TAGORE ENGINEERING COLLEGE

RATHINAMANGALAM, CHENNAI-127.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CCS370 - UI AND UX DESIGN

Name : _____

Reg. No : _____

Branch : _____

Year : _____

Semester : _____

Anna

University : : Chennai

Regulation 2021

TAGORE ENGINEERING COLLEGE

RATHINAMANGALAM, CHENNAI-600127

LABORATORY RECORD

UNIVERSITY REGISTER NUMBER: | |

Certified that this is the bonafide record of work done by Mr./Ms._____of_____ Department in the_____laboratory and submitted for the University Practical Examination conducted on _____ at TAGORE ENGINEERING COLLEGE, CHENNAI-127.

Record Marks: _____

Lab In-charge **Principal** **Head of the Department**

External Examiner **Internal Examiner**

TABLE OF CONTENTS

VISION

To create an environment which is conducive to produce competent Computer Science Engineers through quality education and research-oriented education and equip them for the needs of the industry and society.

MISSION

The Department strives to contribute to the expansion of knowledge in the discipline of Computer Science and Engineering by

- Adopting an efficient teaching learning process in concurrence with increasing industrial demands.
- Ensuring technical proficiency, facilitating to pursue higher studies and carry out Research & Development activities.
- Developing problem solving and analytical skills with deep knowledge in thorough understanding of basic sciences.
- and Computer Science Engineering.
- Infusing managerial and entrepreneurship skills to become ethical, socially responsible, and competitive professionals.

PROGRAM SPECIFIC OBJECTIVES (PSOs)

1. Exhibit design and programming skills to build and automate business solutions using cutting edge technologies
2. Strong theoretical foundation leading to excellence and excitement towards research, to provide elegant solutions to complex problems.
3. Ability to work effectively with various engineering fields as a team to design, build and develop system applications.

PROGRAM OUTCOMES POs

Engineering Graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one‘s own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change

COURSE OUTCOMES

CO1: Build UI for user Applications
CO2: Evaluate UX design of any product or application
CO3: Demonstrate UX Skills in product development
CO4: Implement Sketching principles
CO5: Create Wireframe and Prototype

EX.NO: 01 DESIGNING A RESPONSIVE LAYOUT FOR A SOCIETAL
DATE: APPLICATION

AIM:

To design a responsive layout for a societal application.

ALGORITHM:

1. HTML and CSS Setup:

- Create an HTML5 document with character encoding and viewport settings.
- Use internal CSS to style the layout components.

2. Reset Default Styles:

- Reset margins, padding, and specify a font-family for better control.

3. Style Header, Navigation, Content, and Footer:

- Apply background colors, text colors, and alignment to the header, navigation, and footer.
- Style navigation links as inline elements with spacing.
- Center-align text in header, navigation, and footer.

4. Implement Responsive Design:

- Use a media query for screens up to 768px wide.
- Adjust navigation for mobile display (block-level elements with margin).

5. Add Content:

- Place your application's content within the .container div.

PROGRAM:

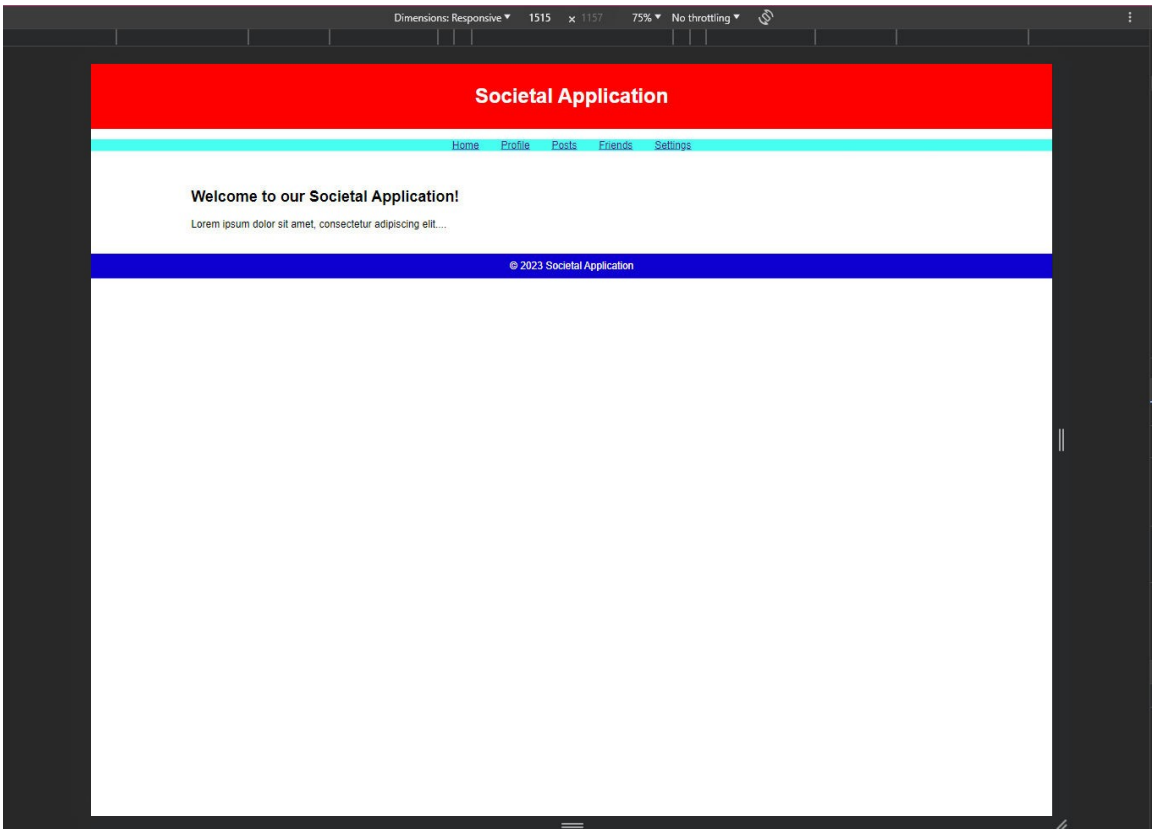
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial scale=1.0">
  <title>Societal Application</title>
  <style>
    /* Reset some default styles */
    body, html {
      margin: 0;
      padding: 0;
      font-family: Arial, sans-serif;
    }

    /* Header styles */
    header {
      background-color: #ff0000;
      color: #fff;
      padding: 10px;
      text-align: center;
    }

    /* Navigation styles */
    nav {
      background-color: #47fff0;
      color: #fff;
      text-align: center;
    }

    nav ul {
      list-style: none;
      padding: 0;
    }

    nav li {
```

```
      display: inline;
      margin: 0 15px;
    }

    /* Main content styles */
    .container {
      max-width: 1200px;
      margin: 0 auto;
      padding: 20px;
    }

    /* Responsive design */
    @media (max-width: 768px) {
      nav {
        display: block;
        text-align: center;
      }

