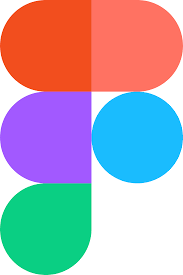
**Blackbox Testing Report**

**Product: Figma**



**Done By:**

Zerlish Burhan

**Contents**

[Introduction: 3](#_Toc203947479)

[Features: 4](#_Toc203947480)

[Quality Attributes: 5](#_Toc203947481)

[5 Selected Traditional Testing Techniques: 6](#_Toc203947482)

[Test Scenarios and Test Cases: 8](#_Toc203947483)

[TS1: Check authentication 8](#_Toc203947484)

[TS2: Check registration 9](#_Toc203947485)

[TS3: Check real time collaboration 10](#_Toc203947486)

[TS4: Check design components and variants 11](#_Toc203947487)

[TS5: Check auto layout 12](#_Toc203947488)

[TS6: Check Prototype Interaction 13](#_Toc203947489)

[TS7: Check File Sharing and Access Permissions 14](#_Toc203947490)

[TS8: Check add comments 15](#_Toc203947491)

[TS9: Check Prototype Presentation Mode 16](#_Toc203947492)

[TS10: Check Team Creation 18](#_Toc203947493)

[TS11: Check typography tools. 19](#_Toc203947494)

[TS12: Check frame creation. 20](#_Toc203947495)

[TS13: Check solid color fill 21](#_Toc203947496)

[TS14: Check Alignment Tools 22](#_Toc203947497)

[TS15: Check exporting assets 23](#_Toc203947498)

[Traceability Matrices: 24](#_Toc203947499)

[Forward Traceability Matrix: 24](#_Toc203947500)

[Backward Traceability Matrix: 29](#_Toc203947501)

[Bi-Directional Traceability Matrix: 34](#_Toc203947502)

[Validation Traceability Matrix 39](#_Toc203947503)

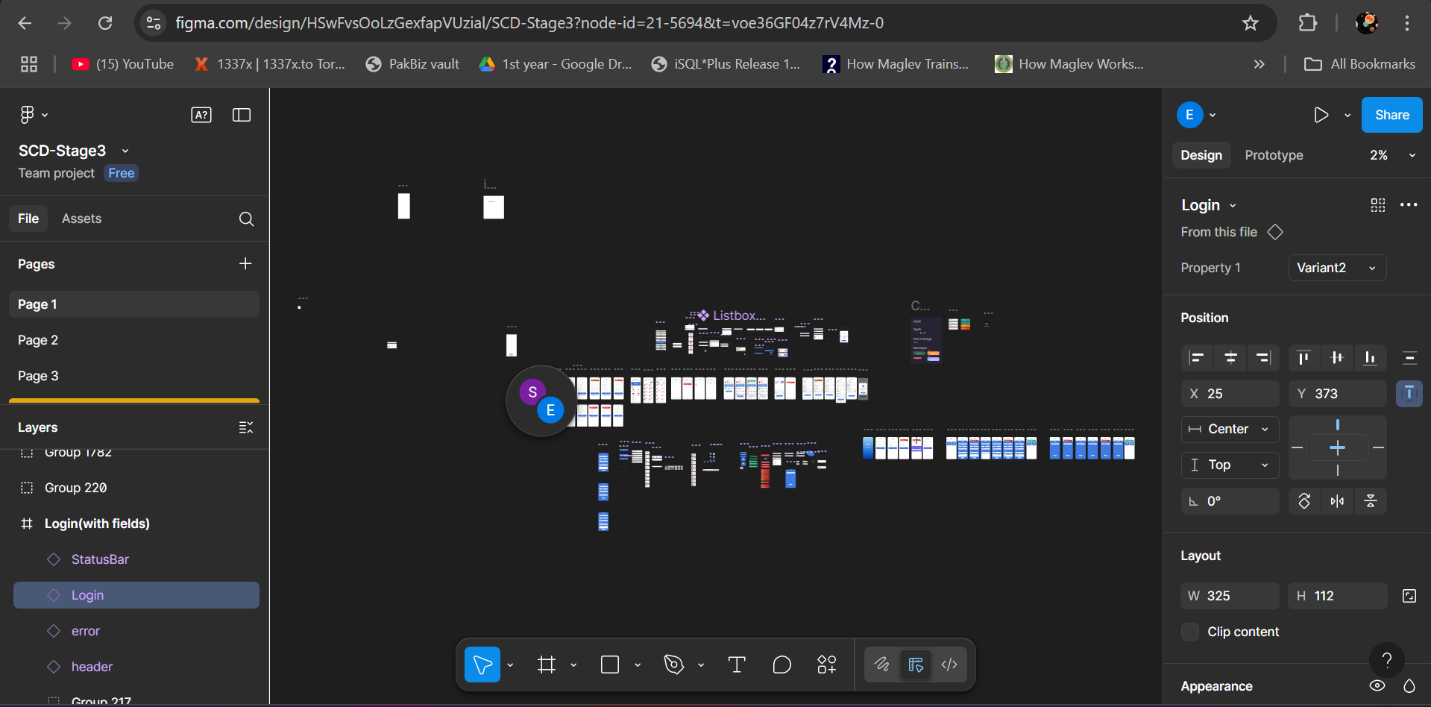
[Horizontal Traceability matrix 40](#_Toc203947504)

[Vertical Traceability Matrix 40](#_Toc203947505)

[Quality Matrix: 41](#_Toc203947506)

# Introduction:

Figma is a design tool used to design user interfaces and prototypes. It offers real-time collaboration, allowing users to work in groups by being online from anywhere in the world at the same time. It is a web-based and cloud solution. Users can create frames, components and variants, text, and shapes. They can use color, typography, and alignment tools. Users can create teams and share their files with other users and control access. The design can be made interactive using prototype interaction and can be displayed in different devices. Users can give comments to improve teamwork. Figma also allows users to export the design with just one click in different formats.

****

# Features:

1. Authentication

User shall be able to log in.

1. Registration

User shall be able to register.

1. Real-Time Collaboration

User shall be able to collaborate in real-time.

1. Design Components & Variants

User shall be able to create and manage design components and variants.

1. Auto Layout

User shall be able to apply auto layout.

1. Prototype Interaction

User shall be able to create interactive prototypes.

1. File Sharing & Access Permissions

User shall be able to share files and set access permissions.

1. Adding Comments

User shall be able to add comments.

1. Prototype Presentation Mode

User shall be able to view prototypes in presentation mode.

1. Team Creation

User shall be able to create teams.

1. Typography Tools

User shall be able to use typography tools.

1. Frame Creation

User shall be able to create frames.

1. Solid Color Fill

User shall be able to apply solid color fills.

1. Alignment Tools

User shall be able to use alignment tools.

1. Exporting Assets

User shall be able to export design assets.

# Quality Attributes:

**1. Usability**

There is a vast targeted audience of Figma due to its live collaboration feature and it is being used by designers as well as developers so it is very important to make it usable for all type of users from beginner to advanced level and its usability is the most important quality attribute to be considered for testing. Usability is a major nonfunctional requirement of Figma.

**2. Learnability**

Figma should be quick for the users to learn. So that the naive users learn to use basic and advanced features in less time. Learnability is very important for user retention because if the system is not learnable the users would give up early and the product will fail to grow its users.

**3. Efficiency**

Efficiency is an important quality attribute for Figma because it ensures that tasks like designing, exporting, and collaborating are completed without any delay. Figma supports real time collaboration, so efficiency of the product is very important for its smooth usage. This helps in saving time, reducing frustration, and improving the user experience.

**4. Security**

Figma is a highly collaborative platform for professionals and requires a lot of security. It should protect the user data and design files from unauthorized access. Figma should ensure appropriate access control (edit and view). Users use Figma to create creative work and expect it to be protected from unauthorized access as they put their time and effort into it.

**5. Correctness**

As far as correctness it concerned it is one of the most important quality attributes, if the behavior of the system is incorrect or unexpected, it impacts the work of the user. It will eventually lead the user to stop using the product. User won’t be satisfied if features don’t work as expected.

**Reasons for Choosing Traditional Testing Techniques:**

1. We are testing at the user level and don’t have access to the actual code. Traditional testing is more suitable in this case as it allows us to focus on how the user interacts with the system.
2. The features we selected in Figma will be static, as we are only testing the current version. We will not be dealing with versions in the future. This makes it easier to test them by simply running the design in Figma and observing how each element performs.
3. Traditional testing also helps us assess the overall efficiency of the system. By manually testing the design, we can validate whether each functionality behaves expectedly and meets the user requirements.
4. We can identify features that may be difficult for users to use. This will allow us to improve the usability of the design and make the interface more user-friendly.
5. Figma is a released software product.

# 5 Selected Traditional Testing Techniques:

1. **Usability Testing:**

**Reason:** This testing technique tests how easy it is for the user to interact with and understand the user interface (UI). This will help in determining how fast the users can complete tasks, or which steps slow the users down or are confusing for them. It evaluates the **Usability** and **Learnability** quality attributes for Figma. It is very important to check the intuitiveness of the UI for a design tool like Figma. This testing will help in making Figma more user-friendly.

1. **Performance Testing:**

**Reason**: This testing technique measures response time of Figma features like how long it takes to load large design files. It will measure the **efficiency** of Figma features which is very important because if the software takes a long time to perform actions, then the users will become frustrated and stop using our product. So, we need to ensure Figma works efficiently under different conditions.

