



**FACULTAD DE INGENIERIA**  
**ESCUELA DE COMPUTACIÓN**  
**DISEÑO Y PROGRAMACIÓN DE SOFTWARE**  
**MULTIPLATAFORMA (DPS 104)**  
**CICLO ACADEMICO: 02 2024**

Título:

**DESARROLLO DE UN CHATBOT CON REACT Y GITHUB COPILOT  
PARA EL SOPORTE AUTOMATIZADO EN INSTITUCIONES EDUCATIVAS**

Docente:

Ing. Alexander Alberto Siguenza Campos.

Presentado por:

Rodrigo Humberto Aguilera Mendoza AM201939

Campus Soyapango, 09 de Agosto de 2024.

## Contenido

<b>DESARROLLO DE UN CHATBOT CON REACT Y GITHUB COPILOT PARA EL SOPORTE AUTOMATIZADO EN UNA INSTITUCIÓN EDUCATIVA .....</b>	<b>3</b>
<b>Situación Problemática .....</b>	<b>3</b>
<b>Objetivo General .....</b>	<b>3</b>
<b>Objetivos Específicos .....</b>	<b>3</b>
<b>Metodología empleada en React .....</b>	<b>4</b>
<b>Stack tecnológico .....</b>	<b>4</b>
<b>Funcionalidades Principales .....</b>	<b>4</b>
<b>Infraestructura .....</b>	<b>5</b>
<b>Configuración inicial .....</b>	<b>5</b>
<b>Configuración del proyecto frontend .....</b>	<b>6</b>
<b>Desarrollo del Chatbot: .....</b>	<b>6</b>
<b>Interfaz.....</b>	<b>6</b>
<b>Configuración del proyecto backend .....</b>	<b>7</b>
<b>Despliegue del chatbot .....</b>	<b>7</b>

# **DESARROLLO DE UN CHATBOT CON REACT Y GITHUB COPILOT PARA EL SOPORTE AUTOMATIZADO EN UNA INSTITUCIÓN EDUCATIVA**

*Agosto 2024*

## **Situación Problemática**

Una institución de educación superior busca optimizar el servicio de atención al estudiante implementando un chatbot capaz de manejar preguntas frecuentes y ofrecer asistencia básica de manera eficiente. Este chatbot se enfocará en diversas áreas clave, incluyendo información académica y administrativa, recursos y servicios del campus, apoyo y bienestar estudiantil, orientación sobre carrera y desarrollo profesional, y vida estudiantil.

## **Objetivo General**

Diseñar y desarrollar un chatbot utilizando React y GitHub Copilot que pueda proporcionar respuestas automatizadas a preguntas frecuentes de los estudiantes.

## **Objetivos Específicos**

1. Evaluar la efectividad del chatbot en el manejo de consultas de estudiantes en diversas áreas.
2. Analizar la integración y uso de GitHub Copilot en el desarrollo del chatbot, explorando cómo la inteligencia artificial puede asistir en la programación.

## Desarrollo de chatbots con React

Los chatbots han sido cada vez más utilizados en instituciones educativas para mejorar la interacción y soporte a los estudiantes. Los chatbots permiten una atención continua y personalizada, reduciendo la carga administrativa y mejorando la satisfacción estudiantil [1]

El uso de React en la creación de chatbots resulta muy eficiente gracias al desarrollo de interfaces que esta librería de JavaScript nos permite construir gracias a las interfaces de usuario interactivas. Su capacidad para manejar componentes reutilizables y su integración con herramientas como GitHub Copilot lo hacen ideal para el desarrollo de chatbots educativos [2].

## Metodología empleada en React

### Stack tecnológico

El stack tecnológico para desarrollar el chatbot con React y GitHub Copilot es el siguiente:

#### 1. Frontend:

- React: Biblioteca de JavaScript para construir la interfaz de usuario.
- React Router DOM: Para la navegación entre diferentes vistas en la aplicación.
- CSS/SCSS: Para el diseño y la estilización de la interfaz de usuario.
- GitHub Copilot: Como herramienta asistida por IA para sugerencias de código y productividad.

#### 2. Backend:

- Node.js: Entorno de ejecución de JavaScript del lado del servidor.

#### 3. Herramientas de desarrollo y despliegue:

- Visual Studio Code: Editor de código compatible con GitHub Copilot.
- Npm: Gestor de paquetes para instalar dependencias.
- Git: Para control de versiones.

## Funcionalidades Principales

- Responder preguntas frecuentes relacionadas con información académica, servicios de campus, apoyo estudiantil, y más.
- Integración con bases de datos institucionales para obtener información actualizada.

## Infraestructura

- **Sistema Operativo:** Windows, macOS, o Linux.
- **RAM:** Mínimo 8GB, recomendado 16GB.
- **Almacenamiento:** Al menos 10GB de espacio libre.
- **Conexión a Internet:** Necesaria para la instalación de paquetes y el uso de GitHub Copilot.

## Configuración inicial

### 1. Instalar Node.js y NPM:

Node.js será esencial porque nos ayudará a construir el backend del chatbot y manejar solicitudes HTTP, interacciones con la base de datos, y lógica del servidor, por otro NPM (Node Package Manager) es la herramienta que se usaremos para gestionar las dependencias y paquetes necesarios para el desarrollo y ejecución de scripts que automatizan tareas de desarrollo como iniciar el servidor o compilar el código.

