.g: provider found/not found, when a required filter is not selected etc.

7. **Dark/Light mode toggle:** A toggle button enables users to switch between dark and light themes, ensuring accessibility and comfort in different lighting conditions.

8. **Filters with `select` elements:** Dropdown menus provide predefined options for each filter (e.g “Wind”, “Solar” for energy resources).

9. **Map integration:** GGSEMP’s most recent platform is focused on the map interface, which was brought to life by using the Leaflet.js library, which is a lightweight and open-source JavaScript library for interactive maps. This integration allows users to explore sustainable energy providers across Germany in an intuitive and user-friendly way. OpenStreetMap is used as the tile provider for displaying map data. Users can pan and zoom across the map to explore various locations. They can also click on the location pins which will then display a compact information on the selected provider. The map is accessible only to authenticated user. If a user is not logged in, they are redirected to the log in page.

10. **Call-to-action buttons:** The form includes buttons for submitting search, resetting search and accessing the main entry platform, styled with hover actions to indicate interactivity and click actions to perform certain processes.

Note that none of the filtering criteria is thought/implemented as `<required>`. So, the user can simply see the total number of pins in the map without selecting any filters. Or, the user can also simply select a combination of preferred criteria to his/her liking (so, not all of the criteria listed on the form) and get a number of limited pins on the map. This will ensure a better non-tiring, user-friendlier experience while using the GGSEMP’s Provider Selector. Selecting more filters will leave the user with less possibilities to choose from. This decision was made after having gone through our experiment results, we came up with this idea to provide a better experience for the user while using our map selector interface.

A.4 Linking Figma prototype with “GGSEMP’s Provider Selector”

Since our team’s goal was to extend functionalities and interactivity only on the map page, we focused on developing such platform. We also thought to adjust and link our existing prototype with the Figma version as well. By clicking the button displayed above, the user will be taken to the Figma prototype. Nevertheless, the controlled experiments and testing where only focused on the map interface of the two prototypes. This link was also used by one of the groups that participated on our controlled experiments (more on that on: **B. Controlled Experiments**).

B.1 Quantitative evaluation

Quantitative evaluation aims to measure the differences between the old and new prototypes using both objective and subjective metrics. This method ensures that the impacts of the changes made to the prototype are measurable, comparable, and statistically significant. The primary goal of quantitative evaluation is to assess whether the new features and enhancements introduced in the code-based platform improve user performance and satisfaction compared to the Figma prototype. This is done by analyzing measurable outcomes such as task completion time, click counts, and user feedback on the usability of the interface.

A/B Testing: Participants are asked to complete identical tasks on both the old and new platforms, allowing for a direct comparison.

Objective Metrics: Data is collected on metrics like the time taken to complete tasks, the number of clicks needed, ***(and the frequency of hover usage).***

Subjective Metrics: User preferences, satisfaction levels, and perceived ease of use are evaluated through a user survey.

Task Assignment: Participants are given tasks such as locating a provider, applying filters, or viewing provider details on both platforms.

Data Collection: Performance data (time, clicks,user feedback) along with subjective feedback is gathered during the task completion process.

Statistical Analysis: Normality tests (Shapiro-Wilk) are conducted to assess the data distribution. Appropriate tests ***(e.g., two-sample t-tests and Mann-Whitney U tests)*** are utilized to identify significant differences in performance metrics.

B. 2 Conditions and variables

B. 2. 1 Independent variables (IV)

Listed below are the IVs we decided to manipulate and focused on while performing the controlled experiments, each with 2 values:

- *Map Visualization 1*: **Both** Zoom in/out and Move left/right function **vs** **Only** Move left/right function

The updated platform includes a zoom in/out feature that simplifies map navigation for users. This zoom functionality enables them to focus on particular areas, unlikely the previous prototype, where users need to first filter the options to see the map of providers. By zooming into users' chosen location, users can easily access information about providers by clicking on the pins displayed on the map.

- *Map Visualization 2*: Heat Map **vs** Standard Map

The updated platform introduces a heat map feature that displays the density of providers in different regions. This improvement enables users to quickly pinpoint areas with the highest number of providers, simplifying the process of comparing regions. The previous version did not have this feature, which meant users had to search manually for these insights.

- *Provider Details:* Compact view **vs** Detailed view

The updated platform presents provider details in a concise format. This design simplifies the information, emphasizing the key details and allowing users to focus on the main points without extra effort. The compact view helps users in saving time while reviewing provider details and improves overall usability.

- *Layout:* Dark mode **vs** Light mode

A dark mode option has been introduced added the existing light mode. This feature allows users to select a layout that matches with their visual preferences, helping to reduce eye strain, especially in dimly lit environments. This update improves accessibility and comfort for longterm use.

- *Button Shape*: Square buttons **vs** Round buttons

The updated platform contains rectangular buttons instead of round ones. This design choice was made for its sleek and contemporary look, improving the interface's overall polished and official appearance.

- *Filtering Options*: Detailed filtering **vs** Compact filtering

The updated platform contains new filtering options that significantly boost the user's ability to narrow down their provider searches. Opposite to the previous prototype, which only had basic filters like "State" and "Energy Type," the updated prototype offers more comprehensive options such as ***price range, region, availability, sustainability preferences, and additional criteria.*** These improvements enable users to find suitable providers with increased efficiency.

- *Hover Descriptions:* Enabled **vs** Disabled

Hover descriptions were added to give users more information about the filter criteria. When users hover over some of the filters, short explanations pop up, helping them understand how to use the filters properly. This feature enables users to make better choices when applying filters, reducing the need for guesswork.

B. 2. 2 Dependent variables (DV)

The evaluation of the prototype changes is based on two categories of dependent variables, which are objective DVs and subjective DVs.

Objective DV
These metrics define user performance during the tasks:

Time taken to complete tasks: The duration required by participants to accomplish specific actions on both the previous and updated platforms.

Number of clicks to achieve a specific action: The total number of clicks (interations) recorded while completing each task.

Subjective DV
Subjective feedback was collected using a Google Form and focusing on the user perceptions of the interface:

Ease of Use: The users were asked how easy and uncomplicated the platforms were for performing a number of actions.

Responsiveness: The users were asked how responsive and fast the exchanges were by the two platforms.

