Roll No. 45 Exam Seat No.

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

Hashu Advani Memorial Complex, Collector's Colony, R. C. Marg,
Chembur, Mumbai – 400074. Contact No. 02261532532

Since 1962

CERTIFICATE

Certified that Mr. Pushkar Prasad Sane

of FYMCA/A has

satisfactorily completed a course of the necessary experiments in

User Interface Lab under my supervision

in the Institute of Technology in the academic year 20 23 – 2024.

Principal Head of Department

Lab In-charge Subject Teacher

V.E.S. Institute of Technology, Collector Colony,
Chembur, Mumbai, Maharashtra 400047
Department of M.C.A

INDEX

Sr. No.	Contents	Date of Preparation	Date of Submission	Marks	Faculty Sign
1	UI Lifecycle and UI Tools.	05-02-2024	09-02-2024		
2	Project proposal and Requirement Gathering.	05-02-2024	09-02-2024		
3	Analysis of “Scorify” Website.	05-02-2024	09-02-2024		
4	Design Creation of Scenario and Story Write a scenario and story that involves all three of the tasks identified for the chosen project. Explain the Scenario, story, Sketch the scenario and story (use any tool or hand sketches).	05-02-2024	09-02-2024		
5	Creation of Persona.	12-02-2024	19-02-2024		
6	Mental Model.	12-02-2024	19-02-2024		
7	Low Fidelity Prototype Creating a Paper Prototype Prototype (Wire Frame) using any UI tool. • Paper prototype • Wireframe	12-02-2024	19-02-2024		

8	Creating a High Fidelity prototype using the Figma tool.	05-03-2024	09-03-2024		
9	Usability Evaluation of the Design Testing of User Interface from Third Party (Test scripts) Output Test Script.	05-03-2024	09-03-2024		
10	Comparative study of user interface design between Flipkart and Amazon website.	05-03-2024	09-03-2024		

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 1**	
Title of Lab Assignment: Introduction to UI Life Cycle and UI Tools.		
DOP: 14/01/2024	**DOS:**	
CO Mapped: **CO1**	**PO Mapped:** **PO2, PO3, PSO1**	**Signature:**

Practical No. 1

Aim: Introduction to UI Life Cycle and UI Tools.

Description:

- **Introduction to UI Life Cycle**

  The User Interface (UI) Lifecycle is a dynamic and systematic process that guides the creation, evolution, and optimization of user interfaces, ensuring a seamless and engaging experience for users throughout the development and usage journey. User Interface development process can be categorized into 7 phases as below:

  1. **Research and Analysis:** During this phase, we conduct user research to understand the target audience's preferences, behaviors, and needs. We may conduct interviews, surveys, or focus groups to gather valuable insights. Additionally, we analyze competitors' interfaces to identify trends and best practices. This research helps us create user personas and define the goals and requirements for the UI.

  2. **Planning and Strategy:** In this phase, we create a detailed plan for the UI project. We define project scope, set measurable goals, and establish a clear strategy. We also create a timeline, allocate resources, and identify key stakeholders. By having a well-defined plan and strategy, we ensure that the UI development process stays on track and aligns with the overall project objectives.

  3. **Design:** The design phase is where the visual and interactive elements of the UI come to life. We start by creating wireframes, which are basic layouts that outline the structure and placement of different UI elements. Then, we move on to creating high-fidelity prototypes that simulate the actual user experience. This phase involves choosing color schemes, typography, and creating a consistent visual language that reflects the brand identity.

  4. **Development:** Once the design is finalized, the development phase begins. Front-end developers use HTML, CSS, and JavaScript to code the UI elements

and implement the desired functionality. Back-end developers work on integrating the UI with the necessary server-side components and databases. This phase requires collaboration between designers and developers to ensure a seamless translation of the design into a fully functional interface.

5. **Testing and Quality Assurance:** In this phase, we thoroughly test the UI to identify any usability issues, bugs, or performance problems. Usability testing involves observing users as they interact with the interface to gather feedback and make necessary improvements. We also conduct accessibility testing to ensure the UI is usable by people with disabilities. Additionally, we perform performance testing to optimize the UI's speed and responsiveness.

6. **Deployment:** Once the UI has passed all the necessary tests and quality checks, it's time to deploy it to the production environment. This involves setting up servers, configuring databases, and ensuring that all the necessary infrastructure is in place. We carefully monitor the deployment process to ensure a smooth transition from development to the live environment.

7. **Maintenance and Updates:** Once the UI is live, it requires ongoing maintenance and updates to address any issues, incorporate user feedback, and keep up with technological advancements. Regular updates and improvements are essential to ensure a seamless user experience.

- **Introduction to UI Tools**
    1. **Adobe XD:**

       Adobe XD is a powerful design and prototyping tool that enables designers to create interactive and visually stunning user interfaces. Known for its seamless integration with other Adobe Creative Cloud applications, XD facilitates the design process from wireframing to interactive prototypes. Its collaborative features make it a popular choice for teams working on UI/UX projects, allowing for real-time collaboration and feedback.
    2. **Figma**

Figma is a cloud-based design tool that has gained popularity for its collaborative features and platform independence. Designers can create and prototype interfaces in real-time, fostering efficient team collaboration. With its browser-based interface, Figma enables designers to work seamlessly across different operating systems, making it a versatile choice for distributed teams. The shared components and design systems in Figma enhance consistency and streamline the design process.

3. **Sketch**

   Sketch is a vector-based design tool specifically crafted for UI and UX designers working on macOS. It is known for its simplicity, speed, and an extensive library of plugins that enhance its functionality. Sketch allows designers to create high-fidelity interfaces with ease and offers features like symbols and artboards for efficient design iteration. Its popularity among designers is also attributed to its straightforward learning curve.

4. **Marvel**

   Marvel is a user-friendly prototyping tool that simplifies the process of turning static designs into interactive prototypes. With Marvel, designers can link screens, add animations, and create realistic user flows. Collaboration features enable stakeholders to provide feedback directly on the prototypes. Marvel's focus on simplicity and quick prototyping makes it a valuable tool for designers aiming to test and iterate their designs swiftly.

5. **Fluid**

   Fluid is a design and prototyping tool designed for creating responsive and interactive user interfaces. It supports adaptive layouts for various screen sizes and devices, making it suitable for designing interfaces that work seamlessly across platforms. Fluid emphasizes the importance of responsive design, allowing designers to preview and test their interfaces in real-time for different screen resolutions and orientations.

6. **Pencil**

Pencil is an open-source GUI prototyping tool that is widely used for creating diagrams, UI mockups, and interactive prototypes. Available as a Firefox extension or a standalone application, Pencil provides a range of stencils and drawing tools. While it may not have the advanced features of some commercial tools, Pencil is a valuable option for those seeking a lightweight and accessible tool for basic UI prototyping and diagramming.

Conclusion: Introduction to User Interface (UI) life cycle and UI tools, design and theory done successfully.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 2**	
Title of Lab Assignment: Project Proposal and Requirement Gathering.		
DOP: 15-01-2024	**DOS: 09-02-2024**	
CO Mapped: **CO1, CO2**	**PO Mapped:** **PO2, PO3, PO5, PSO1, PSO2**	**Signature:**

Practical No. 2

Aim: Project proposal and requirement gathering.

Description:

A comprehensive software program called the Cricket Scoring System was created to completely transform the way cricket matches are scored at different levels. The technology guarantees accurate recording of ball-by-ball information, including runs, extras, wickets, and player statistics, with an intuitive interface and real-time updates. With its comprehensive statistical analysis features, it offers insights into individual and team performance. Scorers, umpires, and administrators may all access the platform thanks to its secure multi-level access and device flexibility.

Constructed with a strong technological foundation, the system aims to optimize the scoring procedure, minimize mistakes, and provide cricket fans with a more effective and captivating experience.

Project Proposal:

The goal of this project is to design a cricket scoring system that is easy to use and effective for keeping score of cricket matches at different levels. The system will have a real-time scoring interface capturing precise ball-by-ball information, coupled with extensive statistical analysis tools for player and team performance evaluation.

The project also highlights how crucial a responsive design is to ensure cross-platform compatibility and accessibility for administrators, umpires, and scorers. The Cricket Scoring System intends to improve the precision, speed, and overall experience of cricket match scoring, contributing to the modernization of the sport. It has a defined development timeframe, a strong technological stack, and a budgetary allotment.

Types of users:

1. Developers
2. Project Managers
3. Users

Objectives:

1. To reduce multiple manual calculations during the match such as net run rate, batsman's analysis, bowler's analysis.
2. To provide a ball-by-ball update of the match.
3. To provide a detailed scorecard.
4. Develop a user-friendly interface for scoring cricket matches.
5. Implement real-time scoring updates accessible to players, officials, and spectators.

Tools:

1. **Software:** Visual Studio Code, PHP, MySQL.
2. **Web Browser:** Chrome, Brave, Firefox.

Feasibility Study:

1. **Technical Feasibility:** Two popular and well-established web development technologies are PHP and MySQL. Their compatibility guarantees the project's technical viability, and a sizable development community offers resources and assistance. These technologies make it easy to integrate responsive design and real-time changes, and they also make hosting on conventional web servers or cloud platforms simple.

2. **Economic Feasibility:** The project's economic viability is reinforced by the utilization of open-source technologies (PHP, MySQL), which lowers license expenses. The development, hosting, and testing cost estimates taken together are affordable. It is anticipated that the advantages—such as enhanced scoring effectiveness and a better user experience—will exceed the expenses, making the project financially feasible.

3. **Legal Feasibility:** In order to ensure clear terms of service and agreements, the Cricket Scoring System project must comply with data protection and intellectual property legislation. It should be compliant with jurisdictional consequences and accessibility norms, necessitating strong contractual agreements and legal landscape knowledge. It is advised to consult legal advice.

4. **Operational Feasibility:** Because PHP and MySQL are extensively used and have a big user base, operational feasibility is high. Operational requirements are met via the creation of a user-friendly interface, multi-level access controls, and device interoperability. Developers' experience with MySQL and PHP guarantees that scorers, umpires, and administrators will find it easy to use and maintain.

Project Requirements:

1. **Software Requirements:**
    - System O.S: Window or Linux (Debian or Arch).
    - Front-end: HTML, JS, CSS.
    - Back-end: PHP.
    - Database: MySQL.

2. **Hardware Requirements:**
    - Processor: Intel Core 3.0 2.3 GHz or more.
    - RAM: 4GB or more.
    - Monitor: 17 CRT or LCD, Plasma, etc.
    - Hard-Disk: 256 or more (SSD preferable)
    - Keyboard: Normal or multimedia.
    - Mouse: Compatible

3. **Performance Requirements:**

    The proposed Cricket Scoring System will have following performance requirements:
    - Responsiveness: The user interface should be highly responsive, allowing scorers and administrators to input data quickly, and users to access information seamlessly across different devices, including smartphones, tablets, and desktops.
    - Scalability: The system should be capable of handling a scalable user base, accommodating varying levels of match complexity and user load, ranging from local tournaments to international events.
    - Reliability: The system should be reliable and available during matches, minimizing downtime and ensuring consistent performance to avoid disruptions to the scoring process.

- Availability: The system should be available 24/7, with minimal downtime for maintenance for updates. Any planned maintenance should be communicated to users in advance.

4. **Safety Requirements:**

   The proposed Cricket Scoring System will have following safety requirements:

   - User Safety: The system should prioritize the safety of users by providing accurate and up-to-date information about the score.
   - Data Safety: The system should ensure that user data is stored and processed securely, with appropriate measures in place to protect against data breaches, theft, or loss. This includes implementing user authentication, encryption of sensitive data and regular backups of data to prevent loss in case of a system failure.
   - Compliance: The system should comply with all relevant safety regulations, such as those related to user privacy and data protection. This includes ensuring that user data is collected, stored and processed in accordance with relevant laws and regulations.
   - User education: The system should provide users with relevant safety information and guidelines to ensure that they are aware of potential issues and best practices while scoring.

5. **Security Requirements:**

   The proposed Cricket Scoring System will have following security requirements:

   - User Authentication: Implement secure user authentication mechanisms, including strong password policies and optional multi-factor authentication, to ensure that only authorized individuals can access the system.
   - Access Control: Enforce role-based access controls to limit system access based on user roles (scorer, user), preventing unauthorized users from gaining inappropriate access to sensitive information.
   - Encryption: Utilize encryption protocols (such as SSL/TLS) to secure data transmission between users and the system, protecting against eavesdropping and unauthorized interception of sensitive information.

- Security Monitoring: The system should have security monitoring in place to detect and respond to any security threats and anomalies. This includes logging and monitoring all user activity.

Conclusion: Project requirements and requirement gathering for “Cricket Scoring System” is done successfully.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 3**	
Title of Lab Assignment: Analysis of “Scorify” application.		
DOP: 25-01-2024	**DOS: 09-02-2024**	
CO Mapped: **CO1, CO2**	**PO Mapped:** **PO2, PO3, PO5, PSO1**	**Signature:**

Practical No. 3

Aim: Analysis of "Scorify" application.

