

Can GitHub Copilot Replace Automation Testers?

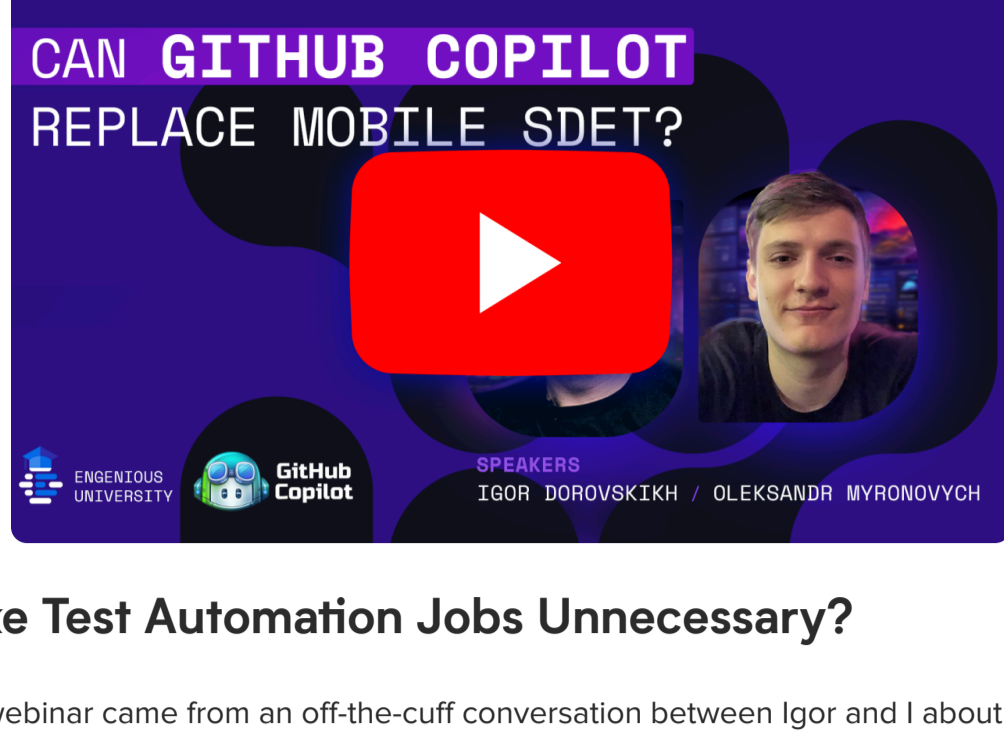
Apr 1, 2024 • AI, Career, Copilot, Mobile Testing, QA, Software Testing

Author: [Olena Vashchenko](#)

With the explosion of AI in just about every sector of the technology industry, many mobile SDETs are asking themselves if test automation tools like [GitHub Copilot](#) will eventually put them out of a job in the future. It's a fair question to ask given the big moves tech companies are making to incorporate AI into their processes more and more. However, it should seem that GitHub is banking big on their AI test automation tool Copilot, so the real question is "Should automation testers be worried?"

Recently, our own Engenious Founder and CEO Igor and I hosted a webinar called "[Can GitHub Copilot Replace Mobile SDET?](#)" During this session, we discussed the capabilities, applications, and implications of using GitHub Co-Pilot in the realm of mobile test automation. Before we continue, a little about me—I am Oleksandr, an iOS test automation guru with many years of experience working with many big-name brands on intense automation and iOS development.

In case you missed it, check out the webinar below:



Will AI Make Test Automation Jobs Unnecessary?

