Dominando la automatización de pruebas con ChatGPT

La automatización de pruebas es esencial para que el software no presente errores y garantice la experiencia de usuario deseada. A medida que avanza la tecnología, también lo hace la complejidad de la automatización de pruebas.

Mantenerse al día con las últimas herramientas y técnicas puede ser una tarea abrumadora, especialmente para aquellos que no están familiarizados con la codificación o son nuevos en el campo.

Sin embargo, la tecnología de inteligencia artificial moderna está cruzando el horizonte y entrando en el mundo convencional con herramientas poderosas como ChatGPT, que puede, entre otras miles de cosas, hablar contigo y enseñarte a leer y escribir código como un humano.

En este documento, analizaremos el proceso de dominar la automatización de pruebas utilizando la nube híbrida BrowserStack y la tecnología OpenAI GPT.

**Tabla de contenido**

* ¿Qué es ChatGPT?
* ¿Qué es BrowserStack Automate?
* Ejemplos de uso de ChatGPT para pruebas de automatización
  + Planificación de pruebas con ChatGPT
  + Pruebas unitarias con ChatGPT
  + Escriba automatización de pruebas en cualquier marco con ChatGPT
  + Operaciones avanzadas a nivel de script con ChatGPT
  + Análisis y resumen de datos
  + ChatGPT como asistente de inteligencia artificial para todo uso
* Limitaciones de ChatGPT

¿Qué es ChatGPT?

[ChatGPT](https://chat.openai.com/) es un modelo de lenguaje de última generación desarrollado por [OpenAI](https://openai.com/) . Mediante el uso de inteligencia artificial basada en el procesamiento del lenguaje natural, permite la automatización de tareas que implican interacciones similares a las humanas, como un chat Bot de IA o un asistente virtual.

OpenAI pretende utilizar diferentes interfaces como Playground, ChatGPT y API para hacer que sus capacidades de IA sean más accesibles y fáciles de usar para una amplia gama de usuarios y aplicaciones.

La interfaz de juegos permite a los usuarios experimentar e interactuar con los modelos de IA de OpenAI de forma visual e intuitiva. Esta interfaz está diseñada para ser fácil de usar y proporciona controles más precisos sobre la dinámica del modelo y la E/S.

ChatGPT es una aplicación web basada en un modelo de IA que permite a los usuarios interactuar con la IA de OpenAI de forma natural y conversacional. Puede comprender y responder a entradas de lenguaje natural y realizar un seguimiento de conversaciones anteriores para brindar respuestas más precisas y relevantes.

ChatGPT puede generar texto en varios formatos, como datos estructurados, fragmentos de código y anotaciones, que pueden usarse para automatizar tareas como la generación y prueba de código.

A continuación, se muestran algunas formas en las que se puede utilizar ChatGPT para aprender a codificar:

1. **Generación de código:** puede generar fragmentos de código basados ​​en indicaciones en lenguaje natural, lo que proporciona una excelente manera para que los principiantes aprendan la sintaxis y la estructura básicas de un lenguaje de programación.
2. **Completado de código:** se puede utilizar para completar código parcialmente escrito, proporcionando orientación sobre los próximos pasos y ayudando a los principiantes a comprender cómo resolver un problema particular.
3. **Explicación del código:** se puede utilizar para explicar el propósito y la función de líneas específicas de código, ayudando a los principiantes a comprender cómo funcionan juntos los diferentes componentes del código.
4. **Depuración:** se puede utilizar para identificar y explicar errores en el código, ayudando a los principiantes a aprender cómo depurar y solucionar problemas de su propio código.
5. **Orientación del proyecto:** puede proporcionar orientación y sugerencias sobre la estructura del proyecto, las mejores prácticas y las bibliotecas que pueden utilizar los principiantes que quieran crear un proyecto.

En general, es un avance positivo y su amplio uso demuestra el creciente interés de los usuarios por tener herramientas de IA de alta capacidad como parte de su arsenal.

Dicho esto, tenga en cuenta que **ChatGPT es un modelo de lenguaje potente, pero no reemplaza a los profesores, mentores o desarrolladores experimentados.**Puede ser una gran ayuda para aprender a codificar, pero debe usarse junto con otros recursos y la orientación de desarrolladores experimentados.

¿Qué es BrowserStack Automate?

[BrowserStack Automate](https://www.browserstack.com/automate) es una plataforma de pruebas que permite a los usuarios ejecutar pruebas automatizadas en una variedad de dispositivos y navegadores. Con una amplia gama de configuraciones y capacidades, es una excelente herramienta para probar aplicaciones web y móviles.

Una de las mayores ventajas de utilizar BrowserStack Automate es su plataforma en la nube, que permite un fácil acceso a una amplia gama de dispositivos y navegadores para realizar pruebas.

Esto elimina la necesidad de un laboratorio de dispositivos físicos y permite realizar pruebas en una variedad de configuraciones, lo que garantiza que su aplicación funcione sin problemas en diferentes dispositivos y navegadores.

Otra ventaja de BrowserStack es su integración con Selenium, Appium y otros frameworks de automatización de pruebas populares. Esta integración permite la automatización de aplicaciones web y móviles y proporciona una amplia gama de capacidades para probar su aplicación de manera eficaz.

¿Cómo utilizar ChatGPT y Automate para la automatización de pruebas?

La combinación del poder de BrowserStack Automate y ChatGPT puede aportar mejoras significativas a su proceso de automatización de pruebas.

Antes de comenzar, deberá configurar una cuenta en [BrowserStack](https://www.browserstack.com/) .

Y necesitará una cuenta [OpenAI](https://openai.com/) para acceder a ChatGPT.

