## EJALAS_TESTING

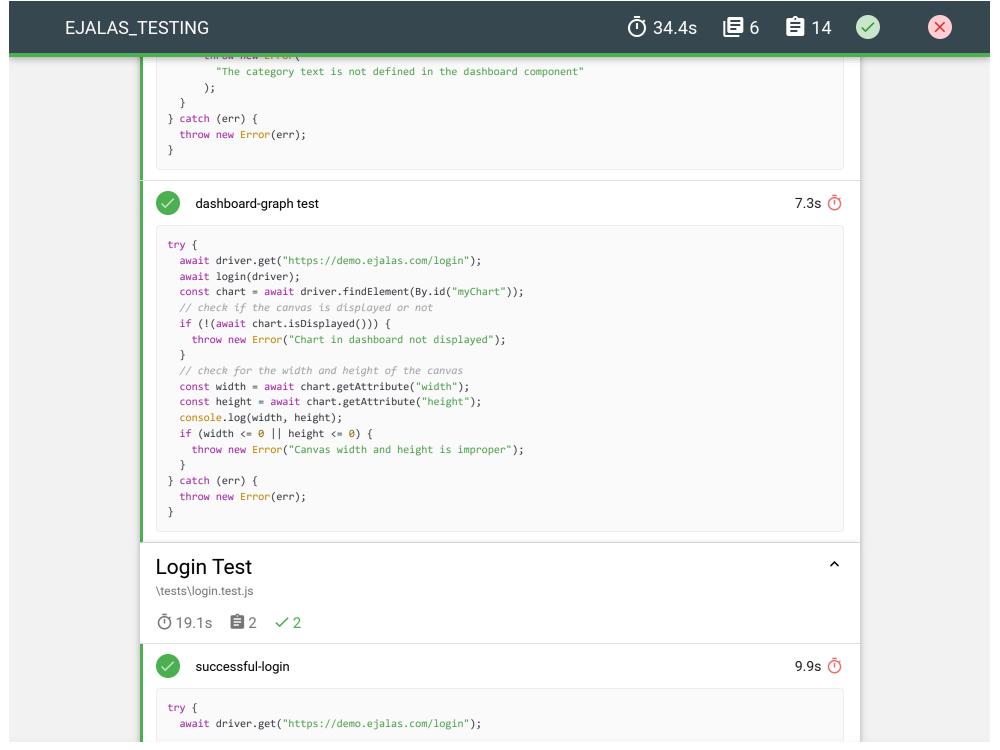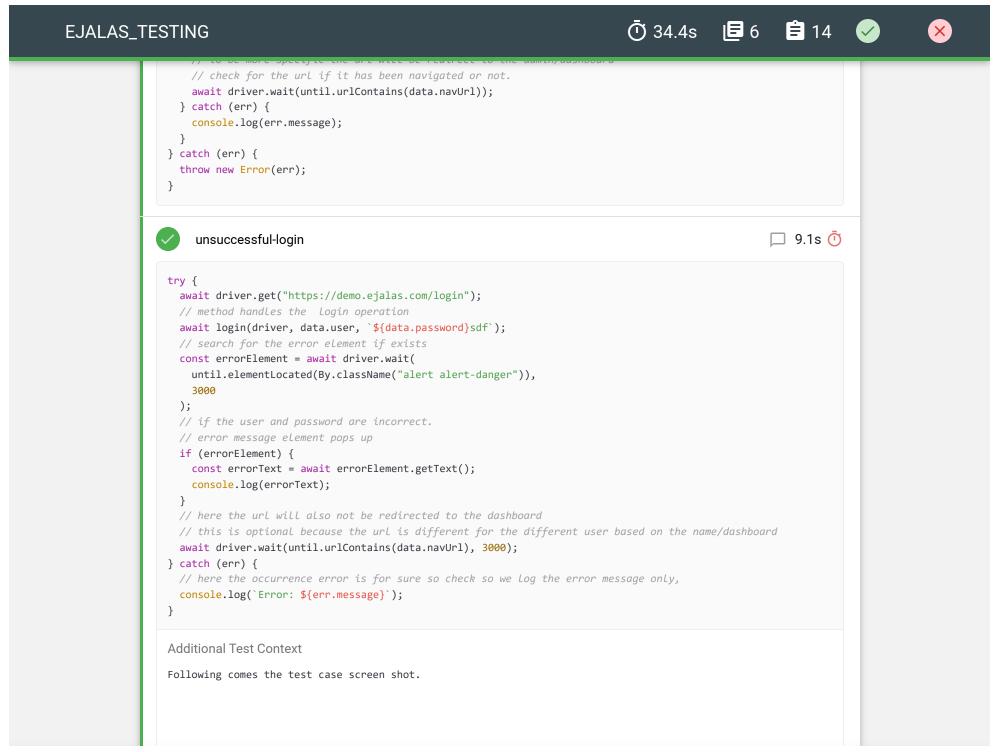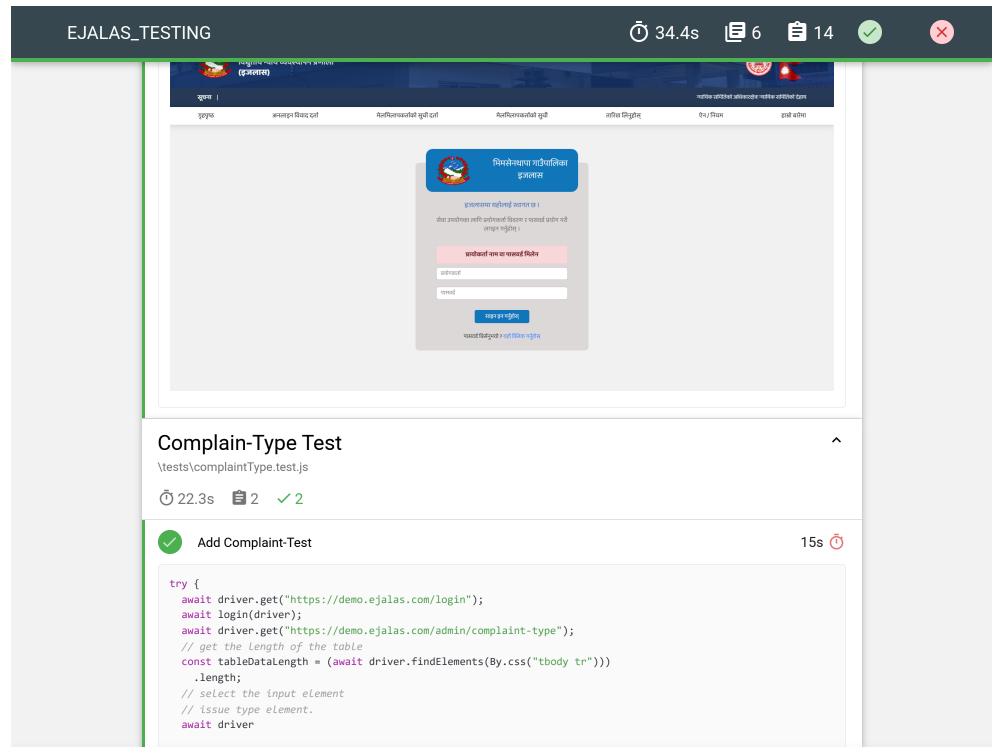⏱ 34.4s    📰 6    📋 14    ✅ 14    ❌ 0