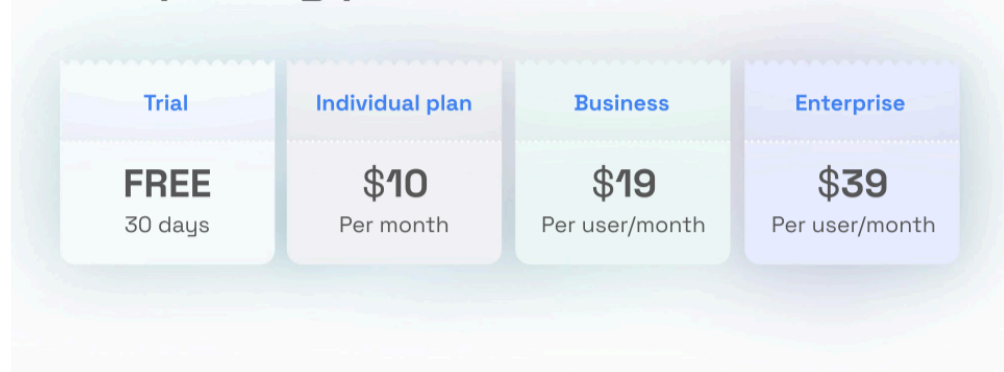
The idea for the webinar came from an off-the-cuff conversation between Igor and I about the possibility that such a powerful AI-driven test automation tool could possibly replace engineers and revolutionize test automation services moving forward. Understandably, many engineers are nervous about being replaced by software. Igor and I decided to tackle this scary topic by mapping out precisely what GitHub Copilot is, its practical uses, and possible limitations of the technology.

To put these fears about being replaced by AI from perspective, I decided to try out GitHub Copilot for myself to see what all the hype is about, how it handles test automation framework and automated UI testing tasks, and how it compares with a knowledgeable iOS development ninja like myself who wields XCUITest with finesse.

What is GitHub Copilot?

The best way I can describe Copilot is an AI-driven test automation tool that suggests code completions as developers type out code fragments by turning natural language prompts into coding suggestions based on a project's context and style conventions. It uses technology pioneered by Open AI to suggest code suggestions that align with what you want to create based on your input and predetermined conditions. GitHub Copilot was created in 2021 with the purpose of providing intelligent code completion suggestions in real time to the user.

Pricing



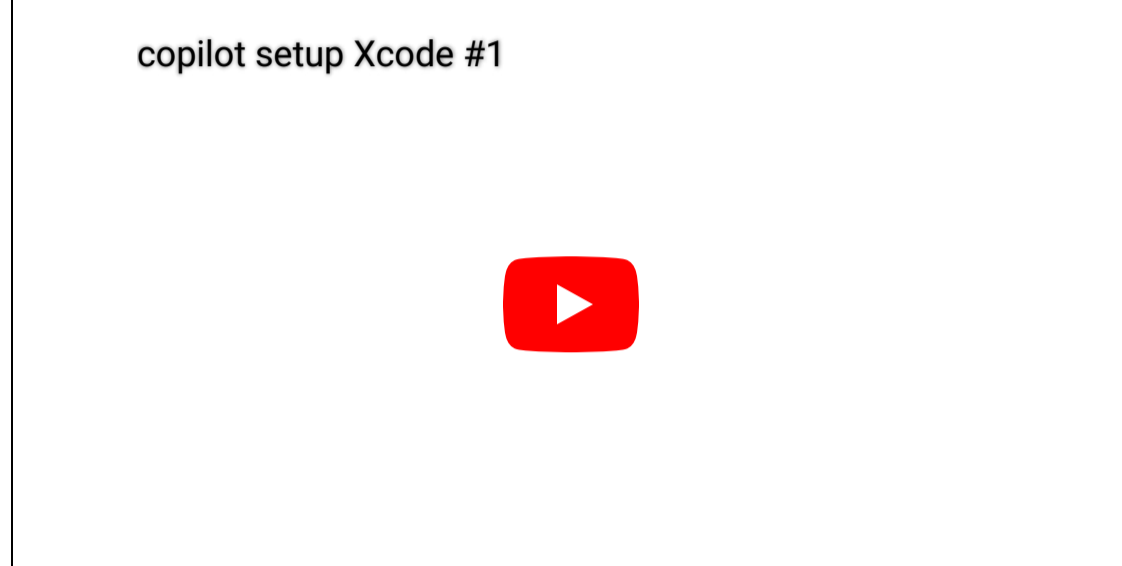
Key Features of GitHub Copilot

As with most AI-powered tools, Copilot comes loaded with some cool features that any developer is sure to like:

- **Intelligent code suggestions:** Code completion suggestions populate based on the context of what you are coding for as well as what code has already been written by a user.
- **Wide-ranging language support:** Copilot will be compatible with many popular programming languages and frameworks, including Javascript, Python, Ruby, and others. Additionally, the tool is trained to accept languages that appear in public repositories. It's important to note that the quality of code suggestions may also be affected by individual language data training and volume.
- **Adaptation to personal coding style and preferences:** Copilot will study the nuances, habits, and actions you take when coding to better anticipate your preferences when making coding suggestions for line completion. This level of personalized code suggestion helps users to code faster with less errors and inconsistencies.
- **Continuous learning:** The program learns your preferences more quickly from user feedback and the more experience it has observing individual coding styles.

How to Integrate GitHub Copilot into Xcode

- **Step 1:** Sign into GitHub to enable the Copilot feature
- **Step 2:** Download [CopilotForXcode](#) extension from the Releases tab
- **Step 3:** Enable the extension in your computer settings by following this sequence:
 - **Settings > Privacy & Security > Extensions > Xcode Source Editor > Select Done**
- **Step 4:** Link the extension to GitHub Copilot account



Use Cases for Test Automation

GitHub Copilot has proven itself to have many aces up its sleeve in terms of capability and flexibility in its usage by developers to make their work more efficient:

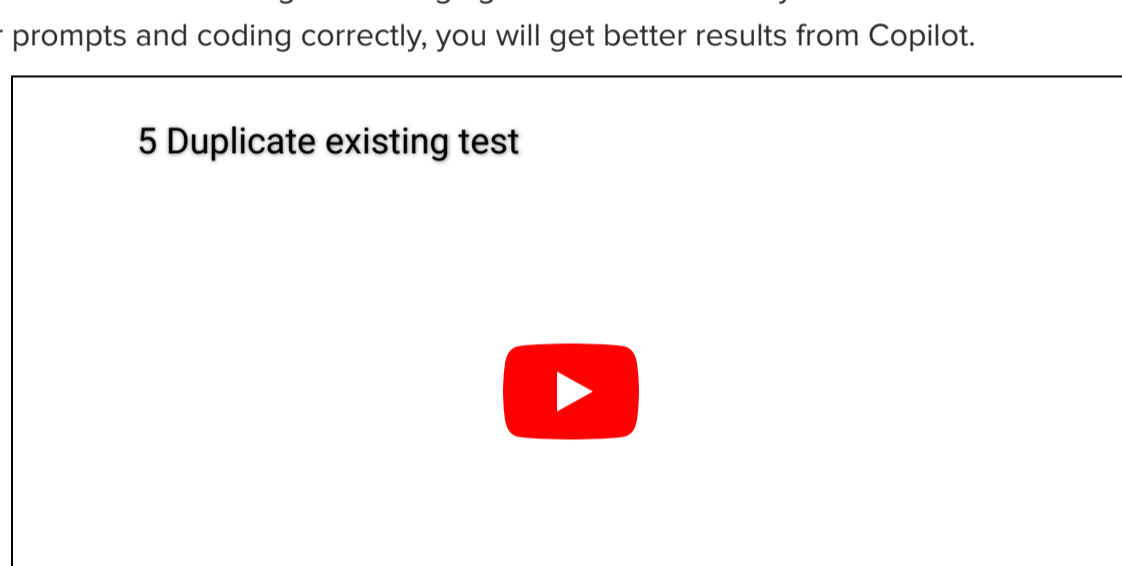
- **Rapid prototyping:** Helps developers generate code snippets on the fly and enable faster experimentation and iteration during the prototyping phase.
- **Learning new languages:** Acts as an educational tool to assist developers in learning new programming languages or frameworks by providing real-time coding suggestions based on syntax and best practices.
- **Code exploration and discovery:** Assists users in exploring code basics by suggesting relevant code snippets for increased understanding of existing projects or when contributing to open source repositories.
- **Boilerplate generation:** Common task boilerplate code generation saves time as well as helps with setting up project structures, handling configuration files, or creating standard code patterns.
- **Consistent code styling:** Maintain consistent code styling patterns across projects by generating code suggestions that align with established coding standards to reduce style inconsistencies.
- **Documentation generation:** Suggests descriptive and well-structured comments for documentation purposes to increase overall readability of code base.
- **Refactoring purposes:** Suggests alternative code snippets that are potentially more efficient to aid in code optimization with less manual effort required.