ChatGPT se puede utilizar de varias maneras al intentar aprender o implementar la automatización de pruebas para cualquier tipo de escenario, incluidos, entre otros:

1. **Pruebas unitarias**: para proyectos front-end creados en varios lenguajes como JavaScript, Python, Java, Swift, etc. ChatGPT se puede utilizar para generar referencias de sintaxis útiles para escribir pruebas unitarias correctamente.
2. **Pruebas de integración:** ChatGPT se puede utilizar para generar scripts de prueba para pruebas de integración con la ayuda de un marco como Selenium. Esto se puede utilizar para probar aplicaciones web front-end creadas con JavaScript, HTML y CSS.
3. **Pruebas de API:** ChatGPT se puede utilizar para generar scripts de prueba para pruebas de API con la ayuda de un marco como Rest Assured o Postman. Esto se puede utilizar para probar API de backend creadas con lenguajes como Java, Python, NodeJS, etc.
4. **Pruebas móviles**: ChatGPT se puede utilizar para generar scripts de prueba para pruebas móviles con la ayuda de un marco como Appium. Esto se puede utilizar para probar aplicaciones móviles creadas con lenguajes como Java, Swift, etc.
5. **Pruebas de extremo a extremo**: ChatGPT se puede utilizar para generar scripts de prueba para pruebas de extremo a extremo con la ayuda de un marco como Selenium, [Puppeteer](https://www.browserstack.com/guide/puppeteer-framework-tutorial" \o "Tutorial de Puppeteer Framework: conceptos básicos y configuración" \t "_blank) , etc. Esto se puede utilizar para probar escenarios de extremo a extremo para aplicaciones web creadas con JavaScript, HTML y CSS.

ChatGPT funciona en hilos de conversación de forma muy similar a una aplicación de mensajería y mantiene el contexto para futuras referencias en el historial de chat.

Ejemplos de uso de ChatGPT para pruebas de automatización

**Planificación de pruebas con ChatGPT**

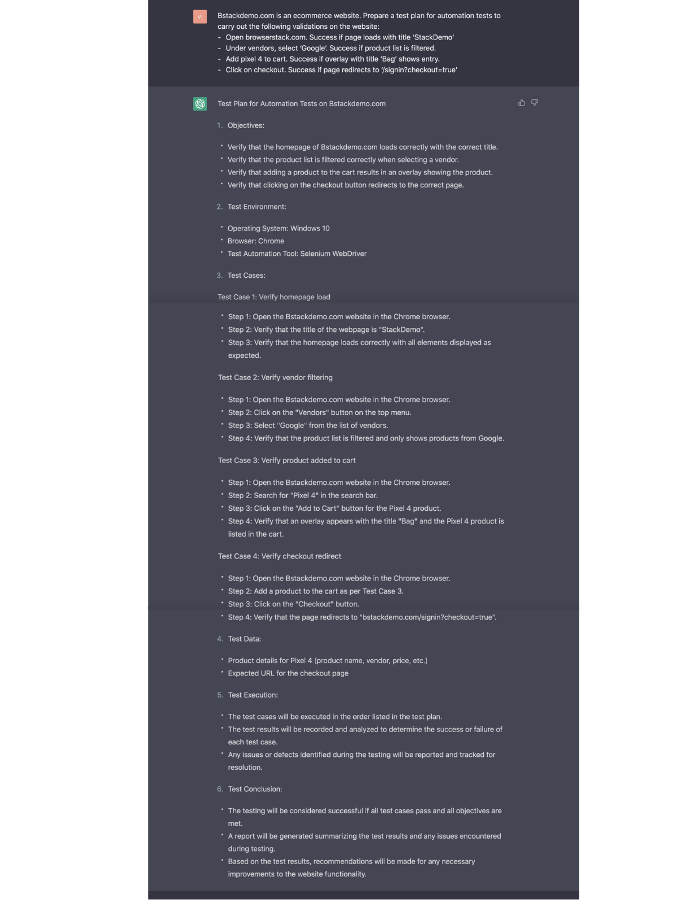
**Ejemplo 1:**Supongamos que tienes un sitio web bstackdemo.com en el que has creado algunas funcionalidades y quieres planificar la automatización de las pruebas.

Puede utilizar GPT en estos casos proporcionándole detalles sobre los componentes funcionales y su uso y pidiéndole que genere un plan de prueba.

**Aviso:** “Bstackdemo.com es un sitio web de comercio electrónico. Prepare un plan de pruebas para realizar pruebas de automatización para realizar las siguientes validaciones en el sitio web:

* Abra browserstack.com. Se carga correctamente si la página tiene el título 'StackDemo'
* En la sección de proveedores, seleccione "Google". La búsqueda se ha realizado correctamente si la lista de productos está filtrada.
* Agregue Píxel 4 al carrito. Se realizó correctamente si la superposición con el título "Bolsa" muestra la entrada.
* Haga clic en finalizar compra. Se realizó correctamente si la página redirecciona a '/signin?checkout=true'”

**ChatGPT:**



Y así, ya tienes suficiente material de plantilla para empezar a crear un documento de prueba. Puedes compartirlo con tu equipo y empezar a desarrollar este plan básico.

Tenga en cuenta que no solo entendió la tarea a partir de información limitada, sino que también identificó una plataforma y una herramienta de automatización adecuadas para realizar pruebas. Una vez que su plan esté listo, puede probarlo en BrowserStack.