Information Clarity: Feedback included the clarity and understandibility of the information displayed, including details about providers and the filtering options.

Feature Effectiveness: The users evaluated the effectiveness of different new functions incorporated in the code-based platform like compact views, hover descriptions, and more advanced filters.

Platform Preference: The users expressed their preferences of the previous protoype and the updated prototype with regard to different interactions.

Satisfaction and Appeal: The users expressed their levels of satisfaction towards the platforms' appeal, layout, and the final completion of the tasks.

B.2.3 User Tasks for the Experiment

Task 1: Finding a Provider by Location
This task evaluates how well users can find a provider in a specific area by using the filtering and navigation tools available on both platforms.

Steps:
Figma Prototype:
- Set the "State" filter to "Berlin."
- Select any type of energy using the "Energy Type" filter (for instance, "Solar Energy").
- Display a provider on the map within Berlin.

Code-Based Platform:
- Set the "Region" filter to "Berlin."
- Choose random or default options for the other filters to proceed with the task.
- Display a provider on the map within Berlin.

Comparison Questions:
1. Was it easier to select the desired region in the code-based platform compared to the Figma prototype
2. Did the map respond faster in the code-based platform compared to the Figma prototype?
3. Were the provider's details easier to view in the coded platform?
4. Which platform do you prefer for finding a provider by location?

Task 2: Viewing Provider Details
Evaluate how easily users can access and understand provider information, considering the differences in workflow between the Figma prototype and the code-based platform.

Steps:
Figma Prototype:
-Use the "State" filter to select a specific state (e.g., Lower Saxony).
-Apply the "Energy Type" filter to choose a spesific energy type (e.g., Wind Energy).
-Click "Go to Map" and select any provider on the map.
-Review the details of the provider.

Code-Based Platform:
-Click on any provider on the map to view its details without needing to pre-select filters.

Comparison Questions:
1. Did the coded platform allow you to access provider details faster compared to the Figma prototype (which required pre-selecting filters)?
2. Did the compact view in the code-based platform provide enough information to identify providers quickly?
3. Was the workflow in the coded platform more straightforward than in the Figma prototype?
4. Did you find the layout of provider details clearer and more appealing in the coded platform?
5. Which platform's provider details feature do you prefer overall?

Task 3: Filtering Options
Assess how easy and effective it is to use the additional filtering options in the code-based platform compared to the simpler filtering options available in the Figma prototype.

Steps:
Figma Prototype:
-Use the "State" and "Energy Type" filters to find providers that offer "Wind Energy" in Lower Saxony.
-Take note of how long it takes to apply these filters and locate the relevant providers.

Code-Based Platform:
-Make use of the advanced filtering options to refine the list of providers to those that:
  - Provide "Wind Energy." which are located in Lower Saxony. and have a price cap of €0.25/kWh.
  - Focus on community impact or have carbon offset initiatives.
-Log the time taken to apply these filters and identify the relevant provider.

Comparison Questions:
1. Did the additional filtering options in the code-based platform make it easier to find a suitable provider compared to the Figma prototype?
2. Was the filtering process in the code-based platform more intuitive despite having more options?
3. Did the limited filtering options in the Figma prototype make the task less efficient?
4. Which platform provided a better overall experience for filtering providers?

Task 4: Price Selection Function
Assess whether the hover descriptions in the code-based platform enhance users' understanding of filter options compared to the Figma prototype, which does not conatin any hover descriptions.

Steps:
-Code-Based Platform:
-Hover over additional filter options (e.g., "Community Impact" or "Carbon Offset Programs") to see their descriptions.
-Use the "Region" and "Energy Type" filters to find providers offering "Solar Energy" in Hamburg.
-Choose random or default options for the other filters to proceed.

Figma Prototype:
-Use the "State" and "Energy Type" filters to select "Solar Energy" in Hamburg.
Comparison Questions:
1. Did the hover descriptions in the coded platform make the filtering options easier to understand?
2. Was the simplicity of the Figma prototype more helpful for quick filtering?
3. Did the additional hover-based filters in the code-based platform feel more comprehensive than the limited options in the Figma prototype?

B. 4 Introduction to our Null Hypothesis
This part includes usertesting and performance metrics between a Figma prototype and a coded platform in-depth.
To assure the results would be as accurate as possible, two-sample t-tests were done. Analysis is guided by clearly stated hypotheses on each of the tasks that will look into user efficiency, comprehension, and interaction patterns across the two platforms.

Finding a Provider by Location Description

Null Hypothesis (H_0): There is no significant difference in the time taken in a specific region between the Figma prototype and the coded platform.
Alternative Hypothesis (H_1): There is a significant difference of clicks for finding a provider in a specific region between the Figma prototype and the coded platform.

Null Hypothesis (H_0): There is no significant difference of clicks for finding a provider in a specific region between the Figma prototype and the coded platform. (Note that all participants were obligated to fulfill the selection form completely. Hence there are significant more clicks and filters compared to Figma prototype.)
Alternative Hypothesis (H_1): There is a significant difference in the time taken in a specific region between the Figma prototype and the coded platform

Viewing Provider Details

Null Hypothesis (H_0): Users will not access or comprehend provider details significantly faster or more intuitively in the code-based platform compared to the Figma prototype.
Alternative Hypothesis (H_1): Users will access and comprehend provider details significantly faster and more intuitively in the code-based platform compared to the Figma prototype.

Filtering Options Description

Null Hypothesis (H_0): The additional filtering options in the code-based platform will significantly reduce the number of clicks to find suitable providers compared to the limited options in the Figma prototype. . (Note that all participants were obligated to fulfill the selection form completely. Hence there are significant more clicks and filters compared to Figma prototype.)
Alternative Hypothesis (H_1): The additional filtering options in the code-based platform will not significantly reduce the number of clicks to find suitable providers compared to the limited options in the Figma prototype.

Null Hypothesis (H_0): The additional filtering options in the code-based platform will not significantly reduce the time taken to find suitable providers compared to the limited options in the Figma prototype.
Alternative Hypothesis (H_1): The additional filtering options in the code-based platform will significantly reduce the time taken to find suitable providers compared to the limited options in the Figma prototype.