GitHub Copilot in Action

During the webinar, I went through three separate demos to showcase the functionality of GitHub Copilot in real time. Click on the links below to see the demos:

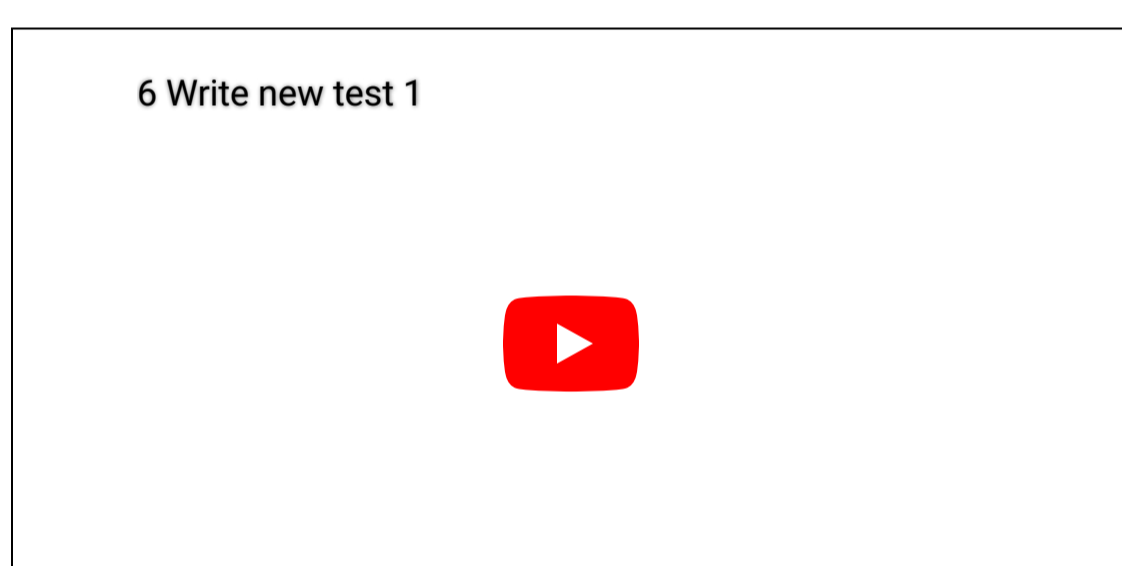
Duplicate Existing Test

In the second demo, I show how the same functionality of the first test can be duplicated for another restaurant. By initiating a function definition command and a restaurant name change, Copilot fills in the code from the previous test with the new restaurant name to match the new request. This ability to replicate lines of code from another test is game-changing in terms of efficiency. This demo shows how if you organize your prompts and coding correctly, you will get better results from Copilot.



Create a UI Test

For the third demo, I begin a brand new coding template within Copilot that will not be informed by the previous demos—I start with a clean slate. From the beginning, Copilot is able to use its analysis of what I coded before in terms of style and format to make suggestions based on the context of where I am in the coding sequence. For example, in typing my first function definition, it knew that there should be an application initialized at the beginning. Copilot uses precedents set by the user to make better suggestions as code is typed in.