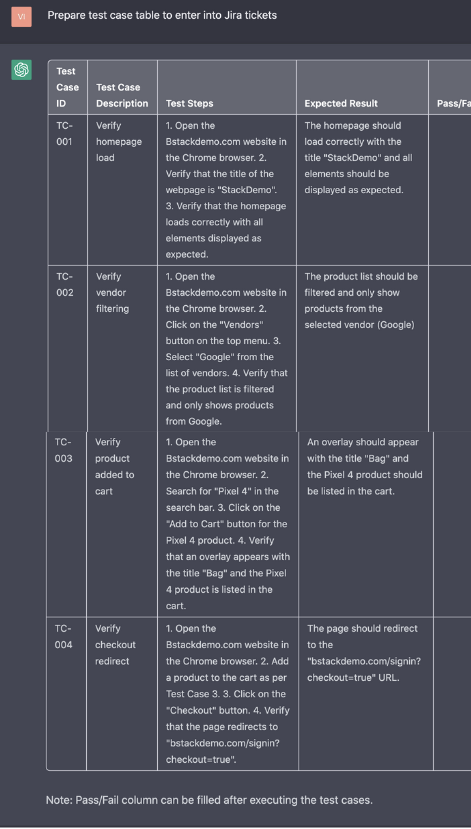
**Ejemplo 2:**Puede utilizar ChatGPT para reformatear y preparar cualquier dato para cualquier otro formato.

Digamos que desea el plan de prueba en un formato que le resulte fácil ingresarlo en una plataforma de seguimiento de errores como Jira o GitHub.

En tal caso, puedes usar ChatGPT solicitando más información en el mismo hilo:

**Indicación:** “Preparar la tabla de casos de prueba para introducirla en los tickets de Jira”

**ChatGPT:**

****

**Pruebas unitarias con ChatGPT**

**Ejemplo 1:**Puedes darle a ChatGPT un fragmento de código y escribirá pruebas unitarias para él en el formato deseado.

**Indicación:** “Escriba pruebas unitarias para probar el siguiente código:

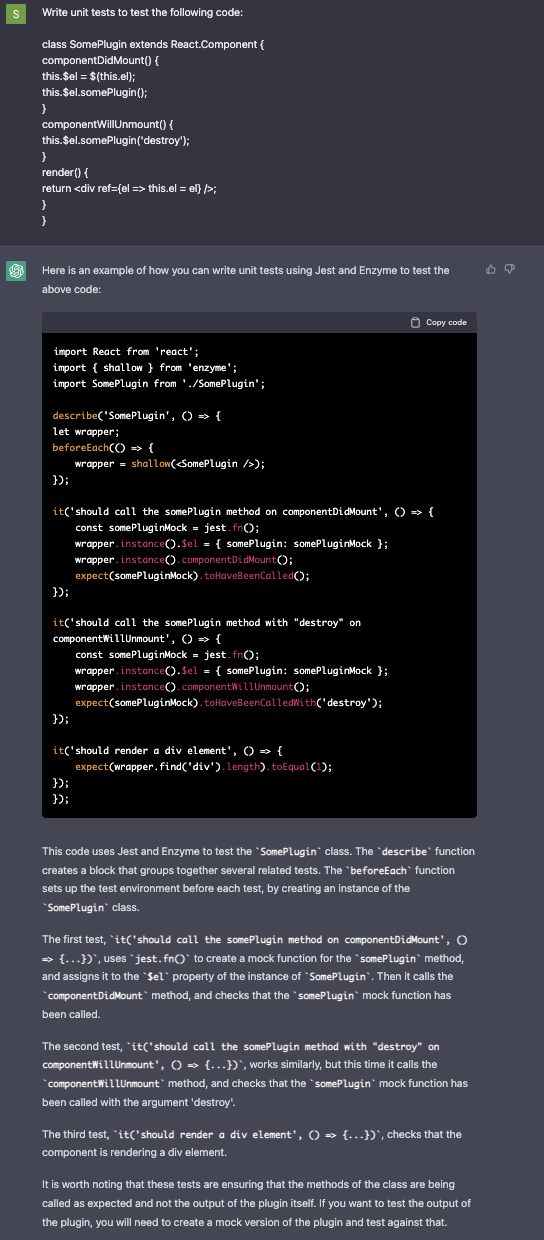
clase SomePlugin extiende React . Component {

componentDidMount () { this . $el = $ ( this . el ); this . $el . somePlugin (); }

componentWillUnmount () { this . $el . somePlugin ( 'destruir' ); }

render () { return < div ref ={ el => this . el = el } />; } }

**ChatGPT:**



No solo puede reconocer el idioma del código de entrada sin ninguna información de nuestra parte, sino que también utiliza una biblioteca correcta llamada [Enzyme](https://enzymejs.github.io/enzyme/" \t "_blank) y continúa explicando los ejemplos con mayor profundidad con las exenciones de responsabilidad correspondientes.

**Ejemplo 2:** Si eres un principiante y solo quieres aprender conceptos, puedes pedirle a ChatGPT que te dé ejemplos y explique cómo funcionan las pruebas unitarias en cualquier lenguaje o marco popular.

**Indicación:** “¿Cómo escribir pruebas unitarias para iOS con XCtest? Explícalo con un ejemplo”.

**ChatGPT:** responde con definiciones, ejemplos y una explicación detallada:



El poder de presentar conceptos, ejemplos de código y explicaciones de código de manera fluida es lo que hace de ChatGPT una herramienta poderosa para implementar en varios escenarios de codificación, incluida la automatización de pruebas.