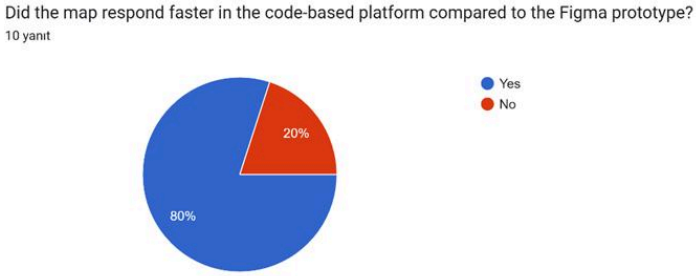
Price Selection Function

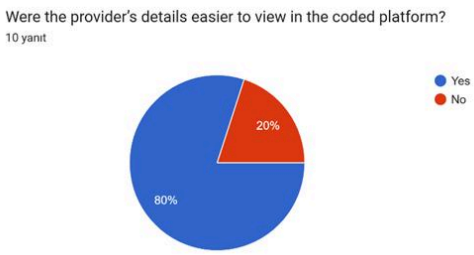
Null Hypothesis (H_0): Price selection function in the code-based platform will not significantly reduce time taken for users to find and compare the prices of certain providers of a certain area.
Alternative Hypothesis (H_1): Price selection function in the code-based platform will significantly reduce time taken for users to find and compare the prices of certain providers of a certain area.

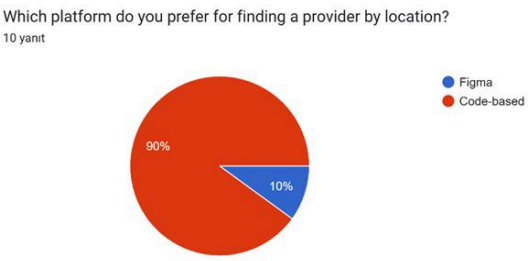
C. Analysis of the Experiments

C. 1 User Survey Results

TASK 1



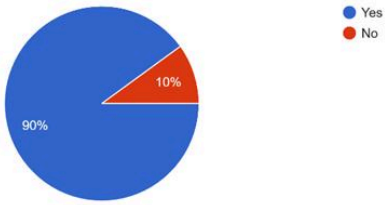




TASK 2

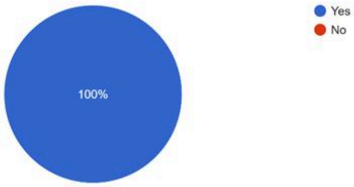
Did the coded platform allow you to access provider details faster compared to the Figma prototype (which required pre-selecting filters)?

10 yanıt



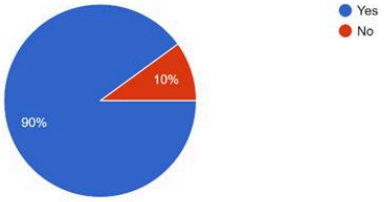
Did the compact view in the code-based platform provide enough information to identify providers quickly?

10 yanıt



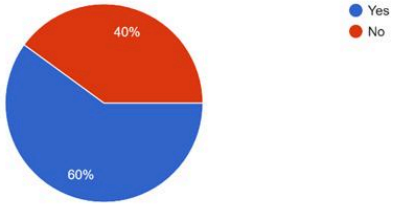
Was the workflow in the coded platform more straightforward than in the Figma prototype?

10 yanıt



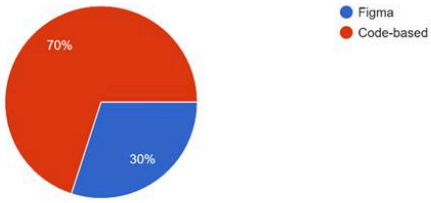
Did you find the layout of provider details clearer and more appealing in the coded platform?

10 yanıt



Which platform's provider details feature do you prefer overall?

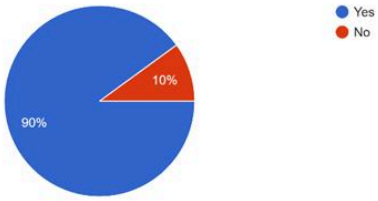
10 yanıt



TASK 3

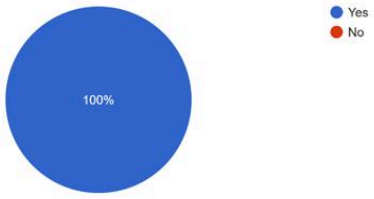
Did the additional filtering options in the code-based platform make it easier to find a suitable provider compared to the Figma prototype?

10 yanıt



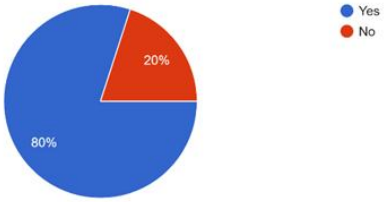
Was the filtering process in the code-based platform more intuitive despite having more options?

10 yanıt



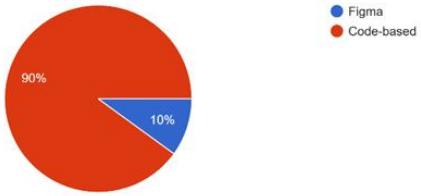
Did the limited filtering options in the Figma prototype make the task less efficient?

10 yanıt

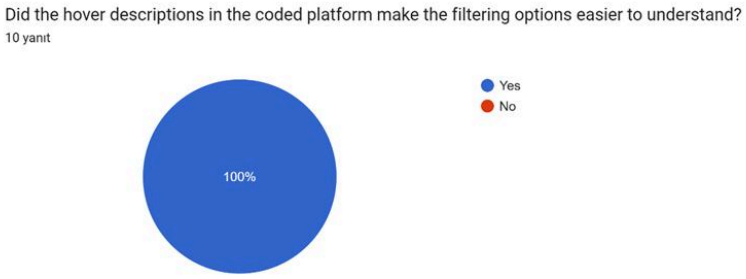


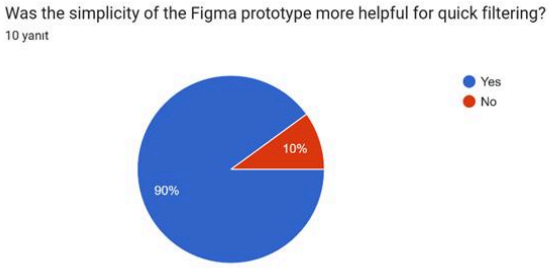
Which platform provided a better overall experience for filtering providers?

