

Ex no:3)	Continuous Integration/Continuous Deployment of a React Application using
	GitHub and Jenkins Freestyle Project

Aim:

To automate the build, test, and deployment process of a React application using GitHub for version control and Jenkins Freestyle Project for CI/CD..

Procedure:**Step 1: Create GitHub Repository and Clone**

1. Create a new repository in GitHub "TripFreeStyle".
2. Clone the repository into your local system inside a folder (e.g., devops) using:
`git clone https://github.com/Aswini-30/TripFreeStyle.git`
3. Open the folder in Visual Studio Code.

Step 2: Create a New React Project

1. Open the terminal in VS Code inside the cloned repository folder.
2. Run the following command to create a React project using **Vite**:
`npm create vite@latest TripFreeStyle`
3. Choose the required framework (**React**) and variant (**JavaScript**).
4. Commit and push the project files to GitHub:
`git add .`
`git commit -m "feat:Files added"`
`git push origin main`

Step 3: Configure Jenkins Job

1. Open Jenkins in your browser.
2. Create a **New Item** → choose **Freestyle Project** → name it (Trip Free Style).
3. In **Source Code Management**, select **Git** and paste your GitHub repository URL.
4. Update the **Branch Specifier** (default: */main).

Step 4: Add Build Steps

In the **Build Steps** section, add the following commands:

```
call npm ci || call npm install
call npm run build
mkdir "C:\ProgramData\Jenkins\.jenkins\userContent\react-app"
xcopy /E /I /Y "dist\*" "C:\ProgramData\Jenkins\.jenkins\userContent\react-app\"
```

where,

- `call npm ci || call npm install` → Installs dependencies (ci preferred, fallback to install).
- `call npm run build` → Builds the production-ready app into the dist folder.

- `mkdir "C:\ProgramData\Jenkins\.jenkins\userContent\react-app"` → Creates a folder in Jenkins to serve the app.
- `xcopy /E /I /Y "dist*" "C:\ProgramData\Jenkins\.jenkins\userContent\react-app\"` → Copies the build files to Jenkins for deployment.

Step 5: Build and Deploy

1. Save the job and click **Build Now**.
2. After a successful build, open your app in the browser .

Step 6: Run Locally with Serve

1. Open **Command Prompt**.
2. Navigate to the dist folder:
`cd C:\ProgramData\Jenkins\.jenkins\workspace\React-project\frontend\dist`
3. start `serve -s . -l 8081`
4. Copy the provided local link and paste it into the browser to access the app locally.

Output:

```
PS C:\Users\Asus\Desktop\Agile> git init
Initialized empty Git repository in C:/Users/Asus/Desktop/Agile/.git/
PS C:\Users\Asus\Desktop\Agile> git remote add origin https://github.com/jerina9558/Voting.git
```

```
PS C:\Users\Asus\Desktop\Agile> git commit -m "Frontend Project"
[master (root-commit) a3e15c3] Frontend Project
13 files changed, 3143 insertions(+)
create mode 100644 frontend/.gitignore
create mode 100644 frontend/README.md
create mode 100644 frontend/eslint.config.js
create mode 100644 frontend/index.html
create mode 100644 frontend/package-lock.json
create mode 100644 frontend/package.json
create mode 100644 frontend/public/vite.svg
create mode 100644 frontend/src/App.css
create mode 100644 frontend/src/App.jsx
create mode 100644 frontend/src/assets/react.svg
create mode 100644 frontend/src/index.css
create mode 100644 frontend/src/main.jsx
create mode 100644 frontend/vite.config.js
```

```
C:\Devops\TripFreeStyle>git branch
* main

C:\Devops\TripFreeStyle>git push
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 12 threads
Compressing objects: 100% (21/21), done.
Writing objects: 100% (22/22), 183.17 KiB | 4.95 MiB/s, done.
Total 22 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Aswini-30/TripFreeStyle.git
* [new branch]      main -> main
```

Command

See [the list of available environment variables](#)

```
call npm ci || call npm install
call npm run build
mkdir "C:\ProgramData\Jenkins\.jenkins\userContent\react-app"
xcopy /E /I /Y "dist\*"
"C:\ProgramData\Jenkins\.jenkins\userContent\react-app\"
```

