

Unit Testing (Frontend)

[Plans](#)

[Progress + Results](#)

Plans

Objective: UI component tests (checking rendering, user interaction).

Scope:

- Test rendering of individual UI components under different props, states and user interactions
- Most functions will be mocked to focus on the internal logic of components and functions.

Test Cases (UI components):

- **BackButton:** verify button renders correctly based on state
- **DisplayFile:** verify component renders file only when it is uploaded
- **LoadingButton:** verify button does not allow interactions when in loading state
- **LoadingModal:** verify modal renders correctly based on state
- **ProceedButton:** verify user can only interact with button when a valid file is uploaded
- **SkipButton:** verify skip button will only enable when a solution is present in database
- **UploadPopUp:** verify component contains all sub-components and works correctly with simulated user interactions / events

Tools:

- Vitest
- React Testing Library
- JSDom

Acceptance Criteria: Refer to Jira.

Progress + Results

Link: [jetedge/frontend/src/tests/components at develop · hotungkhanh/jetedge](https://github.com/hotungkhanh/jetedge/tree/develop/frontend/src/tests/components)

There are multiple test cases implemented per UI component.

- **BackButton component:**
 - Renders as an active button by default
 - Renders as a disabled button given the appropriate props
- **DisplayFile component:**
 - Renders an empty Box when fileChosen is null
 - Renders the file name and FilePresentIcon when fileChosen is not null
- **LoadingButton component:**
 - Renders a disabled button with circular loading progress and no text when in loading state
 - Renders an enabled button with specified text when not in loading state
- **LoadingModal component:**
 - Renders the modal with a circular loading progress when it is set to opened
 - Does not render the modal or a circular loading progress when open is set to false

- **ProceedButton component:**
 - Renders a disabled button when no files have been uploaded
 - Renders an enabled button when a file have been uploaded
 - Triggers navigation given the user click this component
- **SkipButton component:**
 - Renders a disabled button when there are no timetable solutions found
 - Renders an enabled button when a timetable solution is found
 - Triggers navigation given the user click this component
- **UploadPopUp component:**
 - Renders the "Generate Timetable" button
 - Opens the upload file modal when "Generate Timetable" button is clicked
 - Closes the upload file modal when user click outside
 - Renders a disabled ProceedButton when no file is uploaded
 - Renders a disabled ProceedButton when an invalid file is uploaded