



Universidad  
**CATÓLICA**  
de Pereira

ESPECIALIZACIÓN EN  
**Desarrollo  
de Software**

# Proceso de Desarrollo de Software

*Andrés Mauricio Martínez Hincapié*

---

## Taller Práctico: React + GitHub + Vercel + CI/CD Automático

---

**Objetivo:** Aprender a contribuir a proyectos open-source mediante forks, pull requests y convenciones profesionales.

### Requisitos Previos

- Node.js (v18+)
- Cuenta en [GitHub](#)
- Cuenta en [Vercel](#)

---

## 1. Crear una Aplicación React Básica

### Paso 1: Instalar React

Ejecuta en tu terminal:

```
npx create-react-app ucp-app
```

```
cd ucp-app
```

```
Success! Created ucp-app at /Users/andresmauriciomartinezhincapie/u/ucp/eds/git_hub/ucp-app
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd ucp-app
  npm start

Happy hacking!
bash-3.2$
```

## Paso 2: Crear un Componente Simple

Edita `src/App.js`:

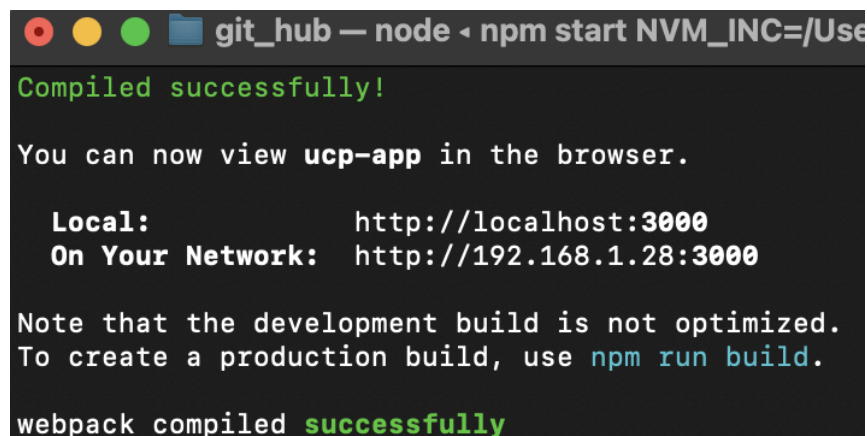
```
import React from 'react';

function App() {
  return (
    <div>
      <h1>¡Universidad Católica de Pereira !</h1>
      <p>Listado de integrantes - Proceso de Desarrollo de Software I</p>
    </div>
  );
}

export default App;
```

## Paso 3: Ejecutar Localmente

`npm start`



```
git_hub — node < npm start NVM_INC=/Use
Compiled successfully!

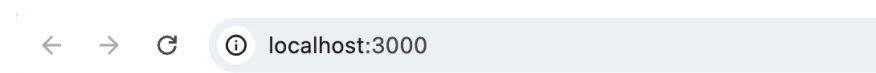
You can now view ucp-app in the browser.

  Local:            http://localhost:3000
  On Your Network:  http://192.168.1.28:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

Abre <http://localhost:3000> para ver tu app.



# ¡Universidad Católica de Pereira !

Listado de integrantes - Proceso de Desarrollo de Software I

---

## 2. Configurar Repositorio en GitHub

### Paso 1: Crear Repo

1. Ve a [github.com/new](https://github.com/new)
2. Nombre: `ucp-app-react`
3. **No inicialices con README** (ya tenemos archivos locales).

### Paso 2: Subir Código

Ejecuta:

```
git init
git add .
git commit -m "Initial commit UCP"
git branch -M main
git remote add origin https://github.com/amartinezh/ucp-app-react.git
git push -u origin main
```