1. **Acceptance Testing:**

**Reason:** This testing technique tests Figma from users’ perspectives to confirm that it meets user needs. It will verify Figma’s **Correctness**, **Usability**, and **Learnability.** If the product fails to provide features that the users require then the product is useless so it is very important to test if it meets user needs.

1. **Black Box Testing:**

**Reason:** This testing technique tests how the system performs when different inputs are given to it. It checks if the system works correctly by seeing if the outputs are the same as expected. It is very important to test the correctness of Figma features to ensure they work as expected.

1. **Exploratory Testing:**

**Reason:** This testing technique allows us to find the defects that may remain hidden by the other selected testing techniques. This technique will help in testing the **Correctness** of Figma features. It also helps us in finding **Usability** issues that were missed during Usability Testing.

# Test Scenarios and Test Cases:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 1 | | | | | | **Test Case ID** | | | T1, T2, T3, T4, T5 | | |  | | |
| **Test Case Description** | | 1. User login with valid email and correct password. 2. User login with valid email and wrong password. 3. User logins with empty email and correct password. 4. User logins with valid email and empty password. 5. User enters invalid email format. | | | | | | **Test Priority** | | | HIGH | | |
| **Pre-Requisite** | | * User should open Figma web or app to get logged in. | | | | | | **Post-Requisite** | | | The user is Logged in successfully. | | |
| **Test Execution Steps:**  **T1**   1. Open Figma 2. Open the login screen. 3. Enter a valid email (e.g., user@gmail.com). 4. Enter a valid password. 5. Click the Login button. | | | | **T2**   1. Open Figma 2. Go to the login screen. 3. Enter a valid email. 4. Enter an incorrect password. 5. Click Login button | | **T3**   1. Open Figma 2. Open the login page. 3. Leave the email field empty. 4. Enter a valid password. 5. Click on the Login button. | | | | **T4**   1. Open Figma 2. Launch the login screen. 3. Enter a valid email. 4. Leave the password field empty. 5. Press the Loginbutton. | | | | | **T5**   1. Open Figma 2. Open the login form. 3. Enter an invalid email (e.g., User123gmail.com – missing '@'). 4. Enter a valid password. 5. Click on **Login button.** | |
| **Sr** | **Action** | | **Inputs** | | | | **Expected Output** | | **Actual Output** | | | **Test Browser** | **Test Result** | | | **Test Comments** | |
| **Email** | | **Password** | |
| T1 | Authentication | | Valid | | Valid | | Redirect user to the home page. | | User redirected to the homepage. | | | Google | Pass | | |  | |
| T2 | Authentication | | Valid | | Invalid | | Error: “Incorrect password” | | Error: “Log in with Google or reset password with Reset password”. | | | Google | Fail | | |  | |
| T3 | Authentication | | Empty | | Valid | | Error: “Please provide email.” | | Error: “Please provide your email.” | | | Google | Pass | | |  | |
| T4 | Authentication | | Valid | | Empty | | Error: “Please provide password.” | | Error: “Please provide your password.” | | | Google | Pass | | |  | |
| T5 | Authentication | | User123gmail.com | | Valid | | Error: “Invalid email address format.” | | Error:  Please include an ‘@’ in the email address. User123gmail.com is missing an ‘@’ | | | Google | Pass | | |  | |

## TS1: Check authentication

## TS2: Check registration

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | T2 | | | | | | **Test Case ID** | T1, T2, T3, T4, T5 |  | |
| **Test Case Description** | | 1. The user enters invalid email and a valid password. 2. The User enters valid email and an invalid password. 3. The User enters valid email and a valid password. 4. The User enters an invalid email and an invalid password. 5. The User chooses a Google account to create Figma account. | | | | | | **Test Priority** | High |
| **Pre-Requisite** | | User should have Figma open | | | | | | **Post-Requisite** | Account is successfully created |
| **Test Execution Steps:**  **T1-4**   1. Go to Figma 2. Click on Create Account 3. Enter email and password. 4. Click on “Open Email” to verify the email by clicking on the “Verify email” button. | | | | | | **T5**  1. Go to Figma  2. Click on Create Account  3. Click on Continue with Google  4. Select a Google account | | | | | |
| **Sr#** | **Action** | | **Inputs** | | **Expected Output** | | **Actual Output** | **Test Browser** | **Test Result** | | **Test Comments** |
| **Email** | **Password** |
| T1 | Registration | | Invalid email | Valid password | Error message “Please include @ in the email address” | | Error message “Please include @ in the email address” | Chrome | Pass | |  |
| T2 | Registration | | Valid Email | Invalid password (password: pa) | Error message “Please enter a password longer than seven characters.” | | Error message “Please enter a password longer than seven characters.” | Chrome | Pass | |  |
| T3 | Registration | | Valid email | Valid password | User is navigated to the Figma homepage. | | User is navigated to Figma homepage. | Chrome | Pass | |  |
| T4 | Registration | | Invalid email (email: emanfaisal003gmail.com) | Invalid password (password: pas) | Error message “Please include @ in the email address”  Error message “Please enter a password longer than seven characters.” | | Error message “Please include @ in the email address” | Chrome | Fail | |  |
| T5 | Registration | | Choose a Google account | | User is navigated to the Figma homepage. | | User is navigated to the Figma homepage. | Chrome | Pass | |  |

## TS3: Check real time collaboration

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | 3 | | | | **Test Case ID** | | | T1, T2, T3, T4 | | |  | |
| **Test Case Description** | 1. All the users access the same file through a single link to collaborate with each other. 2. If one user renames any element in the design file that will be update in all the files of other users also. 3. If User A is editing a shape, User B shouldn't be able to edit it until User A is done. 4. Both users’ cursors appear in different colors with labels, indicating real-time presence. | | | | **Test Priority** | | | HIGH | | |
| **Pre-Requisite** | * The user should be logged in and the design file must be shared with the collaborators via link etc. | | | | **Post-Requisite** | | | All the changes are regularly updated after the collaboration is made. | | |
| **Test Execution Steps:**  **T1:**   1. Open Google Chrome. 2. Paste and open the shared Figma file link: https://www.figma.com/design. 3. Log in with different user accounts. 4. Check if all users are able to open the file. 5. Confirm that each user’s online status is visible. | | | **Test Execution Steps:**  **T2:**   1. Open the Figma file in Chrome with multiple users. 2. Double-click the frame name from one user's account. 3. Type a new name and hit enter. 4. On other users' screens, observe the name change without refreshing. 5. Confirm the update is shown instantly to all users. | | | **Test Execution Steps:**  **T3:**   1. Open the shared design file in Chrome with User A and User B. 2. Let User A select and start editing a shape. 3. At the same time, User B tries to edit the same shape. 4. Check if User B is restricted from editing. 5. Confirm that only one user can edit the shape at a time. | | | **Test Execution Steps:**  **T4:**   1. Open the shared Figma file in Chrome with both User A and User B. 2. Make sure both users are logged in. 3. Move each user's cursor around the canvas. 4. Observe the cursor colors for each user. 5. Verify that each cursor is shown in a different color. | | | |
| **Sr** | **Action** | **Inputs** | | **Expected Output** | | | **Actual Output** | **Test Browser** | | **Test Result** | | **Test Comments** |
| T1 | Real Time Collaboration | File Link: https://www.figma.com/design | | All the users can access the file and their status will be shown as online | | | All the user accessed the file in real-time | Chrome | | Pass | |  |
| T2 | Real Time Collaboration | Double click on frame name + Add text | | The rename will be updated and visible to all the users without refreshing the page. | | | Frame name updated successfully on all the users file in real time in refreshing the page. | Chrome | | Pass | |  |
| T3 | Real Time Collaboration | User A edits a shape; User B tries to edit the same shape | | User A edits a shape; User B tries to edit the same shape. Editing should not be allowed. | | | |  | | --- | | User A edits a shape; User B tries to edit the same shape. Editing is not allowed. | | Chrome | | Pass | |  |
| T4 | Real Time Collaboration | User A and User B open the shared design file. | | Both users' cursors should appear in different colors, showing real-time interaction | | | |  | | --- | | Both users' cursors appear in different colors, showing real-time interaction | | Chrome | | Pass | |  |

## TS4: Check design components and variants

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 4 | | | | | | **Test Case ID** | T1, T2, T3, T4, T5 | | |  | | |
| **Test Case Description** | | 1. User creates component by right click. 2. User creates component by button 3. User instantiates component by dragging. 4. User instantiates components by insert instance button. 5. User adds variant. | | | | | | **Test Priority** | Medium | | |
| **Pre-Requisite** | | * User should be logged in. * User should have a project open. * User should have some element. | | | | | | **Post-Requisite** | * Component, instance, variant should be created successfully | | |
| **Test Execution Steps:**  **T1**   1. Open Figma. 2. Open a project. 3. Select any element(s) 4. Right click on the screen 5. Click on create component | | | **T2**  (1-3)  4. Click on create component icon in the right panel | | | **T3**  (1-2)  3. Switch to assets in the left panel.  4. Select components  5. Select page  6. Drag the selected component on the page | | | | **T4**  (1-2)  3. Switch to assets in the left panel.  4. Select components  5. Select page  6. Click on component  7. Click insert instance | | | **T5**  (1-2)  3. Select created component  4. Click on add variant icon in the right panel  5. Name the variant | |
| **Sr** | **Action** | | | **Inputs** | **Expected Output** | | **Actual Output** | | **Test Browser** | | **Test Result** | | | **Test Comments** |
| **Selected Element (s)** |
| T1 | Creation of a component by right click | | | Rectangle and text | Component should be created successfully | | Component is created successfully | | Chrome | | Pass | | |  |
| T2 | Creation of a component by button | | | frame | Component should be created successfully | | Component is created successfully | | Chrome | | Pass | | |  |
| T3 | Component instantiation by drag | | | Created component | Instance should be created successfully | | Instance is created successfully | | Chrome | | Pass | | |  |
| T4 | Component instantiation by insert instance button | | | Created component | Instance should be created successfully | | Instance is created successfully | | Chrome | | Pass | | |  |
| T5 | Add variant | | | Created component | Variant should be added successfully | | Variant should be added successfully | | Chrome | | Pass | | |  |