### Conciliator test ⌃

\tests\conciliatorRegister.test.js

⏱ 18.1s    📋 1    ✓ 1

✅ conciliator-inspection-verify test      18.1s ⏱

```
try {
  driver.get("https://demo.ejalas.com/login");
  // login
  await login(driver);
  // navigate to the url
  await driver.get("https://demo.ejalas.com/admin/melmilapkarta-darta");
  // target the table data
  const targetElement = await driver.findElement(
    By.css(`tbody tr:first-child`)
  );
  // first get the data of the darta that you are going to vew
  // this is the username that we want to vew the content of
  const name = await targetElement.findElement(By.css("td")).getText();
  // now see the darta info by clicking on view/eye icon
  const viewButton = await targetElement.findElement(
    By.css(`:last-child a`)
  );
  await driver.executeScript("arguments[0].click();", viewButton);
  //now target the table and get the text and note the text should not be empty
  // in most case, we consider the test that does contains the info in the table
  const tableElement = await driver.wait(
    until.elementLocated(By.css("tbody tr:first-child"))
  );
  const tableName = await tableElement.findElement(By.css("td")).getText();
  if (name != tableName)
    throw new Error(
      "The user name that we viewed the content of didn't match"
    );
  if (name == tableName) {
    // click reject or approved we check for the reject
    const btn = await driver.findElement(
      By.css(".buttons-container a:last-child")
```

