

PASSING DATA BETWEEN COMPONENTS

IIHT

Time To Complete: 10 to 12 hr

CONTENTS

1	Project Abstract	3
2	Problem Statement	3
3	Proposed Passing Data Between Components Application Wireframe	4
3.1	Screenshots	5
4	Business-Requirement:	6
5	Constraints	6
6	Mandatory Assessment Guidelines	7

1 PROJECT ABSTRACT

In modern front-end development, seamless communication between components is critical for building dynamic applications. This assignment focuses on implementing data passing and event handling between parent and child components using Angular.

The objective is to develop a **User Profile Single Page Application (SPA)** where the user information is displayed and can be edited. Data binding and event emission techniques will be used to facilitate interaction between components, ensuring clear separation of concerns and maintainability.

2 PROBLEM STATEMENT

You are tasked with building a **User Profile SPA** using Angular. The application should:

- Display user details such as **Name, Email, and Bio**.
- Allow editing of user details via form inputs.
- Properly pass user data between components.
- Update the profile information dynamically based on user actions.

The primary focus is to utilize Angular's **@Input()** and **@Output()** decorators to pass data and handle events between components.

3 PROPOSED PASSING DATA BETWEEN COMPONENTS APPLICATION WIREFRAME

UI needs improvisation and modification as per given use case and to make test cases passed.

3.1 SCREENSHOTS

localhost:4200/

User Profile

Name: John Doe

Email: john.doe@example.com

Bio: Software Developer

[Edit](#)

Edit

localhost:4200/

User Profile

Name: Email: Bio:

[Save](#) [Cancel](#)


localhost:4200/

User Profile

Name: Email: Bio:

[Save](#) [Cancel](#)

After Edit

localhost:4200/

User Profile

Name: John Dominic

Email: john.dom@example.com

Bio: Senior Software Developer

Edit

4 BUSINESS-REQUIREMENT:

As an application developer, develop the Passing Data Between Components (Single Page App) with below guidelines:

User Story #	User Story Name	User Story
US_01	Welcome Page	<p>As a user I should be able to visit the welcome page as the default page.</p> <p>Acceptance criteria:</p> <p>UserProfileComponent:</p> <ol style="list-style-type: none">1. Create an object with name user having fields as name, email, bio with values as John Doe, john.doe@example.com, Software Developer respectively.2. A method with name onSave(updatedUser: any) should be defined to accept an updated user values and it should be updated to already created user object.3. Please add the app-user-info component in the user profile component template. Pass the user object as an input, and handle the edit output by calling onSave(\$event) when it emits. <p>Edit Functionality (UserInfoComponent):</p> <ul style="list-style-type: none">• Accept a user object from the parent using @Input().• Emit an edit event using @Output() when changes are saved.• Show user details when not in edit mode (as shown in screenshot).• On clicking Edit, switch to edit mode and populate a form with the user's current data.• On clicking Save, emit the updated user object via the edit output.• On clicking Cancel, discard changes and exit edit mode.• When not editing, show user info and an "Edit" button.• When editing, show inputs for name, email, and bio, along with "Save" and "Cancel" buttons. <p>** Kindly refer to the screenshots for any clarifications. **</p>

5 CONSTRAINTS

1. You should be able to press the "TAB" key and "SHIFT + TAB" to navigate from top field to bottom field and vice-versa.

6 MANDATORY ASSESSMENT GUIDELINES

1. All actions like build, compile, running application, running test cases will be through Command Terminal.
2. To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) -> Terminal -> New Terminal.
3. This editor Auto Saves the code.
4. If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.
5. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.
6. This is a web-based application, to run the application on a browser, use the internal browser in the workspace. Click on the second last option on the left panel of IDE, you can find Browser Preview, where you can launch the application.

Note: The application will not run in the local browser

7. You can follow series of command to setup Angular environment once you are in your project-name folder:
 - a. `npm install --no-bin-links --unsafe-perm` -> Will install all dependencies -> takes 10 to 15 min.
 - b. `npm run start` -> To compile and deploy the project in browser. You can press the <Ctrl> key while clicking on localhost:4200 to open the project in the browser -> takes 2 to 3 min.
 - c. `npm run test` -> to run all test cases. **It is mandatory to run this command before submission of workspace** -> takes 5 to 6 min.
8. Once you are done with development and ready with submission, you may navigate to the previous tab and submit the workspace. It is mandatory to click on **"Submit Assessment"** after you are done with code.

9. You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.