Advanced ▾

← → ↺ localhost:8080/job/voter/



Jenkins / voter

Status

✓ voter

Changes

Workspace

Build Now

Configure

Delete Project

Rename

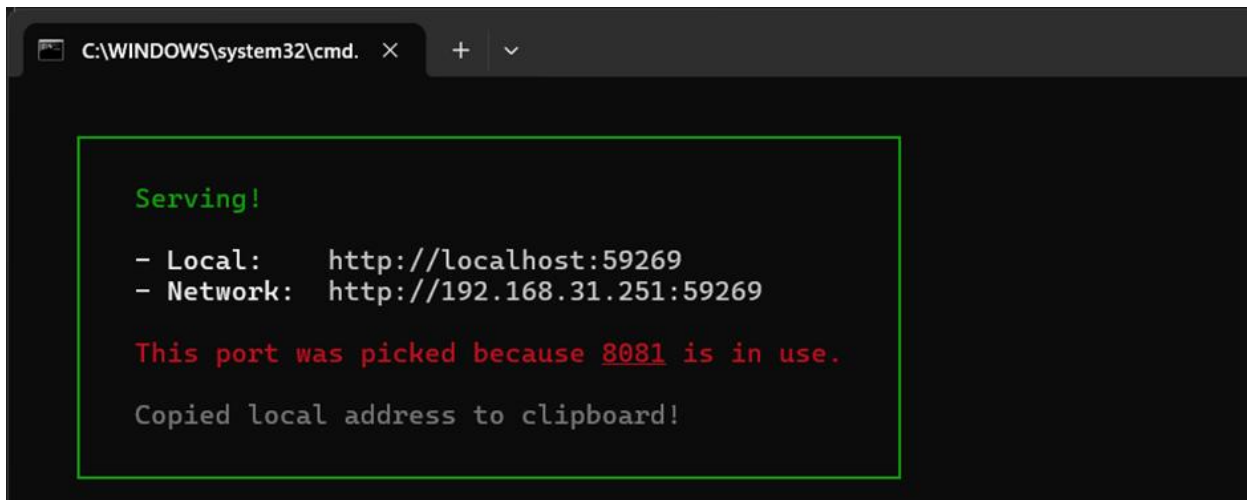
Permalinks

- [Last build \(#1\), 25 min ago](#)
- [Last stable build \(#1\), 25 min ago](#)
- [Last successful build \(#1\), 25 min ago](#)
- [Last completed build \(#1\), 25 min ago](#)

Builds

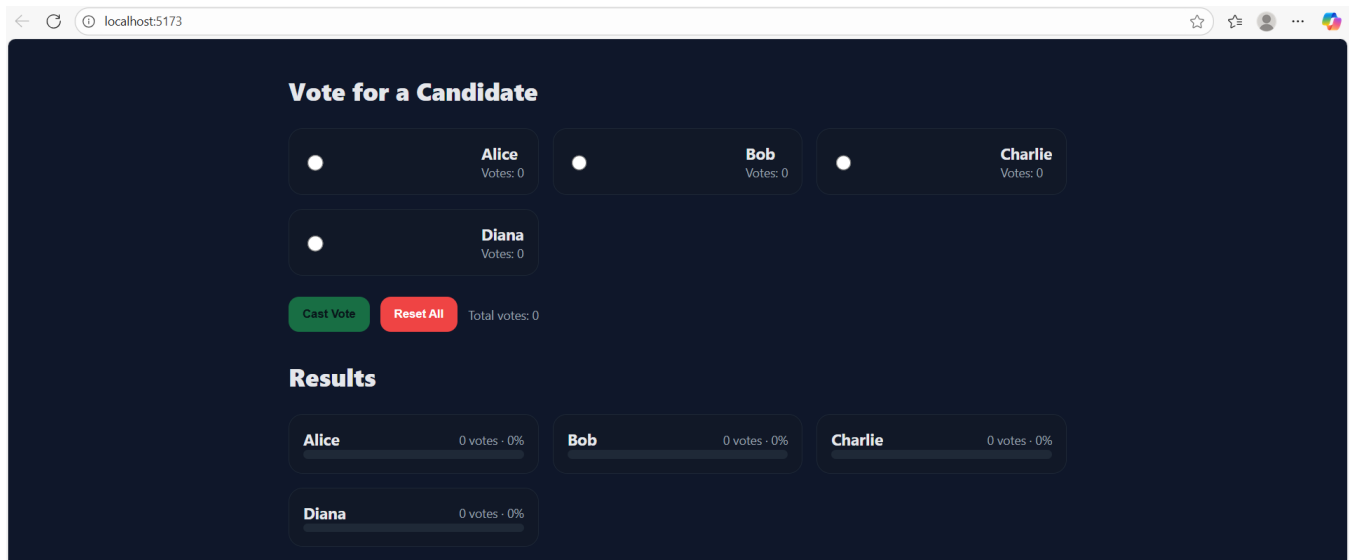
...