**EJALAS_TESTING**          ⏱ 34.4s   📑 6   📋 14   ✓   ✗

```
// target the element that we clicked previously
    await driver.sleep(2000);
    const targetElement = await driver.wait(
      until.elementLocated(By.css(`tbody tr:first-child`))
    );
    const isChanged = await targetElement
      .findElement(By.css(`td:nth-child(7)`))
      .getText();
    if (isChanged != "अस्वीकार गरियो")
      throw new Error("conciliator registered was not rejected");
  }
} catch (err) {
  throw new Error(err);
}
```

## DashBoard-Test                                              ⌃

\tests\dashboard.test.js

⏱ 17.5s   📋 2   ✓ 2

✓   dashboard-component test                              10.2s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
  await login(driver);
  // checking the dashboard content
  const contentElement = await driver.findElements(
    By.className("number-of-cases-item")
  );
  for (content of contentElement) {
    const icon = await content.findElement(By.xpath(".//i")).isDisplayed();
    const categoryText = await content.findElement(By.xpath(".//h3"));
    const caseQuantity = await content.findElement(By.xpath(".//span"));
    // test for the icon render in the component dashboard.
    if (!icon)
      throw new Error("Error: component icon not rendered in dashboard");
    // number of cases cannot be negative.
    if ((await Number(caseQuantity)) <= 0 || caseQuantity == "")
      throw new Error(
        "The number defined in the dashboard component is not displayed or inappropriate format"
```

**EJALAS_TESTING**    ⏱ 34.4s    📑 6    📋 14    ✓    ✕

```
         "The category text is not defined in the dashboard component"
    );
  }
} catch (err) {
  throw new Error(err);
}
```

✓  dashboard-graph test                                        7.3s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
  await login(driver);
  const chart = await driver.findElement(By.id("myChart"));
  // check if the canvas is displayed or not
  if (!(await chart.isDisplayed())) {
    throw new Error("Chart in dashboard not displayed");
  }
  // check for the width and height of the canvas
  const width = await chart.getAttribute("width");
  const height = await chart.getAttribute("height");
  console.log(width, height);
  if (width <= 0 || height <= 0) {
    throw new Error("Canvas width and height is improper");
  }
} catch (err) {
  throw new Error(err);
}
```

## Login Test                                                       ⌃

\tests\login.test.js

⏱ 19.1s    📋 2    ✓ 2

✓  successful-login                                             9.9s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
```

**EJALAS_TESTING**      ⏱ 34.4s   📖 6   📋 14   ✓   ✕

```
                                                                    // to be more specifie the url will be redirect to the admin/dashboard
    // check for the url if it has been navigated or not.
    await driver.wait(until.urlContains(data.navUrl));
  } catch (err) {
    console.log(err.message);
  }
} catch (err) {
  throw new Error(err);
}
```

✓  **unsuccessful-login**                                    💬  9.1s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
  // method handles the  login operation
  await login(driver, data.user, `${data.password}sdf`);
  // search for the error element if exists
  const errorElement = await driver.wait(
    until.elementLocated(By.className("alert alert-danger")),
    3000
  );
  // if the user and password are incorrect.
  // error message element pops up
  if (errorElement) {
    const errorText = await errorElement.getText();
    console.log(errorText);
  }
  // here the url will also not be redirected to the dashboard
  // this is optional because the url is different for the different user based on the name/dashboard
  await driver.wait(until.urlContains(data.navUrl), 3000);
} catch (err) {
  // here the occurrence error is for sure so check so we log the error message only,
  console.log(`Error: ${err.message}`);
}
```

Additional Test Context

Following comes the test case screen shot.

