

**Deployment**

**Activity 17 – PostIT App – Deployment**

The objectives of this activity are to:

1. Create and utilize environment variables on the server side for secure and dynamic configuration.
2. Create configuration settings in the client side for easier management.
3. Create a GitHub repository and push their React app code to it.

**GitHub**

GitHub is an essential tool for hosting and sharing code. In this activity, students will learn how to create a GitHub repository and push their React app code to it. This will teach them version control practices and how to collaborate effectively on projects.

**Step 1: Set Up Your React App**

* Ensure your React app is complete and functioning correctly.
* Open the terminal or command prompt in your React app folder.

**Step 2: Install Git for Windows**

If Git is not installed, you'll need to download and install it.

* **Download git**: Visit the [official Git website](https://git-scm.com/download/win) to download the latest version of Git for Windows.
* **Run the installer**: Execute the downloaded file and follow the installation prompts. Default options are typically sufficient for most users.
* **Verify installation**: Reopen the Command Prompt or PowerShell and type git --version to confirm Git is now recognized.

**Step 3: Initialize a Local Git Repository**

* Run the following command to initialize Git:

**git init**

* Add **.gitignore** to exclude unnecessary files (e.g., **node\_modules**).
* Add all project files to the staging area:

**git add .**

* Commit the changes with a meaningful message:

**git commit -m "Initial commit"**

**Step 4: Create a Repository on GitHub**

1. Log in to your [GitHub account](https://github.com/).
2. Click the **New Repository** button. Name the repository (e.g., **"postit-app"**).
3. Set the repository to **Public** or **Private** based on preference.
4. Click **Create Repository**.

**Step 5: Link Local Repository to GitHub**

1. Copy the repository’s remote URL (shown on GitHub after repository creation).
2. Add the remote URL to your local repository

**git branch -M main**

**git remote add origin <repository-url>**

1. Push your code to the GitHub repository

**git push -u origin main**

**Step 6: Verify the Code on GitHub**

Go to your GitHub repository URL in a browser.

**Deploy the React App to Render.com via GitHub account**

1. Register in <https://render.com/> via your GitHub account.
2. Login to render.com, click New->Static Site. Your GitHub repositories should be automatically displayed. Select and click Connect.

**A screenshot of a web page

Description automatically generated**

Specify the Settings for the Static Site.

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

1. Deploy the Static Site. If there are no errors in building the site. You should see the message “Your site is live”.

**A black screen with white text

Description automatically generatedzA screenshot of a computer

Description automatically generated**

1. You can now check your live site. For example: <https://postit-app-al2g.onrender.com/>
2. Deploying the server. **Click New->Web Service**. Select the GitHub repository. Click Connect.
3. Provide the Settings.

**A screenshot of a web service

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**A close-up of a number

Description automatically generatedA screenshot of a computer

Description automatically generated**

1. For the Environment Variables, you can manually create or your choose “Add from .env”. Since we have .env file, you can choose “Add from .env”.

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

1. Click Deploy Web Service. If there are no errors in building the site. You should see the message “Your site is live”.
2. We need to change our application settings in the development environment to production settings. In your Web Service, copy the URL.

A screenshot of a computer

Description automatically generated

1. In the client folder, update the **.env** file.

#Server URL-development(local server)

#REACT\_APP\_SERVER\_URL=http://localhost:3001

#Server URL-production(deployment)

REACT\_APP\_SERVER\_URL=https://postit-app-server.onrender.com

1. In your git bash, execute the command to push the changes to the remote.

git add .

git commit -m "update server url"

git push -u origin main

1. Your application should be updated on the cloud.