Prepare for Coding Interviews with GitHub Copilot

Potential employer trick that GitHub Copilot offers users is the ability to compile work examples to present to another employees during challenge interviews. We understand that going through the interview process is already nerve-wracking on its own, so throwing in requests to demonstrate your coding skills is just the cherry on top to make things that much more complicated.

However, being able to create and compile real-world work examples with AI-guided assistance might just lead to automation testing jobs that you might not have considered. Nothing quite beats evidence of real-world experience!

- **Code examples and snippets:** Create coding examples that you can present to your potential employers that are relevant to their business or tasks you expect to complete to show you can add value to their team and understand their industry.
- **Code understanding and explanation:** Demonstrate your understanding of complex coding actions and algorithms by providing vivid explanations and comments alongside the very same generated code you are explaining.
- **Practice:** Get some practice in by having Copilot generate solutions to coding problems so you can sharpen your skills and gain experience solving such problems; additionally, you can work on solving common questions you might encounter in interviews to ensure you are well-prepared for a variety of coding techniques.
- **Code review:** Learn to better review existing code to look for ways to improve or optimize what you have already created prior to showing it off in an interview setting; additionally, being able to review code

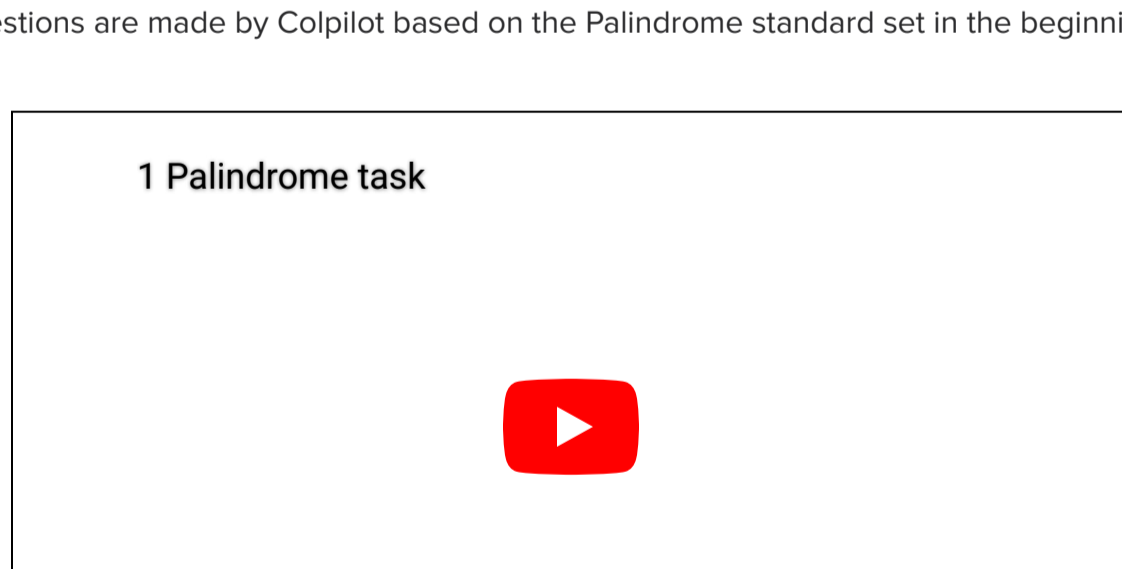
PLEASE NOTE: While Copilot can certainly be helpful in giving you ample opportunity to practice your coding craft and develop workable examples to win your interviews, do not depend on it to act as a replacement of your hard-earned skills or your understanding of principles and concepts that govern code creation. Interviewers will certainly ask you to explain such concepts to them without the assistance of any AI. It is a best practice to only treat Copilot as a helpful tool like many others, and should be treated as such to amplify the skills and knowledge you already possess.