## TS5: Check auto layout

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | 5 | | | | | | **Test Case ID** | | T1, T2, T3, T4, T5 |  | | | |
| **Test Case Description** | 1. User adds auto layout to a frame or a group of elements. 2. User adjusts padding, spacing between items, or alignment. 3. User nests auto layouts inside other auto layouts. 4. User duplicates a layout and all constraints are retained. 5. User removes auto layout from a frame. | | | | | | **Test Priority** | | MEDIUM |
| **Pre-Requisite** | * User must be logged in. * A Figma file must be open with at least one frame created. | | | | | | **Post-Requisite** | | Layout should behave as per user adjustments. |
| **Test Execution Steps:**  **T1:**   1. Open the Figma file in Chrome. 2. Select a frame on the canvas. 3. Click on the “+ Auto layout” button from the right panel. 4. Observe the frame background change to a grey-black color. | | | **Test Execution Steps:**  **T2:**   1. Select a frame that already has auto layout applied. 2. In the right panel, set Padding = 10 and Spacing = 20. 3. Observe the spacing between child elements and padding inside the frame. 4. Check if the frame layout reflects these values correctly. | | | **Test Execution Steps:**  **T3:**   1. Create two separate frames and apply auto layout to both. 2. Place one auto layout frame inside the other. 3. Adjust elements inside both frames to test alignment and spacing. 4. Observe if the nested layout behaves as expected. | | **Test Execution Steps:**  **T4:**   1. Select a frame with auto layout. 2. Press Ctrl + D to duplicate the frame. 3. Click on the duplicated frame. 4. Check if the auto layout properties are the same as the original. | | | | **Test Execution Steps:**  **T5:**   1. Select a frame that has auto layout enabled. 2. Click on the “Remove auto layout” option in the right panel. 3. Observe the layout change. 4. Check if the frame behaves like a normal group. | |
| **Sr** | | **Action** | | **Inputs** | **Expected Output** | | **Actual Output** | | **Test Browser** | | **Test Result** | | **Test Comments** |
| T1 | | Auto Layout | | Select frame + “+ Auto layout” | Frame should become an auto layout solid color grey-black | | Frame becomes auto layout | | Chrome | | Pass | |  |
| T2 | | Auto Layout | | Padding: 10 + Spacing: 20 | Frame should reflect padding and spacing changes | | Padding and spacing updated correctly | | Chrome | | Pass | |  |
| T3 | | Auto Layout | | Create two frames with auto layout | Nested layout should behave properly | | Nested layout works properly | | Chrome | | Pass | |  |
| T4 | | Auto Layout | | Ctrl + D | New layout should retain all auto layout properties | | Properties retained in duplicate | | Chrome | | Pass | |  |
| T5 | | Auto Layout | | Click “Remove auto layout” | Frame should convert to normal group | | Frame is no longer auto layout | | Chrome | | Pass | |  |

## TS6: Check Prototype Interaction

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 6 | | | | **Test Case ID** | | T1, T2, T3, T4, T5 | |  | | |
| **Test Case Description** | | 1. User applied on click interaction between 2 frames. 2. User applied on drag interaction between 2 frames. 3. User applied while hovering interaction between 2 frames. 4. User applied while pressing the interaction on the same frame. 5. User applied interaction between a frame and a non-frame element. | | | | **Test Priority** | | High | |
| **Pre-Requisite** | | * User is logged in. * User has a project with at least 2 frames | | | | **Post-Requisite** | | * Successful navigation between frames | |
| **Test Execution Steps:**   1. Open Figma in browser 2. Open a design project 3. Click on prototype in the right top corner 4. Select any frame 5. Drag the corner of frame A and join it with frame B 6. Select the trigger | | | | | | | | | | | | |
| **Sr** | **Action** | **Inputs** | | | **Expected Output** | | **Actual Output** | | **Test Browser** | | **Test Result** | **Test Comments** |
| **Source Frame** | **Destination Frame** | **Trigger** |
| T1 | Adding interaction | Frame A | Frame B | On click | Navigated from frame A to frame B with On Click trigger | | Navigated from frame A to frame B with On Click trigger | | Chrome | | Pass |  |
| T2 | Adding interaction | Frame A | Frame B | On drag | Navigated from frame A to frame B with On Drag trigger | | Navigated from frame A to frame B with On Drag trigger | | Chrome | | Pass |  |
| T3 | Adding interaction | Frame A | Frame B | While hovering | Navigated from frame A to frame B with While Hovering trigger | | Navigated from frame A to frame B with While Hovering trigger | | Chrome | | Pass |  |
| T4 | Adding interaction | Frame A | Frame A | While pressing | Navigated from frame A to frame A with While Pressing trigger | | Navigated from frame A to frame A with While Pressing trigger | | Chrome | | Pass |  |
| T5 | Adding interaction | Frame A | Non frame element | - | Figma shouldn’t apply interaction | | Figma didn’t apply interaction between frame and shape | | Chrome | | Pass |  |

## TS7: Check File Sharing and Access Permissions

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 7 | | | | **Test Case ID** | | T1, T2, T3, T4, T5 | |  | |
| **Test Case Description** | | 1. The user enters a valid email and gives Can Edit access permission. 2. The user enters a valid email and gives Can View access permission. 3. The user enters an invalid email and gives Can Edit access permission. 4. The user enters an invalid email and gives Can View access permission. 5. The user copies link of the file. | | | | **Test Priority** | | Medium | |
| **Pre-Requisite** | | * User should be logged in * User should have a project open | | | | **Post-Requisite** | | The invite is sent successfully with the specified access | |
| **Test Execution Steps:**  **(T1-T4)**   1. Open Figma. 2. Open any project. 3. Click on Share. 4. Enter a/n email/s. 5. Select can edit/can view. 6. Click on Invite. | | | | | | **Test Execution Steps:**  **(T5)**  (1-3)  4. Click on copy link | | | | | |
| **Sr** | **Action** | | **Inputs** | | **Expected Output** | | **Actual Output** | **Test Browser** | **Test Result** | | **Test Comments** |
| **Email** | **Access** |
| T1 | Share file | | Valid email | Can edit | Invitation successfully sent with edit access to the receiver | | Invitation successfully sent with edit access to the receiver | Chrome | Pass | |  |
| T2 | Share file | | Valid email | Can view | Invitation successfully sent with only view access to the receiver | | Invitation successfully sent but receiver can also copy the contents of the file | Chrome | Fail | |  |
| T3 | Share file | | Invalid email | Can edit | Email will be highlighted in red and user would not be able to send invitation | | Email was highlighted in red and invitation could not be sent | Chrome | Pass | |  |
| T4 | Share file | | Invalid email | Can view | Email will be highlighted in red and user would not be able to send invitation | | Email was highlighted in red and invitation could not be sent | Chrome | Pass | |  |
| T5 | Share file | | Link of file | | Link successfully copied | | Link successfully copied | Chrome | Pass | |  |

## TS8: Check add comments

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 8 | | | | | | **Test Case ID** | | | T1, T2, T3, T4, T5 | | |  | |
| **Test Case Description** | | 1. User add comments on Figma design. 2. User view comments of other team members on Figma. 3. User reply to team members through comments. 4. User resolves issues through comments that are occurring in their ongoing project on Figma. 5. User deletes a comment by clicking on the delete button. | | | | | | **Test Priority** | | | MEDIUM | | |
| **Pre-Requisite** | | * User should be logged in the Figma Design. * The user should be in the particular directory where they are working on collaborative project. * The user should also have the access to do comments on the elements of Figma file and edit the file. | | | | | | **Post-Requisite** | | | The comments will be added to the project. | | |
| **Test Execution Steps:**  T1:   1. Open the design file in Chrome. 2. Click the comment tool and select a spot on the canvas. 3. Type a comment and mention a valid team member using @teamMember. 4. Post the comment. | | | | **Test Execution Steps:**  T2:   1. Open the Figma file in Chrome. 2. Select the comment tool. 3. Click on a specific part of the design. 4. Type "Please increase the font size" and post it. | | **Test Execution Steps:**  T3:   1. Open the comment tool in the design file. 2. Type a comment and mention a non-existent user like @nonExistentMember. 3. Post the comment. | | | **Test Execution Steps:**  T4:   1. Find an existing comment in the file. 2. Click reply and type “The color is not correct.” 3. Post the reply. | | | **Test Execution Steps:**  T5:   1. Find an existing comment on the canvas. 2. Click the **three dots** (more options) next to the comment. 3. Select **Delete**. | | | |
| **Sr** | **Action** | | **Inputs** | | **Expected Output** | | **Actual Output** | | | **Test Browser** | | | **Test Result** | | **Test Comments** |
| T1 | Add comments | | @teamMember | | Comment should be added and the team member should receive a notification | | Comment is added and the team member receives a notification. | | | Chrome | | | Pass | |  |
| T2 | Add comments | | Text: “Please increase the font size” | | Comment should appear exactly where the user placed the comment. | | Comment appears exactly where the user placed the comment. | | | Chrome | | | Pass | |  |
| T3 | Add comments | | @nonExistentMember | | The Comment box should warn the user that the member does not exist | | The Comment is added without showing any warning. | | | Chrome | | | Fail | |  |
| T4 | Add comments | | Text “The color is not correct.” (The user types the message to reply back to a specific team member) | | The replied Comment should be visible or highlighted as replied comment. | | The replied comment appears as simple comments not as highlighted | | | Chrome | | | Fail | |  |
| T5 | Delete comment | | Click “Delete” | | Comments should be permanently removed from all the logged in users of that project. | | Comment deleted successfully. | | | Chrome | | | Pass | |  |