      nav li {
        display: block;
        margin: 10px 0;
      }
    }

    /* Footer styles */
    footer {
      background-color: #0e00d1;
      color: #fff;
      text-align: center;
      padding: 10px;
    }
  </style>
</head>
<body>
  <header>
    <h1>Societal Application</h1>
  </header>
  <nav>
    <ul>
```

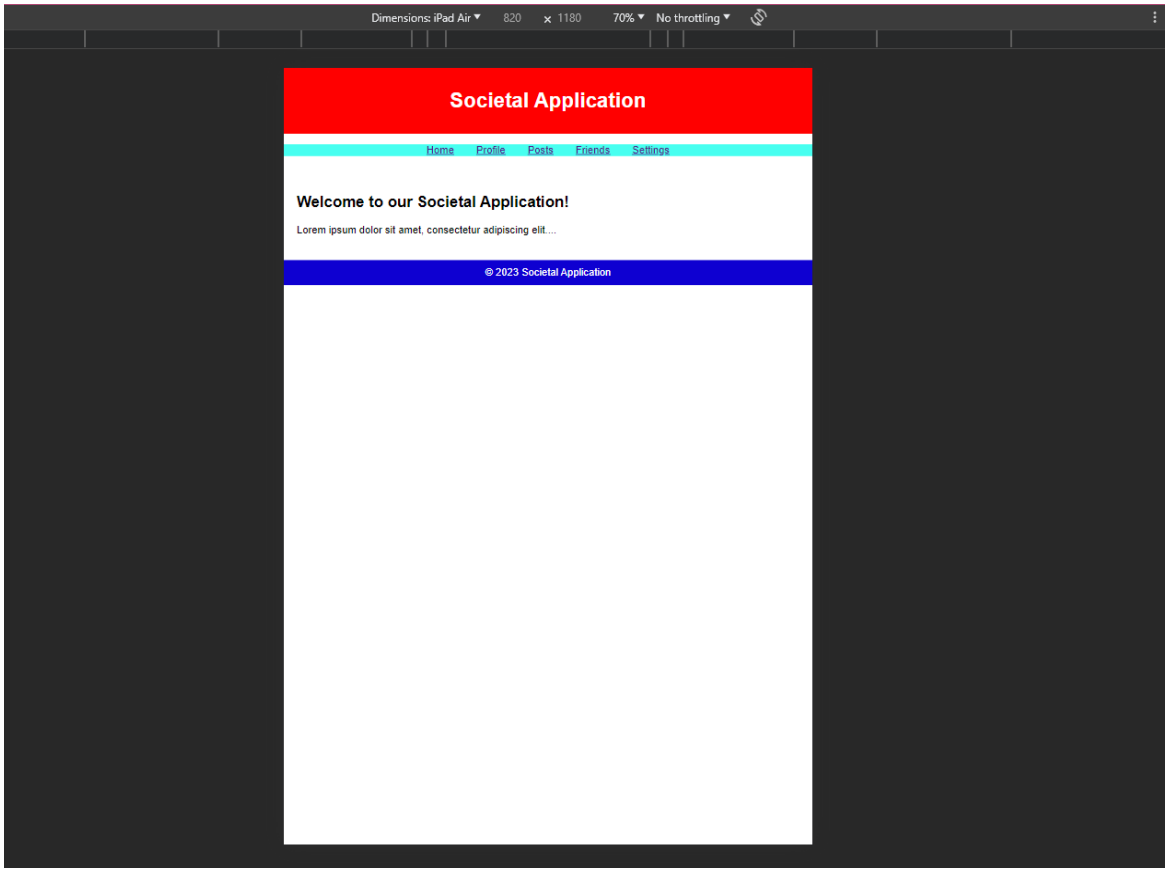
```
        <li><a href="#">Home</a></li>
        <li><a href="#">Profile</a></li>
        <li><a href="#">Posts</a></li>
        <li><a href="#">Friends</a></li>
        <li><a href="#">Settings</a></li>
      </ul>
    </nav>
    <div class="container">
      <!-- Your content goes here -->
      <h2>Welcome to our Societal Application! </h2>
      <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit....</p>
    </div>
    <footer>
      &copy; 2023 Societal Application
    </footer>
  </body>
</html>
```

OUTPUT:

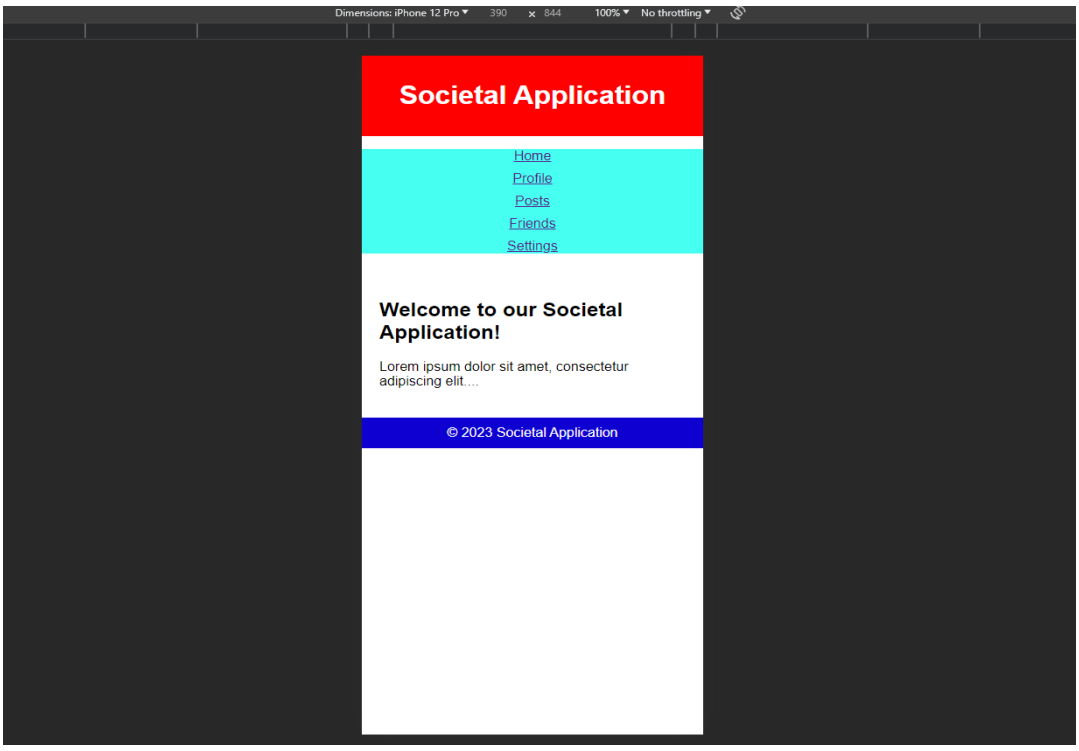
DESKTOP VIEW:



TABLET VIEW:



MOBILE VIEW:



RESULT:

Thus, the responsive layout for societal application has been designed successfully.

EXP.NO: 02 **EXPLORING VARIOUS UI INTERACTION**

DATE: **PATTERNS**

AIM:

To explore various UI interaction patterns.

ALGORITHM/PROCEDURE:

1. **Identify UI Elements:**

Recognize all UI components such as buttons, input fields, and dropdown menus.

2. **Understand User Flows:**

Grasp how users navigate, interact, and input information.

3. **Interact with UI Elements:**

Click buttons, input text, and select options to observe system responses.

4. **Test Input Validation:**

Validate input fields by entering both valid and invalid data.

5. **Explore Error Handling:**

Trigger errors to see how the system responds and displays error messages.

6. **Navigate Through Flows:**

Test various actions like adding, editing, and deleting items to ensure smooth functionality.

7. **Verify Responsiveness:**

Test UI on different devices and screen sizes to ensure adaptability.

8. **Test Accessibility:**

Check if the UI is accessible to users with disabilities, including keyboard navigation and screen reader compatibility.

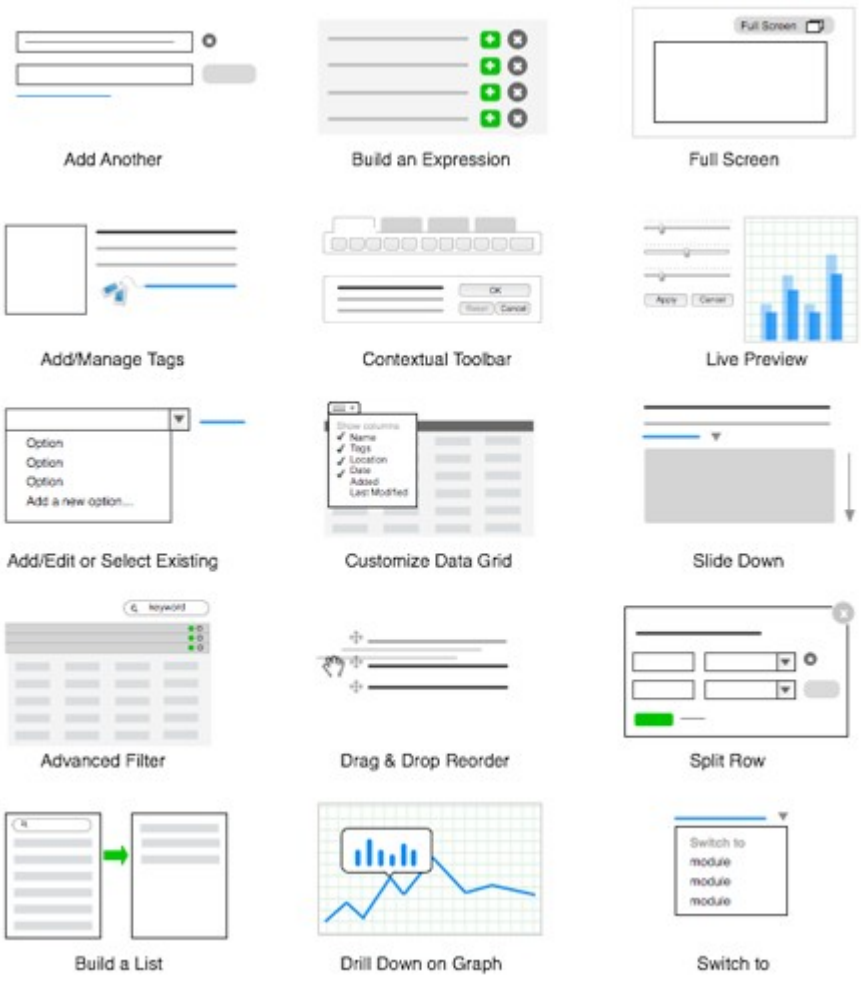
9. **Evaluate User Feedback:**

Consider user feedback and analytics to identify areas for improvement.

10. **Document Findings:**

Record observations, issues, and suggestions for UI enhancement.

OUTPUT:



RESULT:

Thus, variuos UI interactive designs has been explored successfully.

EXP.NO: 03 **DEVELOPING AN INTERFACE WITH PROPER UI**

DATE: **STYLE GUIDES**

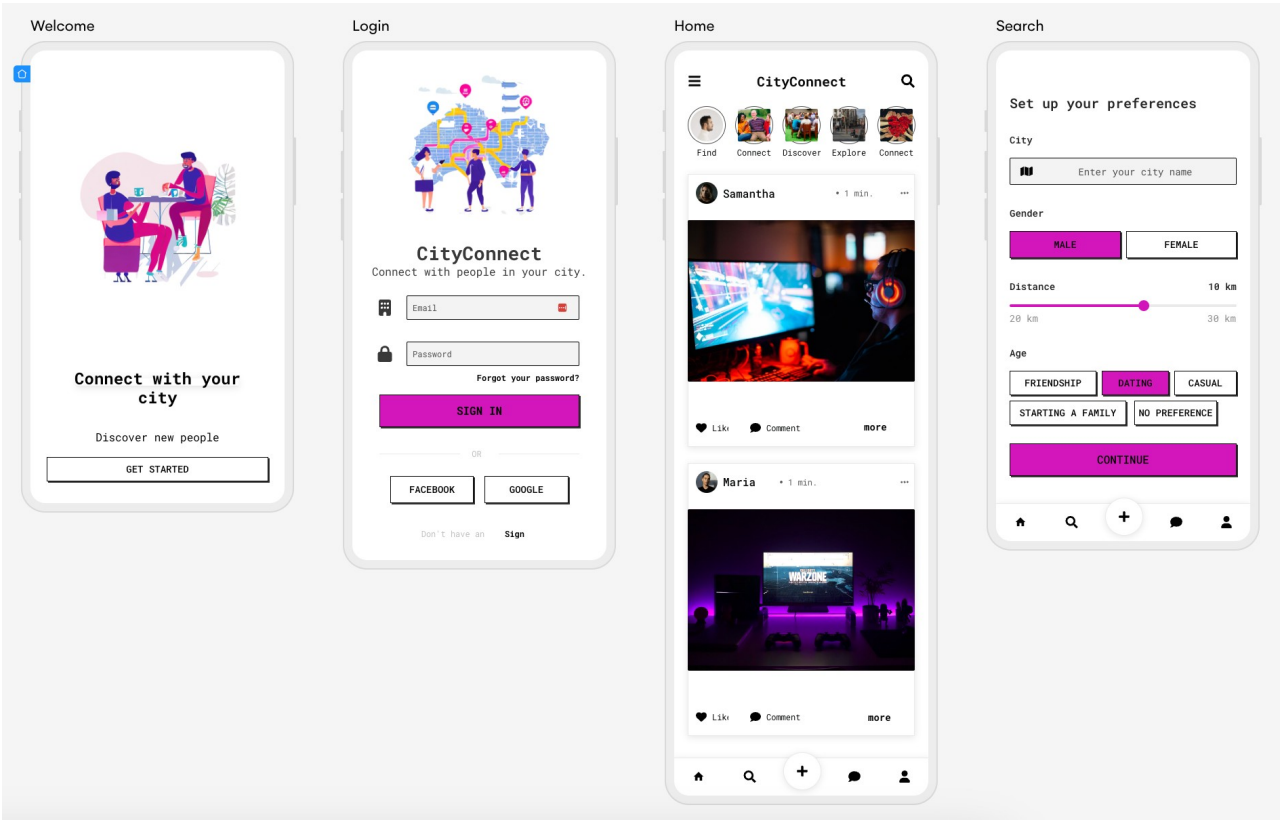
AIM:

To develop an interface with proper UI style guides.

ALGORITHM / PROCEDURE:

1. **Define Goals and Audience:** Understand the project's purpose and target users.
2. **Research and Inspiration:** Gather industry insights and design inspiration.
3. **Create a Figma Project:** Start a new Figma project.
4. **Workspace Setup**: Organize Figma files and create sections for style guides and components.
5. **Brand Guidelines**: Set color, typography, and brand-related guidelines.
6. **UI Components**: Create component libraries for buttons, forms, icons, and navigation.
7. **Typography:** Define font styles, sizes, and spacing.
8. **Iconography:** Design and organize icons as components.
9. **Color System**: Document primary, secondary, background, and text colors.
10. **Grids and Layouts:** Establish grid systems for different devices.
11. **Accessibility Guidelines:** Ensure WCAG-compliant contrast and text sizing.
12. **UI Elements:** Document usage guidelines for various UI elements.
13. **Wireframes:** Create wireframes and user flows for screen layout and interaction.
14. **Visual Design:** Develop screens based on wireframes using components and styles.
15. **Interactive Prototyping:** Build interactive prototypes for user testing and feedback.

OUTPUT:



RESULT:

Thus, an interface with proper UI guidelines has been developed successsfully.

EX.NO:04 DEVELOPING WIREFLOW DIAGRAM FOR APPLICATION
DATE: USING OPEN-SOURCE SOFTWARE

AIM:

To develop Wireflow diagram for application using open-source software

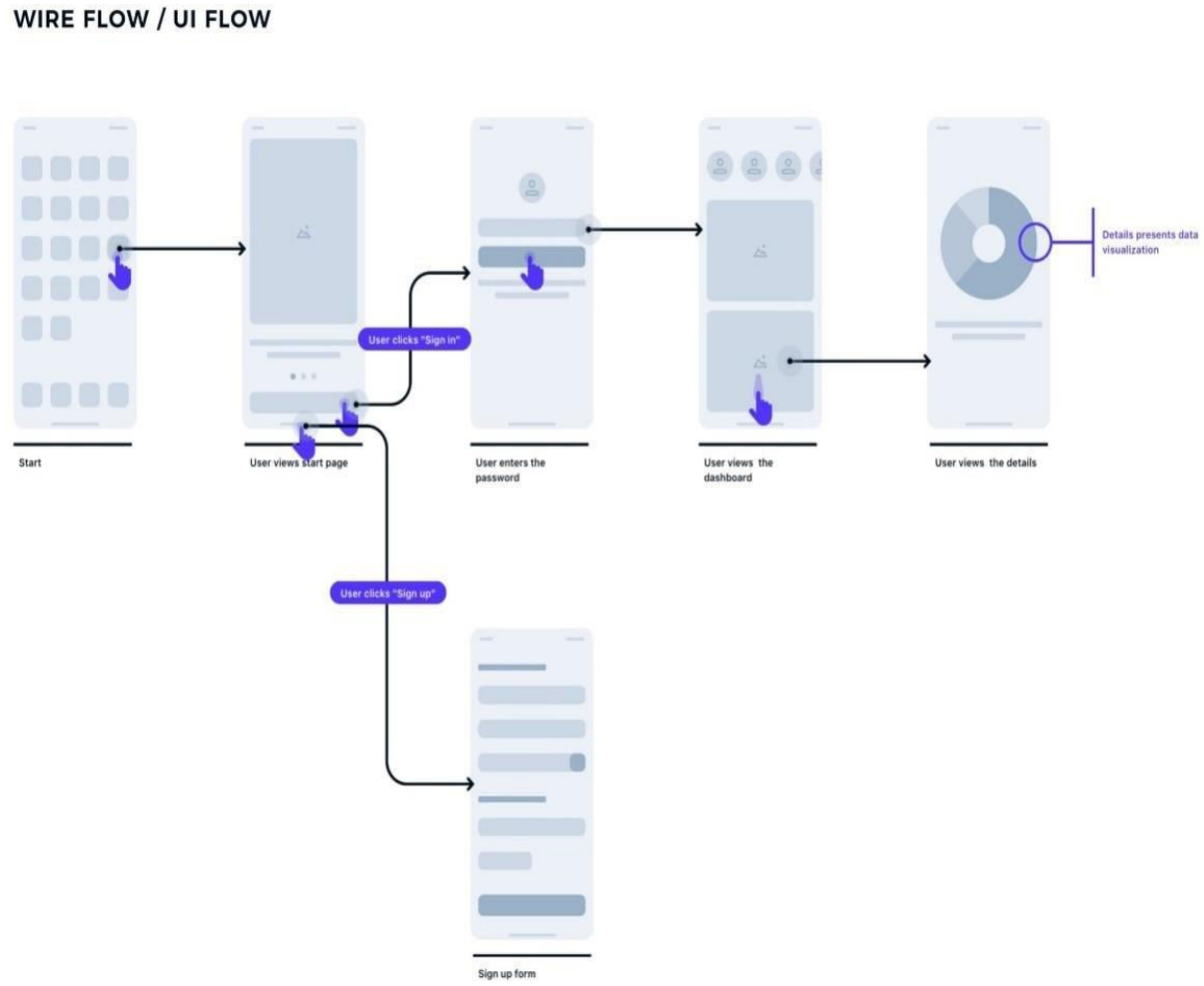
