Experiment NO. 5 : Flask Application using render_template() function.

Name of Student	**Anuprita Mhapankar**
Class Roll No	**28**
D.O.P.	**06/03/2025**
D.O.S.	**13/03/2025**
Sign and Grade	

AIM : To create a Flask application that demonstrates template rendering by dynamically generating HTML content using the `render_template()` function.

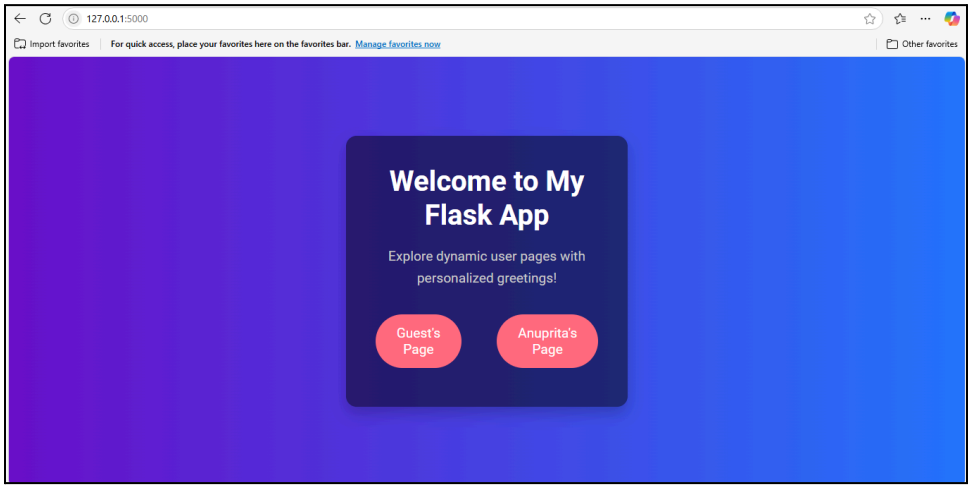
OVERVIEW OF TASKS PERFORMED :

The experiment involves developing a **Flask application** with multiple routes and dynamic content rendering. A **homepage route ("/")** displays a welcome message with navigation links. A **dynamic route ("/user/<username>")** renders an HTML template with a personalized greeting using Jinja2 templating. Jinja2 features such as variables and control structures are utilized to enhance the templates, enabling dynamic content generation based on user input.

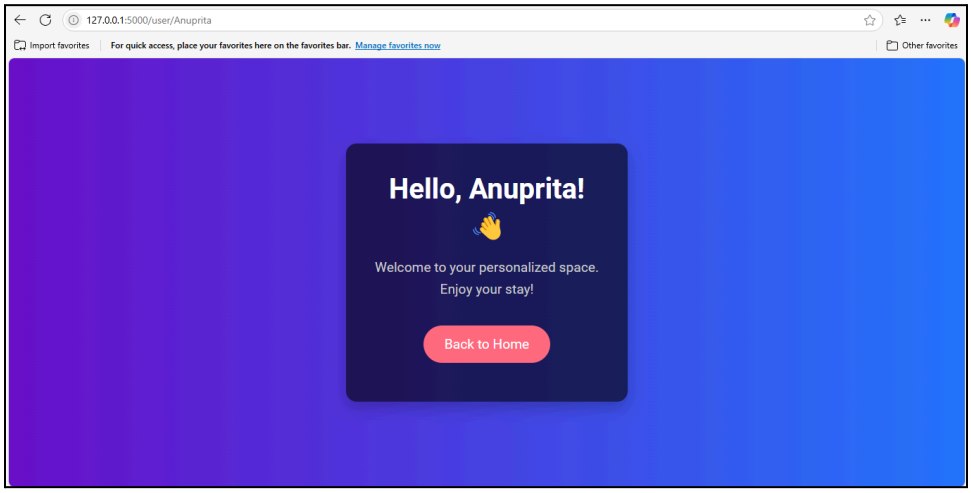
GITHUB LINK - https://github.com/Anuprita2022-26/WebX_Exp5

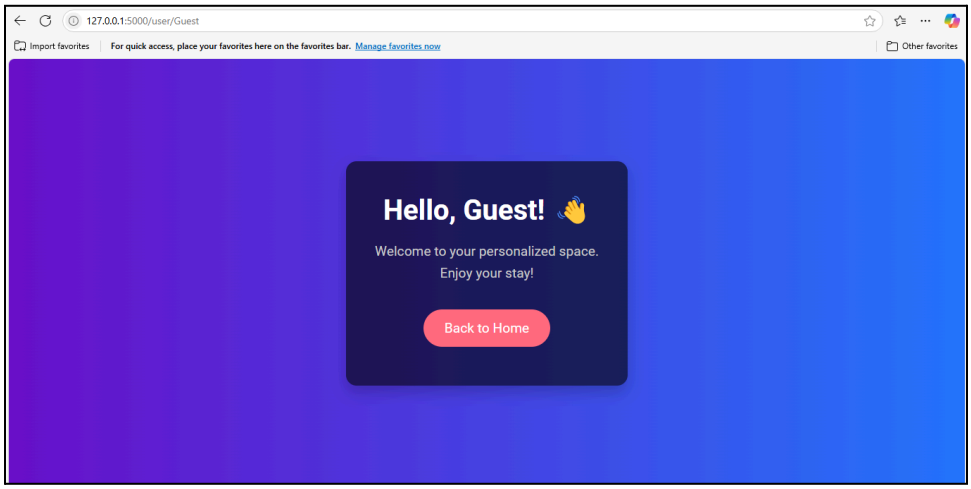
OUTPUT

- **Homepage (`/`)**: The homepage displays a welcome message along with two links for user-specific pages (e.g., Sneha's Page and Anuprita's Page).



- **User Page (`/user/<username>`)**: When clicking on any of the user links, the app renders a personalized greeting with the username passed as a URL parameter.





CONCLUSION

The experiment successfully demonstrated the use of the **render_template()** function in Flask to dynamically generate HTML content. A **homepage (/)** was created with links to user-specific pages, and a **dynamic route (/user/<username>)** was implemented to personalize greetings using Jinja2 templating.

This experiment highlighted key Flask concepts such as **template rendering, Jinja2 syntax, variable passing, and dynamic content generation**, showcasing how Flask efficiently separates business logic from presentation logic to create interactive web applications.