## TS9: Check Prototype Presentation Mode

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 9 | | | | | | | **Test Case ID** | | T1, T2, T3, T4, T5 | |  | | |
| **Test Case Description** | | 1. User selects iPhone 13 mini (initial frame), iPhone 13 mini (device), portrait, white device outline, and 000000 background color 2. User selects iPhone 13 mini (initial frame), iPhone 16 (device), landscape axis, pink device outline, and white (FFFFFF) background color. 3. User selects iPhone 13 mini (initial frame), MacBook Air (device), silver device outline, and yellow (FFEA00) background color. 4. User presses “r” key to restart presentation. 5. User presses forward arrow key to move to the next frame. | | | | | | | **Test Priority** | | High | |
| **Pre-Requisite** | | * User should be logged in. * Figma project should be open. * There should be at least one frame. | | | | | | | **Post-Requisite** | | * Design is presented successfully according to selected inputs. | |
| **Test Execution Steps:**  **(T1-T3)**   1. Open Figma in browser. 2. Open a design project. 3. Click on prototype in the right panel. 4. Click on show prototype settings. 5. Select device, axis, device outline and background color. 6. Click on play button on the top right corner. | | | | | **T4**  (1-2)  3. Click on play button in the top right corner.  4. Press forward key.  5. Press r on the keyboard. | | | | | **T5**  (1-2)  3. Click on play button in the top right corner.  4. Press forward key. | | | | | |
| **Sr** | **Action** | **Inputs** | | | |  |  | **Expected Output** | **Actual Output** | | | **Test Browser** | **Test Result** | **Test Comments** |
| **Device** | **Initial Frame** | **Axis** | | **Device Outline** | **Background Color** |
| T1 | Presentation of prototype | iPhone 13 mini | iPhone 13 mini | Portrait | | White | 000000 | The initial iPhone 13mini should fit properly in the iPhone 13 mini and the device should appear in portrait mode with device outline as white and background as 000000. | The initial iPhone 13mini fits properly in the iPhone 13 mini and the device appears in portrait mode with device outline as white and background as 000000. | | | Chrome | Pass |  |
| T2 | Presentation of prototype | iPhone 16 | iPhone 13 mini | Landscape | | Pink | FFFFFF | The initial iPhone 13mini should fit properly in the iPhone 16 and the device should appear in landscape mode with device outline as pink and background as FFFFFF. | There is space in the iPhone 16 as the initial frame was iPhone 13mini. The iPhone 13mini covers only the space in the vertical mode, the rest is empty space in the device with pink outline and FFFFFF background. | | | Chrome | Fail |  |
| T3 | Presentation of prototype | MacBook Air | iPhone 13 mini | - | | Silver | FFEA00 | Error message “Mobile device cannot be displayed as Laptop device” | The design is displayed on the iPhone 13mini dimensions, the rest of the space in MacBook Air is empty. The device outline is Silver and the background color is FF | | | Chrome | Fail |  |
| T4 | Restart of prototype | ‘r’ key | | | | | | Prototype should restart from the first frame. | Prototype restarts from the first frame. | | | Chrome | Pass |  |
| T5 | Presentation of prototype | ‘forward arrow’ key(->) | | | | | | Prototype should move to the next frame. | Prototype moves to the next frame. | | | Chrome | Pass |  |

## TS10: Check Team Creation

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | | T10 | | | | | **Test Case ID** | T1, T2, T3, T4, T5 |  | |
| **Test Case Description** | | | 1. The User enters any team name, invalid email, and chooses starter plan for team creation. 2. The User enters any name, skips email, and chooses starter plan for team creation. 3. The User does not enter any team name and clicks Create Team. 4. The User enters any team name, enters the same email two times, chooses starter plan, and clicks on Create Team. 5. The Users enter a team name that already exists, skips email, and chooses starter plan and clicks Create Team. | | | | | **Test Priority** | Medium |
| **Pre-Requisite** | | | User must be logged into his/her Figma account. | | | | | **Post-Requisite** | Team is successfully created |
| **Test Execution Steps:**   1. Go to Figma. 2. Log in to Figma. 3. Click on Create team 4. Enter a suitable name for the team. 5. Enter email/s of team member/s 6. Choose a plan (Starter, Professional, Organization). | | | | | | | | | | | |
| **Sr#** | **Action** | **Inputs** | | | | **Expected Output** | **Actual Output** | **Test Browser** | **Test Result** | | **Test Comments** |
| **Team Name** | | **Email** | **Plan** |
| T1 | Team Creation | Any name | | Invalid email | Option “Starter” | Error message “Invalid Email” | Error message “Invalid Email” | Chrome | Pass | |  |
| T2 | Team Creation | Any name | | “Skip for now” | Option “Starter” | Navigated to Home Page | Navigated to the Home Page | Chrome | Pass | |  |
| T3 | Team Creation | Not entered | | - | - | Error message “Your team name is empty.” | Error message “Your team name is empty.” | Chrome | Pass | |  |
| T4 | Team Creation | Any name entered | | Same email address was entered two times | Option “Starter” | Error message “Duplicate email addresses are not allowed” | Email sent to member and navigated to the Home Page | Chrome | Fail | |  |
| T5 | Team Creation | Team name entered that already exists | | “Skip for now” | Option “Starter” | Error message “Team with this name already exists” | Navigated to the Home Page | Chrome | Fail | |  |

## TS11: Check typography tools.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 11 | | | | | **Test Case ID** | T1, T2, T3, T4, T5 | |  | |
| **Test Case Description** | | 1. User inputs valid font style 2. User inputs invalid font style 3. User inputs valid font weigh. 4. User inputs valid font size 5. User inputs invalid font size | | | | | **Test Priority** | High | |
| **Pre-Requisite** | | * User should be logged in * User should have a design project open * User should have some text written in the text box | | | | | **Post-Requisite** | * The text should be styled successfully | |
| **Test Execution Steps:**   1. Open Figma in browser 2. Open Figma project. 3. Select a text to apply typography. 4. Apply any valid typography (font style, font weight, font size) to the text element. | | | | | | | | | | | |
| **Sr** | **Action** | | **Inputs** | | | **Expected Output** | **Actual Output** | **Test Browser** | **Test Result** | | **Test Comments** |
| **Font Style** | **Font Weight** | **Font Size** |
| T1 | Apply font style | | Valid | - | - | Font style should apply on text | Font style was applied on text | Chrome | Pass | |  |
| T2 | Apply font style | | Invalid | - | - | Font style should not apply on text | Font style was not applied on text | Chrome | Pass | |  |
| T3 | Apply font weight | | - | Valid | - | Font weight should apply on text | Font weight was applied on text | Chrome | Pass | |  |
| T4 | Apply font size | | - | - | Valid | Font size should apply on text | Font size was applied on text | Chrome | Pass | |  |
| T5 | Apply font size | | - | - | Invalid | Font size should not apply and error message “Font size cannot be >=0” should be displayed. | Font size 1 is applied | Chrome | Fail | |  |

## TS12: Check frame creation.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 12 | | | | | | | **Test Case ID** | | T1, T2, T3, T4, T5 | | |  | |
| **Test Case Description** | | 1. A customized-sized frame is created manually. 2. A standard device frame is inserted. 3. The Frame appears as a single unit after grouping them into one. 4. The frame can be renamed. 5. Borders are added to the frame after creating them to visualize them properly. | | | | | | | **Test Priority** | | HIGH | | |
| **Pre-Requisite** | | * User should be logged in. | | | | | | | **Post-Requisite** | | Frame will be created successfully. | | |
| **Test Execution Steps:**  **T1:**   1. Open the Figma file in Chrome. 2. Press the “F” key to activate the Frame tool. 3. Click and drag on the canvas to draw a custom frame. 4. Release the mouse to set the frame size. 5. Confirm that a custom-sized frame appears. | | | **Test Execution Steps:**  **T2:**   1. Open Figma in Chrome. 2. Click on the **Frame Tool icon** in the toolbar. 3. From the right panel, choose a preset like iPhone. 4. Click on the selected frame type. 5. Confirm that the selected standard device frame is added to the canvas. | | **Test Execution Steps:**  **T3:**   1. Press **“F”** and drag twice to create two frames on the canvas. 2. Place both frames inside a larger parent frame. 3. Select both frames. 4. Press **Ctrl + G** to group them. 5. Confirm that the grouped frames appear as a single frame. | | | **Test Execution Steps:**  **T4:**   1. Select any frame on the canvas. 2. Double-click on the frame name in the layers panel or directly on the canvas. 3. Type a new name for the frame. 4. Press Enter to apply the name. 5. Confirm that the frame is renamed successfully. | | | | **Test Execution Steps:**  **T5:**   1. Select a frame on the canvas. 2. Go to the right properties panel. 3. Click on the “Stroke” checkbox to enable the border. 4. Adjust stroke color or size if needed. 5. Confirm that the frame has a visible border. | | | |
| **Sr** | **Action** | | | **Inputs** | | **Expected Output** | **Actual Output** | | | **Test Browser** | | | **Test Result** | | **Test Comments** |
| T1 | Frame Creation | | | Press “F” + Drag | | Creates a customize-sized frame manually. | Created a customized-sized frame manually. | | | Chrome | | | Pass | |  |
| T2 | Frame Creation | | | Click “Frame Tool Icon” + select frame type (e.g., iPhone) | | Inserts a standard device frame. | Inserted a standard device frame. | | | Chrome | | | Pass | |  |
| T3 | Frame Creation | | | Press “F” + Drag inside the parent frame + Group these 2 frames by pressing ctrl + “G”. | | Appears as a single frame. | Appeared as a single frame. | | | Chrome | | | Pass | |  |
| T4 | Frame Creation | | | Double click on frame name + Add text | | Frame name is renamed. | Frame name is renamed | | | Chrome | | | Pass | |  |
| T5 | Frame Creation | | | Click on frame + select “Stroke” option in right panel. | | Adds Borders to the frame. | Border is added to the frame. | | | Chrome | | | Pass | |  |