**Escriba automatización de pruebas en cualquier marco con ChatGPT**

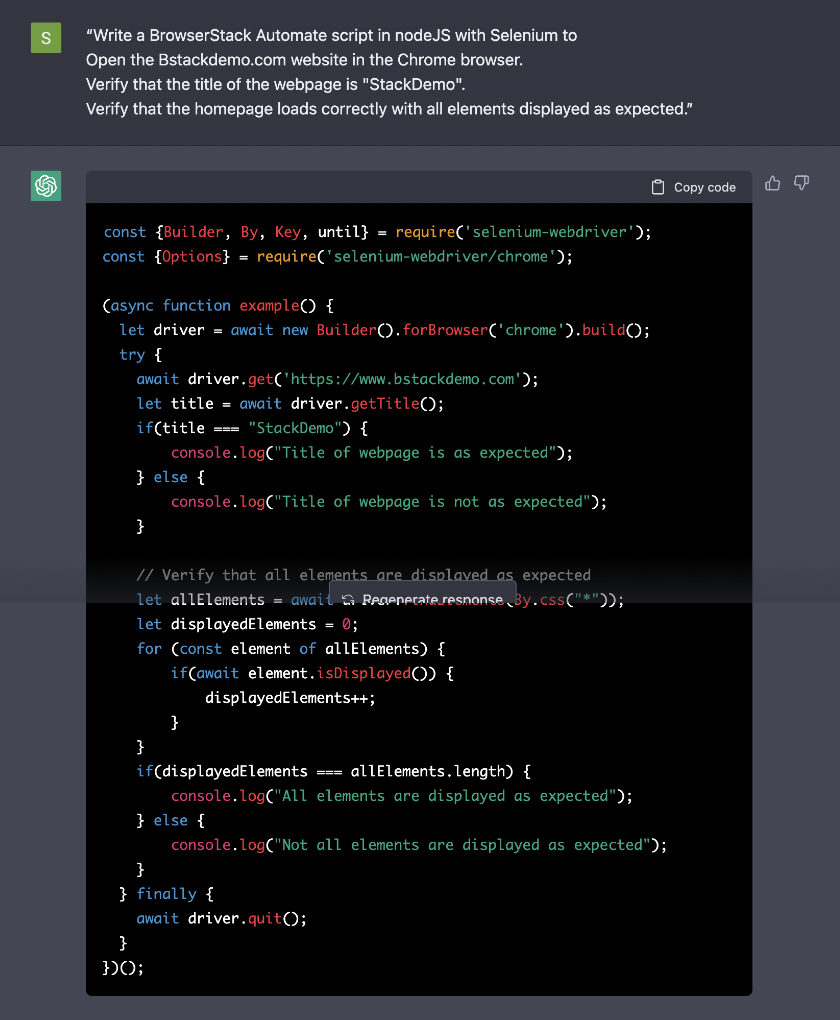
**Ejemplo 1:** Pruebas web con [Selenium](https://www.selenium.dev/)

Continuando con los planes de prueba generados anteriormente, ahora puedes usar ChatGPT para escribir código de automatización de pruebas para el mismo.

**Indicación:** “Escriba un script de BrowserStack Automate en NodeJS con Selenium para

1. Abra el sitio web Bstackdemo.com en el navegador Chrome.
2. Verifique que el título de la página web sea “StackDemo”.
3. Verifique que la página de inicio se cargue correctamente y que todos los elementos se muestren como se espera”.

**ChatGPT:**

****

**Ejemplo 2:** Pruebas móviles con [Appium](https://appium.io/)

**Indicación:** “¿Cómo escribir una prueba de automatización en BrowserStack usando Appium? Explícalo con un breve ejemplo”.

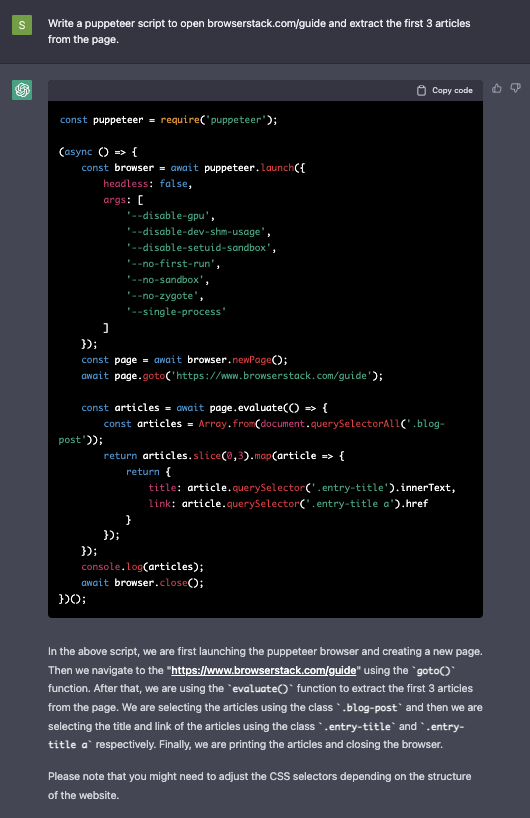
**ChatGPT:**

****

**Ejemplo 3:** Rastreo web con [Puppeteer](https://github.com/puppeteer/puppeteer" \t "_blank)

**Indicación:** “Escribe un script de puppeteer para abrir browserstack.com/guide y extraer los primeros 3 artículos de la página”.

**ChatGPT:**



De esta manera, al variar las indicaciones apropiadamente, puedes generar ejemplos de código para el lenguaje de programación y el marco de automatización que elijas.

Estos fragmentos de código nunca son 100% sintácticos, pero son suficientes para tener una estructura básica que se puede seguir trabajando y refinando con la ayuda de guías oficiales y, nuevamente, ChatGPT.