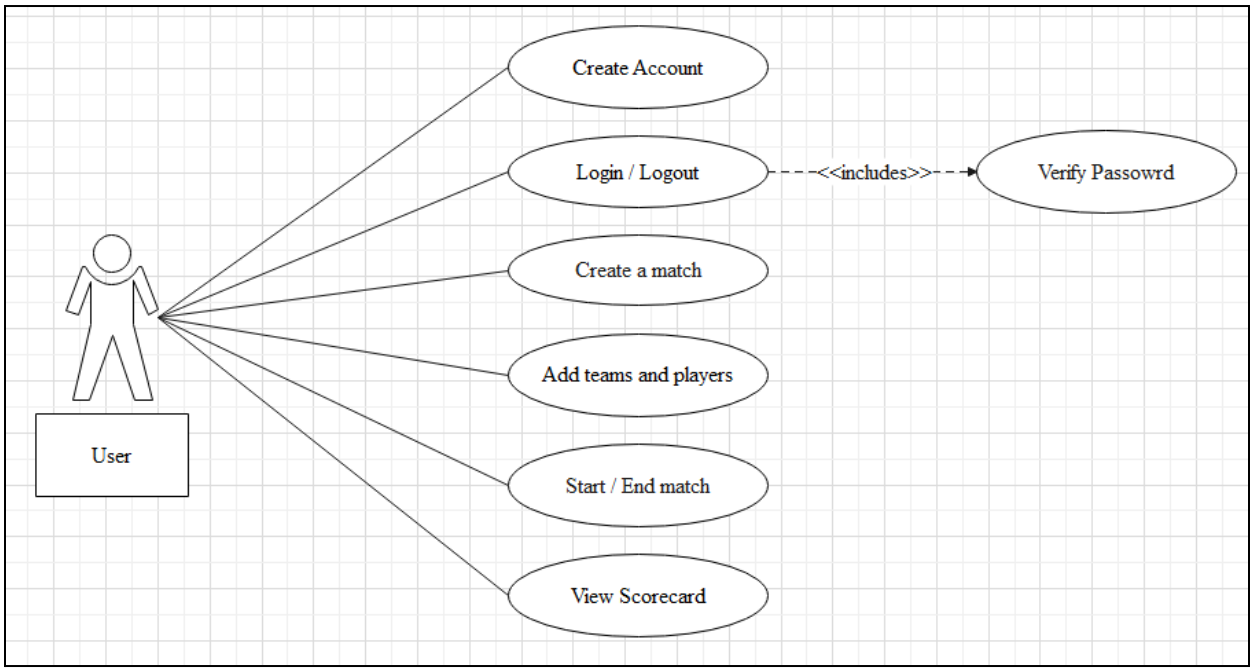
Description:

1. **User Analysis:**
    a. Target Audience: The target audience includes the individuals who play cricket matches and maintain record of it. This may include a diverse range of users such as Players, Coaches, Umpires and Scorers.
    b. User Profiles: The application includes profiles such as scorer and spectator where the scorer maintains the record of each and every event happening in the game and spectator who can get updates of the game.
    c. User Goals: Users of the application will have different goals such as
        i. Player Development: To track the performance of players during practice matches to identify areas for improvement.
        ii. Player Evaluation: To provide constructive feedback of players by analyzing their individual statistics, such as batting averages, bowling figures, and fielding performance.
        iii. Opponent Analysis: To scout opposing teams by reviewing their past performances and analyzing key players' strengths and weaknesses to develop effective match strategies.
    d. User Preferences: Users of this application prefer a user-friendly interface with customization options, mobile compatibility, and real-time updates during matches. Seamless data synchronization, offline capabilities, and strong data security measures are also crucial considerations.
    e. Feedback and Reviews: The application typically focus on the user experience, highlighting aspects such as ease of use, reliability, and the effectiveness of features like scoring, player statistics tracking, and team management. Users often appreciate intuitive interfaces, mobile compatibility, and timely updates during matches

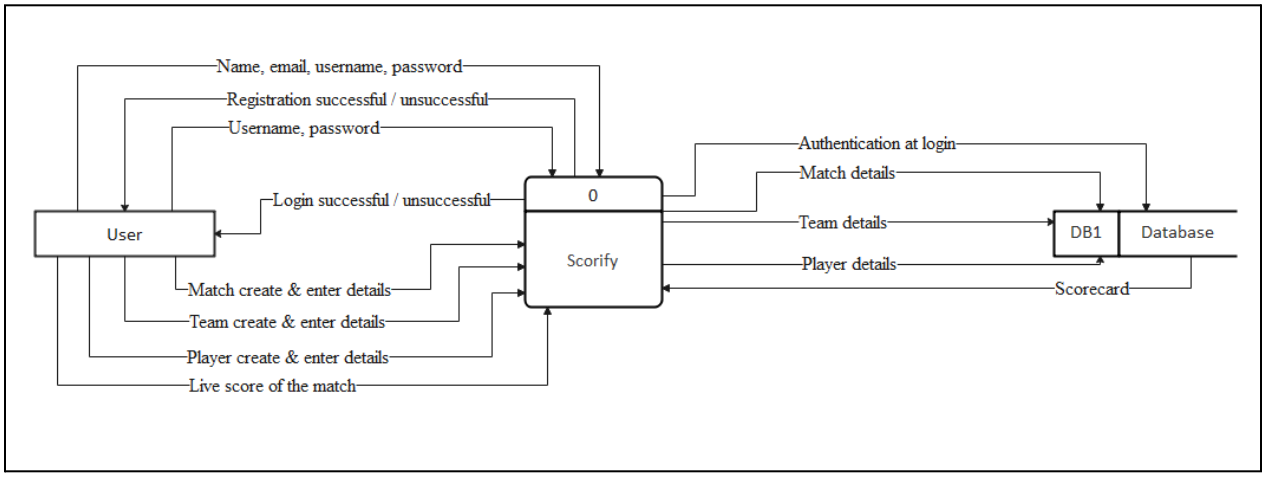
2. **Problem Areas:**
    a. User Interface Complexity: If the interface is overly complex or cluttered, users may struggle to navigate the application efficiently, leading to frustration and reduced usability.
    b. Technical Issues: Users may encounter bugs, glitches, or performance issues that disrupt their experience with the application, such as crashes, slow loading times, or errors in scoring calculations.
    c. Scoring Accuracy: Users may encounter issues with scoring inaccuracies, such as incorrect counts of runs, wickets, or overs, impacting the integrity of match data and statistics.
    d. Complexity of Scoring: The scoring process may be overly complex or cumbersome, particularly for users who are unfamiliar with cricket scoring conventions or lack experience with the application.
    e. Data Management Challenges: Managing large volumes of match data and player statistics may become cumbersome without efficient organization, search, and filtering capabilities, making it difficult for users to extract meaningful insights.

3. **Solution Goals:**
    a. Enhanced Scoring Accuracy: Implement algorithms and validation checks to ensure accurate scoring calculations, reducing errors and discrepancies in match data.
    b. Streamlined User Interface: Simplify the scoring process with an intuitive and user-friendly interface, incorporating clear prompts, tooltips, and visual aids to guide users through the scoring process efficiently.
    c. Flexible Customization Options: Expand customization features to allow users to tailor scoring formats, match settings, and player attributes to align with specific match requirements and user preferences.
    d. Efficient Data Management: Introduce advanced data management functionalities, such as robust search, filtering, and sorting capabilities, to facilitate the organization and retrieval of match data and player statistics effectively.