## TS13: Check solid color fill

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | 13 | | | | | | **Test Case ID** | | | T1, T2, T3, T4, T5 | |  | | |
| **Test Case Description** | 1. Users creates the shape and it fills with default solid color. 2. User chooses a color from color picker. 3. Users apply color code in fill section. 4. User removes the color of the shape. 5. User hides the shape in the fill section. | | | | | | **Test Priority** | | | MEDIUM | |
| **Pre-Requisite** | * User should be logged in. * User should be in design file. * User should draw a shape (Rectangle, Circle etc.) | | | | | | **Post-Requisite** | | | Shape is filled with solid color. | |
| **Test Execution Steps:**  **T1:**   1. Open Figma in Chrome. 2. Draw or select a rectangle on the canvas. 3. Go to the right panel and click on “Fill”. 4. Observe that a default solid grey-black color is applied. | | | **Test Execution Steps:**  **T2:**   1. Open Figma in Chrome. 2. Select a rectangle on the canvas. 3. Click on the **“Fill”** option in the right panel. 4. Open the color picker and choose red. 5. Apply the selected color. | | **Test Execution Steps:**  **T3:**   1. Open Figma in Chrome. 2. Select the rectangle on the canvas. 3. Click **“Fill”** in the right panel. 4. In the color input, type the hex code #FF5733. 5. Press Enter to apply the color. | | | | **Test Execution Steps:**  **T4:**   1. Open Figma in Chrome. 2. Select a rectangle that already has a fill color. 3. In the right panel, click the **“-” (**minus**)** icon next to Fill. 4. The color should be removed. 5. Check that the rectangle has no fill color. | | **Test Execution Steps:**  **T5:**   1. Open Figma in Chrome. 2. Select a rectangle with a fill color. 3. In the Fill section of the right panel, click the “eye (hide)” icon. 4. The fill color should disappear from view. 5. Confirm that the shape is still there but without visible fill. | | | |
| **Sr** | | **Action** | | **Inputs** | | **Expected Output** | | **Actual Output** | | **Test Browser** | | | **Test Result** | **Test Comments** |
| T1 | | Solid Fill Color | | Select Rectangle +  Click “Fill” on right panel | | Rectangle fills with default solid color grey-black | | Rectangle is filled with default solid color grey-black | | Chrome | | | Pass |  |
| T2 | | Solid Fill Color | | Select Rectangle +  Click “Fill” on right panel + Pick red color | | Rectangle is fills with red color | | Rectangle is filled with red color | | Chrome | | | Pass |  |
| T3 | | Solid Fill Color | | Select Rectangle + Apply code “#FF5733” in fill on right panel | | Rectangle fills with #FF5733 color. | | Rectangle fills with #FF5733 color. | | Chrome | | | Pass |  |
| T4 | | Solid Fill Color | | Select Rectangle + click “-” in fill on right panel | | Solid color of shape removes. | | Solid color is removed. | | Chrome | | | Pass |  |
| T5 | | Solid Fill Color | | Select Rectangle + click “hide” icon in fill on the right panel | | Hides the shape color. | | Shape color hides successfully. | | Chrome | | | Pass |  |

## TS14: Check Alignment Tools

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | | 14 | | | | **Test Case ID** | TI, T2, T3, T4, T5 | |  | |
| **Test Case Description** | | | 1. User inputs left in horizontal and top in vertical alignment 2. User inputs right in horizontal and bottom in vertical alignment 3. User inputs center in horizontal and center in vertical alignment 4. User inputs left in horizontal alignment 5. User inputs bottom in vertical alignment | | | | **Test Priority** | High | |
| **Pre-Requisite** | | | * User should be logged in * User should have a design project open * User should have some element | | | | **Post-Requisite** | * The element should be aligned successfully | |
| **Test Execution Steps:**   1. Open Figma in browser 2. Open Figma project. 3. Select an element to align. 4. Apply horizontal alignment to the element. 5. Apply vertical alignment to the element. | | | | | | | | | | | |
| **Sr** | **Action** | **Inputs** | | | **Expected Output** | **Actual Output** | | **Test Browser** | **Test Result** | | **Test Comments** |
| **Horizontal Alignment** | | **Vertical Alignment** |
| T1 | Align element | Left | | Top | Element should align horizontally left and vertically at top | Element aligns horizontally left and vertically at top | | Chrome | Pass | |  |
| T2 | Align element | Right | | Bottom | Element should align horizontally right and vertically at bottom | Element aligns horizontally right and vertically at bottom | | Chrome | Pass | |  |
| T3 | Align element | Center | | Center | Element should align horizontally center and vertically at center | Element aligns horizontally center and vertically at center | | Chrome | Pass | |  |
| T4 | Align element | Left | | - | Element should align horizontally left | Element aligns horizontally left | | Chrome | Pass | |  |
| T5 | Align element | - | | Bottom | Element should align vertically at bottom | Element aligns vertically at bottom | | Chrome | Pass | |  |

## TS15: Check exporting assets

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario ID** | | 15 | | | | | **Test Case ID** | | T1, T2, T3, T4, T5 | |  | |
| **Test Case Description** | | 1. User exports the asset in PNG format and 1.75x. 2. User exports the asset in SVG format and 2x. 3. User exports the asset in PDF format and 3x. 4. User exports the asset in JPG format and 4x. 5. User exports the asset in Default(PNG) format and default(1x). | | | | | **Test Priority** | | MEDIUM | |
| **Pre-Requisite** | | * User must be logged in. * A design component (icon, frame, shape) must be present in the file. | | | | | **Post-Requisite** | | Assists will be exported successfully. | |
| **Test Execution Steps:**   1. Open Figma. 2. Open any file. 3. Select any frame or shape. 4. Click on the “Export” section from the right panel. 5. Choose format and size (PNG, SVG, PDF). 6. Click “Export”. | | | | | | | | | | | | |
| **Sr** | **Action** | | **Inputs** | | **Expected Output** | **Actual Output** | | **Test Browser** | | **Test Result** | | **Test Comments** |
| **Format** | **Size** |
| T1 | Exporting assets | | PNG | 1.75x | |  | | --- | | Image in PNG format at 1.75x resolution | | |  | | --- | | Image in PNG format at 1.75x resolution | | | Chrome | | Pass | |  |
| T2 | Exporting assets | | SVG | 2x | |  | | --- | | Vector SVG image at 2x resolution | | |  | | --- | | Vector SVG image at 2x resolution | | | Chrome | | Pass | |  |
| T3 | Exporting assets | | PDF | 3x | |  | | --- | | Document in PDF format at 3x resolution | | |  | | --- | | Document in PDF format at 3x resolution | | | Chrome | | Pass | |  |
| T4 | Exporting assets | | JPG | 4x | |  | | --- | | Image in JPG format at 4x resolution | | |  | | --- | | Image in JPG format at 4x resolution | | | Chrome | | Pass | |  |
| T5 | Exporting assets | | Default (PNG) | Default (1x) | PNG image at normal 1x resolution | PNG image at normal 1x resolution | | Chrome | | Pass | |  |