**Operaciones avanzadas a nivel de script con ChatGPT**

**Ejemplo 1:**Puedes usar la IA para que te explique una parte del código o el archivo completo; puedes simplemente copiar y pegar y pedirle que te lo explique de la siguiente manera:

**Indicación:** Explique el siguiente código:

const webdriver = require ( 'selenium-webdriver' ) ;

async function runTestWithCaps ( capacitys ) { let driver = new webdriver.Builder () .usingServer ( 'http:// <YOUR-SERVER> .browserstack.com/wd/hub' ) . withCapabilities ({ ... capacitys , ... capacitys [ 'browser' ] && { browserName : capacitys [ ' browser' ]} // Debido a que la vinculación del lenguaje NodeJS requiere que browserName esté definido } ) . build (); try {

await driver.get ( " https://bstackdemo.com/" );

await driver.wait ( webdriver.tilt.titleMatches ( / StackDemo / i ), 10000 ) ; // ubicando el producto en la página web y obteniendo el nombre del producto let productText = await driver

. findElement ( webdriver.By.xpath ( '//\*[@id="1"]/p' ) ) . getText(); // haciendo clic en el botón "Añadir al carrito" await driver.findElement(webdriver.By.xpath( ' // \* [ @ id = " 1

" ] / div [ 4]' ) ) . click (); // esperando hasta que se muestre el panel del carrito en la página web driver.findElement

( webdriver.By.className ( " float-cart\_\_content" )); // localizando el producto en el carrito y obteniendo el nombre del producto en el carrito let productCartText = await driver.findElement

( webdriver.By.xpath ( ' //\*[@id="

\_\_ next"]/div/div/div[2]/div[2]/div [ 2]/div/div[3]/p[1]' ) ) ) . getText ( ) ;

// comprobar si el producto ha sido añadido al carrito comparando el nombre del producto if ( productCartText !== productText ) throw new Error ( "" );

await driver .executeScript ( ' browserstack\_executor : {"action": "setSessionStatus", "arguments": {"status": "passed", "reason": "¡El producto ha sido añadido correctamente al carrito!"}}' ); } catch ( e ) {

await driver .executeScript ( ' browserstack\_executor : {"action": "setSessionStatus", "arguments": {"status": "failed", "reason": "¡Algunos elementos no se pudieron cargar!"}}' ); }

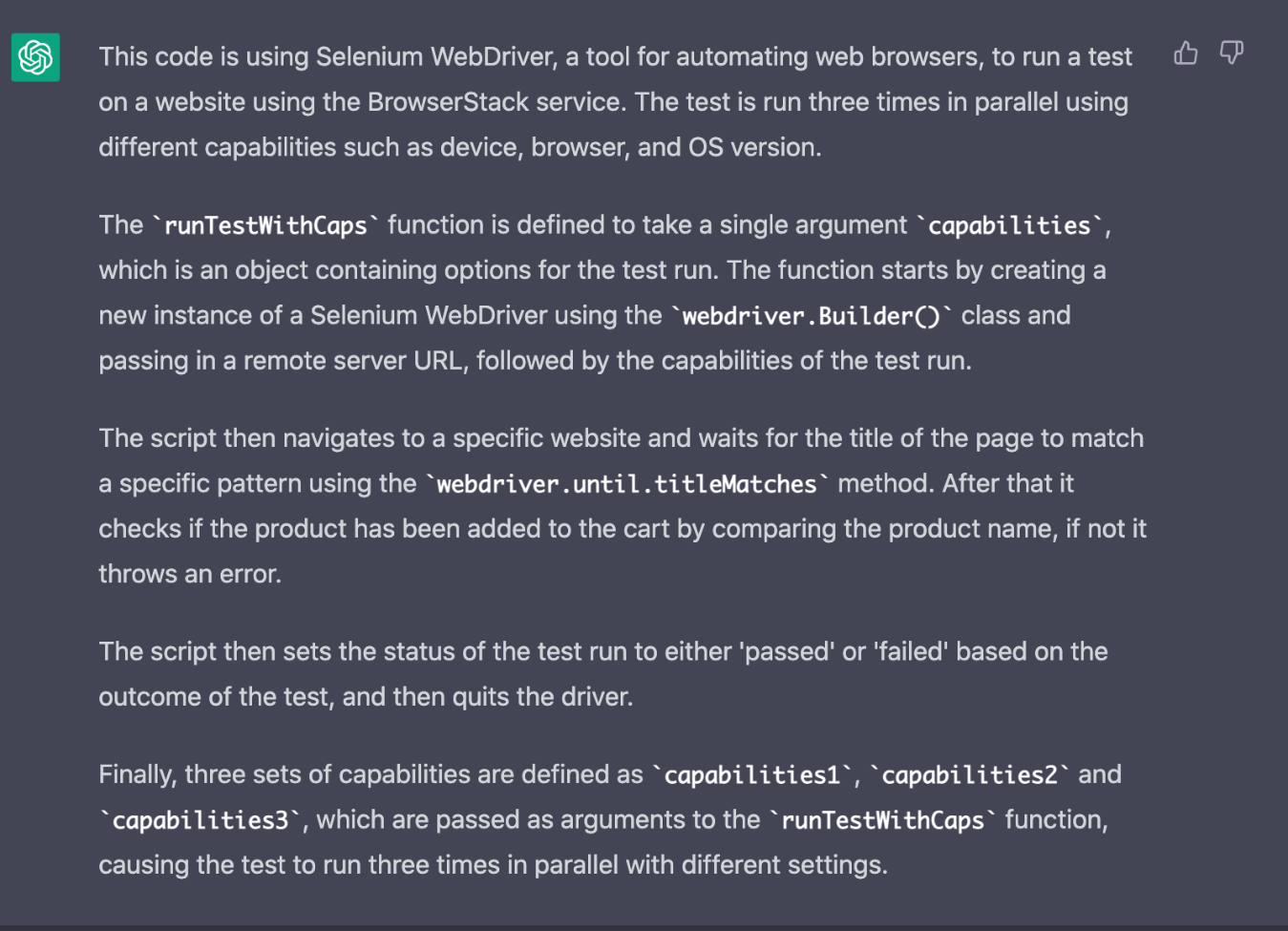
await driver .quit ( ); } const capacidades1 = { 'bstack:opciones' : { "nombreDispositivo" : "iPhone 14 Pro Max" , "versiónOS" : "16" , "orientaciónDispositivo" : "retrato" , "nombreCompilación" : "browserstack-build-1" , "nombreSesión" : "Prueba paralela 1" , }, "nombreNavegador" : "ios" } const capacidades2 = { 'bstack:opciones' : { "nombreDispositivo" : "Google Pixel 7 Pro" , "versiónOS" : "13.0" , "orientaciónDispositivo" : "retrato" , "nombreCompilación" : "browserstack-build-1" , "nombreSesión" : "Prueba paralela 2" , }, "nombreNavegador" : "android" } const capacidades3 = { 'bstack:opciones' : { "os" : "Windows" , "osVersion" : "11" , "browserVersion" : "última versión beta" ,"buildName" : "browserstack-build-1" , "sessionName" : "Prueba paralela 3" , }, "browserName" : "Edge" }