```
bash-3.2$ git init
Reinitialized existing Git repository in /Users/andresmauriciomartinezhincapie/u
bash-3.2$ git add .
bash-3.2$ git commit -m "Initial commit UCP"
[main 806c0cb] Initial commit UCP
Committer: Andres Mauricio Martinez Hincapie <andresmauriciomartinezhincapie@Ma
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 5 insertions(+), 18 deletions(-)
bash-3.2$ git branch -M main
bash-3.2$ git remote add origin https://github.com/amartinezh/ucp-app-react.git
bash-3.2$ git push -u origin main
Enumerating objects: 26, done.
Counting objects: 100% (26/26), done.
Delta compression using up to 8 threads
Compressing objects: 100% (26/26), done.
Writing objects: 100% (26/26), 175.29 KiB | 14.61 MiB/s, done.
Total 26 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), done.
To https://github.com/amartinezh/ucp-app-react.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
bash-3.2$
```


### 3. Desplegar en Vercel

#### Paso 1: Conectar Vercel a GitHub

1. Inicia sesión en [Vercel](#).
2. Haz clic en **Import Git Repository**

#### Import Git Repository

Install the GitHub application for the accounts you wish to Import from to continue


 Install

[Import Third-Party Git Repository →](#)

3. Selecciona tu repo `ucp-app-react`. en el botón Install

☒ **Only select repositories**

Select at least one repository. Also includes public repositories (read-only).

 **Select repositories** ▾

Selected 1 repository.

---

 **amartinezh/ucp-app-react** ×

with these permissions:

- 
- ✓ **Read** access to metadata
  - ✓ **Read** and **write** access to administration, checks, code, commit statuses, deployments, issues, pull requests, and repository hooks

---

**User permissions**

Vercel can also request users' permission to the following resources. These permissions will be requested and authorized on an individual-user basis.

- 
- ✓ **Read** access to email addresses


**Install**

[Cancel](#)


## GitHub Installation Completed

This window will close automatically.

## Import Git Repository

 amartinezh



Search...

 ucp-app-react · 20m ago

Import

Missing Git repository? [Adjust GitHub App Permissions →](#)



## New Project

Importing from GitHub  
 amartinezh/ucp-app-react  main

Choose where you want to create the project and give it a name.



Vercel Team

Project Name

 Andres Mauricio's proje... Hobby 

ucp-app-react

Framework Preset

 Create React App 

Root Directory

./

Edit

> Build and Output Settings

> Environment Variables

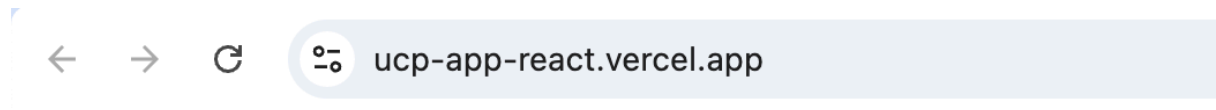
Deploy

4. Configuración automática (Vercel detecta que es React).
5. Haz clic en **Deploy**.

## Paso 2: Obtener Dominio

Vercel asignará una URL como:

`https://mi-app-react.vercel.app`



# ¡Universidad Católica de Pereira !

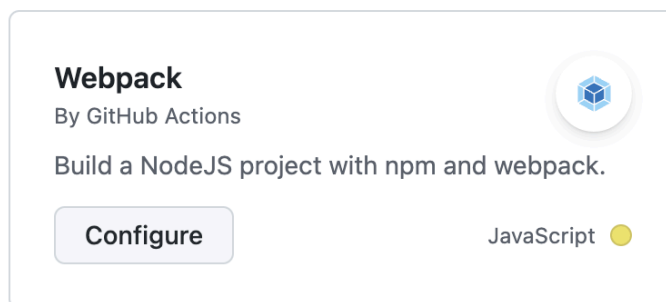
Listado de integrantes - Proceso de Desarrollo de Software I

---

## 4. Automatizar Deploys con GitHub Actions

### Paso 1: Crear Workflow

1. En tu repo, ve a **Actions > Suggested for this repository.**



2. Elige **"Node.js"** como plantilla.
3. Nombra el archivo: `.github/workflows/deploy.yml`.

### Paso 2: Configurar el Workflow

Reemplaza el contenido con:

```
name: Deploy to Vercel

on:
  push:
    branches: [ "main" ]

jobs:
  deploy:
    runs-on: ubuntu-latest
    permissions:
```

```

    contents: read      # Necesario para hacer checkout del código
                        (generalmente es el default)
    deployments: write # Necesario para que la acción cree "GitHub
Deployments"
    steps:
      - name: Checkout code
        uses: actions/checkout@v4