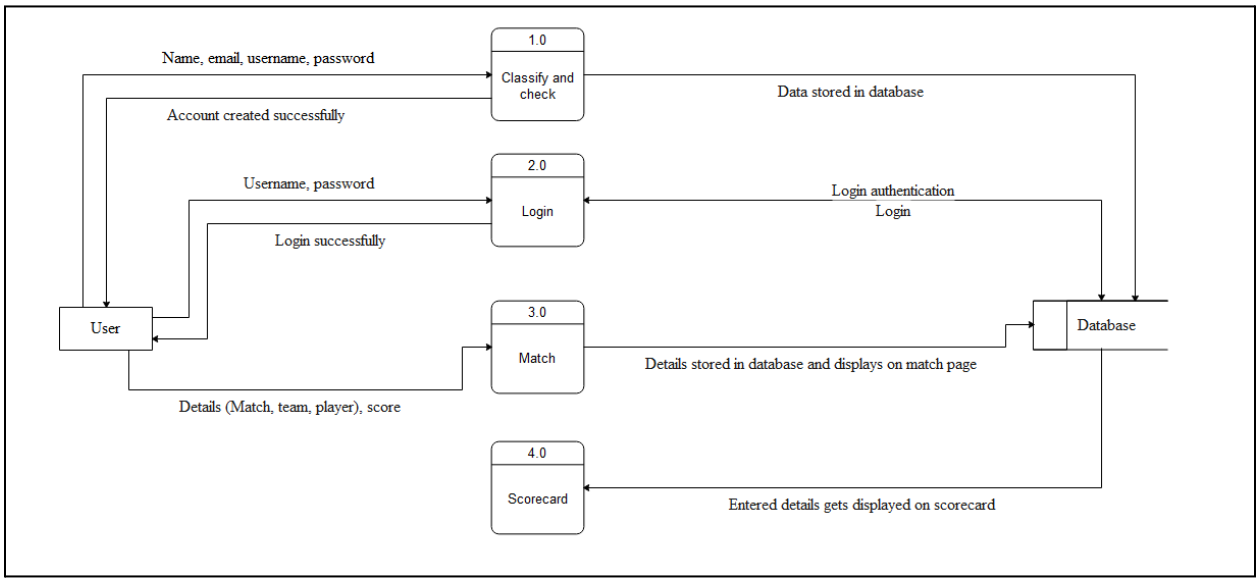
4. Use Case Diagram



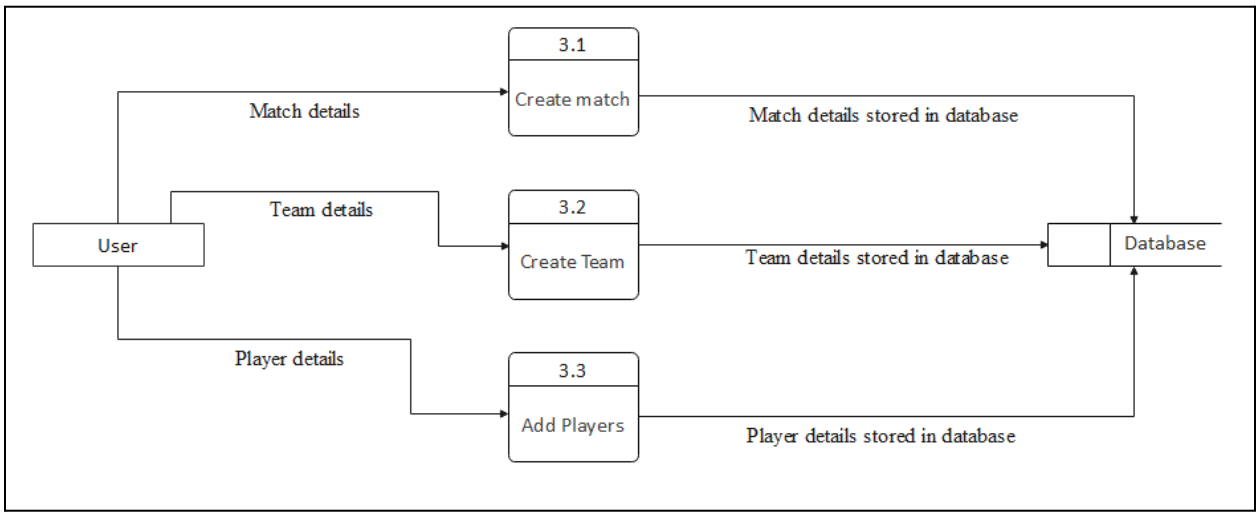
5. Data Flow Diagram



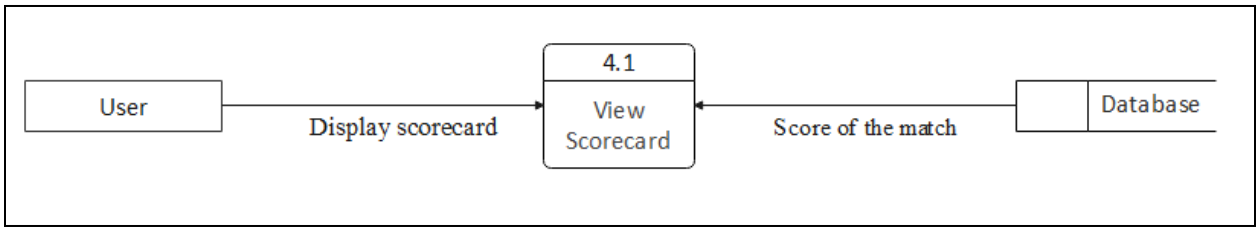
Level 0 DFD



Level 1 DFD

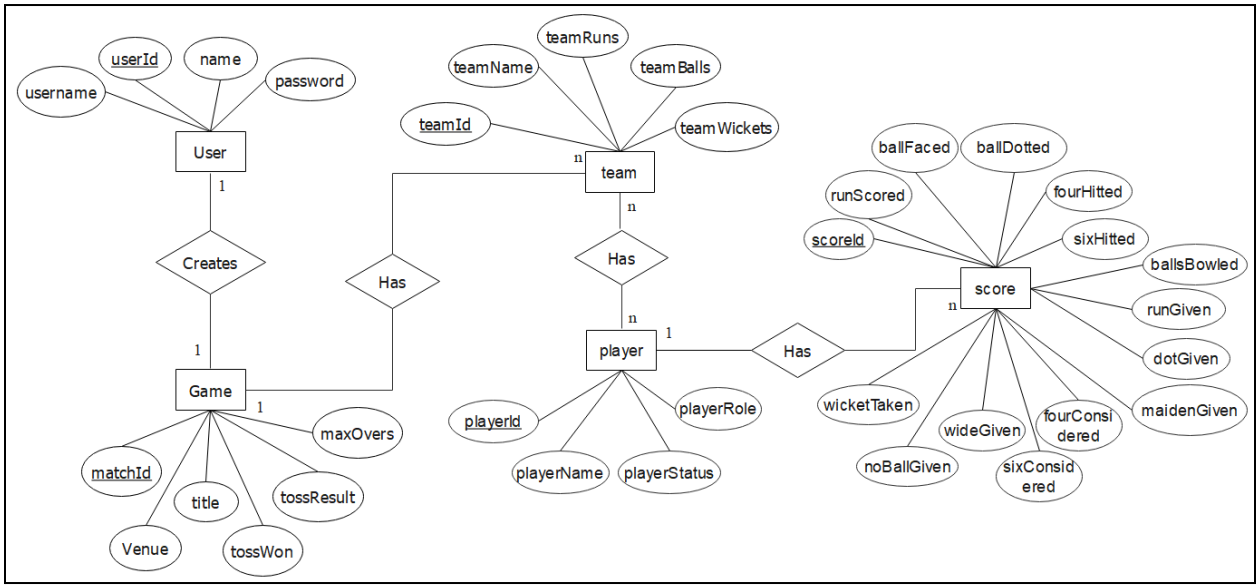


Level 2 DFD for Match



Level 2 DFD for Scorecard

6. ER Diagram



Conclusion: Analysis of “Scorify” done successfully.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 4**	
Title of Lab Assignment: Design creation of scenario. Write a scenario that involves all three of the tasks identified for the chosen project.		
DOP: 29-01-2024	**DOS: 09-02-2024**	
CO Mapped: **CO2**	**PO Mapped:** **PO3, PO5, PSO1, PSO2**	**Signature:**

Practical No. 4

Aim: Design creation of scenario. Write a scenario that involves all three of the tasks identified for the chosen project.

a. **Explain a scenario**
b. **Sketch a scenario**
c. **Draw a mental model**

Theory:

A UI Scenario, also known as user interface scenario, is a description or narrative that outlines the interactions and actions that a user may perform within a user interface (UI) of a software application or system. It provides a step-by-step account of how a user interacts with the UI, including the screens, input fields, buttons and other elements involved in the process.

A UI Scenario helps designers, developers and stakeholders understand and visualize the user's journey through the application's interface. It serves as a means to explore and validate the user experience, identify potential usability issues, and ensure that the interface meets the needs and expectations of the users.

A typical UI scenario includes details such as the user's actions, the system's responses and any relevant input or output data involved in the interaction. It may also include information about the context or purpose of the user's actions, such as the user's goals, motivations or specific tasks they are trying to accomplish.

Overall, a UI scenario provides a structured way to describe and analyze the user interface interaction allowing designers and developers to iteratively improve the UI design and ensure a seamless and intuitive user experience.

UI scenarios offer several advantages in the design and development process of software applications:

1. User Centric Focus: UI scenarios put the user at the center of the design process. By outlining the user's actions, goals and interactions with the interface, UI scenarios help

designers understand the user's perspective and design interfaces that meet their needs effectively.

2. Visualizing User Journeys: UI scenarios provide a visual representation of the user's journey through the application. They help stakeholders, designers and developers to visualize the sequence of screens, interactions and transitions, enabling a better understanding of the overall user flow.
3. Identifying User Interface Issues: By creating UI scenarios, designers can identify potential usability issues or challenges in the user interface early in the design process. It allows for proactive problem solving and refinement of the interface to enhance usability and user experience.
4. Iterative Design and Validation: UI scenarios facilitate an iterative design process. Designers can create, review and iterate on UI scenarios to validate and refine the interface design. It helps ensure that the interface aligns with user expectations, goals and tasks.
5. Collaboration and Communication: UI scenarios serve as a communication tool for designers, developers and stakeholders. They provide a common language and understanding of the user interface, facilitating collaboration and effective communication between team members.
6. Design Consistency: UI scenarios help maintain design consistency across the application. By visualizing the user flow and interface elements, designers can ensure a cohesive and seamless user experience throughout different screens and interactions.
7. Early User Feedback: UI scenarios can be shared with users or stakeholders for feedback at an early stage. This allows for gathering usable insights and incorporating user feedback into the design before significance development efforts have been made.
8. Documentation: UI scenarios can serve as documentation for the design rationale and decision making process. They provide a record of the user interface design and the intended user experience, making it easier to revisit and understand the design choices in the future.

Overall, UI scenarios help designers create user-centered, intuitive and effective interfaces. They enhance collaboration, facilitate interactive design and improve the overall user experience and usability of software applications.

A **Mental Model** refers to an individual's internal representation or understanding of how something works or how a particular system, concept or process functions. It is a cognitive framework that helps people interpret and interact with the world around them, guiding their perception, reasoning, decision-making and problem-solving.

Mental models are based on an individual's knowledge, beliefs, experiences and assumptions. They are constructed through a combination of learning, observation and personal interpretation. Mental models allow individuals to make sense of complex information and situations by simplifying and organizing them into coherent structures.

Here are a few points about mental models:

1. Simplifications and Abstraction: Mental models simplify reality by abstracting and focusing on the essential aspects of a system or concept. They filter out irrelevant details and highlight the most relevant elements for understanding and decision making.
2. Influence on Perception and Interpretation: Mental models shape how individuals perceive and interpret information. They serve as a lens through which people view the world, influencing their understanding of judgements about the environment and the actions they take.
3. Predictive and Inferential Power: Mental models enable individuals to predict and anticipate how things will behave or unfold based on their understanding of underlying principles and casual relationships. They help individuals make inferences and draw conclusions based on their mental representations.
4. Bias and Limitations: Mental models are subjective and can be influenced by biases, misconceptions or incomplete information. They may lead to cognitive biases, such as confirmation bias or anchoring bias, which can impact decision-making and problem-solving.
5. Evolving and Adaptive: Mental models are not fixed but can evolve and adapt over time through new experiences, learning and exposure to new information. They are constantly refined and updated as individuals acquire new knowledge or challenge their existing beliefs.

Mental models play a crucial role in various domains, including psychology, cognitive science, user experience design and decision making. Understanding the mental models of users is particularly important in user-centered design as it helps designers create interfaces and

interactions that align with users' existing mental models, making them more intuitive and user-friendly.

Scenarios for Users while working with the 'Scorify' website:

Scenario 1: Homepage

- Upon entering the website, users are greeted with a visually appealing homepage.
- The user can also see the latest news regarding the game.
- Clear navigation menus direct users to different sections of the website, such as "Login", "Registration,", "About" and "Contact Us".

Scenario 2: Scorer Dashboard

- Upon logging in, the scorer is directed to a dashboard where they can see ongoing matches and matches they have scored previously.
- They can select a match to update or start scoring a new match.
- The dashboard also provides access to settings, help documentation, and account management options.

Scenario 3: Match Details Page

- Each match has a dedicated page where users can find comprehensive details such as teams, venue, date, toss result, playing XI, and match status.
- This page also displays ball-by-ball commentary, milestones, and notable events during the match.

Scenario 4: Match Scoring Interface

- The scorer enters the match details such as teams, venue, date, and type of match (e.g., Test, ODI, T20).
- During the match, the scorer updates scores, wickets, overs, and other relevant information in real-time.
- They can easily switch between innings, overs, and bowlers to update the score accurately.

Scenario 5: Scorecard View:

- Users can view the detailed scorecard of completed matches, including batting and bowling statistics for each player.
- The scorecard includes runs scored, balls faced, strike rate, wickets taken, economy rate, and other relevant metrics.

Component	Description	UI Scenario
User Roles	Scorer: Responsible for updating scores, wickets, overs, and other match details in real-time.	2. Scorer Dashboard 3. Match Details Page 4. Match Scoring Interface 5. Scorecard View
Match Information	Includes details such as teams, venue, date, type of match, toss result, playing XI, and match status.	3. Match Details Page 4. Match Scoring Interface 5. Scorecard View
Detailed Match Analysis	Provides ball-by-ball commentary, milestones, notable events, batting and bowling statistics for each player.	4. Match Details Page 5. Scorecard View

Conclusion: We have successfully implemented & Created a Scenario & StoryBoard Template for Users and problems that need to be addressed.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 5**	
Title of Lab Assignment: Creation of Persona Based on Practical 4 and Design persona as per the users of the application.		
DOP: 12-02-2024	**DOS: 19-02-2024**	
CO Mapped: **CO3**	**PO Mapped:** **PO3, PO5, PO7, PO12, PSO1, PSO2**	**Signature:**

Practical No. 5

Aim: Creation of Persona Based on Practical 4 and Design persona as per the users of the application.

Description:

1. **Persona 1:**

   Name: Anish

   Age: 35

   Occupation: Cricket Coach

   Location: Bangalore, India

   Background and Demographics

   Rahul is a seasoned cricket coach with a passion for the game. He has been involved in cricket since his childhood and has played at various levels before transitioning into coaching. Rahul spends most of his time at the cricket ground, either training his team or analyzing matches.

   Goals and Motivations

   - Maintain accurate records of matches played by his team for analysis and improvement.
   - Access live scores and updates during matches to make real-time decisions and adjustments.
   - Collaborate with other coaches and team members to strategize and share insights.
   - Stay updated with the latest cricket trends, news, and statistics to enhance his coaching techniques.

   Challenges

   - Balancing coaching responsibilities with administrative tasks like maintaining match records.
   - Ensuring the accuracy and reliability of match statistics for effective analysis.

- Need for a user-friendly platform that simplifies the scoring process and allows for easy collaboration with team members.
- Limited time for manual data entry and analysis due to his busy coaching schedule.

How the Application Addresses His Needs

- Provides a user-friendly interface for maintaining detailed match records with ease.
- Offers real-time scoring updates and match statistics, allowing Rahul to make quick decisions during matches.
- Facilitates collaboration with team members through features like sharing match data and discussing strategies.
- Offers automated data entry and analysis tools to streamline the coaching process and save time.

Persona Image

Rahul is a dedicated cricket coach who relies on your cricket scoring website to manage and analyze match data efficiently. He values accuracy, reliability, and convenience in maintaining match records and accessing live scores. Your website serves as an indispensable tool in his coaching arsenal, empowering him to make informed decisions and lead his team to success.

2. **Persona 2:**

Name: Shreya
Age: 25
Occupation: Student
Location: Delhi, India

Background and Demographics

Priya is a cricket enthusiast who grew up watching matches with her family and friends. She is currently pursuing her undergraduate degree in computer science. Cricket is not just a sport for Priya; it's a passion that she loves to indulge in during her free time.

Goals and Motivations

- Stay updated with live cricket scores and match results.
- Access detailed player statistics and performance analysis.
- Engage with fellow cricket fans and participate in discussions about matches.
- Learn more about the technical aspects of the game and scoring techniques.

Challenges

- Limited access to live matches due to her academic commitments.
- Difficulty finding a platform that offers comprehensive cricket coverage and analysis.
- Wants a user-friendly interface that is easy to navigate and understand, especially since she may not be familiar with advanced cricket terminology.
- Prefers a platform that provides educational resources to enhance her understanding of the game.

How the Application Addresses Her Needs

- Provides real-time updates on live matches, allowing Priya to stay connected with the game even during her busy schedule.
- Offers detailed player profiles, match statistics, and analysis, enabling her to deepen her knowledge and understanding of the game.
- Features a simple and intuitive interface that makes it easy for Priya to navigate and engage with the platform.
- Includes educational resources such as articles, videos, and tutorials to help Priya learn more about cricket and scoring techniques.

Persona Image:

Priya is a young cricket enthusiast who is passionate about the game and eager to learn more about it. Despite her academic commitments, she makes time to follow matches and stay updated with the latest cricket news and statistics. Your cricket scoring website serves as her go-to platform for all things cricket, providing her with the information, engagement, and education she seeks to fuel her passion for the game.

3. **Persona 3:**

Name: Mrudula

Age: 30

Occupation:Scorer

Location: Chennai, India

Background and Demographics

Deepak is a former professional cricketer who now works as both a cricket coach and a scorer for local cricket matches. He has a deep understanding of the game, having played at various levels and now sharing his expertise with aspiring cricketers. Deepak is meticulous in his scoring duties, ensuring accurate records are maintained for every match he scores.

Goals and Motivations

- Maintain accurate and detailed records of cricket matches for analysis and reference.
- Provide real-time updates and scores during matches to players, coaches, and spectators.
- Access advanced statistical analysis to identify patterns and trends in player and team performance.
- Collaborate with fellow scorers and officials to ensure smooth conduct of matches and accurate scoring.