# Traceability Matrices:

## Forward Traceability Matrix:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario ID** | **Description** | **Test Case ID** | **Description** | **Test Case Status** |
| TS – 1 | User shall be able to log in. | TC 01 | User login with valid email and correct password. | Pass |
| TS – 1 | User shall be able to log in. | TC 02 | User login with valid email and wrong password. | Fail |
| TS – 1 | User shall be able to log in. | TC 03 | User logins with empty email and correct password. | Pass |
| TS – 1 | User shall be able to log in. | TC 04 | User logins with valid email and empty password. | Pass |
| TS – 1 | User shall be able to log in. | TC 05 | User enters invalid email format. | Pass |
| TS – 2 | User shall be able to register. | TC 01 | The user enters invalid email and a valid password. | Pass |
| TS – 2 | User shall be able to register. | TC 02 | The User enters valid email and an invalid password. | Pass |
| TS – 2 | User shall be able to register. | TC 03 | The User enters valid email and a valid password. | Pass |
| TS – 2 | User shall be able to register. | TC 04 | The User enters an invalid email and an invalid password. | Fail |
| TS – 2 | User shall be able to register. | TC 05 | The User chooses a Google account to create Figma account. | Pass |
| TS – 3 | User shall be able to collaborate in real-time. | TC 01 | All the users access the same file through a single link to collaborate with each other. | Pass |
| TS – 3 | User shall be able to collaborate in real-time. | TC 02 | If one user renames any element in the design file that will be update in all the files of other users also. | Pass |
| TS – 3 | User shall be able to collaborate in real-time. | TC 03 | If User A is editing a shape, User B shouldn't be able to edit it until User A is done. | Pass |
| TS – 3 | User shall be able to collaborate in real-time. | TC 04 | Both users’ cursors appear in different colors with labels, indicating real-time presence. | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 01 | User creates component by right click. | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 02 | User creates component by button | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 03 | User instantiates component by dragging. | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 04 | User instantiates components by insert instance button. | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 05 | User adds variant. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 01 | User adds auto layout to a frame or a group of elements. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 02 | User adjusts padding, spacing between items, or alignment. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 03 | User nests auto layouts inside other auto layouts. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 04 | User duplicates a layout and all constraints are retained. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 05 | User removes auto layout from a frame. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 01 | User applied on click interaction between 2 frames. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 02 | User applied on drag interaction between 2 frames. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 03 | User applied while hovering interaction between 2 frames. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 04 | User applied while pressing the interaction on the same frame. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 05 | User applied interaction between a frame and a non-frame element. | Pass |
| TS – 7 | User shall be able to share files and set access permissions. | TC 01 | The user enters a valid email and gives Can Edit access permission. | Pass |
| TS – 7 | User shall be able to share files and set access permissions. | TC 02 | The user enters a valid email and gives Can View access permission. | Fail |
| TS – 7 | User shall be able to share files and set access permissions. | TC 03 | The user enters an invalid email and gives Can Edit access permission. | Pass |
| TS – 7 | User shall be able to share files and set access permissions. | TC 04 | The user enters an invalid email and gives Can View access permission. | Pass |
| TS – 7 | User shall be able to share files and set access permissions. | TC 05 | The user copies link of the file. | Pass |
| TS – 8 | User shall be able to add comments. | TC 01 | User add comments on Figma design. | Pass |
| TS – 8 | User shall be able to add comments. | TC 02 | User view comments of other team members on Figma. | Pass |
| TS – 8 | User shall be able to add comments. | TC 03 | User reply to team members through comments. | Fail |
| TS – 8 | User shall be able to add comments. | TC 04 | User resolves issues through comments that are occurring in their ongoing project on Figma. | Fail |
| TS – 8 | User shall be able to add comments. | TC 05 | User deletes a comment by clicking on the delete button. | Pass |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 01 | User selects iPhone 13 mini (initial frame), iPhone 13 mini (device), portrait, white device outline, and 000000 background color | Pass |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 02 | User selects iPhone 13 mini (initial frame), iPhone 16 (device), landscape axis, pink device outline, and white (FFFFFF) background color. | Fail |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 03 | User selects iPhone 13 mini (initial frame), MacBook Air (device), silver device outline, and yellow (FFEA00) background color. | Pass |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 04 | User presses “r” key to restart presentation. | Pass |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 05 | User presses forward arrow key to move to the next frame. | Pass |
| TS – 10 | User shall be able to create teams. | TC 01 | The User enters any team name, invalid email, and chooses starter plan for team creation. | Pass |
| TS – 10 | User shall be able to create teams. | TC 02 | The User enters any name, skips email, and chooses starter plan for team creation. | Pass |
| TS – 10 | User shall be able to create teams. | TC 03 | The User does not enter any team name and clicks Create Team. | Pass |
| TS – 10 | User shall be able to create teams. | TC 04 | The User enters any team name, enters the same email two times, chooses starter plan, and clicks on Create Team. | Fail |
| TS – 10 | User shall be able to create teams. | TC 05 | The Users enter a team name that already exists, skips email, and chooses starter plan and clicks Create Team. | Fail |
| TS – 11 | User shall be able to use typography tools. | TC 01 | User inputs valid font style | Pass |
| TS – 11 | User shall be able to use typography tools. | TC 02 | User inputs invalid font style | Pass |
| TS – 11 | User shall be able to use typography tools. | TC 03 | User inputs valid font weigh. | Pass |
| TS – 11 | User shall be able to use typography tools. | TC 04 | User inputs valid font size | Pass |
| TS – 11 | User shall be able to use typography tools. | TC 05 | User inputs invalid font size | Fail |
| TS – 12 | User shall be able to create frames. | TC 01 | A customized-sized frame is created manually. | Pass |
| TS – 12 | User shall be able to create frames. | TC 02 | A standard device frame is inserted. | Pass |
| TS – 12 | User shall be able to create frames. | TC 03 | The Frame appears as a single unit after grouping them into one. | Pass |
| TS – 12 | User shall be able to create frames. | TC 04 | The frame can be renamed. | Pass |
| TS – 12 | User shall be able to create frames. | TC 05 | Borders are added to the frame after creating them to visualize them properly. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 01 | Users creates the shape and it fills with default solid color. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 02 | User chooses a color from color picker. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 03 | Users apply color code in fill section. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 04 | User removes the color of the shape. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 05 | User hides the shape in the fill section. | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 01 | User inputs left in horizontal and top in vertical alignment | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 02 | User inputs right in horizontal and bottom in vertical alignment | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 03 | User inputs center in horizontal and center in vertical alignment | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 04 | User inputs left in horizontal alignment | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 05 | User inputs bottom in vertical alignment | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 01 | User exports the asset in PNG format and 1.75x. | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 02 | User exports the asset in SVG format and 2x. | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 03 | User exports the asset in PDF format and 3x. | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 04 | User exports the asset in JPG format and 4x. | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 05 | User exports the asset in Default(PNG) format and default(1x). | Pass |