**EJALAS_TESTING**  ⏱ 34.4s  📰 6  📋 14  ✓  ✕



## Complain-Type Test ⌃

\tests\complaintType.test.js

⏱ 22.3s  📋 2  ✓ 2

✓  **Add Complaint-Test**                                    15s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
  await login(driver);
  await driver.get("https://demo.ejalas.com/admin/complaint-type");
  // get the length of the table
  const tableDataLength = (await driver.findElements(By.css("tbody tr")))
    .length;
  // select the input element
  // issue type element.
  await driver
```

EJALAS_TESTING                    ⏱ 34.4s    📰 6    📋 14    ✓    ✕

```
        .FindElement(By.className( "common court-btn btn btn success" ))
        .click();
    // check the issue if added to the list or not.
    // it will be contained at the last of the table.
    const lastElement = await driver.wait(
      until.elementLocated(By.css("tbody tr:last-child")),
      3000
    );
    console.log(await lastElement.getText());
    // check if the length has been increase in the table if yes then the
    // data has been added
    const newTableDataLength = await driver.findElements(By.css("tbody tr"));
    if (newTableDataLength == tableDataLength)
      throw new Error("The Data was not added");
  } catch (err) {
    throw new Error(err);
  }
}
```

✓    **Edit Complain-Test**                                    7.3s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/admin/complaint-type");
  // target the element that we want to modify
  // target the last table data because it contains the latest data
  const lastElement = await driver.wait(
    until.elementLocated(By.css("tbody tr:last-child"))
  );
  const text = await lastElement.findElement(By.css("td")).getText();
  // click edit btn
  const editBtn = await lastElement.findElement(
    By.css("td:nth-child(3) a")
  );
  await driver.executeScript("arguments[0].click()", editBtn);
  // now do the modification that you want and submit.
  await driver.findElement(By.id("courtName")).sendKeys(" modification");
  // submit
  await driver
    .findElement(By.className("common-court-btn btn btn-success"))
    .click();
  // check if the data has been modified or not
  const modifiedLastElement = await driver.wait(
    until.elementLocated(By.css("tbody tr:last-child"))
```
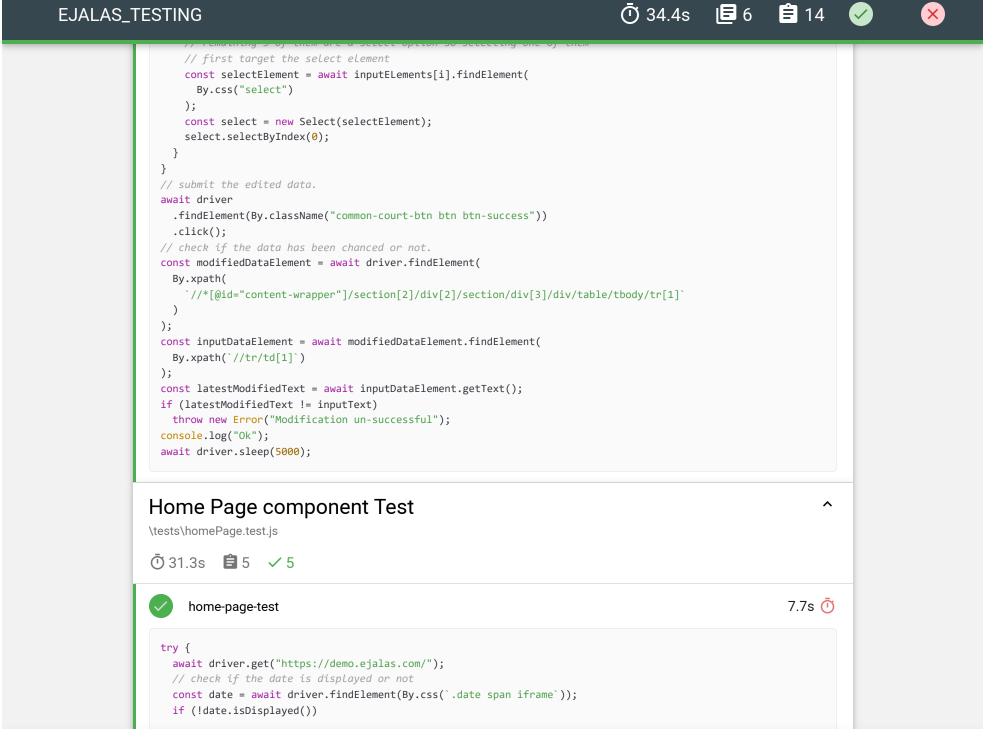
## EJALAS_TESTING