ALGORITHM/PROCEDURE:

1. **Define Purpose and Goals:** Determine the diagram's purpose and goals, focusing on user flows, navigation, and interactions.
2. **Identify User Personas:** If applicable, specify user personas for a user-centric approach.
3. **Gather Requirements:** Collect project information, including existing designs and functionality requirements.
4. **Select Software:** Choose open-source design software, such as Figma, for wireflow creation.
5. **Create a Project:** Begin a new project in your chosen software and set up the canvas to match your project's needs.
6. **Wireframe Screens:** Develop wireframes for each application screen, focusing on visual structure.
7. **Define Interactions:** Add interaction notes or links to illustrate navigation and user interactions.
8. **Create User Flows:** Connect wireframes to illustrate user journeys, navigation paths, and interactions.
9. **Add Annotations:** Include descriptions to clarify elements and interactions in each wireframe.
   .
10. **Collaborate and Share:** Utilize collaboration features to gather feedback from team members and stakeholders.
11. **Iterate and Refine:** Revise the wireflow diagram based on feedback, ensuring alignment with project goals.
12. **Finalize and Export:** Clean up the wireflow diagram and export it to a suitable format for sharing and documentation.

13. **Document the Wireflow:**Create a reference guide to explain the wireflow's purpose and key notes for stakeholders and developers.

14. **Maintain Consistency:**Keep the wireflow diagram in sync with the application's actual design, updating it as needed.

OUTPUT:



RESULT:

Thus, the wire flow diagram for the application has been designed successfully.

EX.NO:05 EXPLORING VARIOUS OPEN-SOURCE COLLABORATIVE
DATE: INTERFACE PLATFORM

AIM:

To Explore Various Open-Source Collaborative Interface Platform.

ALGORITHM /PROCEDURE:

1. **Needs Assessment**:
   Define your team's requirements and goals.

2. **Research Platforms**:
   Identify open-source collaborative tools.

3. **Feature Comparison**:
   Assess features and compatibility.

4. **Community Support**:
   Check for active communities.

5. **Installation**:
   Set up the chosen platform.

6. **User Training**:
   Train and encourage team adoption.

7. **Security**:
   Ensure data security and privacy.

8. **Integration**:
   Check for compatibility with existing tools.

9. **Testing and Feedback**:
   Pilot testing and gather feedback.

10. **Scalability**:
    Ensure the platform can grow with your team.

11. **Documentation**:
    Create user resources and guides.

12.Community Engagement:

Utilize community support.

13.Maintenance:

Keep the platform updated and secure.

14.Feedback Loop:

Encourage ongoing user feedback.

15.Legal Compliance:

Ensure adherence to licensing and Legal requirements.

16.Backup and Recovery:

Implement data safety measures.

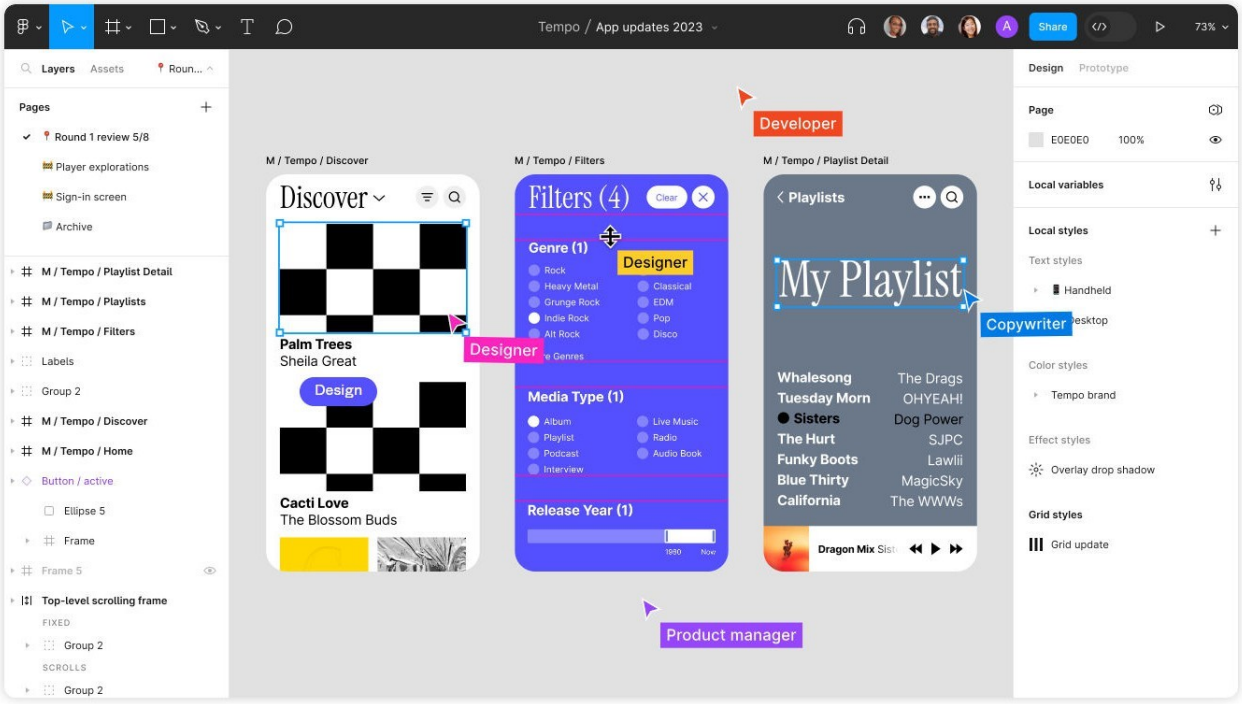
17.Periodic Evaluation:

Continuously assess platform suitability.

18.Migration Plan:

Prepare for possible platform changes.

Exploring different open-source collaborative interface Platform:

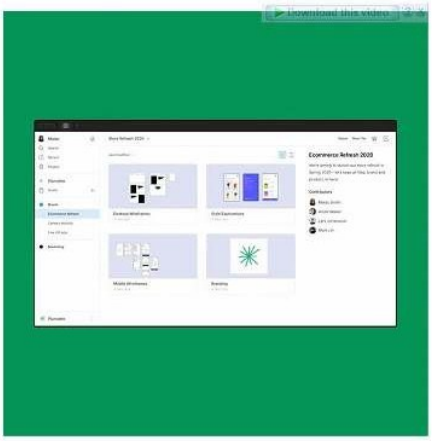


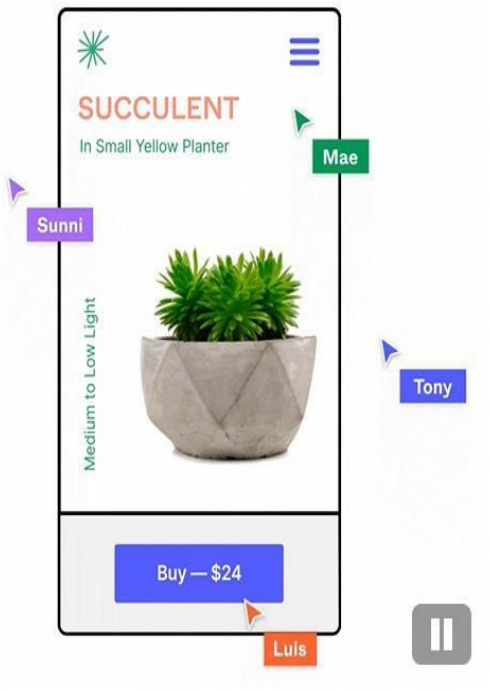
Discover the most important design work:

Discover the most important design work

Figma is where design work gets the visibility it needs across teams.

- **Get the latest updates:** Project Pages serve as your team's hub. Pin files for easy access. Add notes for context.
- **Centered around people:** Easily see contributors and projects on Team and Organization Pages.
- **Discover work by creator:** Browse the latest files each user has contributed to on their profile.





Jump into a file as a team. Bring people on your journey.

Figma makes the design process transparent, which means everyone gets aligned fast—and stays aligned.

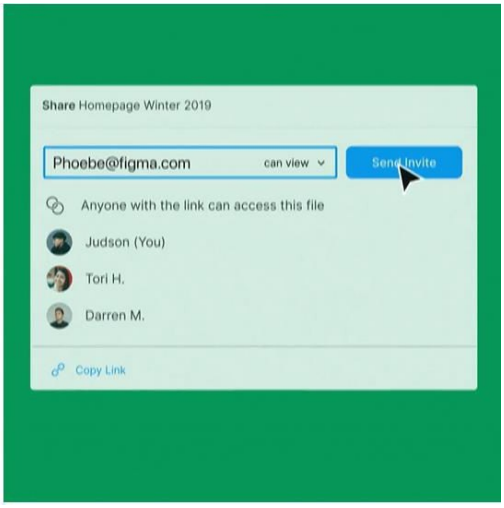
- **Edit together in real-time:** Co-design alongside your co-workers. Run global design critiques.
- **Follow along with Observation Mode:** Shadow the presenter's every move and never get lost.