## Backward Traceability Matrix:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Case Description** | **Scenario ID** | **Scenario Description** | **Test Case Status** |
| TC 01 | User login with valid email and correct password. | TS – 1 | User shall be able to log in. | Pass |
| TC 02 | User login with valid email and wrong password. | TS – 1 | User shall be able to log in. | Fail |
| TC 03 | User logins with empty email and correct password. | TS – 1 | User shall be able to log in. | Pass |
| TC 04 | User logins with valid email and empty password. | TS – 1 | User shall be able to log in. | Pass |
| TC 05 | User enters invalid email format. | TS – 1 | User shall be able to log in. | Pass |
| TC 01 | The user enters invalid email and a valid password. | TS – 2 | User shall be able to register. | Pass |
| TC 02 | The User enters valid email and an invalid password. | TS – 2 | User shall be able to register. | Pass |
| TC 03 | The User enters valid email and a valid password. | TS – 2 | User shall be able to register. | Pass |
| TC 04 | The User enters an invalid email and an invalid password. | TS – 2 | User shall be able to register. | Fail |
| TC 05 | The User chooses a Google account to create Figma account. | TS – 2 | User shall be able to register. | Pass |
| TC 01 | All the users access the same file through a single link to collaborate with each other. | TS – 3 | User shall be able to collaborate in real-time. | Pass |
| TC 02 | If one user renames any element in the design file that will be update in all the files of other users also. | TS – 3 | User shall be able to collaborate in real-time. | Pass |
| TC 03 | If User A is editing a shape, User B shouldn't be able to edit it until User A is done. | TS – 3 | User shall be able to collaborate in real-time. | Pass |
| TC 04 | Both users’ cursors appear in different colors with labels, indicating real-time presence. | TS – 3 | User shall be able to collaborate in real-time. | Pass |
| TC 01 | User creates component by right click. | TS – 4 | User shall be able to create and manage design components and variants. | Pass |
| TC 02 | User creates component by button | TS – 4 | User shall be able to create and manage design components and variants. | Pass |
| TC 03 | User instantiates component by dragging. | TS – 4 | User shall be able to create and manage design components and variants. | Pass |
| TC 04 | User instantiates components by insert instance button. | TS – 4 | User shall be able to create and manage design components and variants. | Pass |
| TC 05 | User adds variant. | TS – 4 | User shall be able to create and manage design components and variants. | Pass |
| TC 01 | User adds auto layout to a frame or a group of elements. | TS – 5 | User shall be able to apply auto layout. | Pass |
| TC 02 | User adjusts padding, spacing between items, or alignment. | TS – 5 | User shall be able to apply auto layout. | Pass |
| TC 03 | User nests auto layouts inside other auto layouts. | TS – 5 | User shall be able to apply auto layout. | Pass |
| TC 04 | User duplicates a layout and all constraints are retained. | TS – 5 | User shall be able to apply auto layout. | Pass |
| TC 05 | User removes auto layout from a frame. | TS – 5 | User shall be able to apply auto layout. | Pass |
| TC 01 | User applied on click interaction between 2 frames. | TS – 6 | User shall be able to apply prototype interactions. | Pass |
| TC 02 | User applied on drag interaction between 2 frames. | TS – 6 | User shall be able to apply prototype interactions. | Pass |
| TC 03 | User applied while hovering interaction between 2 frames. | TS – 6 | User shall be able to apply prototype interactions. | Pass |
| TC 04 | User applied while pressing the interaction on the same frame. | TS – 6 | User shall be able to apply prototype interactions. | Pass |
| TC 05 | User applied interaction between a frame and a non-frame element. | TS – 6 | User shall be able to apply prototype interactions. | Pass |
| TC 01 | The user enters a valid email and gives Can Edit access permission. | TS – 7 | User shall be able to share files and set access permissions. | Pass |
| TC 02 | The user enters a valid email and gives Can View access permission. | TS – 7 | User shall be able to share files and set access permissions. | Fail |
| TC 03 | The user enters an invalid email and gives Can Edit access permission. | TS – 7 | User shall be able to share files and set access permissions. | Pass |
| TC 04 | The user enters an invalid email and gives Can View access permission. | TS – 7 | User shall be able to share files and set access permissions. | Pass |
| TC 05 | The user copies link of the file. | TS – 7 | User shall be able to share files and set access permissions. | Pass |
| TC 01 | User add comments on Figma design. | TS – 8 | User shall be able to add comments. | Pass |
| TC 02 | User view comments of other team members on Figma. | TS – 8 | User shall be able to add comments. | Pass |
| TC 03 | User reply to team members through comments. | TS – 8 | User shall be able to add comments. | Fail |
| TC 04 | User resolves issues through comments that are occurring in their ongoing project on Figma. | TS – 8 | User shall be able to add comments. | Fail |
| TC 05 | User deletes a comment by clicking on the delete button. | TS – 8 | User shall be able to add comments. | Pass |
| TC 01 | User selects iPhone 13 mini (initial frame), iPhone 13 mini (device), portrait, white device outline, and 000000 background color | TS – 9 | User shall be able to view prototypes in presentation mode. | Pass |
| TC 02 | User selects iPhone 13 mini (initial frame), iPhone 16 (device), landscape axis, pink device outline, and white (FFFFFF) background color. | TS – 9 | User shall be able to view prototypes in presentation mode. | Fail |
| TC 03 | User selects iPhone 13 mini (initial frame), MacBook Air (device), silver device outline, and yellow (FFEA00) background color. | TS – 9 | User shall be able to view prototypes in presentation mode. | Pass |
| TC 04 | User presses “r” key to restart presentation. | TS – 9 | User shall be able to view prototypes in presentation mode. | Pass |
| TC 05 | User presses forward arrow key to move to the next frame. | TS – 9 | User shall be able to view prototypes in presentation mode. | Pass |
| TC 01 | The User enters any team name, invalid email, and chooses starter plan for team creation. | TS – 10 | User shall be able to create teams. | Pass |
| TC 02 | The User enters any name, skips email, and chooses starter plan for team creation. | TS – 10 | User shall be able to create teams. | Pass |
| TC 03 | The User does not enter any team name and clicks Create Team. | TS – 10 | User shall be able to create teams. | Pass |
| TC 04 | The User enters any team name, enters the same email two times, chooses starter plan, and clicks on Create Team. | TS – 10 | User shall be able to create teams. | Fail |
| TC 05 | The Users enter a team name that already exists, skips email, and chooses starter plan and clicks Create Team. | TS – 10 | User shall be able to create teams. | Fail |
| TC 01 | User inputs valid font style | TS – 11 | User shall be able to use typography tools. | Pass |
| TC 02 | User inputs invalid font style | TS – 11 | User shall be able to use typography tools. | Pass |
| TC 03 | User inputs valid font weigh. | TS – 11 | User shall be able to use typography tools. | Pass |
| TC 04 | User inputs valid font size | TS – 11 | User shall be able to use typography tools. | Pass |
| TC 05 | User inputs invalid font size | TS – 11 | User shall be able to use typography tools. | Fail |
| TC 01 | A customized-sized frame is created manually. | TS – 12 | User shall be able to create frames. | Pass |
| TC 02 | A standard device frame is inserted. | TS – 12 | User shall be able to create frames. | Pass |
| TC 03 | The Frame appears as a single unit after grouping them into one. | TS – 12 | User shall be able to create frames. | Pass |
| TC 04 | The frame can be renamed. | TS – 12 | User shall be able to create frames. | Pass |
| TC 05 | Borders are added to the frame after creating them to visualize them properly. | TS – 12 | User shall be able to create frames. | Pass |
| TC 01 | Users creates the shape and it fills with default solid color. | TS – 13 | User shall be able to apply solid color fills. | Pass |
| TC 02 | User chooses a color from color picker. | TS – 13 | User shall be able to apply solid color fills. | Pass |
| TC 03 | Users apply color code in fill section. | TS – 13 | User shall be able to apply solid color fills. | Pass |
| TC 04 | User removes the color of the shape. | TS – 13 | User shall be able to apply solid color fills. | Pass |
| TC 05 | User hides the shape in the fill section. | TS – 13 | User shall be able to apply solid color fills. | Pass |
| TC 01 | User inputs left in horizontal and top in vertical alignment | TS – 14 | User shall be able to use alignment tools. | Pass |
| TC 02 | User inputs right in horizontal and bottom in vertical alignment | TS – 14 | User shall be able to use alignment tools. | Pass |
| TC 03 | User inputs center in horizontal and center in vertical alignment | TS – 14 | User shall be able to use alignment tools. | Pass |
| TC 04 | User inputs left in horizontal alignment | TS – 14 | User shall be able to use alignment tools. | Pass |
| TC 05 | User inputs bottom in vertical alignment | TS – 14 | User shall be able to use alignment tools. | Pass |
| TC 01 | User exports the asset in PNG format and 1.75x. | TS – 15 | **User shall be able to export design assets.** | Pass |
| TC 02 | User exports the asset in SVG format and 2x. | TS – 15 | **User shall be able to export design assets.** | Pass |
| TC 03 | User exports the asset in PDF format and 3x. | TS – 15 | **User shall be able to export design assets.** | Pass |
| TC 04 | User exports the asset in JPG format and 4x. | TS – 15 | **User shall be able to export design assets.** | Pass |
| TC 05 | User exports the asset in Default(PNG) format and default(1x). | TS – 15 | **User shall be able to export design assets.** | Pass |