⏱ 34.4s    📖 6    📋 14    ✓    ✕

```
    console.log(text);
    console.log(modifiedText);
    // if modified text and the previous text is same then the data was not updated
    if (modifiedText == text) throw new Error("Data was not modified");
  } catch (err) {
    throw new Error(err);
  }
}
```

## Judiciary Committee Test ⌃

\tests\judiciaryCommittee.test.js

⏱ 28.4s    📋 2    ✓ 2

✓    Addition of judiciary Committee Test      15.5s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/login");
  // login.
  await login(driver);
  await driver.get("https://demo.ejalas.com/admin/court");
  // here the input element is a div tag
  const inputELements = await driver.findElements(By.className("col-md-3"));
  for (let i = 0; i < inputELements.length; i++) {
    // the first one is the input field and other is to be selected from the option element
    if (i == 0) {
      const input = await inputELements[0].findElement(By.css("input"));
      await input.sendKeys(obj[0]);
    } else {
      // remaining 3 of them are a select option so selecting one of them
      // first target the select element
      const selectElement = await inputELements[i].findElement(
        By.css("select")
      );
      const select = new Select(selectElement);
      select.selectByIndex(0);
    }
  }
  // submit the data that we added.
  await driver
```

**EJALAS_TESTING**    ⏱ 34.4s    📰 6    📋 14    ✓    ✗

```
// to check if the element is added or not
const lastElement = await driver.wait(
  until.elementLocated(By.css("tbody tr:last-child")),
  3000
);
// target the tbody to check the length of tr
const tbody = await driver.findElements(By.css("tbody tr"));
const thText = await lastElement.findElement(By.css("th")).getText();
// gets the data in array format
const tdText = (
  await lastElement.findElement(By.css("td")).getText()
).split();
console.log(tdText, thText);
if (tdText[0] != obj[0] && tbody.length != thText) {
  throw new Error(
    "The content didn't match with the input or the index was not matched with the table index"
  );
}
} catch (err) {
  throw new Error(err);
}
```

✓    **Edit Judicial Committee Test**                                    12.8s ⏱

```
await driver.get("https://demo.ejalas.com/admin/court");
//click the edit button
await driver
  .findElement(
    By.xpath(
      `//*[@id="content-wrapper"]/section[2]/div[2]/section/div[3]/div/table/tbody/tr[1]/td[5]/a`
    )
  )
  .click();
// now modify the data
// target the input and change the data.
const inputELements = await driver.findElements(By.className("col-md-3"));
//input text.
const inputText = `test modified ${Math.floor(Math.random() * 10)}`;
for (let i = 0; i < inputELements.length; i++) {
  // the first one is the input field and other is to be selected from the option element
  if (i == 0) {
    const input = await inputELements[0].findElement(By.css("input"));
```

**EJALAS_TESTING**                    ⏱ 34.4s    📖 6    📋 14    ✅    ❌

```
            // remaining 3 of them are a select option so selecting one of them
            // first target the select element
            const selectElement = await inputELements[i].findElement(
              By.css("select")
            );
            const select = new Select(selectElement);
            select.selectByIndex(0);
        }
    }
    // submit the edited data.
    await driver
      .findElement(By.className("common-court-btn btn btn-success"))
      .click();
    // check if the data has been chanced or not.
    const modifiedDataElement = await driver.findElement(
      By.xpath(
        `//*[@id="content-wrapper"]/section[2]/div[2]/section/div[3]/div/table/tbody/tr[1]`
      )
    );
    const inputDataElement = await modifiedDataElement.findElement(
      By.xpath(`//tr/td[1]`)
    );
    const latestModifiedText = await inputDataElement.getText();
    if (latestModifiedText != inputText)
      throw new Error("Modification un-successful");
    console.log("Ok");
    await driver.sleep(5000);
```

## Home Page component Test                                      ^

\tests\homePage.test.js

⏱ 31.3s    📋 5    ✅ 5

✅    home-page-test                                          7.7s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/");
  // check if the date is displayed or not
  const date = await driver.findElement(By.css(`.date span iframe`));
  if (!date.isDisplayed())
```