- **Worry-free version history:** See who changed what and go back in time as needed.

Invite folks into your design process:

Invite folks into your design process

Enable others to add copy, grab specs, and give you feedback, so you ship better work.

- **Invite the people you want:** Figma puts you in the driver seat. Control who can view, add comments, and edit.
- **Share links to files:** Send a link to your Figma file. View from anywhere with the Interwebs.
- **Share or embed prototypes:** Send a link to your prototype or embed it in your preferred tool.



RESULT:

Thus various open-source collaborative interfaces Platform has been explored successfully.

EX.NO:06 **HANDS ON DESIGN THINKING PROCESS FOR A NEW**
DATE: **PRODUCT**

AIM:

To apply the design thinking process for a new product.

ALGORITHM /PROCEDURE:

Empathize:

Begin by conducting user research and interviews to gain insights into potential user needs and pain points related to smartphone usage.

Define:

Analyze the gathered information to define a clear and specific problem statement. For example, "Users need a more efficient way to track their daily fitness activities."

Ideate:

Organize brainstorming sessions with a diverse team to generate a wide range of creative solutions. Encourage free thinking and open collaboration**.**

Prototype:

Create a low-fidelity prototype of the smartphone app. This can be a paper sketch or a digital wireframe that represents the app's basic functionality.

Test:

Conduct user testing sessions with a small group of potential users. Observe how they interact with the prototype and gather feedback. Iterate: Based on user feedback, refine the prototype and make necessary improvements to address user concerns or suggestions**.**

Prototype (Again):

Create a more advanced prototype, closer to the final product. It should incorporate the changes and improvements identified during the initial testing phase.

Test (Again):

Conduct another round of user testing, this time with a larger group of users. Gather data on usability, functionality, and overall user experience.

Refine:

Analyze the results of the second testing phase and make further efinements to the app design and functionality.

Implement:

Develop the final version of the smartphone app, incorporating all the changes and improvements identified during the design thinking process.

Test (Final Testing):

Conduct thorough testing of the fully developed app to ensure it's bug-free and ready for launch.

Launch:

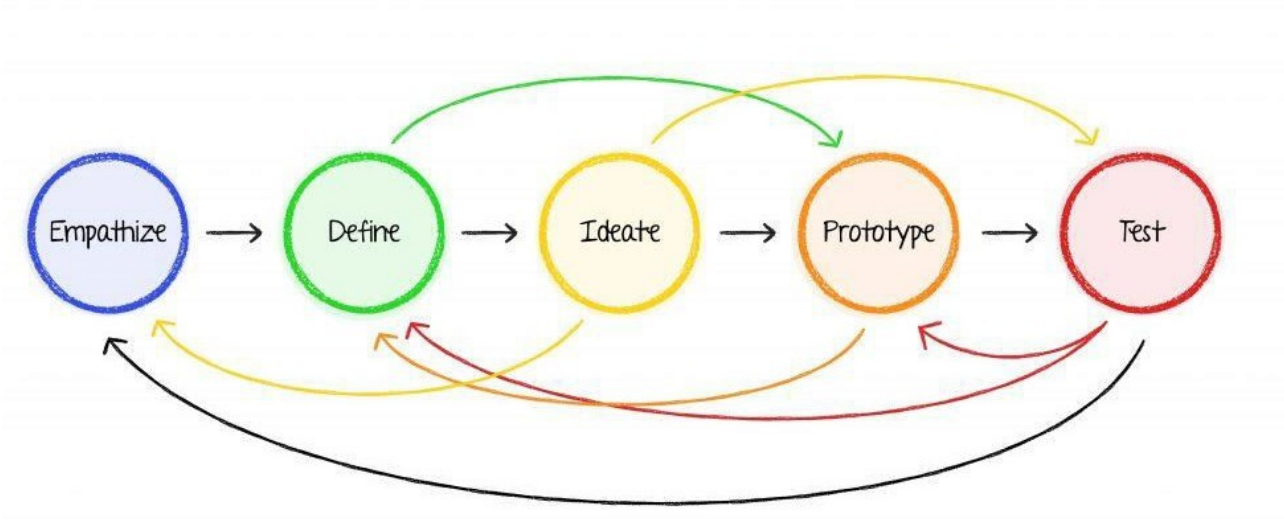
Release the app to the target market, accompanied by marketing and promotion efforts.

Example:

Let's say the team is designing a fitness tracking app. During the "Empathize" phase, they conduct interviews and surveys with potentialusers, discovering that users find it challenging to keep track of their dailyphysical activities. In the "Define" phase, they define the problem as"Users need a more efficient way to track their daily fitness activities."In the "Ideate" phase, the team generates multiple ideas, including features like GPS tracking, step counting, and customizable fitness goals.They create a low-fidelity prototype that represents these features. In the first round of user testing, they observe that users have difficulty navigating the app.

After gathering feedback and identifying navigation issues, the team iterates by redesigning the user interface to improve user experience. They create an advanced prototype with a more intuitive interface and test it with a larger group of users in the second round of testing.

OUTPUT:



RESULT:

Thus the design thinking process for new product has been studied.

EX.NO: 07 **BRAINSTORMING FEATURE FOR PROPOSED**
DATE: **PRODUCT**

AIM:

The aim of this process is to generate brainstorming feature of proposed product.

ALGORITHM/PROCEDURE:

Begin by thoroughly understanding the proposed product's concept, its target audience, and its unique selling points.

Gather a Diverse Team:

Assemble a cross-functional team with members from various departments (e.g., product development, marketing, design) to bring different perspectives to the brainstorming session.