Challenges

- Balancing coaching responsibilities with scoring duties during matches.
- Ensuring accuracy and consistency in scoring under pressure during fast-paced matches.
- Need for a user-friendly scoring platform that streamlines the scoring process and minimizes errors.
- Limited resources for accessing advanced statistical analysis tools for match insights.

How the Application Addresses His Needs

- Provides a user-friendly interface for Deepak to efficiently record and update match scores and statistics in real-time.
- Offers scoring features tailored to cricket matches, including customizable scoring templates for different formats of the game.
- Incorporates built-in error-checking mechanisms to minimize scoring errors and ensure accuracy.
- Includes advanced statistical analysis tools for Deepak to delve deeper into match data and gain valuable insights for coaching and analysis.

Persona Image

Deepak is a dedicated cricket coach and scorer who takes pride in his role in the game. He approaches scoring with the same level of dedication and professionalism as his coaching duties, ensuring that every match is accurately recorded and updated in real-time. Your cricket scoring website serves as an invaluable tool for Deepak, empowering him to fulfill his scoring duties efficiently and effectively while also gaining valuable insights for coaching and analysis purposes.

4. **Persona 4:**

Name: Pushkar
Age: 22
Occupation: University Student (Majoring in Sports Management)
Location: Kolkata, India

Background and Demographics:

Ananya is a passionate cricket enthusiast who has always been involved in sports, both as a player and a volunteer. She is currently pursuing her degree in Sports Management and actively seeks opportunities to gain practical experience in the field. Ananya volunteers at local cricket matches, assisting with various tasks including scoring.

Goals and Motivations

- Gain hands-on experience in sports management, particularly in cricket.

- Assist in maintaining accurate records and scores during cricket matches to support the smooth conduct of the event.
- Learn about the intricacies of scoring and match management to enhance her knowledge and skills in sports management.
- Connect with professionals in the cricket industry to explore potential career opportunities and network with like-minded individuals.

Challenges

- Limited experience in scoring and match management, especially in high-pressure situations.
- Need for guidance and support from experienced scorers and officials to ensure accurate and reliable scoring.
- Desire for a user-friendly scoring platform that simplifies the scoring process and allows for quick learning and adaptation.
- Limited access to educational resources and training materials for improving scoring skills and understanding cricket rules and regulations.

How the Application Addresses Her Needs

- Provides a user-friendly interface with intuitive scoring features designed for volunteers like Ananya, making it easy for her to learn and adapt to scoring tasks.
- Offers guidance and support through tutorials, tips, and online forums where Ananya can connect with experienced scorers and officials for advice and assistance.
- Incorporates error-checking mechanisms and validation tools to minimize scoring errors and ensure accuracy.
- Includes educational resources such as video tutorials, articles, and quizzes to help Ananya improve her scoring skills and deepen her understanding of cricket rules and regulations.

Persona Image:

Ananya is a young and enthusiastic cricket volunteer who is eager to learn and contribute to the sports management field. She approaches her volunteering role with enthusiasm and dedication, seeking opportunities to expand her knowledge and skills in

scoring and match management. Your cricket scoring website serves as a valuable resource for Ananya, providing her with the tools, guidance, and support she needs to excel in her role and pursue her passion for cricket and sports management.

Conclusion: We designed a persona as per the users of the application.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 6**	
Title of Lab Assignment: Design Mental Model Based on Practical 4 and Draw a mental model.		
DOP: 10-02-2024	**DOS: 19-02-2024**	
CO Mapped: **CO3, CO4**	**PO Mapped:** **PO3, PO5, PO7, PO12, PSO1, PSO2**	**Signature:**

Practical No. 6

Aim: Design Mental Model Based on Practical 4. Draw a mental model.

Description:

A Mental Model refers to an individual's internal representation or understanding of how something works or how a particular system, concept or process functions. It is a cognitive framework that helps people interpret and interact with the world around them, guiding their perception, reasoning, decision-making and problem-solving.

Mental models are based on an individual's knowledge, beliefs, experiences and assumptions. They are constructed through a combination of learning, observation and personal interpretation. Mental models allow individuals to make sense of complex information and situations by simplifying and organizing them into coherent structures.

Here are a few points about mental models:

1. Simplifications and Abstraction: Mental models simplify reality by abstracting and focusing on the essential aspects of a system or concept. They filter out irrelevant details and highlight the most relevant elements for understanding and decision making.
2. Influence on Perception and Interpretation: Mental models shape how individuals perceive and interpret information. They serve as a lens through which people view the world, influencing their understanding of judgements about the environment and the actions they take.
3. Predictive and Inferential Power: Mental models enable individuals to predict and anticipate how things will behave or unfold based on their understanding of underlying principles and casual relationships. They help individuals make inferences and draw conclusions based on their mental representations.
4. Bias and Limitations: Mental models are subjective and can be influenced by biases, misconceptions or incomplete information. They may lead to cognitive biases, such as confirmation bias or anchoring bias, which can impact decision-making and problem-solving.
5. Evolving and Adaptive: Mental models are not fixed but can evolve and adapt over time through new experiences, learning and exposure to new information. They are

constantly refined and updated as individuals acquire new knowledge or challenge their existing beliefs.

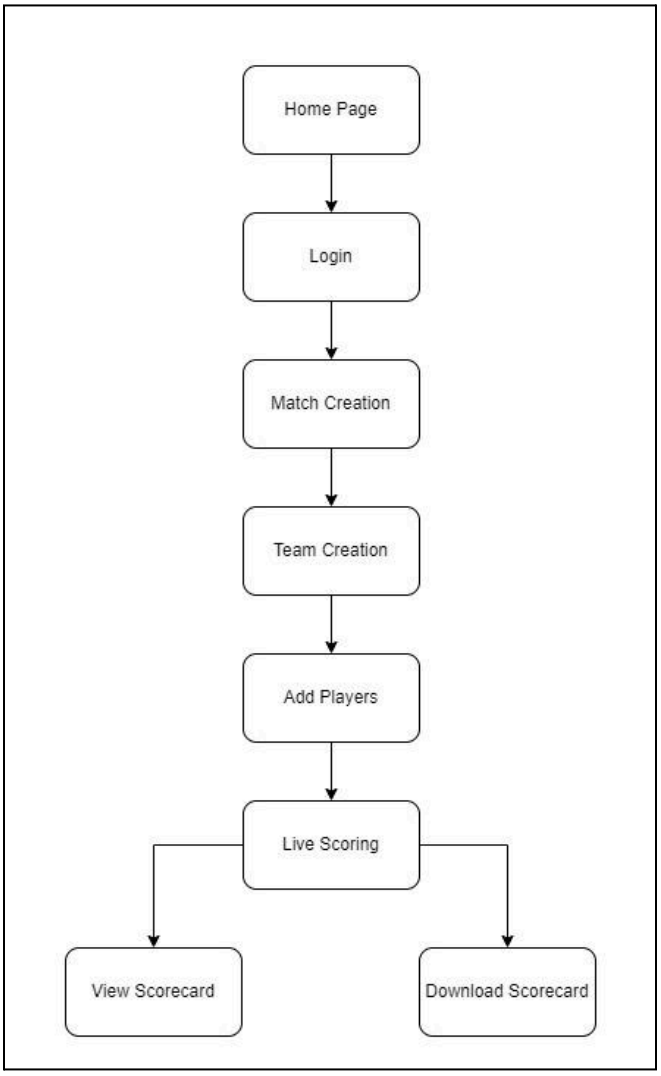
Mental models play a crucial role in various domains, including psychology, cognitive science, user experience design and decision making. Understanding the mental models of users is particularly important in user-centered design as it helps designers create interfaces and interactions that align with users' existing mental models, making them more intuitive and user-friendly.

Table for Mental Model:

Sr. No.	Scenario	Description
1	Accessing Matches	Scorers can access a list of matches from the dashboard. They can filter matches by date, venue, or team and select the match they wish to score.
2	Starting a Match	Upon selecting a match, scorers are directed to the scoring interface where they can input match details such as teams, venue, and toss result. They can start scoring once all necessary information is entered.
3	Recording Scores	Scorers input match data in real-time, recording runs scored, wickets taken, extras conceded, and other relevant statistics for each ball bowled. They can switch between batting and bowling views for both teams.
4	Managing Players	Scorers manage player details, including adding new players, editing existing player information, and assigning batting and bowling orders.
5	Handling Overs	Scorers track overs bowled by each bowler, updating the over count after every six balls. They record maiden overs, wides, no-balls, and other bowling events as they occur.
6	Recording Events	Scorers note significant match events such as boundaries, sixes, milestones (e.g., half-centuries, centuries), dismissals (e.g., catches, run-outs), and player substitutions.
7	Reviewing and Editing	Scorers can review match data at any time during scoring. They have the ability to edit

		scores, correct errors, and make adjustments as needed.
8	Saving and Finalizing	Once the match is complete, scorers finalize the scorecard and save the data. They can generate a printable scorecard, export match statistics, and share match results with stakeholders.

Diagram:



Conclusion: We designed a mental model.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 7**	
Title of Lab Assignment: Low Fidelity Prototype. Creating a Paper Prototype, Prototype (Wire Frame) using any UI tool. ● **Paper prototype** ● **Wireframe.**		
DOP: 10-02-2024	**DOS: 19-02-2024**	
CO Mapped: **CO5**	**PO Mapped:** **PO3, PO5, PO7, PO12, PSO1, PSO2**	**Signature:**

Practical No. 7

Aim: Low Fidelity Prototype. Creating a Paper Prototype, Prototype (Wire Frame) using any UI tool.

- **Paper prototype**
- **Wireframe.**

Theory:

A paper prototype is a low-fidelity representation or simulation of a digital product, such as a website, application, or software interface, created using pen-and-paper or other physical materials like sticky notes, index cards, and sketches. Paper prototypes are typically used in the early stages of the design process to quickly explore and test ideas before investing time and resources into developing a digital prototype or a fully functional product.

The main purpose of a paper prototype is to gather feedback from stakeholders, users, or usability experts regarding the design, layout, and functionality of the product. Designers can easily make changes or iterate on different design concepts during user testing sessions by modifying the paper prototype on the fly.

The key characteristics of a paper prototype include:

1. Low Fidelity: Paper prototypes are intentionally simplistic and low-tech, using basic materials like paper, markers, and sticky notes. This low fidelity allows for quick creation and easy modification during the design process.
2. Physical Representation: Unlike digital prototypes, paper prototypes are physical artifacts that can be easily manipulated by hand. Designers can cut, fold, and rearrange paper elements to simulate different interactions and user flows.
3. Ease of Creation: Paper prototypes can be created rapidly, often within minutes or hours, depending on the complexity of the design. This enables designers to explore multiple design concepts and iterate on ideas quickly.
4. Interactivity: Despite being made of paper, prototypes can simulate interactivity through techniques such as flipping pages, sliding panels, or using sticky notes to represent interactive elements like buttons and menus.
5. User Testing: Paper prototypes are used primarily for user testing and feedback gathering in the early stages of the design process. They allow designers to observe

how users interact with the prototype, identify usability issues, and gather insights to inform further iterations.

6. Collaboration: Paper prototypes encourage collaboration among team members, stakeholders, and users. Designers can easily share and iterate on ideas together, incorporating feedback from various perspectives.
7. Cost-Effectiveness: Paper prototypes are extremely cost-effective compared to digital prototypes or fully developed products. They require minimal resources and can be created using readily available materials, making them accessible to designers with limited budgets.
8. Flexibility: Paper prototypes offer flexibility in terms of design exploration and iteration. Designers can quickly make changes to the prototype based on user feedback, allowing for rapid iteration and improvement.
9. Risk Reduction: By testing design ideas with paper prototypes early in the process, designers can identify and address potential usability issues and design flaws before investing significant time and resources into digital development.