[Descargar e instalar Node.js](#)

### 2. Instalar VS Code:

Visual Studio Code es un editor de código versátil que nos facilitará la escritura, edición, y depuración del código del chatbot. Con soporte para extensiones como GitHub Copilot, nos ayuda a escribir código más rápido y con menos errores mediante sugerencias inteligentes. Además, ofrece integraciones con Git, resaltado de sintaxis, y herramientas de depuración, lo que mejora significativamente la productividad en el desarrollo frontend y backend.

[Descargar e instalar VS Code](#)

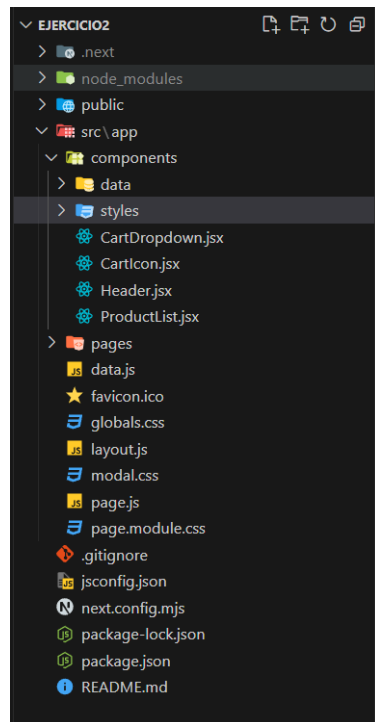
### 3. Instalar GitHub Copilot:

La instalación de GitHub Copilot nos ayudará a acelerar el desarrollo del chatbot al proporcionar sugerencias de código en tiempo real, basadas en inteligencia artificial. Esto incluye autocompletar fragmentos de código, sugerir funciones enteras, y ofrecer soluciones a problemas comunes mientras programas. Es especialmente útil para mejorar la eficiencia, reducir errores, y obtener inspiración para resolver tareas de programación complejas de manera más rápida y precisa.

### 4. Configurar Git: [Instalar y configurar Git](#).

## Configuración del proyecto frontend

1. Crear la aplicación React:  
`npx create-react-app chatbot-frontend`  
`cd chatbot-frontend`
2. Instalar las librerías necesarias:  
`npm install axios react-router-dom`
3. Estructurar el proyecto



**src/components:** Para los componentes del chatbot.

**src/pages:** Para diferentes vistas de la aplicación.

**src/styles:** Para los estilos CSS.

## Desarrollo del Chatbot:

### Interfaz

Se debe crear un componente chatbot que maneje la interacción con el usuario.

## Lógica

Usaremos `useState` y `useEffect` para manejar el estado y las solicitudes a la API.

## Conexión API

Se deberá crear un servicio en `src/services/api.js` para manejar la conexión con el backend, para lo cual se deberá crear una nueva carpeta `services` dentro de la carpeta raíz

## Configuración del proyecto backend

### 1. Estructura el proyecto

`/models`: Definir los modelos de datos.

`/routes`: Definir las rutas de la API.

`/controllers`: Manejar la lógica de las rutas.

`/config`: Configuración de base de datos y variables de entorno.

### 2. Configurar el servidor

`index.js`: Configurar el servidor y las rutas básicas.

Definir las rutas que recibirán las preguntas del chatbot y devolverán las respuestas adecuadas.

### 3. Desarrollo del backend

Ruta `/api/chatbot`: Recibirá las preguntas del frontend y enviará las respuestas.

Integración con IA: Se puede integrar servicios de procesamiento de lenguaje natural.

## Despliegue del chatbot

Desplegar el Frontend:

- Podemos desplegar nuestra aplicación react en plataformas de despliegue como Vercel para lo cual necesitaremos conectar tu repositorio de GitHub y desplegar automáticamente el frontend.

Desplegar el Backend:

- Para esto deberemos conectar nuestro repositorio de GitHub, configurar variables de entorno, y desplegar el backend.

## Resultados Esperados

Se espera que el chatbot desarrollado mejore la experiencia estudiantil al ofrecer respuestas rápidas y precisas a las consultas comunes, liberando recursos humanos para manejar situaciones más complejas. Además, el uso de GitHub Copilot debería demostrar ser una herramienta eficaz en el proceso de desarrollo, facilitando la generación de código de alta calidad.

## Conclusiones

El desarrollo de un chatbot para soporte estudiantil utilizando React y GitHub Copilot tiene el potencial de optimizar los servicios educativos, ofreciendo una solución moderna y eficiente que puede ser replicada en otras instituciones. Este proyecto también sirve como un caso de estudio sobre la efectividad de las herramientas de inteligencia artificial en el desarrollo de software educativo.

## REFERENCIAS

[1] Pérez, J., García, M., & López, A. "Utilización de Chatbots en Instituciones Educativas," *Revista de Tecnología Educativa*, vol. 35, no. 4, pp. 123-135, 2021.

[2] Smith, T. "React and AI: Building Interactive Chatbots," *Journal of Web Development*, vol. 12, no. 2, pp. 89-104, 2020.