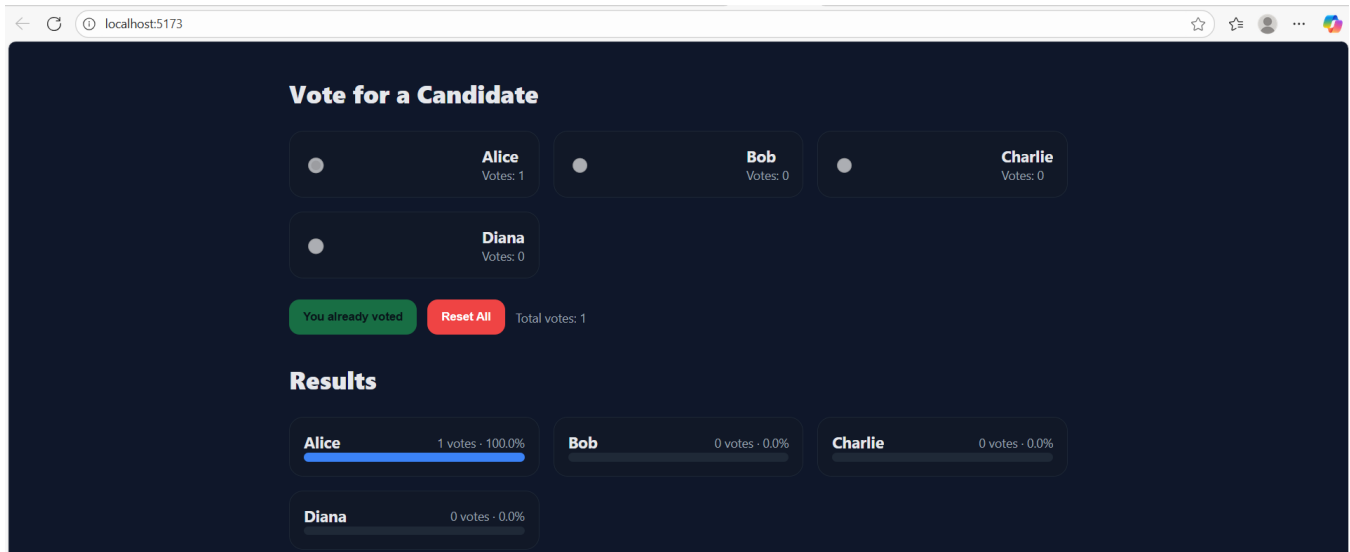
```
C:\Users>cd..  
C:\>cd..  
C:\>cd C:\ProgramData\Jenkins\.jenkins\workspace\React-project\frontend\dist  
C:\ProgramData\Jenkins\.jenkins\workspace\React-project\frontend\dist>start serve -s . -l 8081  
C:\ProgramData\Jenkins\.jenkins\workspace\React-project\frontend\dist>|
```



A terminal window titled "C:\WINDOWS\system32\cmd." showing the output of the serve command. The output is highlighted with a green border. It indicates that the application is serving on localhost:59269 and network address http://192.168.31.251:59269. A note states that port 8081 is in use, and the local address has been copied to the clipboard.

```
Serving!  
  
- Local:    http://localhost:59269  
- Network:  http://192.168.31.251:59269  
  
This port was picked because 8081 is in use.  
  
Copied local address to clipboard!
```



**Rubrics:**

Problem Understanding and Procedure (5)	Implementation and Execution (15)	Testing & Validation (5)	Time Management (5)	Viva (10)	Total (40)

Result:

Thus, successfully automated the build and deployment of a React application using GitHub and Jenkins Freestyle Project. The React app is deployed on the Jenkins server and can also be served locally.