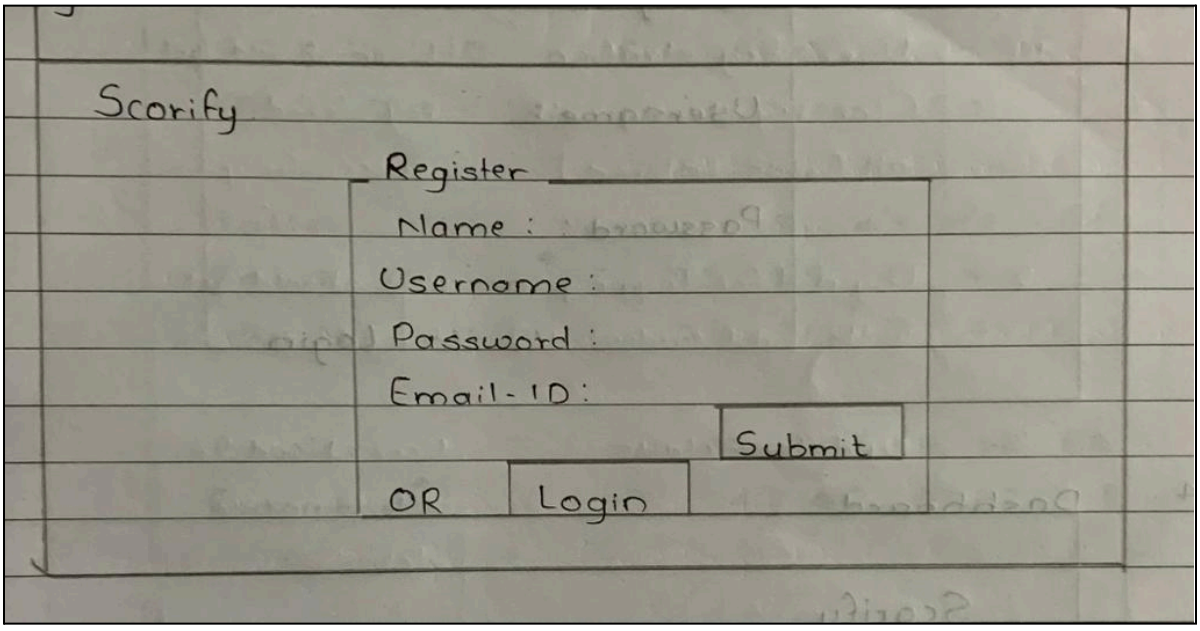
The benefits of using paper prototypes in the design process are numerous, including:

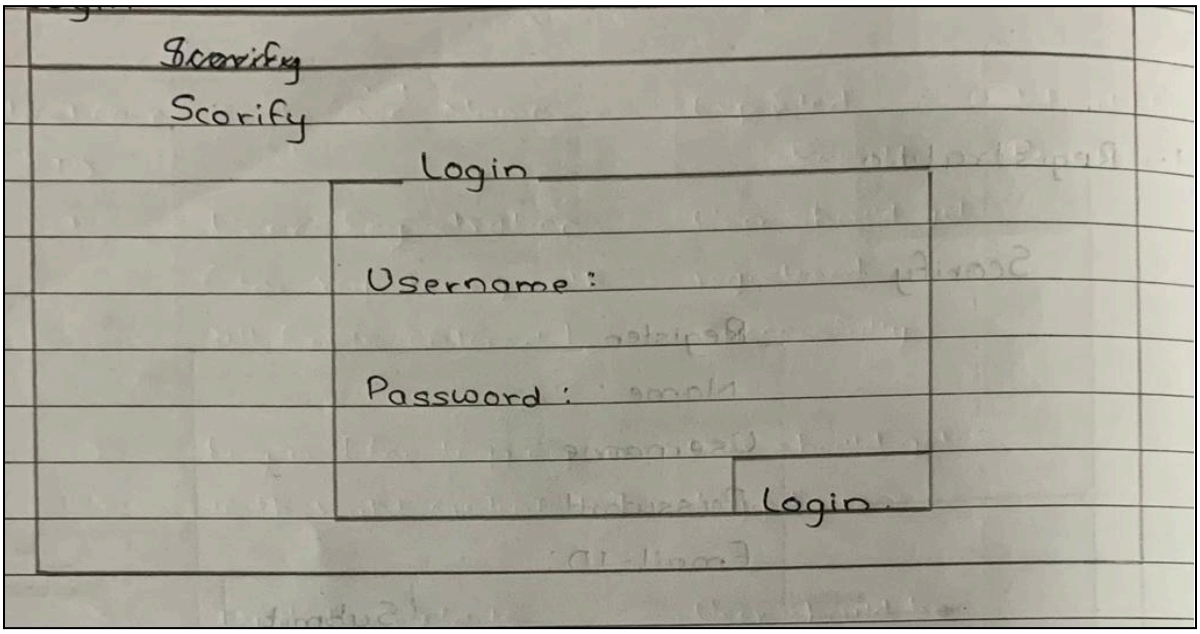
1. Rapid Iteration: Paper prototypes enable designers to quickly create and iterate on design ideas. Changes can be made on the fly during user testing sessions, allowing for rapid refinement of the product.
2. Low Cost: Paper prototypes are inexpensive to create, requiring only basic materials like paper, pens, and sticky notes. This makes them accessible to designers with limited budgets and resources.
3. Early Feedback: Paper prototypes facilitate early user testing, allowing designers to gather feedback on design concepts before investing significant time and resources into development. This helps identify usability issues and refine the user experience early in the process.
4. Flexibility: Paper prototypes are highly flexible and adaptable. Designers can easily make changes to the prototype during testing sessions, exploring different design options and iterating based on user feedback.
5. Collaboration: Paper prototypes encourage collaboration among team members, stakeholders, and users. Designers can involve stakeholders in the design process, gather input from various perspectives, and make informed decisions together.

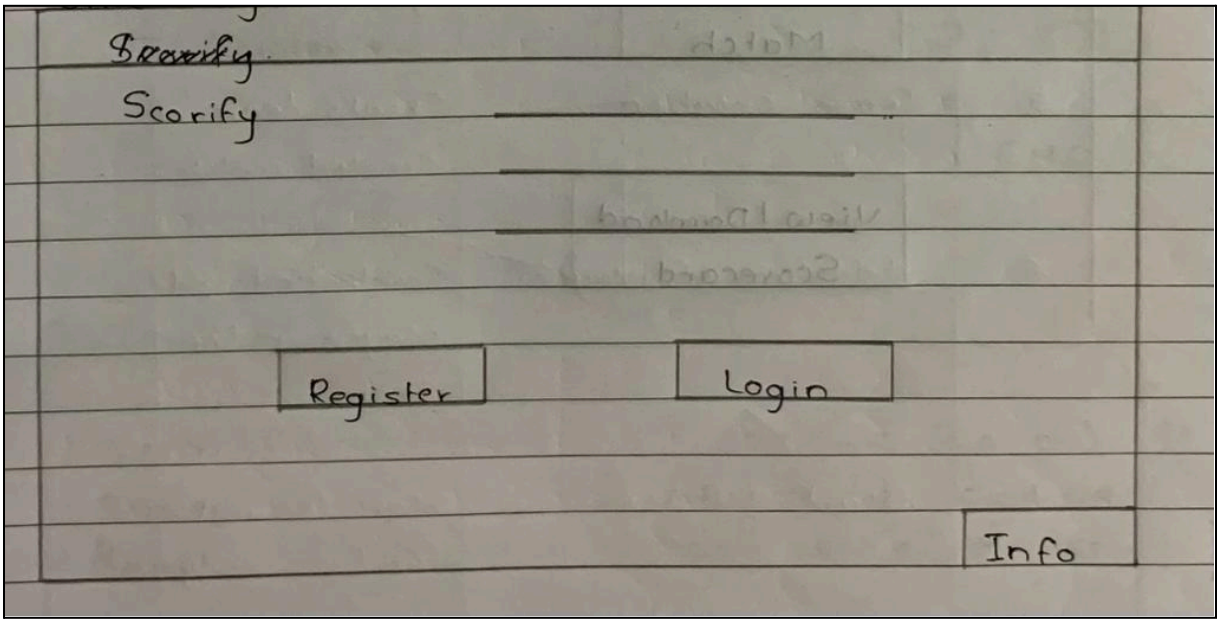
6. Visual Communication: Paper prototypes provide a tangible representation of design ideas that can be easily understood by stakeholders and users, regardless of their technical expertise. This helps facilitate communication and alignment throughout the design process.
7. Risk Reduction: By testing design ideas with paper prototypes early in the process, designers can identify and address potential usability issues and design flaws before investing significant time and resources into digital development. This reduces the risk of costly errors later on.
8. Enhanced Creativity: The low-tech nature of paper prototypes encourages creativity and exploration. Designers are not limited by software constraints and can freely experiment with different design concepts and interactions.
9. User Engagement: Involving users in the design process through paper prototyping fosters a sense of ownership and engagement. Users appreciate being consulted and can provide valuable insights that contribute to a more user-centered design.

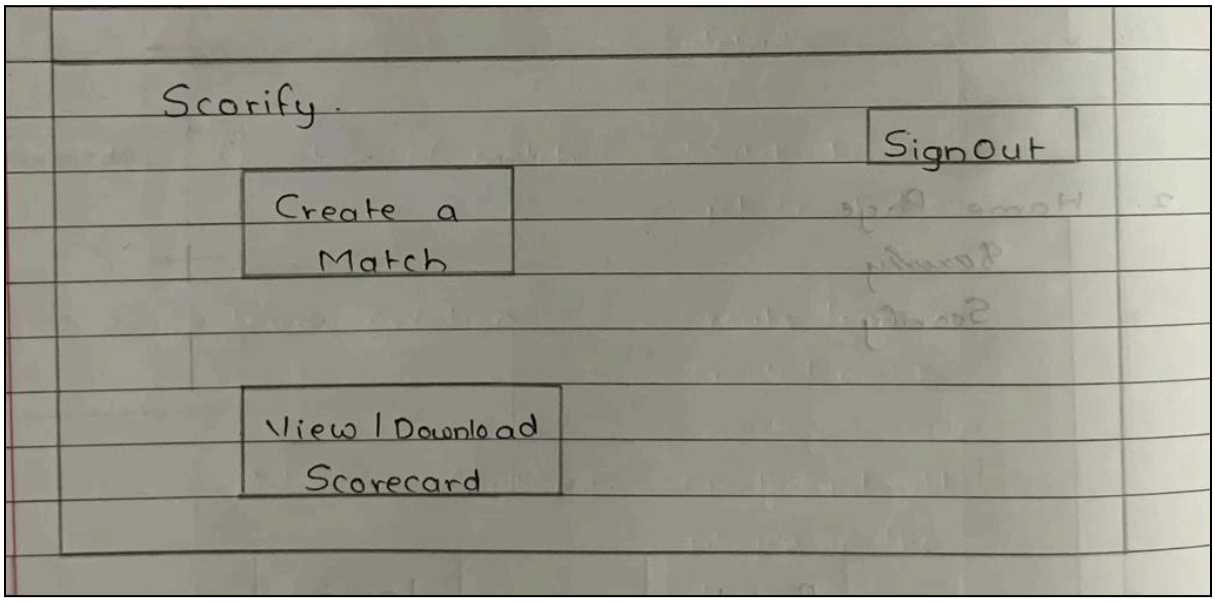
It is important to note that while paper prototypes are valuable tools for early-stage design exploration, they are not meant to replace digital prototypes or development. They serve as a starting point for ideation,feedback and iteration, providing a tangible representation of the user interface and user experience.

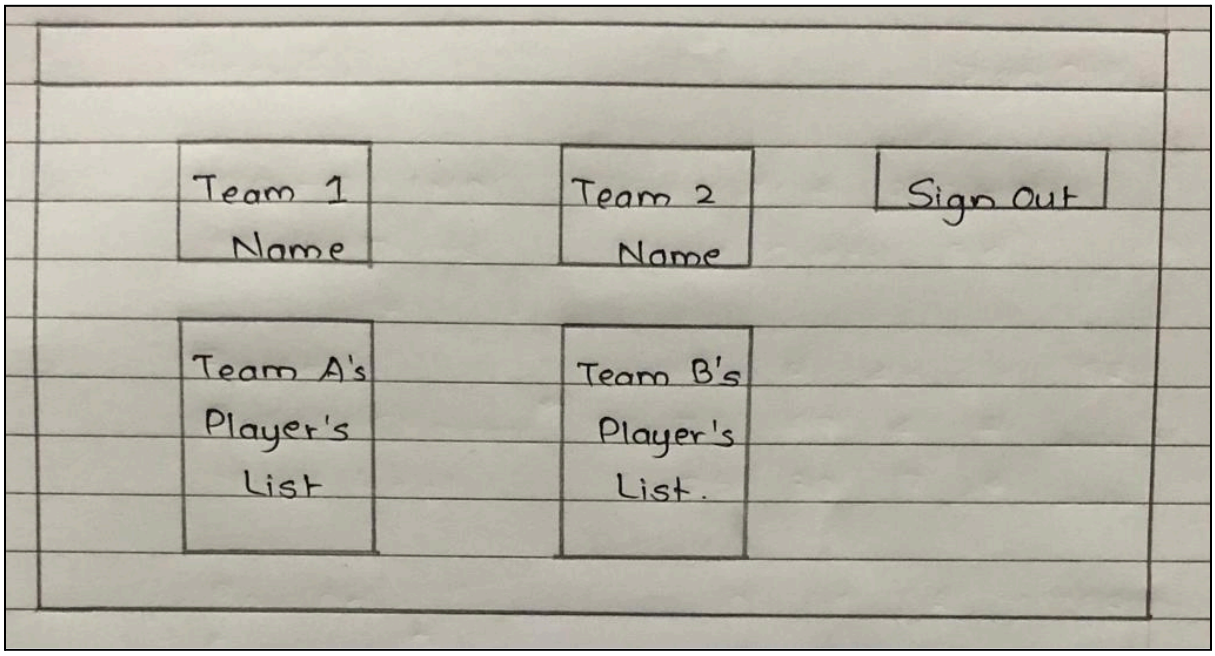
Paper Prototyping:

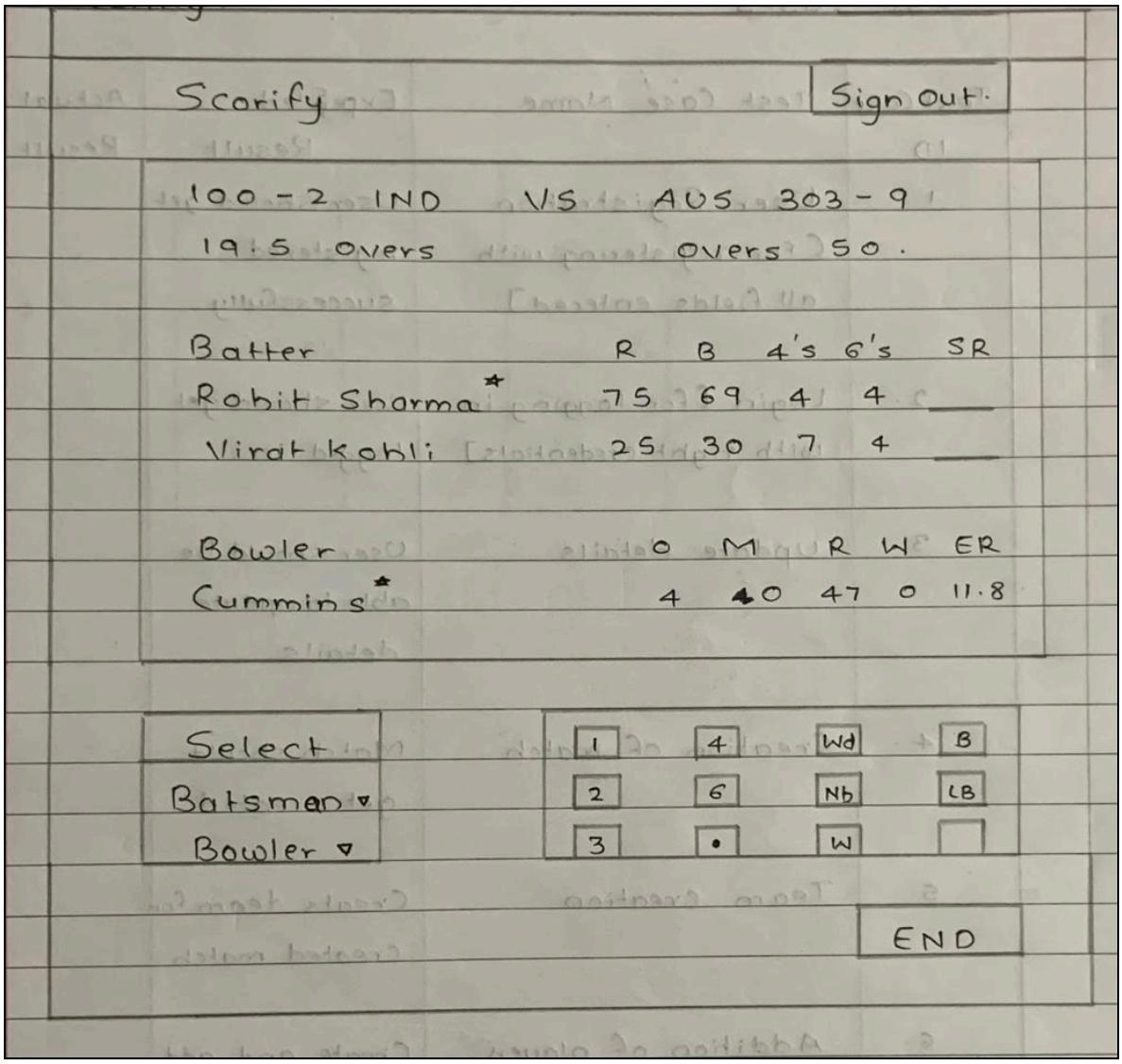


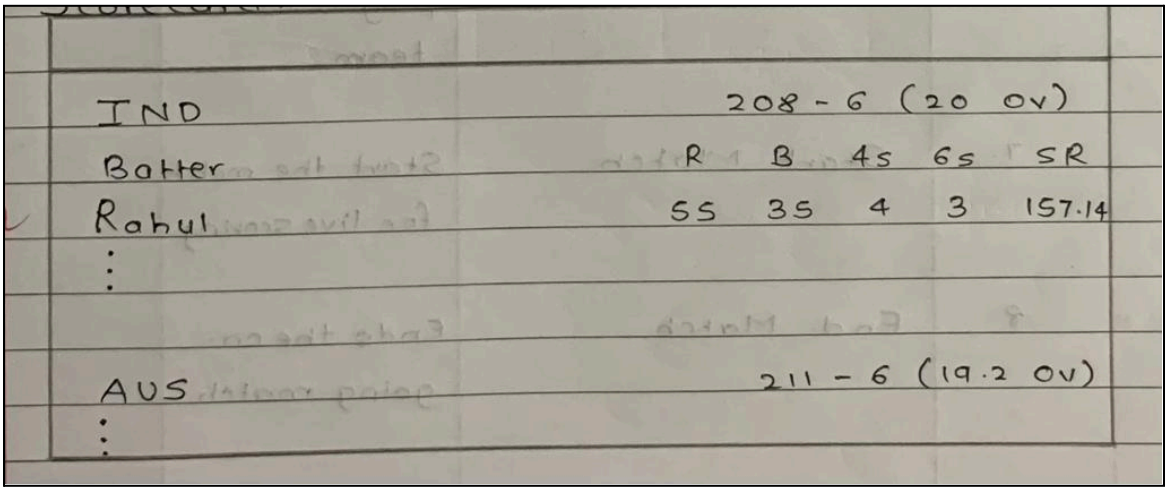

Scarify
Login
Username :
Password :
Login


Scarify
Register
Login
Info


Scorify.
Sign Out
Create a Match
View / Download Scorecard


Team 1 Name
Team 2 Name
Sign Out
Team A's Player's List
Team B's Player's List.


Scorify
Sign out
100 - 2 IND VS AUS 303 - 9
19.5 overs overs 50.
Batter R B 4's 6's SR
Rohit Sharma* 75 69 4 4
Virat Kohli 25 30 7 4
Bowler O M R W ER
Cummins* 4 40 47 0 11.8
Select
Batsman
Bowler
1 4 Wd B
2 6 Nb LB
3 • W
END


IND 208 - 6 (20 ov)
Batter R B 4s 6s SR
Rahul 55 35 4 3 157.14
AUS 211 - 6 (19.2 ov)

Conclusion: I have learned what a low-fidelity prototype is and created a paper prototype.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 8**	
Title of Lab Assignment: Creating a high-fidelity prototype using the Figma tool.		
DOP: 05-03-2024	**DOS: 09-03-2024**	
CO Mapped: **CO5**	**PO Mapped:** **PO3, PO5, PO7, PO12,** **PSO1, PSO2**	**Signature:**

UI Practical 8

Aim: Creating a High-Fidelity prototype using the Figma tool.

Introduction:

Figma is a cloud-based design tool used for interface design, prototyping, and collaboration among teams. It has gained significant popularity due to its versatility, real-time collaboration features, and accessibility across different platforms. When it comes to high-fidelity design in Figma, there are several theories and best practices to consider:

1. **High-Fidelity vs. Low-Fidelity:** High-fidelity design refers to designs that closely resemble the final product in terms of visual appearance and interaction. Low-fidelity designs, on the other hand, are rough sketches or wireframes that focus more on structure and functionality. In Figma, designers can gradually move from low-fidelity to high-fidelity designs as the project progresses.
2. **Visual Design Principles:** High-fidelity designs in Figma should adhere to fundamental design principles such as contrast, alignment, repetition, and proximity (CARP). These principles help create visually appealing and effective designs that communicate the intended message clearly to users.
3. **Typography:** Selecting appropriate typography is crucial in high-fidelity design. Figma offers a wide range of fonts and text styling options. Designers should consider factors such as readability, hierarchy, and brand consistency when choosing fonts for their designs.
4. **Color Theory:** Color plays a significant role in high-fidelity design as it can evoke emotions, convey information, and create visual hierarchy. Figma provides various color tools and features for selecting and managing colors effectively. Designers should understand color theory principles such as complementary colors, analogous colors, and color contrast to create visually cohesive designs.
5. **UI Components and Design Systems:** Figma allows designers to create reusable UI components and design systems, which facilitate consistency and efficiency in high-fidelity design. By using components such as buttons, icons, and input fields, designers can maintain visual coherence throughout the design process and across different screens or projects.

6. **Prototyping and Interactivity:** High-fidelity designs in Figma can include interactive elements and animations to simulate user interactions and demonstrate the flow of the final product. Figma's prototyping features enable designers to create interactive prototypes with transitions, hotspots, and gestures, providing stakeholders and users with a realistic preview of the design's functionality.
7. **User-Centered Design:** Ultimately, high-fidelity design in Figma should prioritize the needs and preferences of the target users. Design decisions should be informed by user research, usability testing, and feedback to ensure that the final product meets user expectations and delivers a positive user experience.

By incorporating these theories and best practices into their design process, designers can create high-fidelity designs in Figma that are visually compelling, functional, and user-friendly.

High-Fidelity Prototyping

High-fidelity prototypes are computer-based, and usually allow realistic (mouse-keyboard) user interactions. High-fidelity prototypes take you as close as possible to a true representation of the user interface. High-fidelity prototypes are assumed to be much more effective in collecting true human performance data (e.g., time to complete a task), and in demonstrating actual products to clients, management, and others.

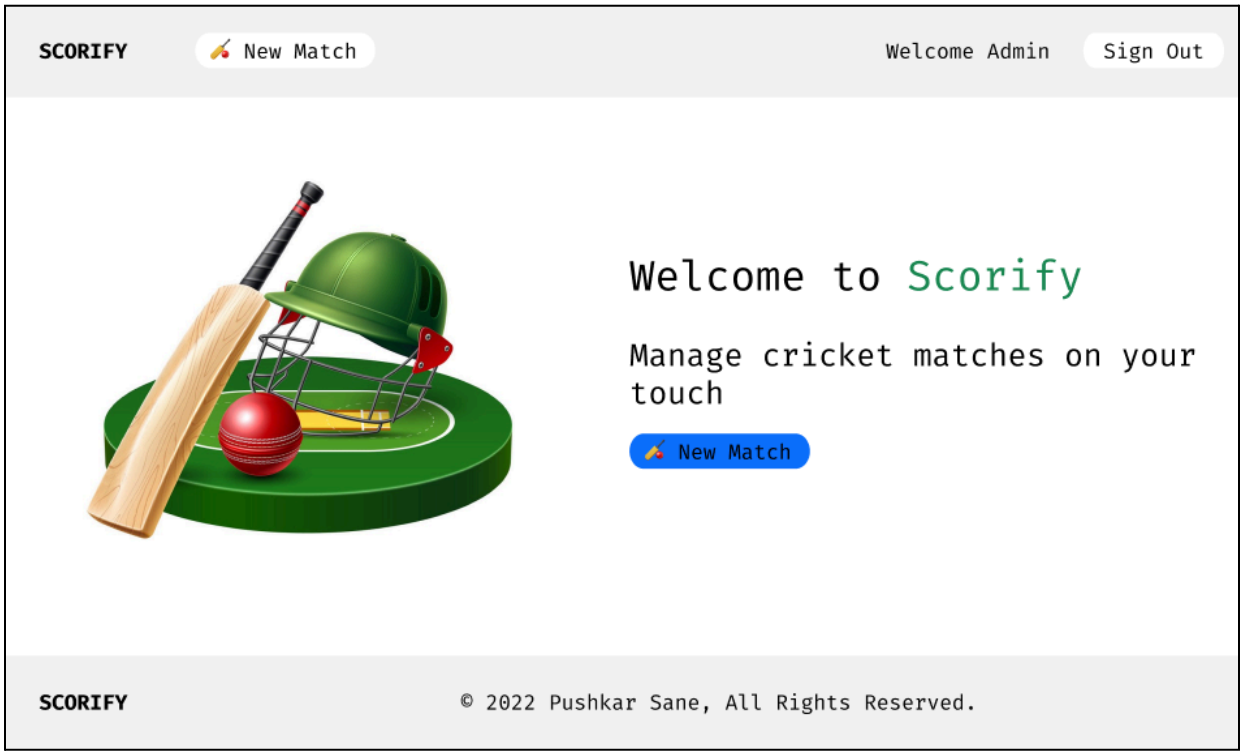
Benefits of high-fidelity prototyping

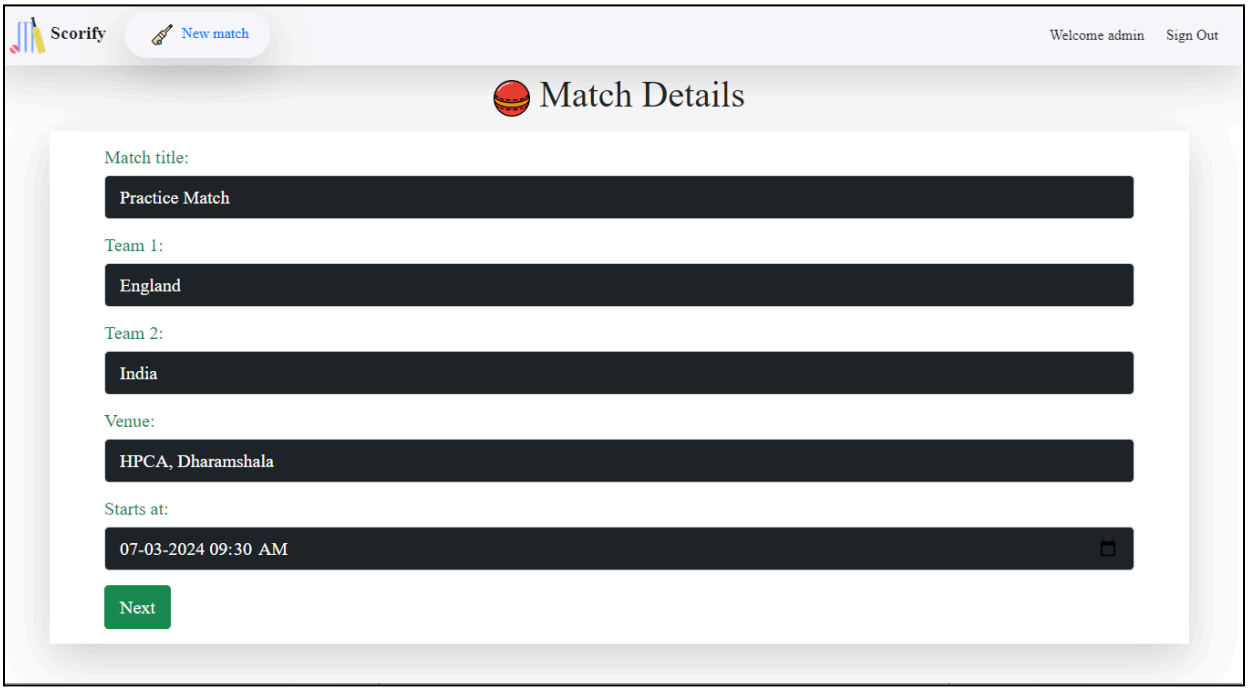
More familiar to users: High-fidelity prototypes look like live software to customers, meaning participants would be more likely to behave naturally during testing.