runTestWithCaps ( capacidades1 );

runTestWithCaps (capacidades2 );

ejecutarPruebaConCaps ( capacidades3 );

**ChatGPT:**



Muy buen resumen de ChatGPT.

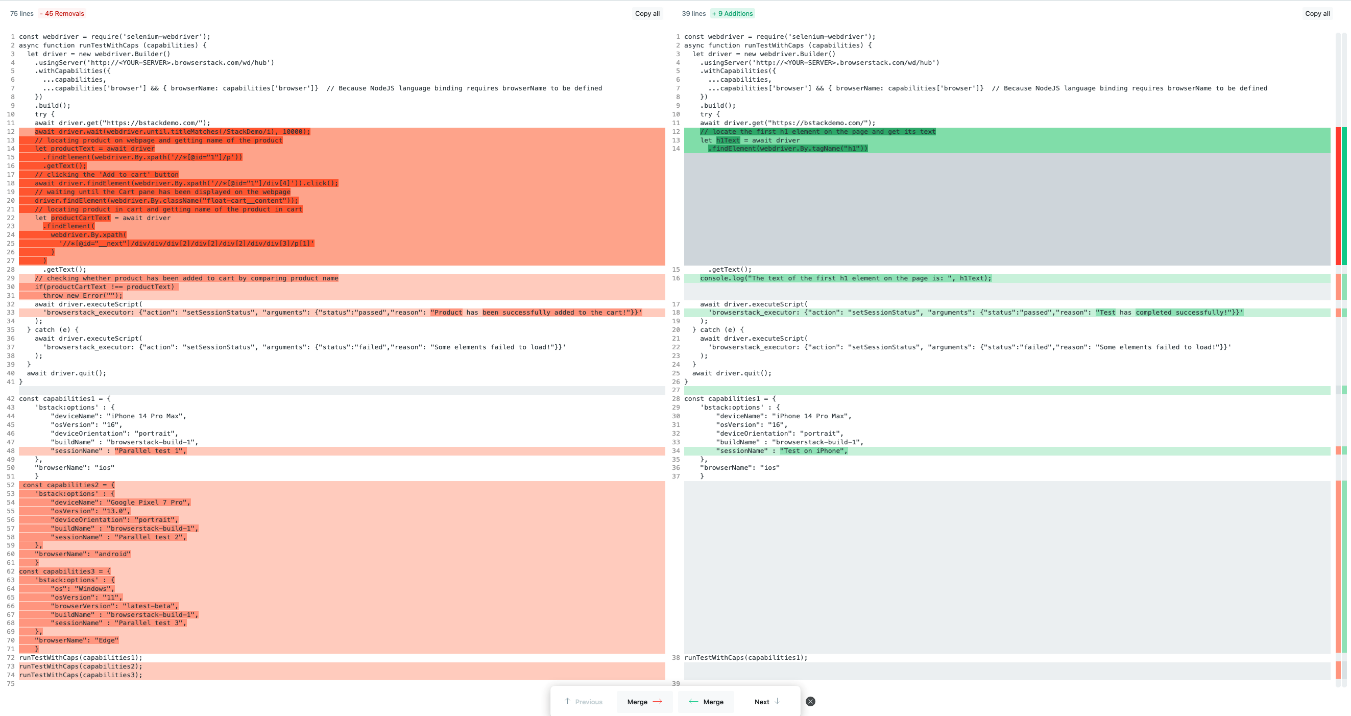
**Ejemplo 2:** Ahora, si desea cambiar algo en el código (por ejemplo, en lugar de ejecutar 3 pruebas, solo ejecutarlas en un dispositivo y cambiar los parámetros de prueba, etc.), puede hacer que ChatGPT lo haga solicitando algo como esto:

**Aviso:** “Cambie el código de ejemplo proporcionado y, en lugar de ejecutar 3 pruebas en paralelo, ejecute solo la primera en el iPhone y elimine las demás. Después de eso, cambie la parte donde coincide con el título de la página y el nombre del producto y elimínela. En su lugar, después de abrir bstackdemo.com, simplemente verifique y registre el primer elemento h1 en la página”.

**ChatGPT:**

****

Comprende perfectamente los comandos intercalados y genera este código, que cuando ejecutamos a través del verificador de diferencias contra el código original obtenemos esto:



Esto no está muy lejos de cómo se vería la revisión de Git de un desarrollador humano antes de enviar el código.

Como puede ver, puede mantener este flujo de ida y vuelta independientemente de la duración de la conversación y realizar los cambios correctos en el lugar correcto.

Lo ideal es que empieces con piezas pequeñas e intentes unirlas para sacar el máximo partido a sus capacidades. Introducirle indicaciones muy grandes podría provocar dificultades a la hora de intentar controlarlo y también un mal tiempo de respuesta del lado del servidor.

**Análisis y resumen de datos**

Los desarrolladores y profesionales de control de calidad pueden utilizar ChatGPT para mejorar sus actividades rutinarias de generación de informes y análisis de datos de varias maneras, especialmente relacionadas con el análisis de informes de automatización de pruebas.