      - name: Install Node.js
        uses: actions/setup-node@v3
        with:
          node-version: 18

      - name: Install dependencies
        run: npm install

      - name: Build project
        run: npm run build

      - name: Pull Vercel Environment Information
        run: npx vercel pull --yes --environment=production --token=${{
secrets.VERCEL_TOKEN }}
        env:
          VERCEL_ORG_ID: ${{{ secrets.VERCEL_ORG_ID }}}
          VERCEL_PROJECT_ID: ${{{ secrets.VERCEL_PROJECT_ID }}}
      - name: Deploy Project to Vercel
        run: npx vercel deploy --prod --token=${{{ secrets.VERCEL_TOKEN }}}
        env:
          VERCEL_ORG_ID: ${{{ secrets.VERCEL_ORG_ID }}}
          VERCEL_PROJECT_ID: ${{{ secrets.VERCEL_PROJECT_ID }}}

```

Cancel changes

Commit changes...

### Paso 3: Configurar Secrets en GitHub

1. En tu repo, ve a **Settings > Secrets > Actions**. para agregar tres variables que se aclaran a continuación:

 **Secrets and variables** 

Actions

2. Agrega estos secrets (los encuentras en Vercel > Project Settings): Ingresa a vercel para obtener el secret, que luego se copiará en la variable que será utilizada en el enlace. A continuación una imagen de la obtención del token.



## Tokens

These tokens allow other apps to control your whole account. Be careful!

### Create Token

Enter a unique name for your token to differentiate it from other tokens.  
Then select the scope for the token.

TOKEN NAME

SCOPE

EXPIRATION

ucp-app-react

Andres Mauricio's projects

1 Year

This token will expire on 5/16/26

Learn more about [Access Tokens](#)

Create

## Token Created

Please copy your token and store it in a safe place.  
**For security reasons we cannot show it again.**

0Nkb7gNvwuSNxnOpy1QYhW7k



Done

- Se agregan las siguientes variables de entorno:
  - i. VERCEL\_TOKEN: Crea uno en [Vercel Account Tokens](#). (como se indica arriba)
  - ii. VERCEL\_ORG\_ID y VERCEL\_PROJECT\_ID: Ingresa a [vercel.com](#)

## Settings

### Project ID

Used when interacting with the Vercel API.

prj\_RQWwin60jeNlv2XZZYTU5y6dQrBV



## Repository secrets

New repository secret

Name ↕	Last updated
VERCEL_ORG_ID	7 minutes ago
VERCEL_PROJECT_ID	now
VERCEL_TOKEN	10 minutes ago

## Repository secrets

This repository has no secrets.

New repository secret

## Actions secrets / New secret

Name \*

VERCEL\_TOKEN

Secret \*

0Nkb7gNvwuSNxnOpy1QYhW7k

Add secret

En caso de presentar inconvenientes, pueden verificar los token.  
con los token.

```
bash-3.2$ npx vercel login
Vercel CLI 41.7.8
? Log in to Vercel Continue with GitHub
> Success! GitHub authentication complete for amartinezh@gmail.com
Congratulations! You are now logged in. In order to deploy something, run `vercel`.
💡 Connect your Git Repositories to deploy every branch push automatically (https://vercel.link/git).
bash-3.2$ vercel link
Vercel CLI 41.7.8
? Set up "~/u/ucp/eds/git_hub/ucp-app"? yes
? Which scope should contain your project? Andres Mauricio's projects
? Link to existing project? no
? What's your project's name? ucp-app
? In which directory is your code located? ./
Local settings detected in vercel.json:
Auto-detected Project Settings (Create React App):
- Build Command: react-scripts build
- Development Command: react-scripts start
- Install Command: `yarn install`, `pnpm install`, `npm install`, or `bun install`
- Output Directory: build
? Want to modify these settings? no
✔ Linked to andres-mauricios-projects/ucp-app (created .vercel and added it to .gitignore)
```

También usar:

```
npx vercel login  
npx vercel whoami  
npx vercel teams ls
```

---

## 5. Probar el Flujo CI/CD

### Paso 1: Hacer un Cambio

Edita `src/App.js`:

```
<h1>Estudiante: <tu nombre></h1>
```

### Paso 2: Push a GitHub


```
git add .  
git commit -m "feat: Update title"  
git pull origin main --rebase  
git push origin main
```

### Paso 3: Ver Resultados

1. En GitHub, ve a **Actions** y verás el workflow ejecutándose.
2. En 1-2 minutos, los cambios aparecerán en tu URL de Vercel.

 Summary

Jobs

 **deploy**

Run details

 Usage

 Workflow file

✓ Build project

```
1 ▶ Run npm run build
4 > ucp-app@0.1.0 build
5 > react-scripts build
8 Creating an optimized production build...
9 Compiled successfully.
10 File sizes after gzip:
11 59.06 kB build/static/js/main.581cde52.js
12 1.77 kB build/static/js/453.ef29bc5c.chunk.js
13 263 B build/static/css/main.e6c13ad2.css
14 The project was built assuming it is hosted at /.
15 You can control this with the homepage field in your package.json.
16 The build folder is ready to be deployed.
17 You may serve it with a static server:
18 npm install -g serve
19 serve -s build
20 Find out more about deployment here:
21 https://cra.link/deployment
```

✓ Pull Vercel Environment Information

```
1 ▶ Run npx vercel pull --yes --environment=production --token=***
7
8 npm warn exec The following package was not found and will be installed: vercel@41.7.8
8 npm warn deprecated path-match@1.2.4: This package is archived and no longer maintained. For support, see https://github.com/expressjs/express/discussions
9 Vercel CLI 41.7.8
10 > NOTE: The Vercel CLI now collects telemetry regarding usage of the CLI.
11 > This information is used to shape the CLI roadmap and prioritize features.
12 > You can learn more, including how to opt-out if you'd not like to participate in this program,
13 > https://vercel.com/docs/cli/about-telemetry
14 Retrieving project...
15 > Downloading 'production' Environment Variables for andres-mauricios-projects/ucp-app
16 Downloading
17 Created .vercel/.env.production.local file [144ms]
18 > Downloading project settings
19 Downloaded project settings to ~/work/ucp-app-react/ucp-app-react/.vercel/project.json [0ms]
```

✓ Deploy Project to Vercel

```
1 ▶ Run npx vercel deploy --prod --token=***
7
8 Vercel CLI 41.7.8
8 Retrieving project...
9 Deploying andres-mauricios-projects/ucp-app
10 Uploading [-----] (0.0B/1.7MB)
11 Uploading [=====] (440.6KB/1.7MB)
12 Uploading [=====] (856.6KB/1.7MB)
13 Uploading [=====] (1.3MB/1.7MB)
14 Uploading [=====] (1.7MB/1.7MB)
15 Inspect: https://vercel.com/andres-mauricios-projects/ucp-app/EahaxgNp2Mr2G2NG1UKLMmkB450S [3s]
16 Production: https://ucp-dn8omm743-andres-mauricios-projects.vercel.app [3s]
17 Queued
18 Building
19 Completing
20 https://ucp-dn8omm743-andres-mauricios-projects.vercel.app
```

✓ Post Install Node.js

```
1 Post job cleanup.
```

## Errores comunes

En caso de denegación por parte de github

### 1. Genera un nuevo PAT con los scopes correctos

1. Ve a [GitHub Settings > Developer Settings > Tokens](#)
2. Haz clic en **"Generate new token"**
3. Selecciona estos scopes:
  - `repo` (acceso completo a repositorios)
  - `workflow` (permite modificar workflows)
4. Copia el token generado (solo se muestra una vez)

### 2. Reconfigura tu repositorio local

Reemplaza la URL remota con el nuevo PAT:

```
git remote set-url origin  
https://TU_TOKEN@github.com/amartinezh/ucp-app-react.git
```

(Reemplaza `TU_TOKEN` con el token que generaste)

### 3. Intenta el push nuevamente

```
git push origin main
```