Set Clear Objectives:

Define clear objectives for the brainstorming session. What problems should the new features solve? What goals should they achieve?

Warm-Up and Icebreaker:

Start the session with a warm-up or icebreaker activity to encourage creative thinking and open communication within the team.

Idea Generation:

Allow team members to freely brainstorm feature ideas. Encourage a "no idea is a bad idea" mindset. Use techniques like mind mapping, brainstorming software, or post-it notes on a whiteboard to record ideas.

Categorize and Prioritize:

Group similar ideas together, and prioritize them based on factors like feasibility, potential impact, and alignment with the product concept.

SWOT Analysis:

Conduct a SWOT (Strengths, Weaknesses, Opportunities, Threats) analysis for each feature idea to evaluate its potential in the market.

Feasibility Assessment:

Assess the technical, financial, and resource feasibility of implementing the proposed features.

Market Research:

Conduct market research to identify user preferences and gather insights that can inform feature development.

Prototype and User Testing:

Create prototypes or mock ups of the proposed features and conduct user testing to gather feedback and refine the ideas.

Cost-Benefit Analysis:

Evaluate the expected cost of development against the projected benefits, such as increased user engagement, retention, or revenue.

Risk Assessment:

Identify potential risks associated with each feature and develop mitigation strategies.

Finalize Feature Set:

Based on the assessment, finalize the set of features to be included in the product. Ensure they align with the product's vision and goals.

Documentation:

Document the chosen features, their objectives, and the rationale behind their selection. This document will guide the development team.

Iterate as Needed:

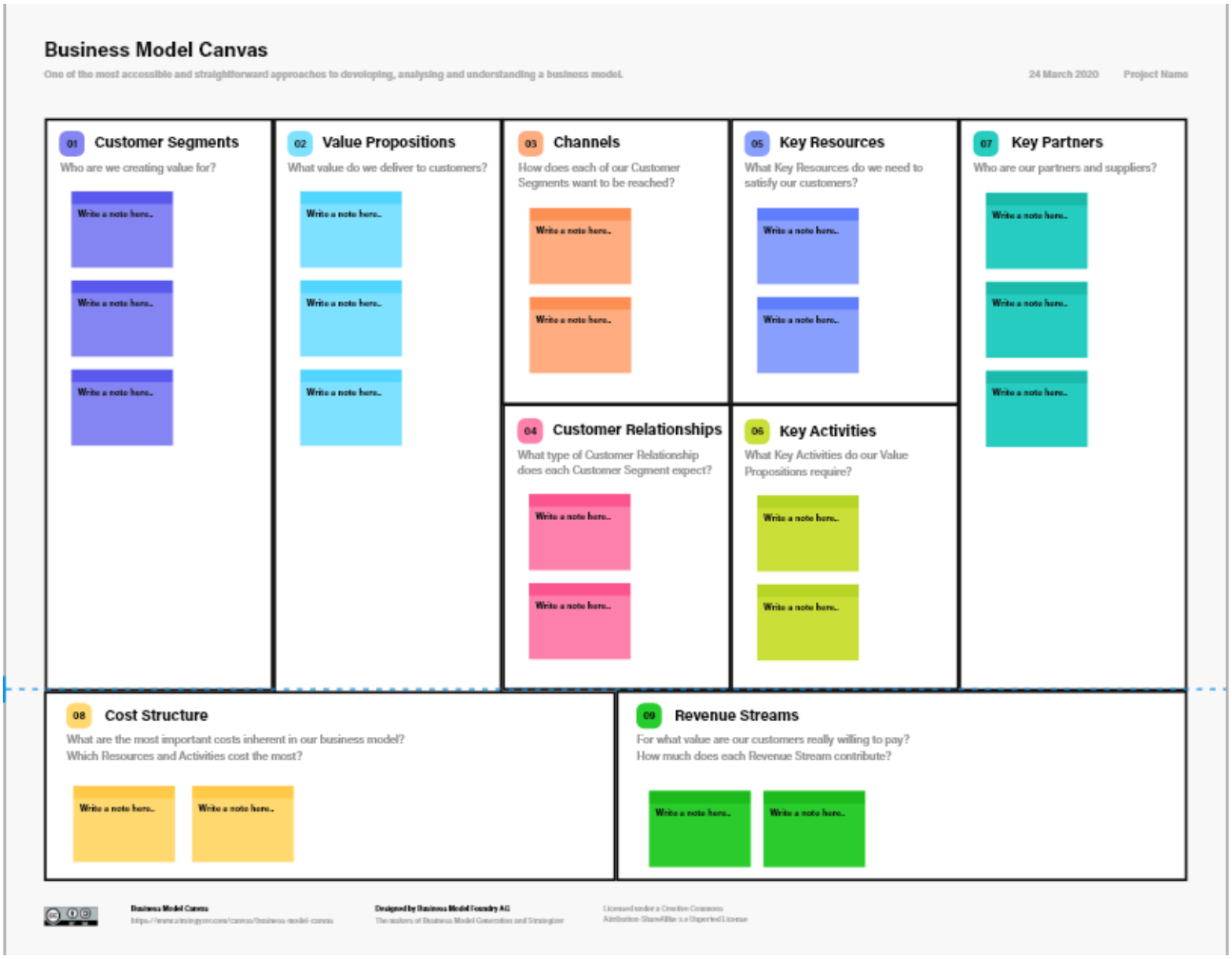
Keep an open line of communication for ongoing feature refinements and iterations, especially as more data and insights become available.

Example:

Suppose a software company is developing a new mobile messaging app. During the brainstorming session, the team generates a wide range of feature ideas, including:

- End-to-End Encryption: To ensure user privacy and data security.
- Message Scheduling: Allowing users to schedule messages to be sent at a specific time.
- Reaction Emojis: A feature that lets users react to messages with emojis for more expressive communication.
- Dark Mode: A night-friendly theme for the app.
- Polls and Surveys: Integration of polls and surveys within the chat for easy decision-making.
- Auto-Translate: Real-time language translation for international communication.

OUTPUT:



RESULT:

Thus brainstorming feature for proposed product has been applied and executed successfully.

EX.NO: 08 **DEFINING THE LOOK AND FEEL OF THE NEW**
DATE: **PROJECT**

AIM:

The aim is to establish the visual design and user experience for a new project, ensuring it aligns with the project's goals and provides an appealing, intuitive, and cohesive interface for users.

ALGORITHM/PROCEDURE:

Project Goal Assessment:

Understand the project's objectives, target audience, and scope. This sets the foundation for design decisions.

Research and Inspiration:

Gather inspiration from existing designs and industry trends. Create mood boards or design boards
to collect visual references.

Define Design Principles:

Determine the core design principles that will guide the project's look and feel. These could include simplicity, consistency, accessibility, and branding.

Wire framing and Prototyping:

Create wire frames or low-fidelity prototypes to plan the layout and structure of the user interface.
Use tools like Figma, Sketch, or Adobe XD for digital projects.

Visual Design:

Develop a color palette, typography choices, and graphic elements (icons, images, logos) that reflect the project's identity. Create high-fidelity designs using design software.

User Interaction Design:

Define user interactions and behaviors, including animations, transitions, and micro-interactions. Ensure a smooth and intuitive user experience.

Responsive Design:

Adapt the design to various screen sizes and devices, focusing on mobile responsiveness.

Accessibility and Usability Testing:

Evaluate the design for accessibility, ensuring it's usable by individuals with disabilities. Conduct usability testing with potential users to gather feedback.

Iteration and Feedback:

Refine the design based on feedback from users and stakeholders.

Documentation:

Create design documentation that includes guidelines for developers to implement the design.

Development Integration:

Collaborate with developers to ensure the design is implemented accurately in the project.

User Testing:

Conduct user testing with real users to verify the design's effectiveness.

Finalization:

Make necessary adjustments and finalize the design elements for deployment.

Example:

Imagine you're tasked with defining the look and feel of a new e- commerce website. You go
through the following steps:

Assess the project's goal: To create a user-friendly, visually appealing online store. Research e-commerce design trends and competitor websites.

Define design principles:

Simplicity, trustworthiness, and a focus on the product. Create wire frames for key pages

like the homepage, product listing, and product detail pages.

Develop a color palette featuring subtle, trust-inducing colors.

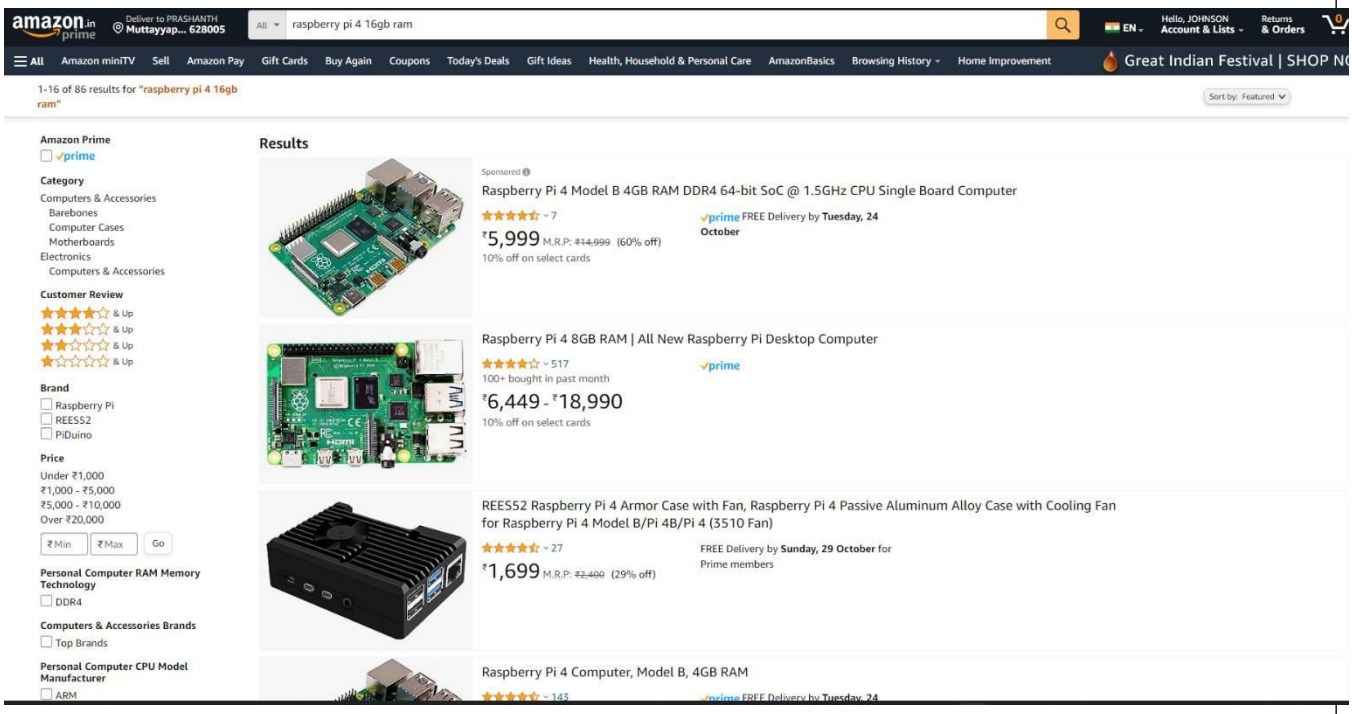
Choose a clear and readable typography for product descriptions and headings. Plan user interactions like product image zoom and streamline checkout process.

Ensure that the design is responsive and mobile-friendly.Conduct accessibility testing to meet WCAG standards.

Iterate on the design based on usability testing feedback. Document the design guidelines for the development team. Collaborate closely with developers to bring the design to life.

Conduct user testing to validate the design's effectiveness. Make final adjustments and prepare for
the website's launch.

OUTPUT:



RESULT:

Thus the Look and Feel of the new Project has been defined successfully.

EX.NO: 09 CREATE A SAMPLE PATTERN LIBRARY FOR THE
DATE: PRODUCT

AIM:

The aim of this project is to create a sample Pattern Library for a product .

ALGORITHM/PROCEDURE:

Define the Scope:

Identify the product for which you are creating the Pattern Library. Understand the product's target audience, brand identity, and design goals.

Gather Inspiration:

Research existing UI designs, competitor products, and design trends to gather inspiration. This will help you create a mood board that captures the desired aesthetics.

Create Mood Boards:

Using design software or tools like Adobe XD or Figma, create mood boards that reflect the desired visual style, mood, and tone of the product. Include images, colors, typography samples,and other visual elements that represent the brand and design principles.

Choose Fonts:

Select fonts that align with the product's brand and design goals. Consider readability, hierarchy, and scalability. Choose a primary font for headings and body text, and possibly secondary fonts for accents.

Define Color Palette:

Create a color palette that includes primary, secondary, and accent colors. Ensure the colors are harmonious and comply with accessibility guidelines. Provide color codes (hex, RGB, or HSL) for each color.

Document UI Principles:

Document the UI design principles that underpin the Pattern Library. This may include guidelines on spacing, layout, typography, and interaction behaviors.

Organize and Label:

Organize the Pattern Library in a clear and accessible manner. Label each component, color, and typography choice, and provide context or usage guidelines.

Mock up Example Screens:

Create example screens or wire frames using the fonts, colors, and components defined in the Pattern Library. These screens should showcase the design principles in action.

Test and Refine:

Share the Pattern Library with designers, stakeholders, or users for feedback. Make refinements based on the feedback to ensure it aligns with the project's goals.