ChatGPT se puede utilizar para resumir los resultados de la ejecución de pruebas, proporcionando una descripción general de los resultados de las pruebas, incluida la cantidad de pruebas aprobadas, fallidas y omitidas.

**Ejemplo 1:** Comprender el informe XML de [Cypress](https://www.cypress.io/) Junit

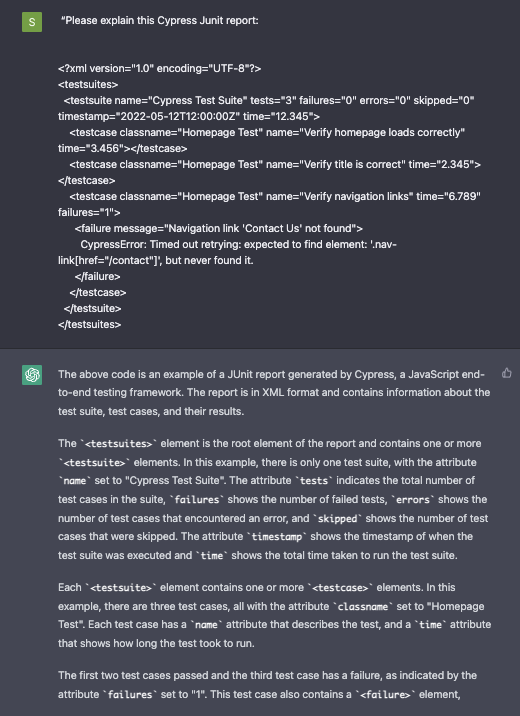
**Indicación:** “Por favor, explique este informe de Cypress Junit:

<? xml version = "1.0" encoding = "UTF-8" ?> <testsuites> <testsuite name = "Cypress Test Suite" tests = "3" failures = "0" errors = "0" skipped = "0" timestamp = "2022-05-12T12:00:00Z" time = "12.345" > <testcase classname = "Prueba de página de inicio" name = "Verificar que la página de inicio se cargue correctamente" time = "3.456" ></testcase> <testcase classname = "Prueba de página de inicio" name = "Verificar que el título sea correcto" time = "2.345" ></testcase> <testcase classname = "Prueba de página de inicio" name = "Verificar enlaces de navegación" time = "6.789" failures = "1" > <failure message = "No se encontró el enlace de navegación 'Contáctenos'" >

CypressError: Se agotó el tiempo de reintento: se esperaba encontrar el elemento: '.nav-link[href="/contact"]', pero nunca lo encontró.

</falla> </caso de prueba> </conjunto de pruebas > < /conjuntos de pruebas >

**ChatGPT:**

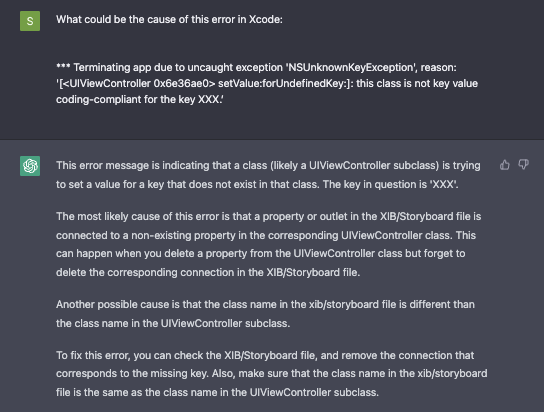
****

**Ejemplo 2:** Dar sentido a los errores del sistema/codificación con ChatGPT.

**Mensaje:** “¿Cuál podría ser la causa de este error en Xcode?

*\*\*\* Finalización de la aplicación debido a una excepción no detectada 'NSUnknownKeyException', motivo: '[<UIViewController 0x6e36ae0> setValue:forUndefinedKey:]: esta clase no cumple con la codificación del valor clave para la clave XXX.'*

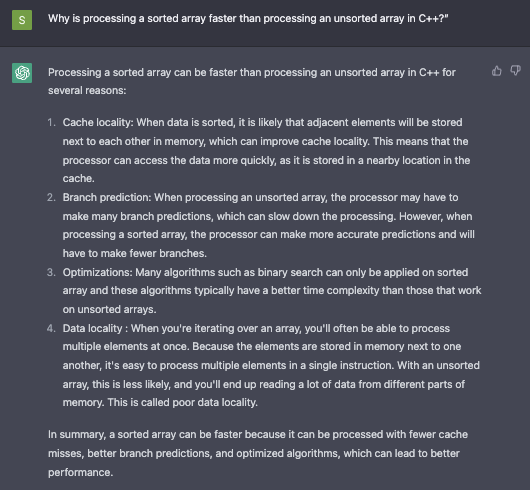
ChatGPT:

****

**Ejemplo 3:** Mejore sus habilidades en el lenguaje de programación con ChatGPT.

**Indicación:** "¿Por qué procesar una matriz ordenada es más rápido que procesar una matriz desordenada en C++?"

**ChatGPT:**

****

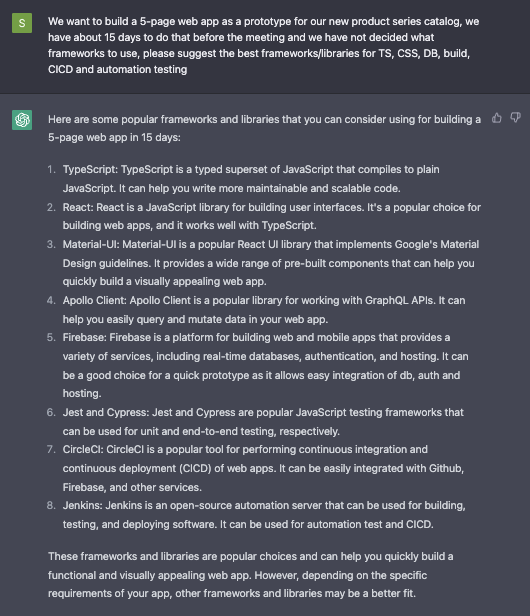
**ChatGPT como asistente de inteligencia artificial para todo uso**

Ahora, para la última sección de nuestra serie de ejemplos, veamos la capacidad más general y poderosa de ChatGPT, que es comportarse como un humano real con pensamientos y opiniones sobre varios aspectos del desarrollo, pruebas de automatización o cualquier otra cosa que desee discutir con él.