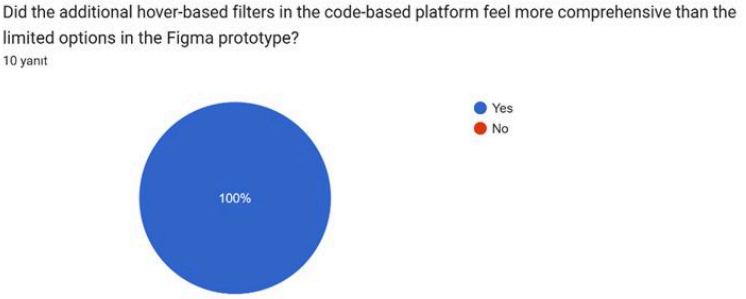
10 yanıt



TASK 4





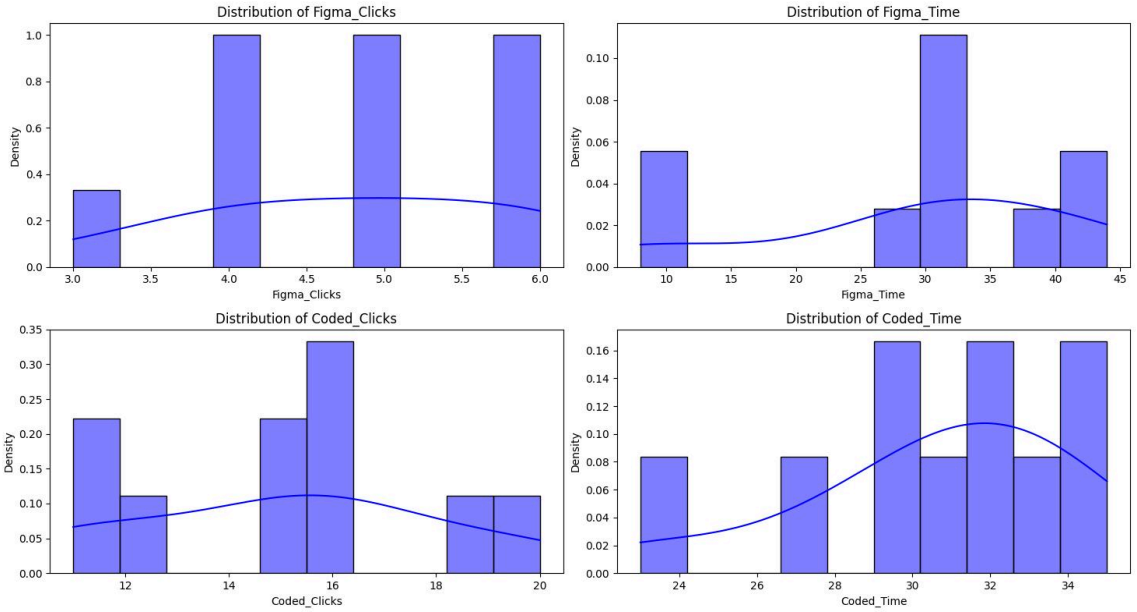


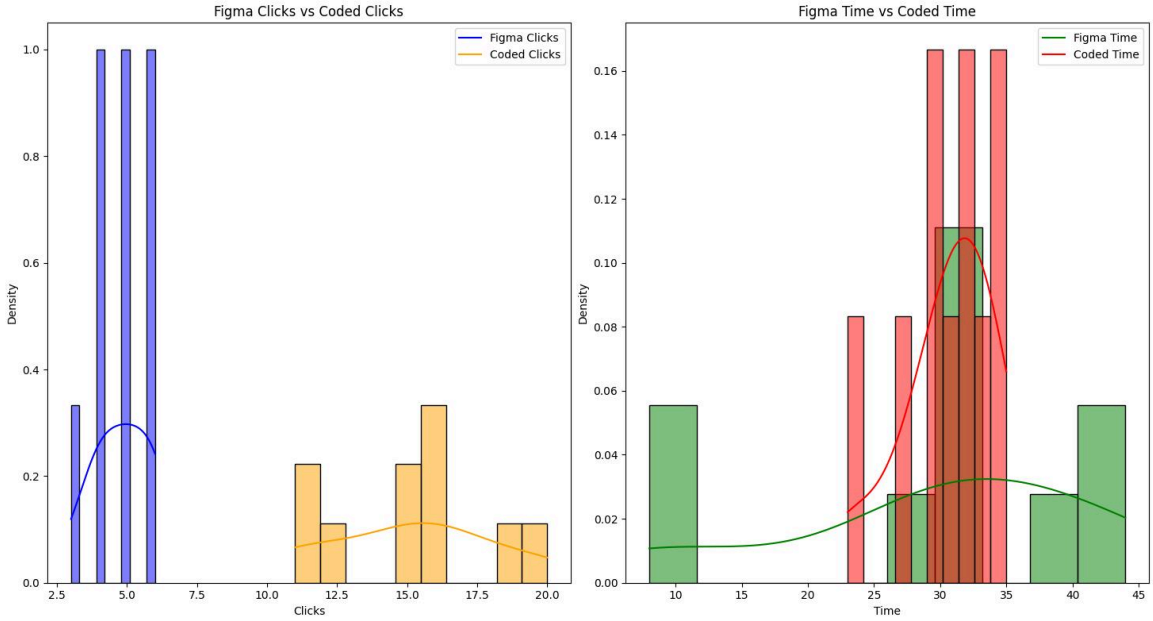
C. 2 Normal (Gaussian) Probability Distribution

Task 1 :

The given graphs represent the distribution and comparison of the number of clicks and time the users recorded while using Figma and coded prototype for Task 1 and the focus is on normal probability distributions. The first four graphs represent the Normal (Gaussian) Probability Distribution separately for Figma/Coded-based Platform for time/clicks measured. The following graph presentation compares the data while being merged into 1 graph for time/clicks. The Figma clicks seem to be shifting closer to the average but the Figma time distribution spread appears to be Gaussian. The distributions for coded clicks and time also seem to demonstrate a wider spread, particularly for coded clicks, which have attained a slightly more symmetric shape characteristic of a Gaussian distribution.
The last two graph analyses Figma against coded clicks and Figma against coded time. From these analysis we can conclude that coded tasks always require more clicks and time than Figma tasks. The density plots demonstrate the more advanced user interaction with complexity of coded tasks, their distributions tend to follow the Gaussian trend with the mean values as value of reference.