GitHub Copilot Solving Problems

In the webinar, I gave three demonstrations of how a developer can utilize GitHub Copilot to solve common examples of practical coding problems you might come across in your work.

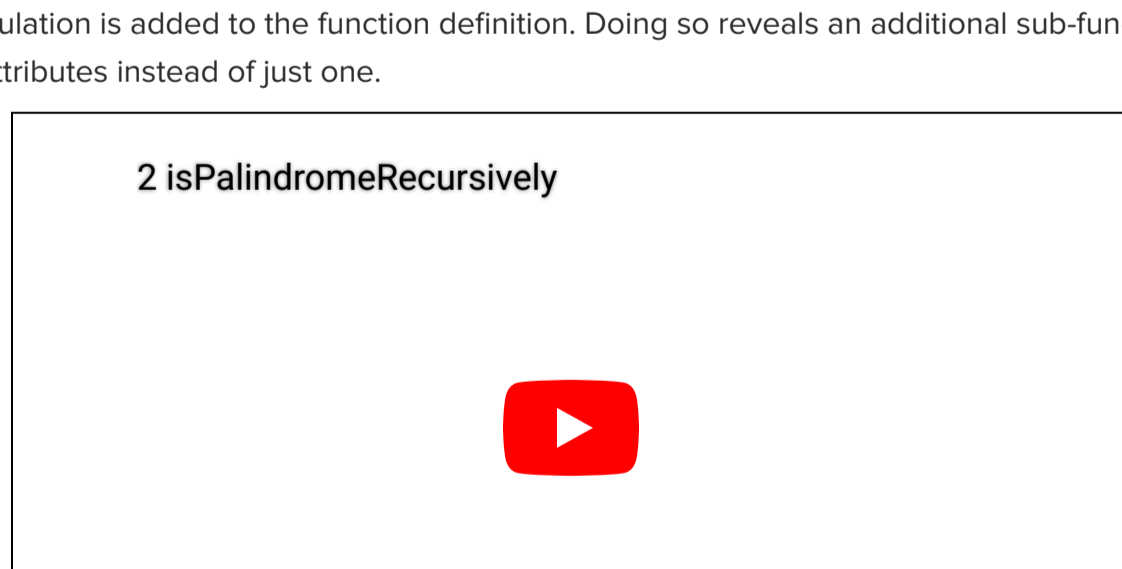
Palindrome Challenge

In the first demo, I prompt Copilot to come up with a solution to the infamous Palindrome problem. I set a standard in the beginning to define what a palindrome is and what to do if one is encountered. From there, Copilot realizes that I am trying to solve the Palindrome problem when I begin typing function definition codes. Suggestions are made by Copilot based on the Palindrome standard set in the beginning to solve the problem.