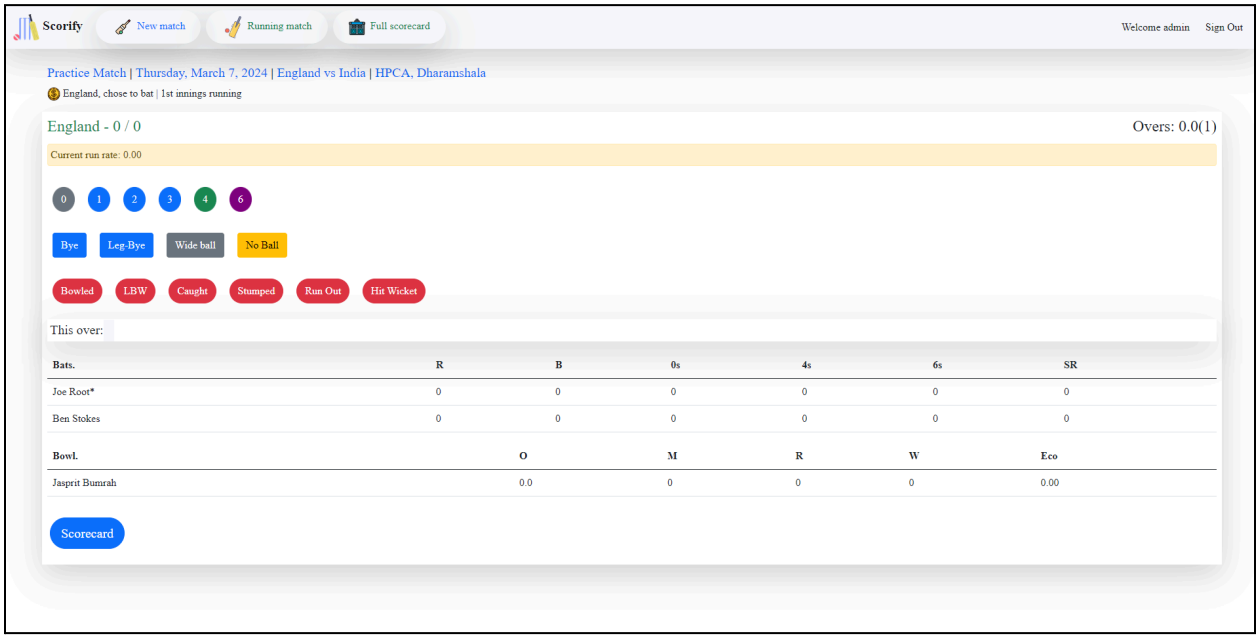
Pinpoint specific components to test: You can dive deep into a single component (like flow, visuals, engagement, or navigation) during user testing. This allows you to get detailed feedback on certain elements of the design that would not be possible with pen and paper.

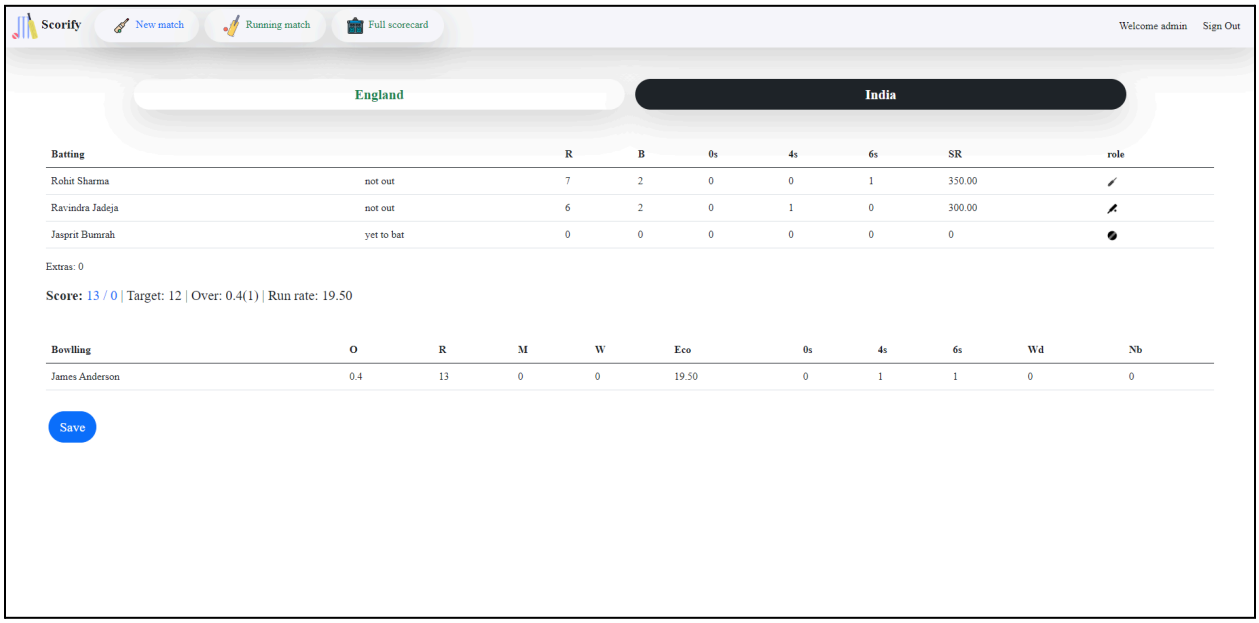
More presentable to stakeholders: Clients and team members will get a clear idea of how the product will look and work before it ever goes live. You can also set clear expectations with developers in the early stages on how much time will be needed to build your prototype and have a finished product.

High Fidelity Prototype:









Conclusions: We have successfully created a High Fidelity (Wire Frame) in this practical.

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 9**	
Title of Lab Assignment: Usability Evaluation of the Design, Testing of User Interface from Third Party (Test scripts) Output Test Script.		
DOP: 04-03-2024	**DOS: 09-03-2024**	
CO Mapped: CO6	**PO Mapped: PO3, PO5, PO7, PO12, PSO1, PSO2**	**Signature:**

Practical No. 9

Aim: Usability Evaluation of the Design. Testing of User Interface from Third Party (Test scripts).

Description:

What is Test Script?

A test script is a document that outlines a series of steps and actions to be taken during software testing. It serves as a guide for testers to systematically verify the functionality of a software application or system. Test scripts are typically created based on test cases or test scenarios and are used to execute tests in a structured manner. Here's how test scripts are used:

Test Execution: Test scripts are used to execute tests on the software under test. Testers follow the steps outlined in the script to interact with the software and verify its behavior.

Repeatability: Test scripts ensure repeatability in testing. By following the same steps each time, the test is executed, testers can ensure consistent results and easily reproduce any issues encountered.

Documentation: Test scripts serve as documentation of the testing process. They provide a record of the tests performed, including the steps taken and the expected results, which can be useful for auditing purposes or for future reference.

Automation: Test scripts can be automated to expedite the testing process. Automated test scripts allow for the execution of tests without manual intervention, saving time and effort, especially for repetitive tests or regression testing.

Collaboration: Test scripts facilitate collaboration among team members involved in testing. They provide a standardized format for documenting tests, making it easier for team members to understand and follow the testing process.

Debugging: Test scripts can help identify and debug issues in the software. If a test fails to produce the expected results, testers can review the steps outlined in the script to identify where the problem occurred and troubleshoot accordingly.

Validation: Test scripts help validate that the software meets the specified requirements and functions correctly according to user expectations. By systematically testing the software against predefined criteria, test scripts help ensure the quality and reliability of the software.

1. **Goal:**
    a. Understanding User Experience
    b. Delve into the users' mindset, motivations, and behaviors when interacting with the website.
    c. Explore how users perceive the website's layout, content, and functionality.
    d. Analyze how well the website meets users' needs and expectations, particularly in accessing farming equipment rental, land rental, and government schemes.
    e. Identifying Pin Points.
    f. Pinpoint specific areas of frustration, confusion, or inefficiency in the user experience related to the core services offered by the website.
    g. Assess the impact of usability issues on users' ability to accomplish tasks related to farming, land rental, and accessing government schemes.
    h. Prioritize identified pain points based on severity and frequency of occurrence to focus improvement efforts.

2. **Gathering Feedback:**
    a. Solicit feedback from users through various channels, including surveys, interviews, and usability testing sessions, specifically targeting the usability of the services offered.
    b. Encourage users to provide both positive feedback and constructive criticism on their experience with farming equipment rental, land rental, and government scheme services.
    c. Actively listen to users' suggestions for improvement and incorporate actionable feedback into future iterations of the website's services.
    d. Improving User Satisfaction:
    e. Measure user satisfaction with the specific services offered by the website through subjective ratings, feedback forms, and Net Promoter Score (NPS) surveys.
    f. Identify factors that contribute to user satisfaction in each service area, such as ease of use, clarity of information, and responsiveness of support.
    g. Implement changes aimed at enhancing user satisfaction and fostering long-term engagement with the website's services.

3. **Enhancing Task Completion:**
    a. Evaluate the effectiveness of the website's services in facilitating task completion for users across different scenarios and use cases.
    b. Identify barriers or obstacles that hinder users from completing tasks related to farming, land rental, and accessing government schemes efficiently.
    c. Streamline service workflows, remove unnecessary steps, and optimize user interfaces to improve task completion rates and user satisfaction.
    d. Benchmarking Performance:
    e. Compare the performance of the website's services against industry benchmarks and best practices, specifically focusing on the usability of farming equipment rental, land rental, and government scheme services.
    f. Benchmark against competitor websites offering similar services to identify areas of competitive advantage and areas needing improvement.
    g. Establish key performance indicators (KPIs) for service usability and track progress over time to measure the impact of usability improvements.

Activities:

1. **Task Analysis:**
    a. Conduct in-depth task analysis to identify the specific goals and subtasks that users need to accomplish when utilizing the website's services.
    b. Analyze the user journey for each service, from initial exploration to task completion, to understand users' behaviors, needs, and pain points.
    c. Document task flows and user journeys for farming equipment rental, land rental, and accessing government schemes to visualize the sequence of steps involved.
    d. User Testing:
    e. Recruit a diverse group of participants representing the target audience for each service, including farmers, agricultural workers, landowners, and government officials.
    f. Develop realistic scenarios and tasks that simulate common use cases and user workflows for each service area.
    g. Capture user interactions and feedback through observation, screen recording, and post-task interviews tailored to each service.

2. **Heuristic Evaluation:**
    a. Engage usability experts or UX professionals to conduct a heuristic evaluation of the website's services.
    b. Evaluate the usability of farming equipment rental, land rental, and government scheme services against established usability principles and best practices.
    c. Generate a list of usability issues and recommendations for improvement based on the evaluation findings for each service area.

3. **Surveys and Interviews:**
    a. Design and administer surveys specific to each service area to gather quantitative data on user satisfaction, ease of use, and perceived usability.
    b. Conduct one-on-one interviews with users who have utilized each service to gain deeper insights into their experiences, preferences, and pain points.
    c. Use open-ended questions to encourage participants to provide detailed feedback and suggestions for improvement related to farming equipment rental, land rental, and government scheme services.

4. **Quantitative Analysis:**
    a. Utilize analytics tools to collect quantitative data on user behavior and service usage patterns for farming equipment rental, land rental, and government schemes.
    b. Analyze metrics such as task completion rates, time to complete tasks, and user engagement with each service to identify areas for improvement.
    c. Use A/B testing to compare different design variations and service features and measure the impact on user satisfaction and task completion rates.

5. **Competitor Analysis:**
    a. Evaluate the usability of competitor websites offering similar services in the agricultural sector to benchmark against industry standards.
    b. Compare key usability metrics, such as service availability, ease of use, and customer satisfaction, for farming equipment rental, land rental, and government scheme services.

c. Identify strengths and weaknesses of competitor services to inform usability improvements for each service area offered by "Scorify."