DV\Participants	1	2	3	4	5	6	7	8	9	10
Figma_Time	41	44	10	27	8	13	31	40	33	31
Coded_Time	33	31	34	32	23	35	27	29	30	32
Figma_Clicks	5	6	5	3	4	5	4	4	6	6
Coded_Clicks	16	16	19	16	20	15	11	12	11	15

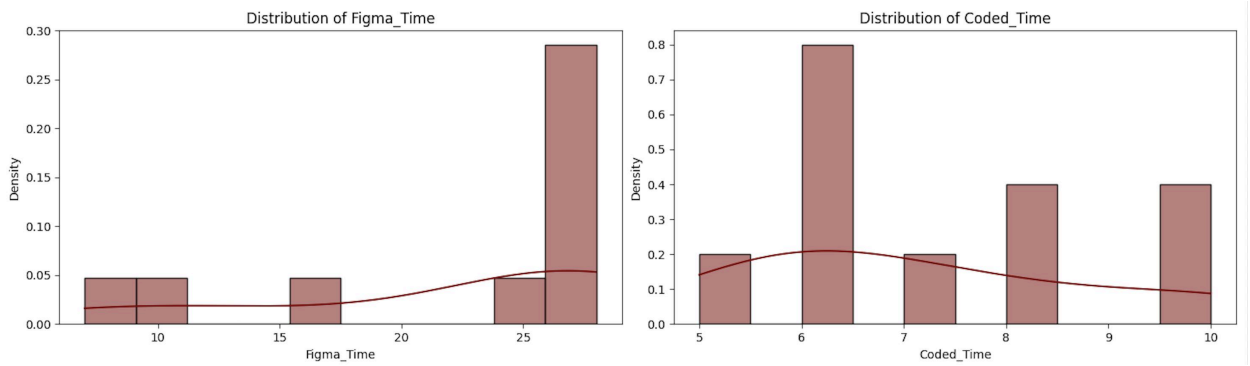


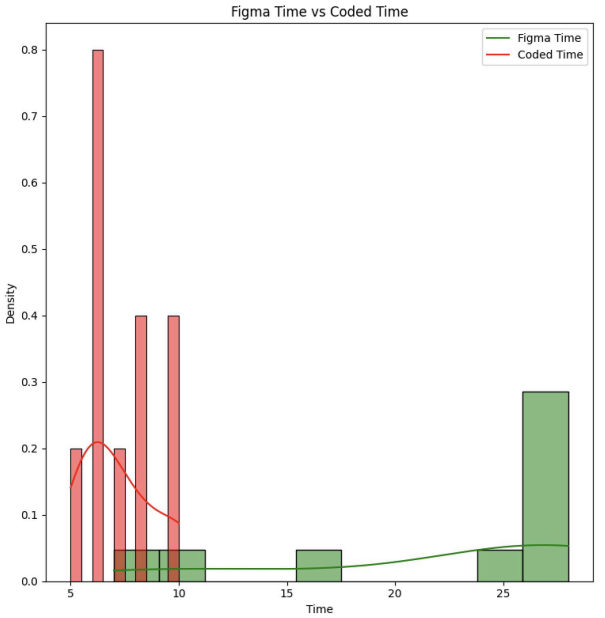


Task 2:

The given graphs represent the distribution and comparison of the taken time, the users recorded for Figma and coded tasks in Task 2 and the focus is on normal probability distributions. The first two graph represent the distribution of Figma and Coded time. The following graph presentation compares the data while being merged into 1 graph for time. Figma time has a right-skewed distribution with most participants taking 20–28 seconds to complete task, while coded time is more evenly spread, taking around 5–10 seconds.
The last graph analyses Figma against coded time merged into 1 graph representation, revealing significant differences in task durations. Figma tasks generally require longer time, as shown by their distribution peaking in the higher ranges, whereas coded time is more consistent and concentrated in the lower range. The density curves indicate that, in general, Figma tasks are longer than coded tasks, indicating perhaps more extensive interaction or complexity than coded tasks.

DV\Participants	1	2	3	4	5	6	7	8	9	10
Figma_Time	10	16	26	28	7	25	27	27	28	28
Coded_Time	6	6	7	8	10	10	5	6	6	8