Maintenance and Version Control:

As the product evolves, maintain and update the Pattern Library to reflect any design changes. User version control systems to keep track of changes and updates.

Example:

Let's say you're creating a Pattern Library for a mobile app focused on wellness and meditation. The aim is to create a soothing and user-friendly design that promotes relaxation. Here's a simplified example:

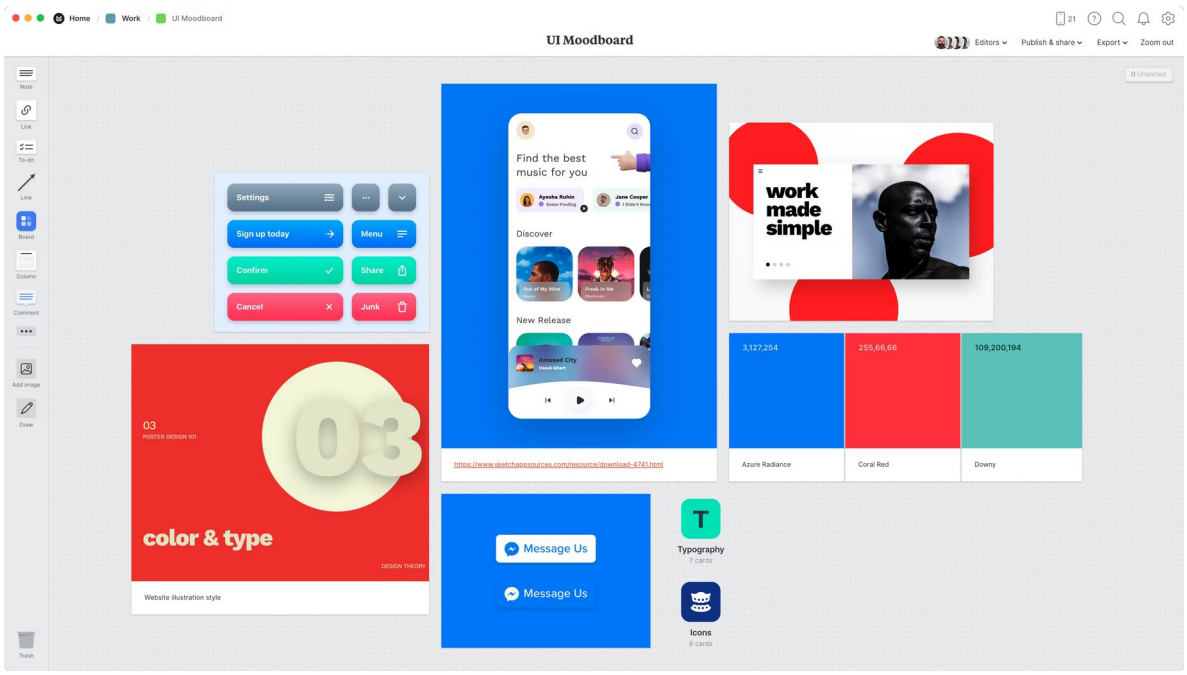
Mood Board: Create a mood board with images of serene landscapes, calming color schemes (e.g., blues and greens), and sample UI elements that convey a sense of tranquility.

Fonts: Choose "Poppins" as the primary font for headings and "Roboto" for body text. These fonts
are clean, readable, and complement the app's aesthetic.

Color Palette: Define a color palette that includes calming colors like "#3CBDBE" for primary elements, "#70C1B3" for secondary elements, and "#EFEFF0" for background. Ensure that
these colors meet accessibility standards.

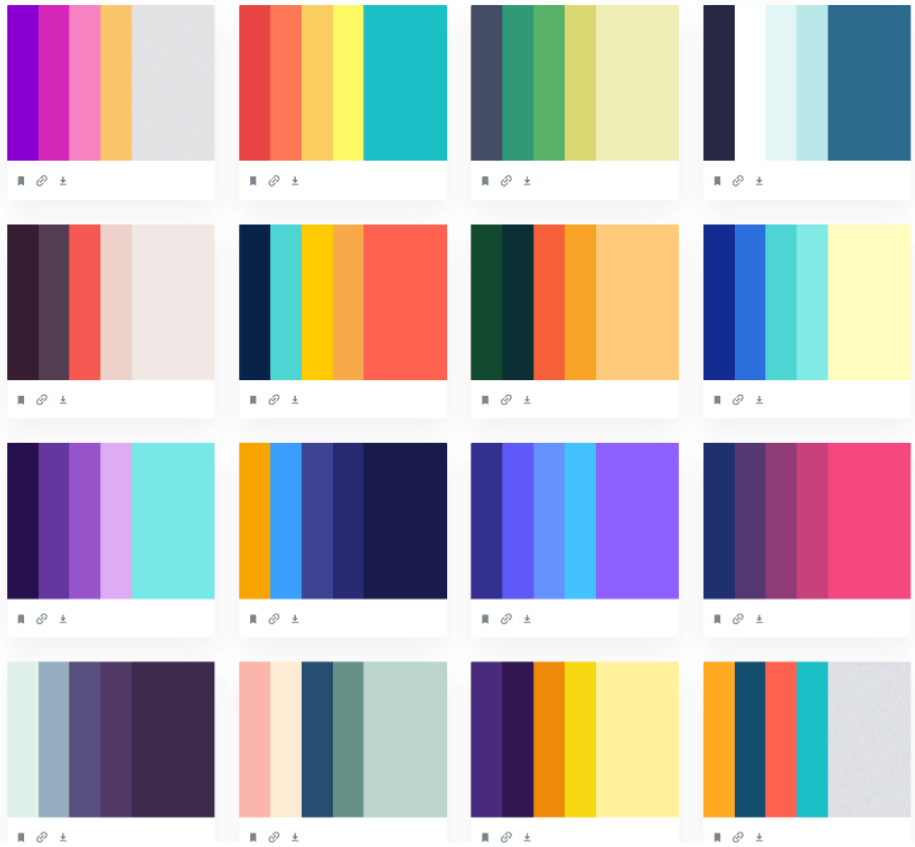
OUTPUT:

MOOD BOARD:



FONTS:

COLOR PALETTE:



RESULT:

Thus a sample pattern library for a product was created successfully.

EX.NO:10 IDENTIFY A CUSTOMER PROBLEM TO SOLVE

DATE:

AIM:

To identify a customer problem to solve it effectively.

ALGORITHM /PROCEDURE:

PROBLEM IDENTIFICATION METRICS:

One of your key metrics is the high rate of customers abandoning their shopping carts before completing the purchase.

DATA ANALYSIS:

Analyze the data and find that a significant number of customers abandon their carts at the payment stage.

PRIORITIZATION:

Prioritize the payment abandonment issue because it directly affects revenue.

ROOT CAUSE ANALYSIS:

Discover that complex payment options and a lack of guest checkout are causing the problem.

SOLUTION IDEATION:

Brainstorm solutions, including simplifying payment options and adding a guest checkout feature.

EXPERIMENT DESIGN:

Design A/B tests to measure the impact of these changes on cart abandonment rates.

IMPLEMENTATION:

Implement the proposed solutions on a small scale for testing.

DATA COLLECTION AND POST-IMPLEMENTATION:

Collect data on cart abandonment rates after the changes are implemented.

ANALYSIS AND VALIDATION:

Analyze the post-implementation data and find that cart abandonment rates have significantly decreased.

FEEDBACK AND ITERATION:

Collect feedback from customers who completed purchases and continue to iterate on the checkout process to further enhance the customer experience and address any remaining issues.

EXAMPLE:

Let's say you're running an e-commerce platform and want to identify a customer problem
related to checkout abandonment. Here's how you could apply the algorithm/procedure:

CUSTOMER SEGMENTATION:

Segment customers based on their demographics and purchase history.

DATA COLLECTION:

Gather data through customer surveys, web analytics, and feedback forms.

RESULT:

Thus a customer problem was identified and understood successfully.

EX.NO:11 CONDUCT END TO END USER RESEARCH
DATE:

AIM:

The aim of this experiment is to conduct end-to-end user research and design process to develop a user-centered solution for a specific problem.

ALGORITHM/PROCEDURE:

DEFINE THE PROBLEM:

Clearly define the problem or challenge you want to address through this user-centered design process

.

USER RESEARCH:

Conduct user interviews, surveys, or observations to gather insights and data about the target users. Analyze the collected data to identify user needs, pain points, and behaviors.

CREATE PERSONAS:

Based on the research findings, create user personas. Personas are fictional representations of your typical users, including their goals, needs, and characteristics.

IDEATION:

Brainstorm creative ideas to address the identified problems and fulfill user needs.Develop user stories and scenarios to articulate how users will interact with the proposed solution.

USER STORIES:

Create user stories using the "As a [type of user], I want [an action] so that [benefit/value]" format. User stories should capture specific user tasks and their motivations.

