



Intuit TurboTax Tests Design System Changes at Scale with Applitools Eyes and Ultrafast Grid

“

One of the really handy pieces of [Ultrafast] Grid is that we can have a screenshot test in Edge, Chrome, and Safari, and we can also do that at different breakpoints. So we can make sure on the smaller screen size everything is responsive the way it should be. That really helps us gauge that any individual change is having the impact that we expected for any of those components.”

Tyler Krupicka
Senior Software Engineer at Intuit

Statistics

50+ UI components completely covered

1000s of potential screen combinations tested

Highlights

- Intuit maintains its design system to help create more consistency and brand recognition, as well **increasing productivity from reusable components**.
- Testing Intuit’s design systems requires thorough documentation of reusable components and their expected behavior and treatment.
- Applitools helps Intuit run visual regression tests on TurboTax’s over 50 UI components and screen configurations across browsers and viewport sizes, **reducing manual testing of thousands of potential screens**.

About Intuit

Intuit creates finance software for its over 100 million customers. Its three main products are tax preparation software TurboTax, small-business accounting software QuickBooks, and personal finance app Mint. This case study focuses on TurboTax, which provides users with an interview-style experience to complete the proper tax forms in a more intuitive way, making filing tax returns easier.

How Intuit Builds Digital Products

TurboTax has thousands of possible screens presented to users during the filing process. These screen configurations use the over 50 components that make up the TurboTax UI. To build a consistent voice in its products, Intuit maintains a well-documented design system.

Design systems consist of components of an application and their documented use patterns. Using these components provides a more consistent user experience and branding of an application. These can include:

- Typography, colors, spacing
- UI elements like form fields and buttons
- Graphics and animations

The reusable code and UI components of Intuit's design system not only increase speed of development for TurboTax, but also make it easier to test its components individually instead of instances across multiple screen possibilities.

How Intuit Tests Its Design System

A change across a design system could be a styling update. For example, buttons may be updated to have different fonts, stroke, fill, hover state, or focus state to align to updated branding or accessibility improvements. Effectively testing changes to a design system requires having thorough documentation of the UI components and their interactions in defined use cases.

It also helps to include automated testing in your CI/CD pipeline to catch bugs before they reach production.

Intuit ensures a consistent and accessible user experience for TurboTax through its thorough testing process for the design system in Storybook:

- **Unit and integration testing:** Intuits uses various tools to test that interactions like button presses and keyboard navigation worked as expected.
- **Accessibility testing:** Intuit uses Axe to run [accessibility tests](#) like contrast checking and validating focus states and tab order.
- **Performance testing:** Intuit tests bundle sizes for its applications, as well as manual responsiveness testing.
- **Visual testing:** Intuit uses Applitoools Ultrafast Grid to run visual tests on its design system updates.

How Intuit Uses Applitoools for Visual Testing Across Screen Configurations

The design system use case mock-ups Intuit builds in Storybook will be screenshot tested between changes in the system. Intuit uses automated visual AI with Applitoools to compare these changes and catch potential visual regression bugs. This visual testing happens on both the component level and the mock-up level before release to ensure individual components are correct and that their changes are having the expected impact across the design system.

The mock-ups are used to compare how component updates will change individual screens of the application. These impacts to the overall design can include spacing shifts, consistency, and nested components.

Ultrafast Grid enables these visual regression tests to be run on all configurations in the Storybook mock-ups. It also lets Intuit test across different browsers and breakpoints to ensure that the screens look correct on mobile devices and full desktop experiences.

Testing Your Own Design System With Intuit and Appliteools

Testing that your application is consistent and accessible starts with your design system. Intuit has a few open source tools that can help developers and designers get started setting up their own design system for their applications.

Test management for design systems is challenging in traditional testing frameworks. [Design System-CLI](#) is an Intuit open-source tool that provides a pre-configured version of Storybook that includes Intuit's tools for creating new components and bundle size tracking for performance testing

Automating testing in Storybook the way you would with your web application isn't possible using just test automation tools like Selenium, Cypress, or WebdriverIO. [Proof](#) is another open-source tool from Intuit that lets you write UI automation tests that target specific stories in Storybook so that you can use WebdriverIO to automate interactions with your components to test. Testing that your design system behaves as expected across different combinations manually would take too much time. [Appliteools Ultrafast Grid](#) supports automated visual testing across different combinations of browsers and viewport sizes.

About Appliteools

Appliteools delivers a next-generation test automation platform through Visual AI and the Ultrafast Test Cloud. We enable engineering teams to release high quality web and mobile apps at incredible speed and at a reduced cost.

Appliteools Visual AI modernizes important test automation use cases to transform the way organizations deliver innovation at the speed of CI/CD at a significantly lower Total Cost of Ownership (TCO):

- Functional Testing
- Visual Testing
- Web and Mobile UI/UX Testing
- Cross-Browser Testing
- Responsive Web Design Testing
- Cross-Device Testing
- PDF Testing
- Accessibility Testing and Compliance Testing

For more information, visit appliteools.com.