EJALAS_TESTING   ⏱ 34.4s   📰 6   📋 14   ✓   ✕

```
const marquee = await driver.findElement(
  By.xpath(`//*[@id="header-wrapper"]/div/div[2]/div/marquee`)
);
// first check if the marquee if displayed or not
if (!(await marquee.isDisplayed()))
  throw new Error("The marquee element/news slide is not displayed");
// now check if the text message inside is present or not?
const marqueeText = await marquee.findElement(By.css("ul")).getText();
if (marqueeText == 0)
  throw new Error("The news-slide content is not displayed");
} catch (err) {
  throw new Error(err);
}
```

✓  ujuri-case-search-result-fail-test                    6.3s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/tarikh");
  // target the input to search for the case number
  // assume the case number is 123 which does not exists should give us error message
  await driver.findElement(By.css(`form input`)).sendKeys("123");
  //target the button and search for it.
  // here we should get the error message because we have an unregister case number
  await driver.findElement(By.css("button")).click();
  // check if the error message pop up or not.
  const errorElement = await driver.wait(
    until.elementLocated(By.className("text-center alert alert-danger"))
  );
  const errorText = await errorElement.getText();
  if (errorText != "मुद्दा नं को तारिख उपलब्ध छैन")
    throw new Error("Error in getting the error element");
  console.log(errorText);
} catch (err) {
  throw new Error(err);
}
```

✓  ujuri-case-search-result-success-test                 5.3s ⏱

```
try {
  const code = "sfj";
```

# EJALAS_TESTING

⏱ 34.4s   🗐 6   📋 14   ✅   ❌

```
await driver.findElement(By.css("form input")).sendKeys(code);
//target the button and search for it.
// here we should get the error message because we have an unregister case number
await driver.findElement(By.css("button")).click();
// here the case number is valid so it should contain some information except the error message
const tableElement = await driver.wait(
  until.elementLocated(By.css("tbody tr"))
);
const codeNumber = await tableElement
  .findElement(By.css("td:last-child"))
  .getText();
if (codeNumber != code)
  throw new Error(
    "The expected ujuri result didn't match with the code that we input"
  );
} catch (err) {
  throw new Error(err);
}
```

✅ **home-page about-us**                                    2.7s ⏱

```
try {
  await driver.get("https://demo.ejalas.com/hamro-barema");
  const textContainer = await driver.findElement(
    By.className("hamro-barema-container")
  );
  // check if the container is displayed or not.
  if (!(await textContainer.isDisplayed()))
    throw new Error("The container is not displayed in the page.");
  // check if the text is empty or not.
  const text = await textContainer.getText();
  if (text == "") throw new Error("The content in the page is not shown");
  console.log(await textContainer.getText());
} catch (err) {
  throw new Error(err);
}
```

✅ **conflict look-up test**                                    9s ⏱

EJALAS_TESTING     ⏱ 34.4s   📰 6   📋 14   ✓   ✕

```javascript
};
await driver.get("https://demo.ejalas.com/index");
// first select the type of the type of case
const selectElement = await driver.findElement(
  By.id("inlineFormCustomSelect")
);
const select = new Select(selectElement);
select.selectByVisibleText(obj.select);
// now select the date of register case.
await driver
  .findElement(By.className("nepali-datepicker ndp-nepali-calendar"))
  .sendKeys(obj.date);
//submit the detail
const submitBtn = await driver.findElement(
  By.className("common-case-btn section-content-item")
);
await driver.wait(until.elementIsVisible(submitBtn), 10000);
await driver.executeScript("arguments[0].click();", submitBtn);
// result will show according to the detail provided
// check for the result
const targetElement = await driver.findElements(By.css(`tbody tr`));
// check if the data is present or not it should not be empty
console.log(targetElement.length);
if (targetElement == 0)
  throw new Error("The search result didn't match for the given input");
await driver.sleep(3000);
```