Session 6

**Test Design Techniques**

* It is used to prepare data for testing.
* So that each and every functionality could be checked.
* By using we can take **minimal data to gain maximum coverage**.

Types of Test Design Techniques

1. Equivalence Class Partitioning (ECP)

* Here we make **partition** of **data** into various **classes** and choose any random number from that class so that entire range of class could be verified by **checking a value**.
* **Ex. of a text box whose range should be in between 1 to 500 or like only number, alphabets etc.**

1. Boundary Value Analysis (BVA)

* It is used to **check the boundaries** of the input
* Here we use certain points like
  + Min, min-1 and min+1
  + Max, max-1 and max+1
* By using which we could reduce the data to be tested.
* **Ex. age between 20-45**

Above two techniques are used as a part of **Input domain testing**

* In which value will be verified in text box/input fields.

1. Decision Table

* Also called as **Cause-Effect table**.
* It is taken into consideration when we’re having more **conditions** and based on condition, we have to perform the **action**.
* It consists of several conditions and actions and based on that we execute different test cases
* **Ex. Of Banking domain like insufficient balance then no transaction.**

1. State Transition

* Here state is transformed
* **Ex. if you are giving valid input credentials then you’ll be directed to the home page of site. And if multiple time incorrect then locks**.

1. Error Guessing

* It’s purely based on Tester’s experience and analytical skill in the domain.

User flow for Health-Tech:

1. Buying Healthcare Device

Ex. ApolloPharmacy

* User visits [www.apollopharmacy.com](http://www.apollopharmacy.com)
* Selects Health Device from Menu Bar.
* Select on Covid Test Kit
* Select the appropriate product out of all available
* Click on Add to Cart button along with specifying number of quantities required which appears in a list, that pop ups after clicking add to cart button.
* Now click View Cart
* Select Proceed Button
* Choose the payment method displayed in form of radio buttons
* Make payment by entering valid PIN.
* A screen will appear showing “Order Placed”.

1. Telemedicine Consultation

Example: [1mg](https://www.1mg.com/)

* User opens the 1mg app or website.
* Navigates to the "Consult Doctors" section.
* Selects the type of consultation (e.g., General Physician, Paediatrician).
* Views a list of available doctors with their profiles and consultation fees.
* Chooses a doctor and clicks on "Consult Now" or "Schedule for Later."
* Completes payment through the integrated payment gateway.
* Initiates the consultation via chat or video call within the app at the scheduled time.
* Receives a digital prescription post-consultation.

### **Medicine Ordering**

**Example:** [Netmeds](https://www.netmeds.com/)

* User visits the Netmeds website or app.
* Searches for required medicines using the search bar.
* Adds selected medicines to the cart.
* Uploads a prescription if required for certain medications.
* Proceeds to checkout and enters delivery address details.
* Chooses a preferred payment method and completes the payment.
* Receives an order confirmation with estimated delivery time.
* Tracks the order status through the app or website until del

**How to know where different sections of a webpage lie?**

* To know different sections like header, footer, main body, etc.
* We can inspect the page to use developer tools.
* Afterwards we can check the HTML code whether where a specific section lies
* Ex. in alianhub.com to know where is the header we can hover and get to know that <header> tag is in a div having id ‘et-boc’.
* **Also learned basics UI/UX elements.** (Accordian, Carousel)

**Possible Tests on AlianHub:**

* GUI Testing

To check the alignments and different resolutions.

* Usability Testing

To check user preference.

* Object Properties Testing

Ex. whether Buy Now button is working or not

* Error Handling Testing

Ex. Error Messages on login page

* Regression Testing

If some features are modified.

* Re-Testing

For ex. change in Message Icon size doesn’t affects the placement of other icons in the Menu.

* Smoke Testing

To verify whether Login Button is clickable or not if not then no requirement of further testing.

**Different Scenarios in which Time tracker might face issues along with its possible solution:**

* Scenario A: Data Loss During Power Cut or System Crash
  + Solution: Local Cache or Logs
  + If the app uses local storage (e.g., files, database, or cache), verify if the last saved state of the timer can be recovered.
  + Ensure that the app saves data periodically (e.g., every minute) during active tracking.
* Scenario B: Manual Adjustments
  + Solution: Manual Time Entry
  + Allow users to manually input the lost time.
  + Provide a reason field (e.g., "Power cut at [time]").
* Scenario C: Server-Side Sync Issues
  + Solution: Resync Mechanism
  + If the app syncs data to the server, implement retry mechanisms to handle unsynced data during power failures or disconnections.

**Use Event Listeners for Power or System Shutdowns**

If the app runs on a system with event listeners (like desktop apps), detect events like shutdowns, sleep mode, or power cuts.