Procedure for Test Script:

1. **Introduction:**
    a. Provide a detailed overview of the usability evaluation process specific to each service area offered by the website.
    b. Explain the purpose, objectives, and scope of the test script in guiding participants through the usability testing sessions for farming equipment rental, land rental, and accessing government schemes.
    c. Set expectations for participants regarding their role, tasks, and responsibilities during the testing sessions for each service area.

2. **Task Instructions:**
    a. Clearly define the tasks that participants will be asked to perform for each service area, providing step-by-step instructions and specific goals to achieve.
    b. Tailor task instructions to reflect common user scenarios and workflows related to farming equipment rental, land rental, and government scheme services.
    c. Clarify any terminology or concepts that may be unfamiliar to participants to ensure a common understanding of the tasks.

3. **Scenario Setup:**
    a. Present participants with realistic scenarios and use cases for each service area, providing context and background information to set the scene.
    b. Customize scenarios to align with the goals and needs of users engaging with farming equipment rental, land rental, and government schemes.
    c. Ensure scenarios are relevant and engaging to participants, capturing a range of user experiences and challenges for each service area.

4. **Navigation Guidance:**
    a. Offer guidance on how to navigate the website and locate relevant information or features specific to each service area.

b. Provide tips and suggestions for efficient navigation tailored to farming equipment rental, land rental, and accessing government schemes.
c. Encourage participants to think aloud and verbalize their thought process as they navigate through each service area of the website.

5. Data Collection:

a. Outline the methods and tools used for data collection during the testing sessions for farming equipment rental, land rental, and government schemes.
b. Specify how observations, user feedback, and performance metrics will be captured and recorded for each service area.
c. Ensure participants understand the importance of providing honest and constructive feedback to improve the usability of each service.

6. Debriefing:

a. Conclude each testing session with a debriefing to gather additional feedback and insights from participants specific to farming equipment rental, land rental, and government schemes.
b. Encourage participants to reflect on their experiences with each service area and share any thoughts or suggestions for improvement.
c. Thank participants for their time and contribution to the usability evaluation process for each service offered by "Scorify."

7. Documentation:

a. Document the findings, observations, and recommendations from the usability testing sessions specific to farming equipment rental, land rental, and government schemes.
b. Summarize key findings, usability issues, and actionable recommendations for improvement for each service area.
c. Provide supporting evidence, such as quotes from participants, screenshots, and performance metrics, to substantiate findings for each service offered by the website.

Test Script:

1. **User Authentication:**

   Scenario:

   A new user wants to register and log in to the cricket scoring website.

   Task:

   1. Navigate to the registration page.
   2. Fill out the registration form with valid details.
   3. Submit the form and create a new account.
   4. Log in using the newly created credentials.

   Evaluation Criteria:

   1. Successful registration without errors.
   2. Proper validation of user input fields.
   3. Successful login with the newly created credentials.
   4. User redirected to the correct page after login.

2. **Match Creation:**

   Scenario:

   A scorer needs to create a new match entry on the website.

   Task:

   1. Log in to the website as a scorer.
   2. Navigate to the match creation section.
   3. Fill out the match details form with accurate information.
   4. Submit the form to create the match entry.

   Evaluation Criteria:

   1. Successful navigation to the match creation section.
   2. Proper validation of match details input fields.
   3. Match entry created and listed correctly on the dashboard.
   4. Confirmation message or notification indicating successful creation.

3. **Match Scoring:**

   Scenario:

   A scorer needs to record ball-by-ball updates for an ongoing cricket match.

   Task:

   1. Log in to the website as a scorer.
   2. Select the ongoing match from the dashboard.
   3. Access the match scoring interface.
   4. Record ball-by-ball updates accurately as per match events.
   5. Save the scoring data.

   Evaluation Criteria:

   1. Successful access to the match scoring interface.
   2. Proper display and arrangement of scoring fields.
   3. Accurate recording of ball-by-ball updates.
   4. Saving the scoring data without errors.

4. **Live Score Display:**

   Scenario:

   A viewer wants to check live scores and match details on the website.

   Task:

   1. Access the website as a viewer.
   2. Navigate to the live scores section or homepage.
   3. Select an ongoing match to view detailed match information.
   4. Verify the accuracy of live scores, run rates, and match status.

   Evaluation Criteria:

   1. Proper navigation to the live scores section.
   2. Correct display of live scores and match details.
   3. Real-time updates of scores without manual refresh.
   4. Accuracy of match information compared to real-world events.

By conducting comprehensive usability evaluations specific to each service offered by "Scorify," the website can identify and address usability issues, prioritize improvements, and enhance the overall user experience for its target audience.

Project Name: Scorify	
Test Case Template	
Test Case ID: Scorify Web	**Test Designed By:** Pushkar Sane
Module Name: Scoring	**Test Designed Date:** 08-03-2024
Test Title: Verify website functionality	**Test Executed By:** Pushkar Sane
Description: Test the Website	**Test Execution Date:** 08-03-2024
Pre-Conditions: User must be able to perform activities	
Dependencies: N.A	

Test Cases:

Test Case No.	Test case Description	Test Case	Expected Output	Actual Output	Remark
1.	Register for user	Name: Pushkar Email id: pushkar@gmail.com Create password: pushkar@ Re-enter password: pushkar@	User has been registered successfully.	User has been registered successfully.	Pass

2.	Register for user	Name: Pushkar Email id: Pushkar.gmail.com Create password: pushkar@ Re-enter password: pushkar@	Please enter valid email id	Please enter valid email id	Pass
3.	Register for user	Name: Pushkar Email id: pushkar@gmail.com Create password: pushkar@ Re-enter password: pushkar	Passwords don’t match	Passwords don’t match	Pass
4	Login for user	Username: Pushkar Password: pushkar	Please enter the correct password!	Please enter the correct password!	Pass
5.	Login for user	Username: pushkar Password: pushkar@	Please enter the correct username!	Please enter the correct username!	Pass
6.	Login for user	Username: Pushkar Password: pushkar@	Redirects users to the dashboard.	Redirects users to the dashboard.	Pass
7.	Create Match	Click on the button	User should get redirect to team creation page	User get redirect to team creation page	Pass

8.	Team Creation	Team-A Name: India Team-B Name: Australia Venue: Hyderabad	User should get redirect to add player page	User get redirect to add player page	Pass
9.	Team Creation	Team-A Name: India Team-B Name:	Team name cannot be blank	Team name cannot be blank	Pass
10.	Team Creation	Team-A Name: India Team-B Name: India Venue: Hyderabad	Names of the teams cannot be the same.	Names of the teams cannot be the same.	Pass
11	Start Match	Click on the button	Users should get redirected to the Scoring page and display entered details.	Users get redirected to the Scoring page and display entered details.	Pass
12.	Scoring	Click on wide	Should add 1 run in the batting team and +1 in extras.	Adds 1 run in batting team and adds 1 ball	Fail
13.	Scoring	Click on no-ball	Should add 1 run in the batting team and +1 in extras.	Add 1 run in the batting team and +1 in extras.	Pass
14.	Scoring	Click on 1-run	Add 1 run in striker runs and 1 run in bowler’s run.	Add 1 run in striker runs and 1 run in bowler’s run.	Pass

15.	Scoring	Click on 4-runs	Add 4 runs in striker runs, 4 runs in bowler's runs and +1 in batsman's 4's column.	Add 4 runs in striker runs, 4 runs in bowler's runs	Fail
16.	Scoring	Click on 0	Add 1 ball to the batsman and bowler.	Add 1 ball to the batsman and bowler.	Pass
17.	Scoring	Click on Wicket	End the inning of batsman, add 1 wicket in batting team as well as bowler.	End the inning of the batsman, add 1 wicket to the batting team.	Fail
18.	Scoring	Selects new batsman	Display on scorecard.	Displays batsman on scorecard	Pass
19.	Scoring	Selects new bowler	Display on scorecard.	Display bowler on scorecard.	Pass
20.	Match	Selects same batsman on strike and non-strike	Both batsmen cannot be same	Redirects to scoring page	Fail

21.	Scoring	Click on Bowled	Add 1 wicket to the batting team and display prompt.	Prompt displayed but no wicket was added to the batting team	Fail
22.	Scoring	Click on Stumped	Add wicket, add bowl played in batsman profile.	No wicket was added. No bowl was added.	Fail
23.	Scoring	Click on 0	Add 1 ball to the batsman and bowler.	Add 1 ball to the batsman and bowler.	Pass
24.	Home	Click on sign-out	Users should get signed-out of their account.	User should log-out of their account and redirect to home page	Pass

Conclusion: We have successfully implemented Usability Evaluation of the Design Testing of User Interface from Third Party (test scripts).

Name of Student: Pushkar Sane		
Roll Number: 45	**Lab Assignment Number: 10**	
Title of Lab Assignment: Comparative study of user interface design between Flipkart and Amazon website.		
DOP: 04-03-2024	**DOS: 09-03-2024**	
CO Mapped: **CO1, CO2, CO3, CO4, CO5, CO6**	**PO Mapped:** **PO3, PO5, PO7, PO12, PSO1, PSO2**	**Signature:**

Practical No. 10

Aim: Comparative study of user interface design between Flipkart and Amazon website.

A Comparative Analysis of Flipkart and Amazon User Interface Design

Homepage Layout:

Both Flipkart and Amazon feature a clean and organized layout.

- Flipkart tends to have a more colorful and vibrant design with large banners showcasing deals and promotions.
- Amazon's homepage focuses more on product recommendations based on user history and preferences, with a simple layout and minimalistic design.

Navigation:

Both websites utilize a top navigation bar for easy access to key categories such as electronics, fashion, books, etc.

- Flipkart often employs a sidebar menu for quick navigation to specific product categories, while Amazon uses a dropdown menu or a hamburger menu for similar functionality.
- Amazon's navigation tends to be more hierarchical, offering multiple layers of subcategories for deeper exploration, while Flipkart may prioritize direct access to popular categories.

Search Functionality:

Both platforms feature a prominent search bar for users to quickly find products.

- Amazon's search functionality is highly sophisticated, often incorporating predictive text, autosuggestions, and filters to refine search results.
- Flipkart's search function is also robust but may emphasize recent searches and popular products more prominently.
-

Product Listings:

Both websites display product listings in a grid format, showcasing product images, titles, prices, and ratings.

- Flipkart may emphasize discounts and deals more prominently within product listings.

- Amazon typically offers more extensive product information, including detailed descriptions, specifications, and customer reviews.

Cart and Checkout Process:

Both platforms have a clear and intuitive cart icon indicating the number of items added.

- Amazon's checkout process is known for its simplicity and efficiency, offering multiple payment options and one-click purchasing for registered users.
- Flipkart's checkout process may involve more steps, including additional offers or promotions presented during the checkout journey.

Responsive Design:

Both websites are optimized for various devices, including desktops, tablets, and smartphones.

- Amazon's responsive design is known for its consistency across different screen sizes and resolutions.
- Flipkart also prioritizes mobile responsiveness, ensuring a seamless shopping experience on smaller screens.

Visual Design:

- Flipkart's design tends to be more visually engaging, with bright colors and bold typography.
- Amazon adopts a more minimalist approach, focusing on clarity and functionality over ornate design elements.

Personalization:

Both platforms utilize user data to personalize the shopping experience, including recommended products and targeted promotions.

- Amazon's personalization algorithms are highly sophisticated, often leading to more accurate product recommendations based on user behavior and purchase history.
- Flipkart also offers personalized recommendations but may prioritize deals and discounts alongside product suggestions.

Feedback Mechanisms:

Both platforms provide feedback mechanisms to inform users about the status of their actions. For example, when adding an item to the cart, users receive confirmation messages or animations.

- Amazon's feedback mechanisms are often more subtle, with changes in button states or progress indicators to signify actions being processed.
- Flipkart may use more prominent animations or pop-up messages to provide feedback, enhancing the user's sense of engagement.

Accessibility:

Both Flipkart and Amazon prioritize accessibility, ensuring their websites are usable by people with disabilities.

- Amazon adheres to WCAG (Web Content Accessibility Guidelines) standards more closely, offering features like alternative text for images, keyboard navigation, and screen reader compatibility.
- Flipkart also provides accessibility features but may not be as comprehensive as Amazon in terms of adherence to WCAG guidelines.

Social Integration:

Both platforms integrate social elements to enhance the shopping experience. This includes features like user reviews, social sharing options, and integration with social media platforms.

- Amazon's user reviews are often more detailed and prominent, influencing purchasing decisions. Additionally, Amazon integrates social login options and allows users to share products on various social media platforms.
- Flipkart similarly incorporates user reviews and ratings but may also leverage social proof through features like trending products or influencer endorsements.

Conclusion:

In conclusion, the comparative analysis of Flipkart and Amazon's user interface design highlights both similarities and distinctions in their approaches to engaging users and facilitating online shopping experiences. While both platforms prioritize functionality, accessibility, and personalization, Flipkart tends to emphasize vibrant visuals and direct promotions, while Amazon focuses on clarity, efficiency, and extensive personalization. Understanding these

nuances provides valuable insights into the strategies employed by these e-commerce giants to cater to diverse user preferences and enhance overall user satisfaction.