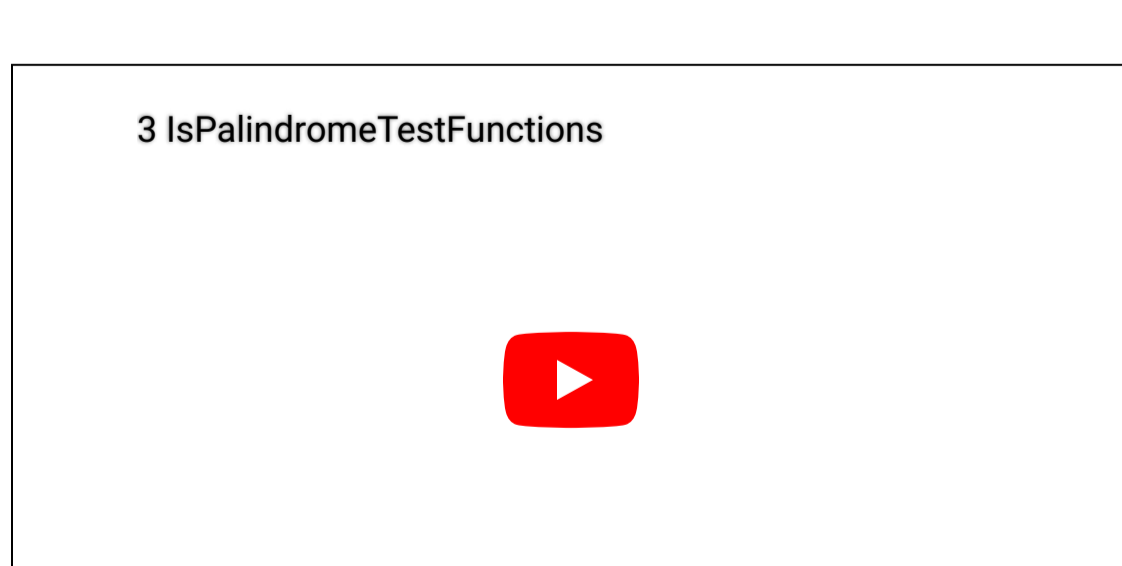
Recursive Palindrome

For the second demo, I explore what happens when the suggestions to the Palindrome problem are not acceptable. We may want to be more specific in our request to get a more recursive solution, so the recursive stipulation is added to the function definition. Doing so reveals an additional sub-function that takes three attributes instead of just one.



Writing Tests

During the third demo, I try out multiple test functions to see how they handle trying to solve the Palindrome problem to see which ones work best. In this example, I test for odd and even numbers of characters in the code.



GitHub Copilot Advantages

Now that we have covered the what GitHub Copilot is, its functionality, and how best to implement it, let's quickly summarize what makes the AI tool great for developers:

- **Intelligent suggestions for faster development:** Copilot provides context-aware code completions utilizing knowledge from vast data sets. Additionally, it learns from your coding style and tailors its suggestions to match the precedents you set in relation to syntax and context.
- **Accelerated onboarding:** New team members can be provided specific tasks that can be generated by Copilot to ensure they understand their role quickly. Additionally, the tool can help to create code completion templates for common tasks to simplify learning of complex code structures.
- **Increased confidence:** Even the most experienced developers can feel unsure of themselves when coding for something out of their wheelhouse. Copilot provides extra guidance when needed to help navigate new, unfamiliar tasks and make intelligent suggestions to boost the confidence of developers who use it to be more efficient and accurate.

Potential Challenges and Solutions

Just as there are many benefits to using auto-generated lines of code when you are in a groove, there are just as many challenges to address and keep in mind when utilizing AI-driven tools like GitHub Copilot. Here are some potential pitfalls to avoid when consulting AI to finish your tasks:

- **AI suggestions can be distracting when creating your logic:** Depending on your level of focus, Copilot may make too many code completion suggestions; too many interventions from the tool can be overwhelming for some developers, forcing them to lose concentration and ultimately slow productivity.
- **Developing an over-reliance on suggestions:** Whether due to complacency or a lack of vigilance, some developers may make the mistake of taking just about every code completion suggestion from Copilot without evaluating its validity or contextual accuracy.

The Verdict: Will AI-Driven Tools Replace Engineers in Test Automation?

The short answer is: no, at least not any time soon. By the end of our conversation, Igor and I had concluded that while AI-powered test automation tools like GitHub Copilot are impressive in its capabilities and ability to adapt to developer coding habits, it's not so smart that it can totally replace the human element in mobile app development. A person is still needed to assess every code completion suggestion to ensure that it fits the context of the task as well as its purpose.

So for now, relax and take solace in the job security of your hard-earned career creating iOS apps, my fellow developers. In fact, I suggest that you at least do a trial run of Copilot to see how it suits your coding style and find out for yourself if AI can help you to be more efficient and productive.