SCENARIOS:

Develop detailed narratives (scenarios) that illustrate how users will use the solution to achieve their goals. Scenarios should provide context, user actions, and expected outcomes.

FLOW DIAGRAMS:

Create flow diagrams to visualize the user's journey through the solution. Use symbols and arrows to represent user actions, system responses, and transitions between different screens or stages.

FLOW MAPPING:

Develop flow maps to provide a holistic view of the user experience. Connect user stories and scenarios to specific steps in the flow, highlighting decision points and potential pain points.

ITERATE AND TEST:

Review and refine your user stories, scenarios, flow diagrams, and flow maps based on feedback
from stakeholders or potential users. Conduct usability testing to validate the proposed user experience.

IMPLEMENTATION:

Work with developers and designers to implement the user-centered solution based on the finalized
flow and design.

EVALUATION:

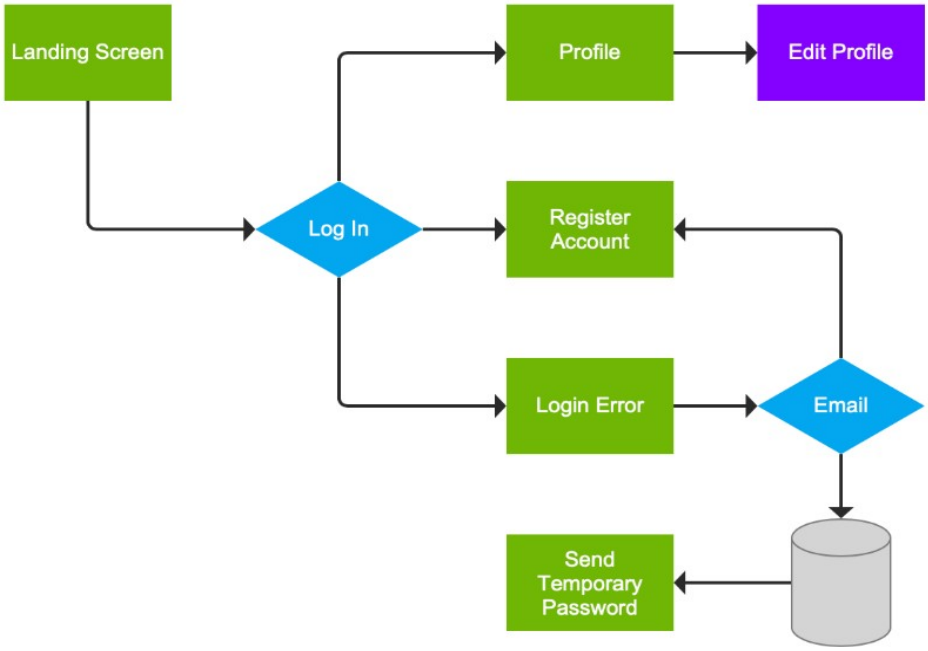
After the solution is implemented, evaluate its effectiveness by gathering user feedback and monitoring key performance metrics.

ITERATE AND IMPROVE:

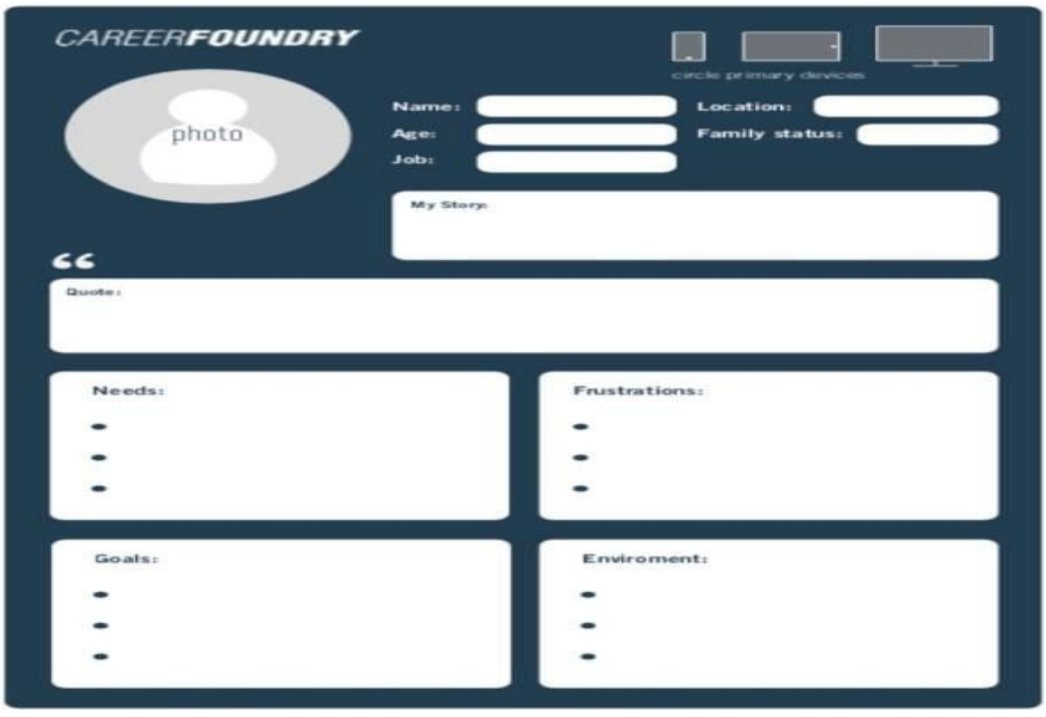
Continuously iterate on the design and user experience based on user feedback and changing needs.

OUTPUT DESIGN:

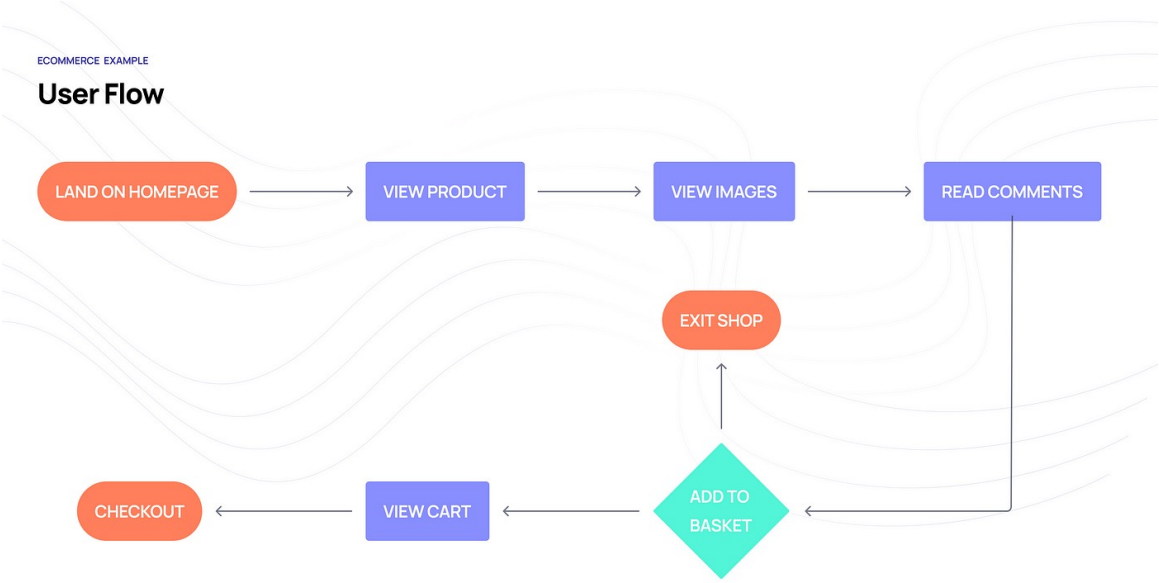
FLOW DIAGRAMS:



USER PERSONAS:



FLOW MAPPING:



RESULT:

Thus , the end-to-end user research has been conducted successfully.

EX.NO:12 SKETCH,DESIGN WITH POPULAR TOOL AND BUILD A
DATE: PROTOTYPE AND PERFORM USABILITY TESTING AND IDENTIFY IMPROVEMENTS

AIM:

The aim of this experiment is to design a user-friendly mobile app.

ALGORITHM / PROCEDURE:

DEFINE OBJECTIVES AND USER PERSONA:

Define the objectives of the task management app. Create a user persona to represent the target audience.

SKETCH AND WIRE FRAMES:

Start with sketching the basic layout and functionality of the app on paper or digitally. Create low fidelity wire frames to visualize the app's structure and layout.

DESIGN WITH A POPULAR TOOL:

Choose a popular design tool such as Adobe XD, Sketch, Figma, or InVision. Create high fidelity designs with attention to visual elements, typography, and color schemes. Implement the user
interface (UI) based on best practices and your user persona's preferences.

PROTOTYPE CREATION:

Use the design tool to create interactive prototypes with clickable elements and transitions. Ensure
that the prototype represents the app's core functionalities.

RECRUIT PARTICIPANTS FOR USABILITY TESTINGS:

Identify potential users or participants who match the user persona. Prepare a usability testing plan, including tasks to be performed within the prototype.

USABILITY TESTING:

Conduct usability testing sessions with participants. The participants are asked to perform specific tasks within the prototype. Observe and record their interactions and gather feedback on their experience.

ANALYZE AND IDENTIFY IMPROVEMENTS:

Analyze the usability testing data to identify pain points and areas of improvement. Look for common patterns and issues encountered by users.

ITERATE ON THE DESIGN:

Implement the necessary design improvements based on the feedback received. Make changes to the prototype to address identified issues.

SECOND ROUND OF USABILITY TESTING:

Conduct a second round of usability testing with new or the same participants to evaluate the impact of the design improvements.

FINALIZE THE PROTOTYPE:

Make any final adjustments based on the results of the second usability testing round.

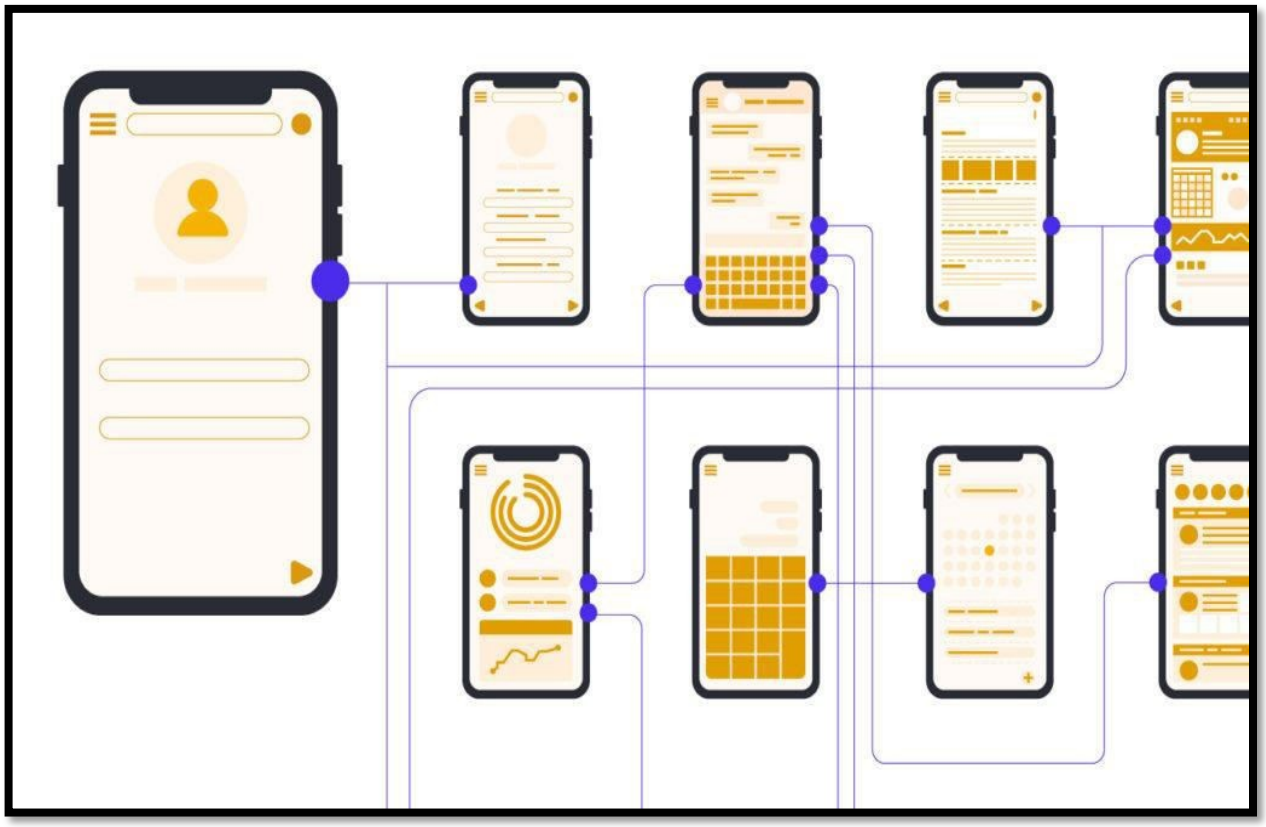
DOCUMENT FINDINGS AND RECOMMENDATIONS:

Document the findings from both rounds of usability testing. Provide clear recommendations for further improvements or development.

CONCLUSION:

Conclude the experiment by summarizing the improvements made to the prototype and how they enhance the user experience.

DESIGN:



RESULT:

Thus, Sketching, building a prototype, performing usability testing and identifying improvements has been executed successfully.