## Bi-Directional Traceability Matrix:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Scenario ID** | **Description** | **Test Case ID** | **Description** | **Test Case Status** |
| TS – 1 | User shall be able to log in. | TC 01 | User login with valid email and correct password. | Pass |
| User shall be able to log in. | TC 02 | User login with valid email and wrong password. | Fail |
| User shall be able to log in. | TC 03 | User logins with empty email and correct password. | Pass |
| User shall be able to log in. | TC 04 | User logins with valid email and empty password. | Pass |
| User shall be able to log in. | TC 05 | User enters invalid email format. | Pass |
| TS – 2 | User shall be able to register. | TC 01 | The user enters invalid email and a valid password. | Pass |
| User shall be able to register. | TC 02 | The User enters valid email and an invalid password. | Pass |
| User shall be able to register. | TC 03 | The User enters valid email and a valid password. | Pass |
| User shall be able to register. | TC 04 | The User enters an invalid email and an invalid password. | Fail |
| User shall be able to register. | TC 05 | The User chooses a Google account to create Figma account. | Pass |
| TS – 3 | User shall be able to collaborate in real-time. | TC 01 | All the users access the same file through a single link to collaborate with each other. | Pass |
| User shall be able to collaborate in real-time. | TC 02 | If one user renames any element in the design file that will be update in all the files of other users also. | Pass |
| User shall be able to collaborate in real-time. | TC 03 | If User A is editing a shape, User B shouldn't be able to edit it until User A is done. | Pass |
| User shall be able to collaborate in real-time. | TC 04 | Both users’ cursors appear in different colors with labels, indicating real-time presence. | Pass |
| TS – 4 | User shall be able to create and manage design components and variants. | TC 01 | User creates component by right click. | Pass |
| User shall be able to create and manage design components and variants. | TC 02 | User creates component by button | Pass |
| User shall be able to create and manage design components and variants. | TC 03 | User instantiates component by dragging. | Pass |
| User shall be able to create and manage design components and variants. | TC 04 | User instantiates components by insert instance button. | Pass |
| User shall be able to create and manage design components and variants. | TC 05 | User adds variant. | Pass |
| TS – 5 | User shall be able to apply auto layout. | TC 01 | User adds auto layout to a frame or a group of elements. | Pass |
| User shall be able to apply auto layout. | TC 02 | User adjusts padding, spacing between items, or alignment. | Pass |
| User shall be able to apply auto layout. | TC 03 | User nests auto layouts inside other auto layouts. | Pass |
| User shall be able to apply auto layout. | TC 04 | User duplicates a layout and all constraints are retained. | Pass |
| User shall be able to apply auto layout. | TC 05 | User removes auto layout from a frame. | Pass |
| TS – 6 | User shall be able to apply prototype interactions. | TC 01 | User applied on click interaction between 2 frames. | Pass |
| User shall be able to apply prototype interactions. | TC 02 | User applied on drag interaction between 2 frames. | Pass |
| User shall be able to apply prototype interactions. | TC 03 | User applied while hovering interaction between 2 frames. | Pass |
| User shall be able to apply prototype interactions. | TC 04 | User applied while pressing the interaction on the same frame. | Pass |
| User shall be able to apply prototype interactions. | TC 05 | User applied interaction between a frame and a non-frame element. | Pass |
| TS – 7 | User shall be able to share files and set access permissions. | TC 01 | The user enters a valid email and gives Can Edit access permission. | Pass |
| User shall be able to share files and set access permissions. | TC 02 | The user enters a valid email and gives Can View access permission. | Fail |
| User shall be able to share files and set access permissions. | TC 03 | The user enters an invalid email and gives Can Edit access permission. | Pass |
| User shall be able to share files and set access permissions. | TC 04 | The user enters an invalid email and gives Can View access permission. | Pass |
| User shall be able to share files and set access permissions. | TC 05 | The user copies link of the file. | Pass |
| TS – 8 | User shall be able to add comments. | TC 01 | User add comments on Figma design. | Pass |
| User shall be able to add comments. | TC 02 | User view comments of other team members on Figma. | Pass |
| User shall be able to add comments. | TC 03 | User reply to team members through comments. | Fail |
| User shall be able to add comments. | TC 04 | User resolves issues through comments that are occurring in their ongoing project on Figma. | Fail |
| User shall be able to add comments. | TC 05 | User deletes a comment by clicking on the delete button. | Pass |
| TS – 9 | User shall be able to view prototypes in presentation mode. | TC 01 | User selects iPhone 13 mini (initial frame), iPhone 13 mini (device), portrait, white device outline, and 000000 background color | Pass |
| User shall be able to view prototypes in presentation mode. | TC 02 | User selects iPhone 13 mini (initial frame), iPhone 16 (device), landscape axis, pink device outline, and white (FFFFFF) background color. | Fail |
| User shall be able to view prototypes in presentation mode. | TC 03 | User selects iPhone 13 mini (initial frame), MacBook Air (device), silver device outline, and yellow (FFEA00) background color. | Pass |
| User shall be able to view prototypes in presentation mode. | TC 04 | User presses “r” key to restart presentation. | Pass |
| User shall be able to view prototypes in presentation mode. | TC 05 | User presses forward arrow key to move to the next frame. | Pass |
| TS – 10 | User shall be able to create teams. | TC 01 | The User enters any team name, invalid email, and chooses starter plan for team creation. | Pass |
| User shall be able to create teams. | TC 02 | The User enters any name, skips email, and chooses starter plan for team creation. | Pass |
| User shall be able to create teams. | TC 03 | The User does not enter any team name and clicks Create Team. | Pass |
| User shall be able to create teams. | TC 04 | The User enters any team name, enters the same email two times, chooses starter plan, and clicks on Create Team. | Fail |
| User shall be able to create teams. | TC 05 | The Users enter a team name that already exists, skips email, and chooses starter plan and clicks Create Team. | Fail |
| TS – 11 | User shall be able to use typography tools. | TC 01 | User inputs valid font style | Pass |
| User shall be able to use typography tools. | TC 02 | User inputs invalid font style | Pass |
| User shall be able to use typography tools. | TC 03 | User inputs valid font weigh. | Pass |
| User shall be able to use typography tools. | TC 04 | User inputs valid font size | Pass |
| User shall be able to use typography tools. | TC 05 | User inputs invalid font size | Fail |
| TS – 12 | User shall be able to create frames. | TC 01 | A customized-sized frame is created manually. | Pass |
| User shall be able to create frames. | TC 02 | A standard device frame is inserted. | Pass |
| User shall be able to create frames. | TC 03 | The Frame appears as a single unit after grouping them into one. | Pass |
| User shall be able to create frames. | TC 04 | The frame can be renamed. | Pass |
| User shall be able to create frames. | TC 05 | Borders are added to the frame after creating them to visualize them properly. | Pass |
| TS – 13 | User shall be able to apply solid color fills. | TC 01 | Users creates the shape and it fills with default solid color. | Pass |
| User shall be able to apply solid color fills. | TC 02 | User chooses a color from color picker. | Pass |
| User shall be able to apply solid color fills. | TC 03 | Users apply color code in fill section. | Pass |
| User shall be able to apply solid color fills. | TC 04 | User removes the color of the shape. | Pass |
| User shall be able to apply solid color fills. | TC 05 | User hides the shape in the fill section. | Pass |
| TS – 14 | User shall be able to use alignment tools. | TC 01 | User inputs left in horizontal and top in vertical alignment | Pass |
| User shall be able to use alignment tools. | TC 02 | User inputs right in horizontal and bottom in vertical alignment | Pass |
| User shall be able to use alignment tools. | TC 03 | User inputs center in horizontal and center in vertical alignment | Pass |
| User shall be able to use alignment tools. | TC 04 | User inputs left in horizontal alignment | Pass |
| User shall be able to use alignment tools. | TC 05 | User inputs bottom in vertical alignment | Pass |
| TS – 15 | **User shall be able to export design assets.** | TC 01 | User exports the asset in PNG format and 1.75x. | Pass |
| **User shall be able to export design assets.** | TC 02 | User exports the asset in SVG format and 2x. | Pass |
| **User shall be able to export design assets.** | TC 03 | User exports the asset in PDF format and 3x. | Pass |
| **User shall be able to export design assets.** | TC 04 | User exports the asset in JPG format and 4x. | Pass |
| **User shall be able to export design assets.** | TC 05 | User exports the asset in Default(PNG) format and default(1x). | Pass |

## Validation Traceability Matrix

|  |  |  |
| --- | --- | --- |
| **Test Scenario ID** | **Validation Activity** | **Validation Result** |
| TS-01 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Pass, Pass, Fail |
| TS-02 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Pass, Pass, Fail |
| TS-03 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Pass, Pass, Pass |
| TS-04 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Fail, Pass, Pass |
| TS-05 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Fail, Pass, Pass |
| TS-06 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Fail, Pass, Pass |
| TS-07 | Black Box Texting, Usability Testing, Exploratory Testing, Performance Testing, Acceptance Testing | Pass, Pass, Fail, Pass, Pass |
| TS-08 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Fail, Pass, Fail |
| TS-09 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Pass, Fail, Fail |
| TS-10 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Fail, Fail, Fail |
| TS-11 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Fail, Pass, Pass, Fail |
| TS-12 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Pass, Pass, Pass |
| TS-13 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Fail, Pass, Pass |
| TS-14 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Pass, Pass, Pass |
| TS-15 | Black Box Texting, Usability Testing, Performance Testing, Acceptance Testing | Pass, Pass, Fail, Pass |

## Horizontal Traceability matrix

|  |  |
| --- | --- |
| **Test Scenario ID** | **Version 1** |
| TS-01 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-02 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-03 | TC-01, TC-02,TC-03, TC-04 |
| TS-04 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-05 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-06 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-07 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-08 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-09 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-10 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-11 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-12 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-13 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-14 | TC-01, TC-02,TC-03, TC-04, TC-05 |
| TS-15 | TC-01, TC-02,TC-03, TC-04, TC-05 |

## Vertical Traceability Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Level** | **Test Scenario ID** | **Design Artifacts** | **Code Modules** | **Test Cases** |
| Requirement | TS-01, TS-02  TS-03, TS-04  TS-05, TS-06  TS-07, TS-08  TS-09, TS-10  TS-11, TS-12, TS-13, TS-14, TS-15 | High-Level Design |  | TC-01, TC-02,TC-03, TC-04, TC-05 |
| Design |  |  |  |  |
| Code |  |  |  |  |
| Testing |  |  |  |  |

## Quality Matrix:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature** | **Learnability** | **Usability** | **Correctness** | **Efficiency** | **Security** |
|  | Yes | Yes | No (Gives wrong error when password is wrong) | Yes, Time taken to login 18s (threshold <= 20s) | Yes |
|  | Yes | Yes | No (Wrong error message when password and email are wrong, says email is wrong) | Yes (Threshold <= 2.5 min  Actual time to register 1.5 min) | Yes |
|  | Yes | Yes | Yes | Yes (Threshold <= 3 sec  Actual time that shows all the changes <= 2 sec) | Yes |
|  | No | Yes | Yes | Yes, executes under the 15-second threshold | Yes |
|  | No | No | Yes | Yes (Threshold <= 1  Actual time that applies the auto layout <= 1 sec) | Yes |
|  | No | Yes | Yes | Yes (Threshold <=10 sec  Actual time applying navigations <= 4sec  applying navigations and changing navigation type <= 6 sec) | Yes |
|  | Yes | Yes | Yes | Yes, executes under 25 seconds threshold | No (With view access user can copy the content) |
|  | Yes | No | No (we can mention member which is not a collaborator in the file) | Yes (Adds comments under 10 seconds threshold) | Yes |
|  | Yes | Yes | No ( Figma allows to select a laptop device for viewing prototype even when the initial device was a phone.) | No (Threshold for medium file is 20 seconds but it took 22 seconds without any settings change, with settings threshold id 35 seconds and it took 36 seconds) | Yes |
|  | No | No | No (Two teams with the same name can be created and same email can be added multiple times when adding a team member) | No (Threshold <=20 sec  Actual time without inviting team members and choosing a plan >=24 sec  Actual time with inviting team members and choosing a plan >=36 sec) | Yes |
|  | Yes | Yes | No (we can input -1 in font size) | Yes( Threshold = 20 seconds and Time taken = 16 seconds) | Yes |
|  | Yes | Yes | Yes | Yes, A new frame is created under a threshold of <= 11 seconds to complete the task. | Yes |
|  | No | Yes | Yes | Yes (The color of the design element is changed in less than 7 sec threshold) | Yes |
|  | Yes | Yes | Yes | Yes( Threshold = 10 seconds and Time taken = 7 seconds.) | Yes |
|  | Yes | Yes | Yes | No (Export is completed in under 10 seconds threshold for standard assets but for large file it took more than 60 seconds to export which is more than the <= 5 sec threshold, so it fails. | Yes |