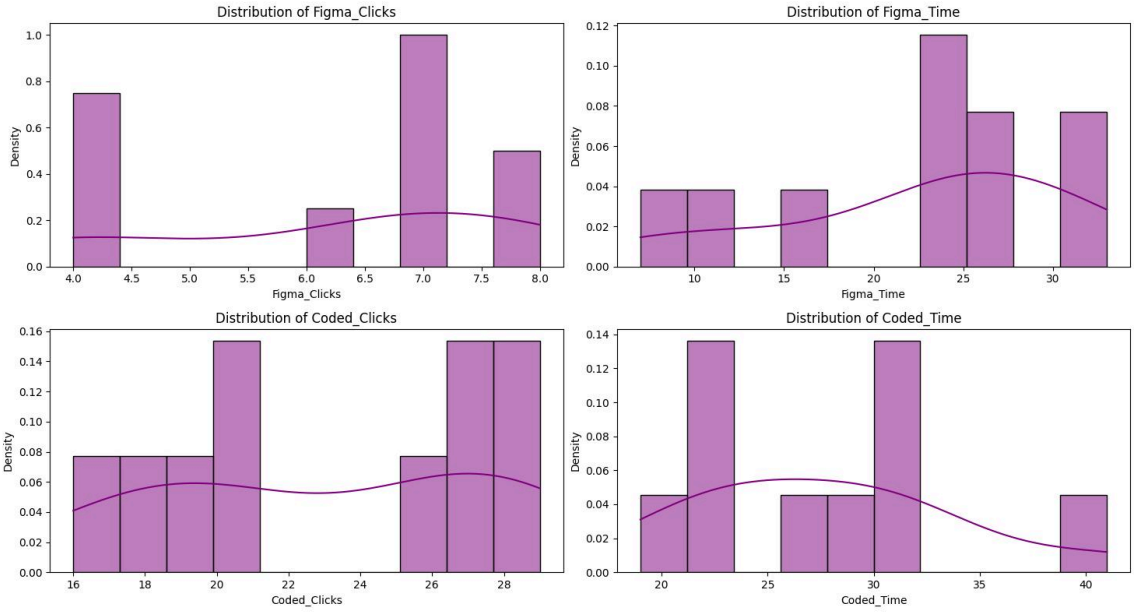


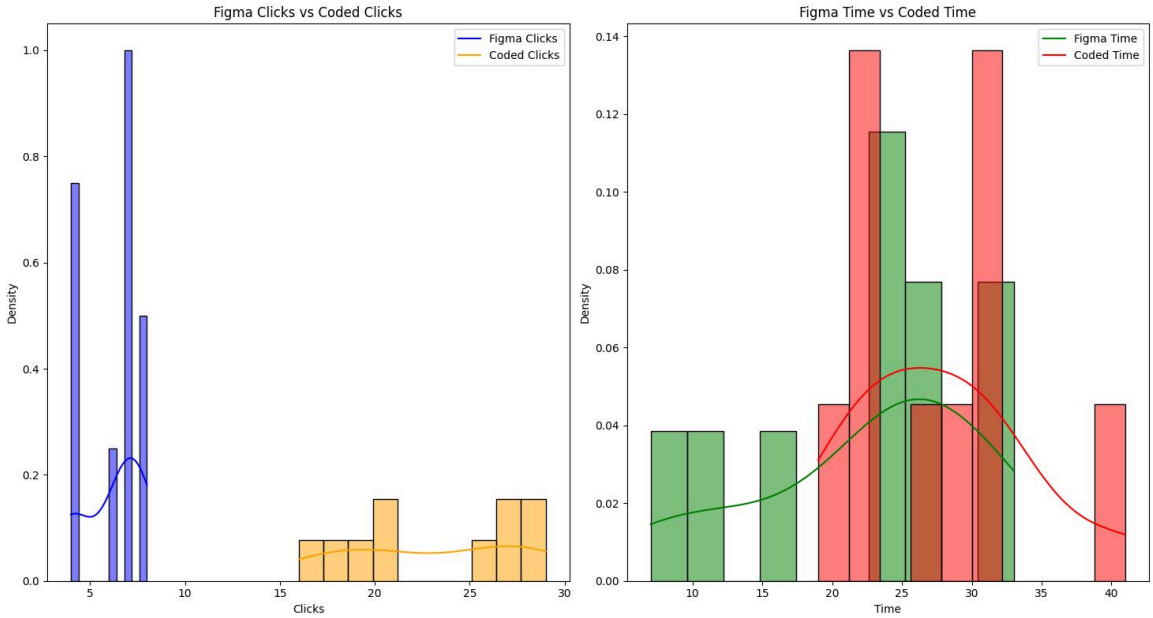
Task 3:

The given graphs represent the distribution and comparison of the number of clicks and time the users recorded while using Figma and coded prototype for Task 3 and the focus is on normal probability distributions. The first four graphs represent the Normal (Gaussian) Probability Distribution separately for Figma/Coded-based Platform for time/clicks measured. Figma clicks are mostly concentrated around 5–7 clicks, whereas Figma time demonstrates a much broader spread, peaking around 25–30 seconds. These patterns partially resemble Gaussian distributions, which are slightly asymmetrical due to participant variability.

The last two graphs compares Figma and coded metrics. Figma tasks generally involve fewer clicks but take more time, whereas coded tasks involve more clicks with durations distributed across a wider range. The density plots show that Figma tasks are centered on efficiency, whereas coded tasks are more complex interactions.

DV\Participants	1	2	3	4	5	6	7	8	9	10
Figma_Time	11	17	26	27	8	25	25	24	33	31
Coded_Time	23	26	41	32	19	23	23	29	30	32
Figma_Clicks	4	7	4	7	4	6	8	8	7	7
Coded_Clicks	20	21	18	16	19	27	27	29	28	26



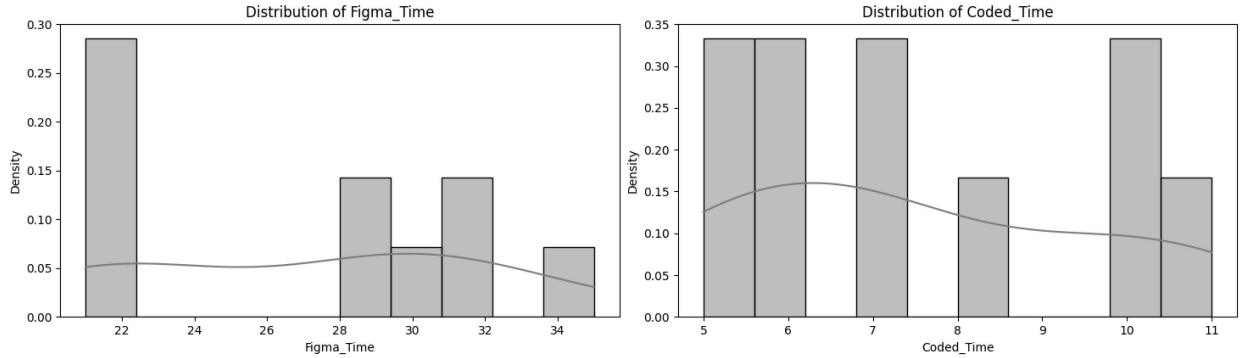


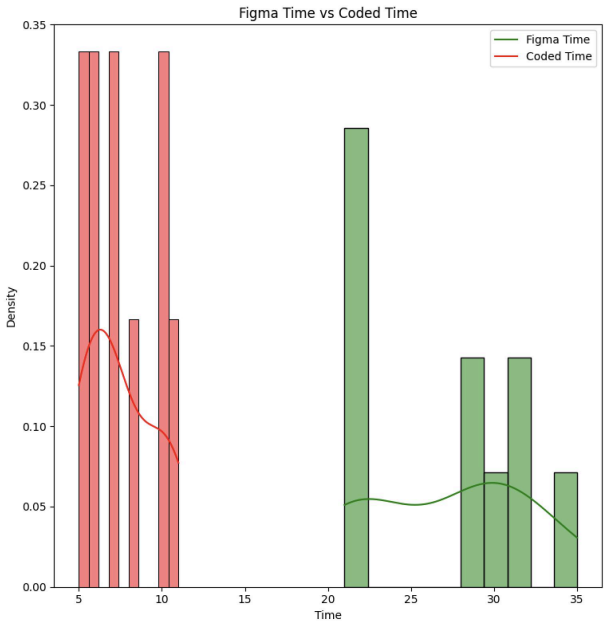
Task 4:

The given graphs represent the distribution and comparison of the taken time, the users recorded while using Figma and coded prototype for Task 4 and the focus is on normal probability distributions. The first two graphs represent the Normal (Gaussian) Probability Distribution separately for Figma/Coded-based Platform for time measured. Figma time demonstrates a right-skewed distribution, with most participants spending between 28 and 32 seconds, while coded time is concentrated around 5–10 seconds, showing a relatively uniform spread with slight Gaussian tendencies.

The last graph compares Figma time to coded time. Figma time consistently requires significantly longer durations compared to coded time, as shown by the density curves. Coded tasks exhibit lower variability and shorter completion times, whereas Figma tasks show more dispersion, reflecting higher interaction complexity or task intensity.

DV\Participants	1	2	3	4	5	6	7	8	9	10
Figma_Time	21	30	35	31	22	31	28	29	22	22
Coded_Time	5	6	7	10	11	6	7	8	5	10

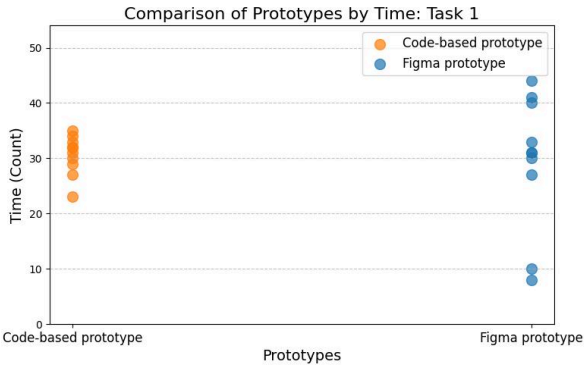


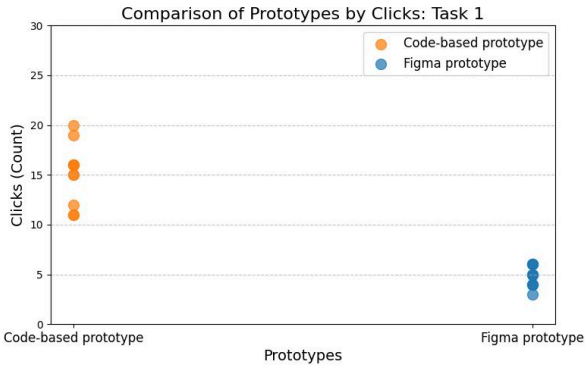


C. 3 Homoscedasticity

Homoscedasticity refers to the assumption that the variability (or spread) of user performance or feedback data remains constant across different levels of an independent variable. In our case, where we are testing and comparing two groups of participants, where each group got in touch with a certain platform we created, having homoscedasticity would mean that the variance in completion times for one interface is similar to the variance in the other one. If the variances are unequal, the data iis said to be heteroscedastic, which may require specialised statistical methods to analyse accurately.

Code-based prototype is the prototype with treatment. Figma prototype is the prototype without treatment.





Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 30.6$; $\sigma^2_1 = 12.7$
$\bar{x}_2 = 29.5$; $\sigma^2_2 = 146.5$

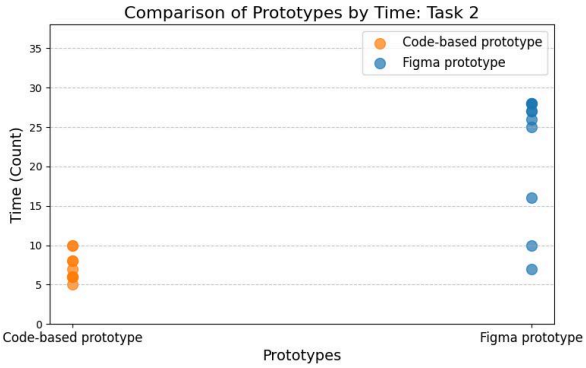
$\Rightarrow \bar{x}_1 > \bar{x}_2$: Shift in means
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**

Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 15.1$; $\sigma^2_1 = 9.44$
$\bar{x}_2 = 4.8$; $\sigma^2_2 = 1.067$

$\Rightarrow \bar{x}_1 > \bar{x}_2$: Shift in means
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**

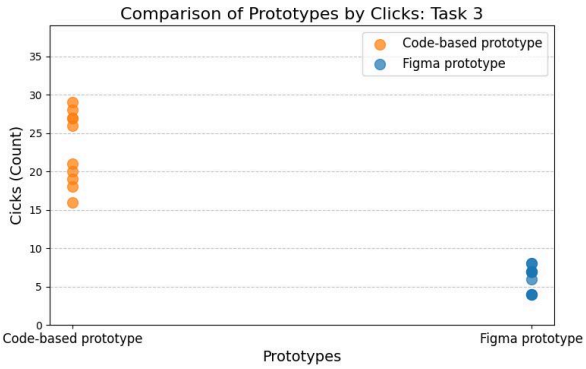


Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 7.2$; $\sigma^2_1 = 3.06$
$\bar{x}_2 = 22.2$; $\sigma^2_2 = 65.25$

$\Rightarrow \bar{x}_1 < \bar{x}_2$
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**

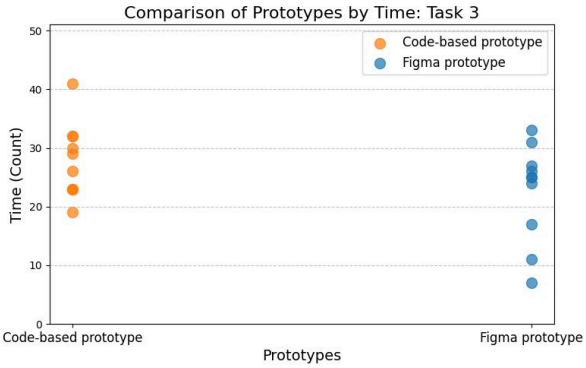


Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 23.1$; $\sigma^2_1 = 22.76$
$\bar{x}_2 = 6.2$; $\sigma^2_2 = 2.62$