**Ejemplo 1:** ChatGPT para ayuda con decisiones técnicas.

**Aviso:** “Queremos crear una aplicación web de 5 páginas como prototipo para nuestro nuevo catálogo de series de productos, tenemos alrededor de 15 días para hacerlo antes de la reunión y no hemos decidido qué marcos utilizar. Sugiera los mejores marcos/bibliotecas para TS, CSS, DB, compilación, CICD y pruebas de automatización”.

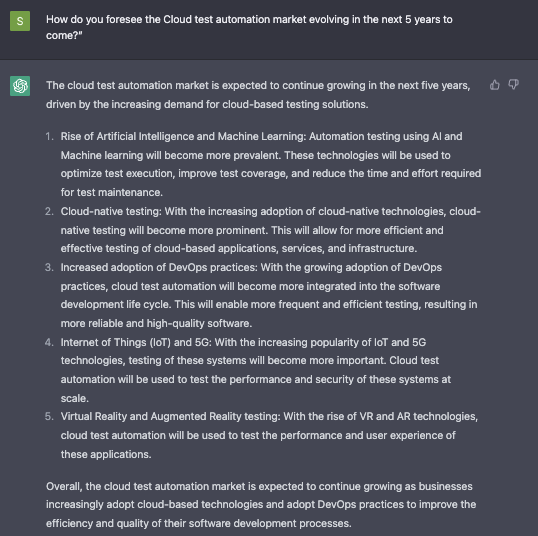
**ChatGPT:**

****

**Ejemplo 2:** ChatGPT para tareas imaginativas

**Pregunta:** “¿Cómo prevé que evolucionará el mercado de automatización de pruebas en la nube en los próximos 5 años?”

**ChatGPT:**



Limitaciones de ChatGPT

ChatGPT tiene ciertas limitaciones y está cambiando y evolucionando constantemente con una adopción más generalizada.

He aquí un resumen rápido de sus limitaciones:

1. **Falta de conocimiento del dominio:** si bien ChatGPT puede generar texto similar a la escritura humana, puede carecer del conocimiento del dominio específico o la experiencia que un experto humano tendría en ciertas áreas.
2. **Errores factuales:** dado que ChatGPT se entrena con una gran cantidad de datos de texto, puede incluir inadvertidamente errores factuales o información errónea presente en los datos de entrenamiento.
3. **Errores de código:** el código generado por ChatGPT puede no funcionar siempre, ya que puede no comprender completamente el contexto específico o los requisitos de la tarea, y puede contener errores o estar menos optimizado. Es importante verificar y probar exhaustivamente el código generado antes de usarlo en cualquier proyecto.
4. **Sesgos:** dado que el modelo se entrena con una gran cantidad de datos de texto de Internet, es posible que haya aprendido sesgos presentes en esos datos. Esto puede generar contenido ofensivo o discriminatorio.
5. **Responsabilidad limitada:** dado que ChatGPT es un modelo de aprendizaje automático, puede resultar difícil comprender cómo genera resultados específicos, lo que dificulta responsabilizar al modelo por cualquier inexactitud o error.
6. **Comprensión limitada del contexto:** ChatGPT está capacitado para generar texto en función de la información que recibe, pero es posible que no comprenda completamente el contexto o la intención de esa información. Esto puede generar confusión o una mala interpretación de la información, y es posible que la información resultante no sea del todo precisa.

OpenAI, la empresa detrás de ChatGPT, ha estado trabajando en formas de mitigar estos problemas. Es esencial siempre verificar la información generada por los modelos GPT, especialmente cuando se utilizan para tareas críticas.

Dado que se trata de un nuevo desarrollo, la funcionalidad proporcionada por ChatGPT no se limita a la automatización de pruebas.

Piense en ello como una suma de todo el conocimiento tecnológico en Internet antes de 2021, accesible a través de un chat Bot altamente funcional que puede procesar y presentar información desde diferentes perspectivas en diferentes contextos.

**Resumen**

Dominar la automatización de pruebas con ChatGPT y BrowserStack Automate es una forma poderosa para cualquier organización que busque mejorar la eficiencia y la eficacia de sus procesos de prueba.

Con ChatGPT, los equipos pueden planificar, ejecutar y analizar fácilmente sus pruebas, mientras que BrowserStack Automate les permite ejecutar esas pruebas en una amplia gama de navegadores y dispositivos.

En conjunto, estas herramientas proporcionan una solución integral para la automatización de pruebas que es fácil de usar y proporciona un control preciso sobre la dinámica de las pruebas.

Al aprovechar el poder de la IA, los equipos pueden automatizar tareas repetitivas y acceder a información más detallada sobre sus datos de prueba, lo que facilita la identificación y el diagnóstico de problemas, la predicción y prevención de defectos y la mejora de la calidad general.

Si bien existen limitaciones en el uso de GPT para la automatización de pruebas, con el enfoque correcto y los casos de uso adecuados, puede ser una herramienta poderosa que puede ayudar a los equipos a mejorar su proceso de pruebas y, en última instancia, a entregar mejores productos.