Example:

Javascript code:

window.onbeforeunload = () => {

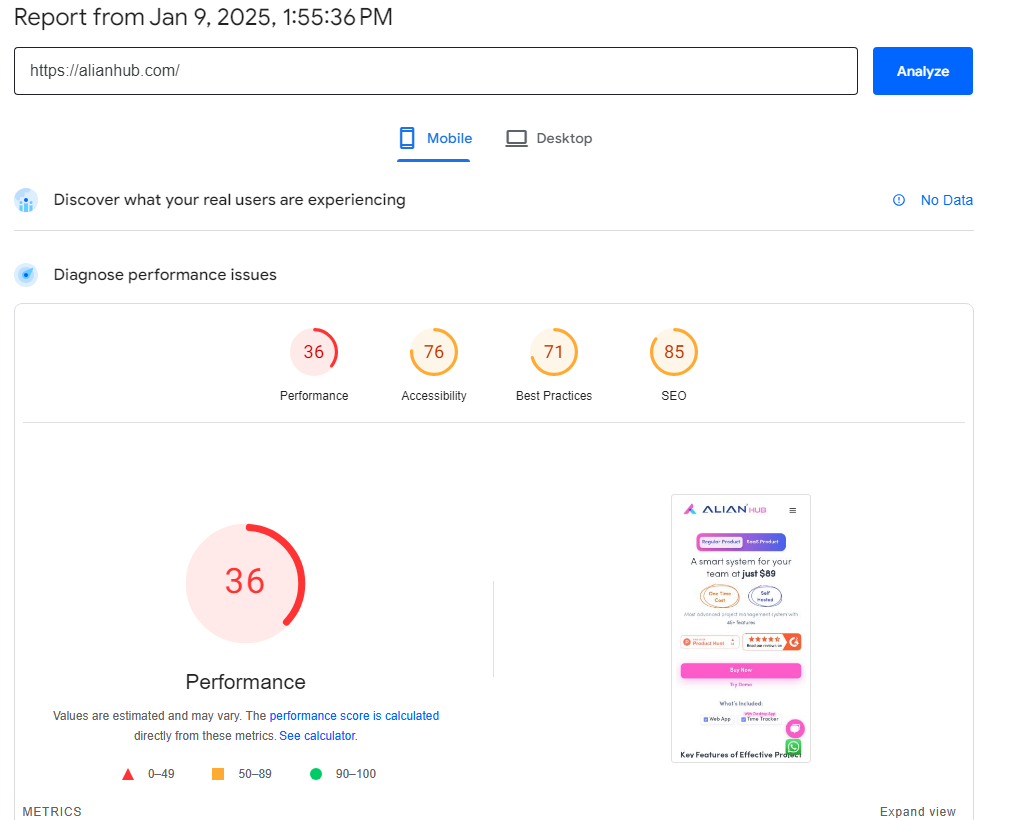
saveCurrentTrackingState ();

};

**Manual Test Case Table for AlianHub Homepage**

| **Test Case ID** | **Test Scenario** | **Test Steps** | **Expected Result** | **Actual Result** | **Status** |
| --- | --- | --- | --- | --- | --- |
| TC\_FUNC\_01 | Verify the "Try Demo" button functionality | 1. Navigate to [alianhub.com](https://alianhub.com/).2. Click on the "Try Demo" button.3. Observe the page redirection. | Clicking the "Try Demo" button should redirect the user to the demo page. | As expected. | Passed |
| TC\_PERF\_01 | Assess homepage loading time | 1. Navigate to [alianhub.com](https://alianhub.com/).2. Measure the time taken for the homepage to fully load. | The homepage should load within 3 seconds for optimal user experience. | The homepage takes approximately 5 seconds to load, which may affect user engagement. | Failed |
| TC\_ACC\_01 | Check for alt text on homepage images | 1. Navigate to [alianhub.com](https://alianhub.com/).2. Inspect the HTML code of the homepage images to verify the presence of alt text attributes. | All images should have descriptive alt text for accessibility purposes. | Several images lack alt text, which may hinder accessibility for users with visual impairments. | Failed |
| TC\_SEC\_01 | Verify the presence of an SSL certificate | 1. Navigate to [alianhub.com](https://alianhub.com/).2. Check for the presence of "https://" in the URL and a padlock icon in the browser's address bar. | The website should have a valid SSL certificate, indicated by "https://" and a padlock icon, ensuring secure connections. | As expected. | Passed |
| TC\_BRO\_01 | Test homepage compatibility on different browsers | 1. Access [alianhub.com](https://alianhub.com/) using various browsers (e.g., Chrome, Firefox, Safari, Edge).2. Observe the layout and functionality of the homepage on each browser. | The homepage should display consistently and function correctly across all major browsers. | The homepage displays and functions correctly on Chrome and Firefox but has layout issues on Safari and Edge. | Failed |
| TC\_CONT\_01 | Check for spelling and grammatical errors on the homepage | 1. Navigate to [alianhub.com](https://alianhub.com/).2. Read through all the text content on the homepage.3. Identify any spelling or grammatical errors present in the content. | The homepage should be free of spelling and grammatical errors to maintain professionalism and clarity. | The phrase "Kickstart your Business with our SaaS Solution at Just $349" should have "Just" in lowercase ("just"). | Failed |

Page not loading :



Some images ain’t havin alt texts

For e.x. Linux icon

How to test a website?

* **First** of all, perform **Exploratory** test
* Then Analyze different menus and modules

For ex. in an E-Commerce website check who could login because may have different personas like Admin, User, Premium User, Seller.

* If we find a bug in UI/UX then it comes under GUI Testing where we check different alignments, colors, etc.
* Let’s take an example, while performing **system testing** it was found that platform fee was not been added in the invoice even though it showing on the Payment gateway. (we can consider it comes under a **calculation/manipulation test**)
* After receiving an updated build, we can perform **Regional Regression Testing** to ensure that the Add to Cart, Payment, Booked Order’s invoice modules are defect free or not.