$\Rightarrow \bar{x}_1 > \bar{x}_2$: Shift in means
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**

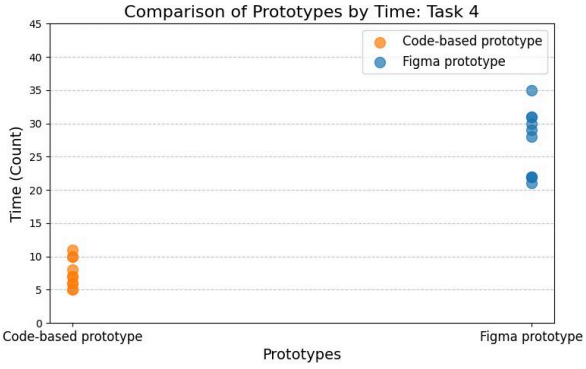


Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 27.8$; $\sigma^2_1 = 40.62$
$\bar{x}_2 = 22.6$; $\sigma^2_2 = 70.26$

$\Rightarrow \bar{x}_1 > \bar{x}_2$: Shift in means
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**



Notes:

X1 - Code-Based Prototype
X2 - Figma Prototype

$\bar{x}_1 = 7.5$; $\sigma^2_1 = 4.72$
$\bar{x}_2 = 27.1$; $\sigma^2_2 = 24.54$

$\Rightarrow \bar{x}_1 > \bar{x}_2$
$\Rightarrow \sigma^2_1$ & σ^2_2 : Markedly different, **does not exhibit homoscedasticity.**

C. 4 Null Hypothesis Significance Testing

C 4.1 Two-Sample T-Test Results

Task 1 - Finding a Provider by Location

Null Hypothesis (H_0): There is no significant difference in the time taken to find a provider in a specific region between the Figma prototype and the coded platform.
Result: The p-value (0.7858) is greater than the significance level (α), meaning H_0 cannot be rejected. The test statistic (T = 0.2758) falls within the 95% region of acceptance, and the effect size (d = 0.12) is small.
There is no statistically significant difference in time taken between the Figma prototype and the coded platform.
The difference in time observed is very minimal and can be considered a variation by chance.

Null Hypothesis (H_0): There is no significant difference in the number of clicks required to find a provider in a specific region between the Figma prototype and the coded platform.
Result: The p-value (8.163e-9) is much smaller than the significance level (α), meaning H_0 is rejected. The test statistic (T = 10.0586) falls outside the 95% region of acceptance, and the effect size (d = 4.5) is large.
The number of clicks for the Figma prototype was statistically different compared to the coded platform. The number of clicks to the coded platform is significantly high and may be contributed by the necessity of completing the selection form and other additional filters.

Task 2 -Viewing Provider Details ; presented on the slides

Null Hypothesis (H_0): Users will not access or comprehend provider details significantly faster or more intuitively in the code-based platform compared to the Figma prototype.
Result: The p-value (0.00001938) is much smaller than the significance level (α), meaning H_0 is rejected. The test statistic (T = -5.7375) falls outside the 95% region of acceptance, and the effect size (d = 2.57) is large.
The time it took to access and understand the details of providers varies very highly statistically between the Figma prototype and the coded platform. Users are much faster and more intuitive on the coded platform.

Task 3 - Filtering Options

Null Hypothesis (H_0): The additional filtering options in the code-based platform will significantly reduce the number of clicks to find suitable providers compared to the limited options in the Figma prototype.
Result: The p-value (3.535e-9) is much smaller than the significance level (α), meaning H_0 is rejected. The test statistic (T = 10.6155) falls outside the 95% region of acceptance, and the effect size (d = 4.75) is large.
There is a highly significant statistical difference between the Figa prototype and the Coded platform regarding the number of clicks; the latter presumably takes much more clicks because of filtering options and form selection to be processed.

Null Hypothesis (H_0): The additional filtering options in the code-based platform will not significantly reduce the time taken to find suitable providers compared to the limited options in the Figma prototype.
Result: The p-value (0.1357) is greater than the significance level (α), meaning H_0 cannot be rejected. The test statistic (T = 1.5622) falls within the 95% region of acceptance, and the effect size (d = 0.7) indicates a moderately large difference, but it is not statistically significant.
There is no significant difference in time required to find good providers between the Figma prototype and the actual coded platform. The difference in time observed was not large enough to conclude there was a real improvement.

Task 4 - Price Selection Function : presented on the slides

Null Hypothesis (H_0): The price selection function in the code-based platform will not significantly reduce the time taken for users to find and compare the prices of certain providers in a specific area.
Result: The p-value (1.044e-9) is much smaller than the significance level (α), meaning H_0 is rejected. The test statistic (T = -11.4678) falls far outside the 95% region of acceptance, and the effect size (d = 5.13) is extremely large.
There is a statistically highly significant difference in time taken for finding and comparing prices between the Figma prototype and the coded platform. On the coded platform, users take significantly less amount of time in finding and comparing the prices.

D. Conclusions

The evaluation conducted between the Figma prototype and the updated code-based platform provides a comprehensive analysis of user performance and satisfaction.
Through A/B testing, measurable differences in task completion time, click counts, and user feedback were identified which highlights the impact of the newly implemented features.

The new features have successfully addressed usability gaps in the earlier version, providing a more intuitive and effective user experience. However, ongoing user testing and optimization are recommended to further enhance areas where improvements were less pronounced.

Overall, the coded platform shows clear advantages in usability, advanced features, and enhanced user satisfaction. However, there are areas, such as filtering options, that need further optimization in order to streamline the user experience without compromising the added functionality. The statistically significant results confirm that the coded platform is indeed a better option than the Figma prototype for most tasks.

Links and resources:

Links:

https://www.statskingdom.com/140MeanT2eq.html

https://www.calculatorsoup.com/calculators/statistics/standard-deviation-calculator.php

Resources:

Lectures/Videos of Section 3: Controlled Experiments: Bachelor Practical Course: Evaluation of User Interfaces

Python Scripts of generating graphs: submitted via ZIP File