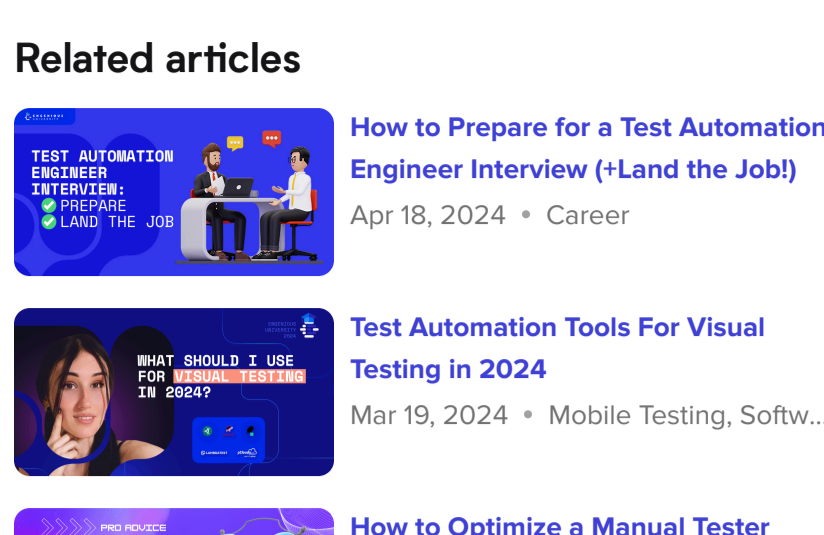
Thanks for joining us on this very critical crossroad of human ingenuity and AI capability. Be sure to sign up early for all [Engenious University webinars](#) to reserve your seat and participate in the conversation live! If you cannot make it, no sweat—registered people get a recording of the webinar sent to them FREE. See you at our next great conversation!

Share on: [Facebook](#) [Twitter](#) [LinkedIn](#) [Email](#)

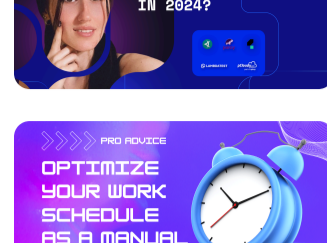
Contents

- [Will AI Make Test Automation Jobs Unnecessary?](#)
- [What is GitHub Copilot?](#)
- [Pricing](#)
- [Key Features of GitHub Copilot](#)
- [How to Integrate GitHub Copilot into Xcode](#)
- [Use Cases for Test Automation](#)
- [GitHub Copilot in Action](#)
- [Prepare for Coding Interviews with GitHub Copilot](#)
- [GitHub Copilot Solving Problems](#)
- [GitHub Copilot Advantages](#)
- [Potential Challenges and Solutions](#)
- [The Verdict: Will AI-Driven Tools Replace Engineers in Test Automation?](#)

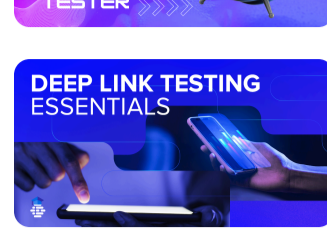
Related articles



How to Prepare for a Test Automation Engineer Interview (+Land the Job!)
Apr 18, 2024 • Career



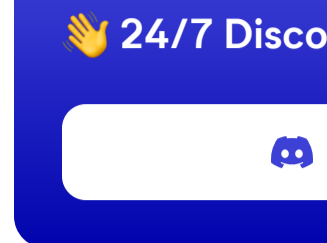
Test Automation Tools For Visual Testing in 2024
Mar 19, 2024 • Mobile Testing, Software



How to Optimize a Manual Tester Overwhelming Work Schedule?
Jan 9, 2024 • Career, Mobile Testing, Software



The Essential Guide to Deep Link Testing: Why It Matters
Dec 21, 2023 • Mobile Testing, QA, Software



November Retrospective. Mobile and Software Testing Industry Digest
Dec 6, 2023 • Mobile Testing, QA, Software

24/7 Discord Support

[Join our Discord](#)

Subscribe to get our latest news

[▶](#)