

Project Plan Presentation

Improve High Contrast Mode for Firefox

The Capstone Experience

Team Mozilla

Noah Pesta

Shaoting Huang

Danielle Lamoureux

Avi Pasula

Jack Ying

Department of Computer Science and Engineering

Michigan State University

Fall 2021



*From Students...
...to Professionals*

Functional Specifications

- Improve on already existing high contrast mode (HCM) within the Firefox desktop app
- Main purpose for this mode is to provide more accessibility to those with visual impairments such as low vision or difficulty distinguishing between colors
- Our team will be fixing several bugs found within HCM to improve on the interface and functionality of the Firefox browser across Windows, macOS, and Linux

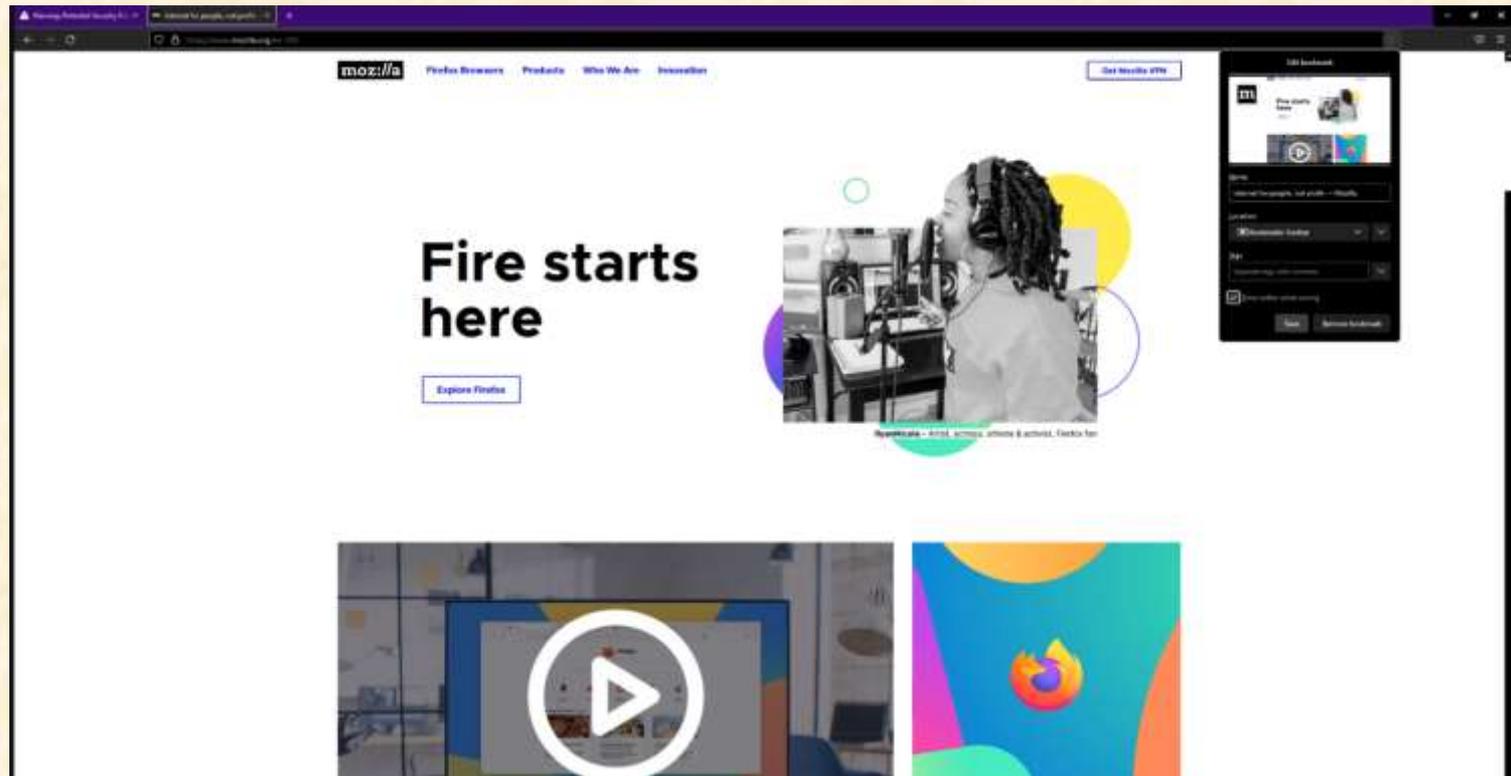


Design Specifications

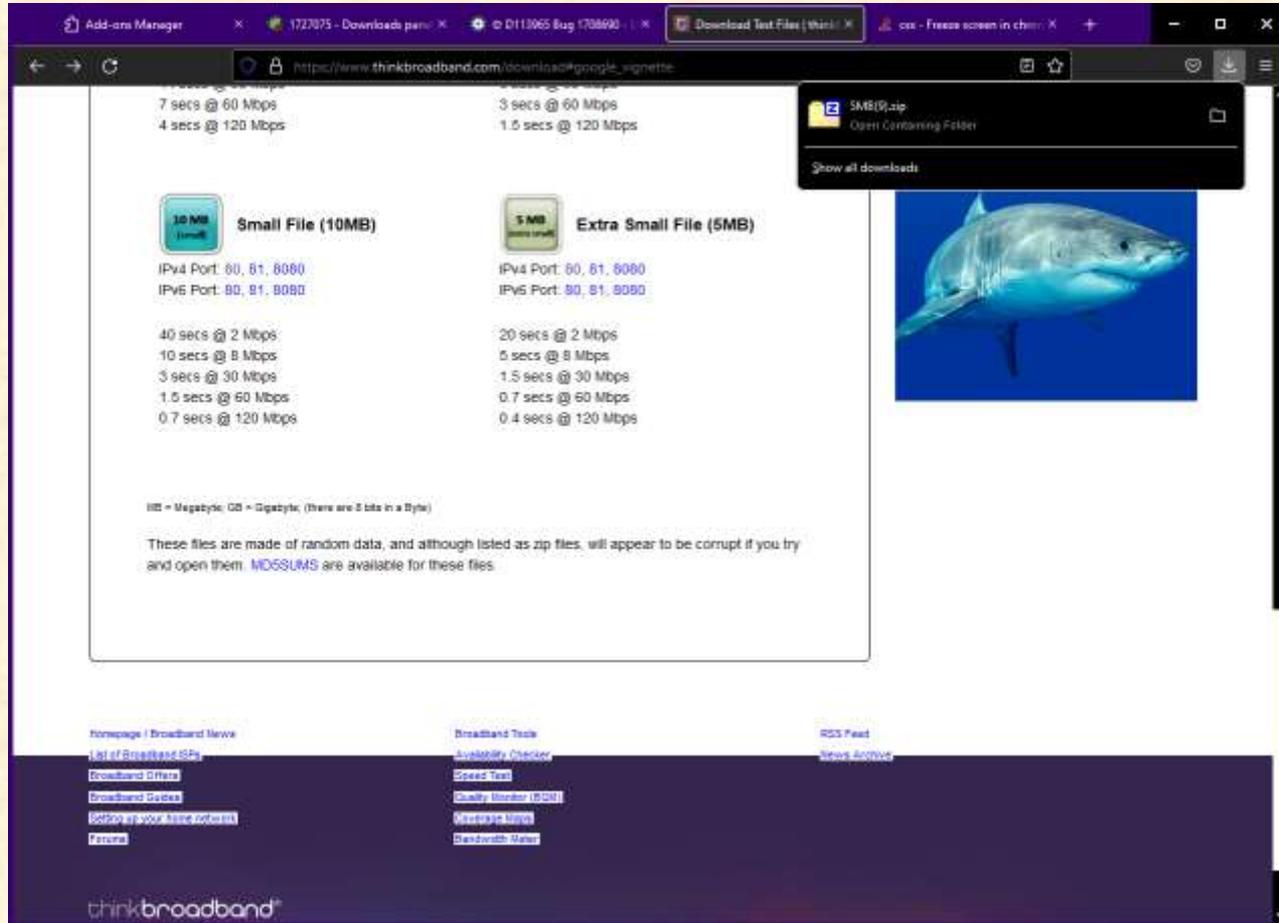
- UI must remain consistent with the non-HCM Firefox standard across all OS
 - Highly contrasting colors and distinction between elements (darker text, outlined elements, etc.)
- Changes will be made to both dark and light theme HCM
- Some features within the browser that will be improved on:
 - Buttons
 - Checkboxes
 - Tabs bar
 - Menus
 - Panels



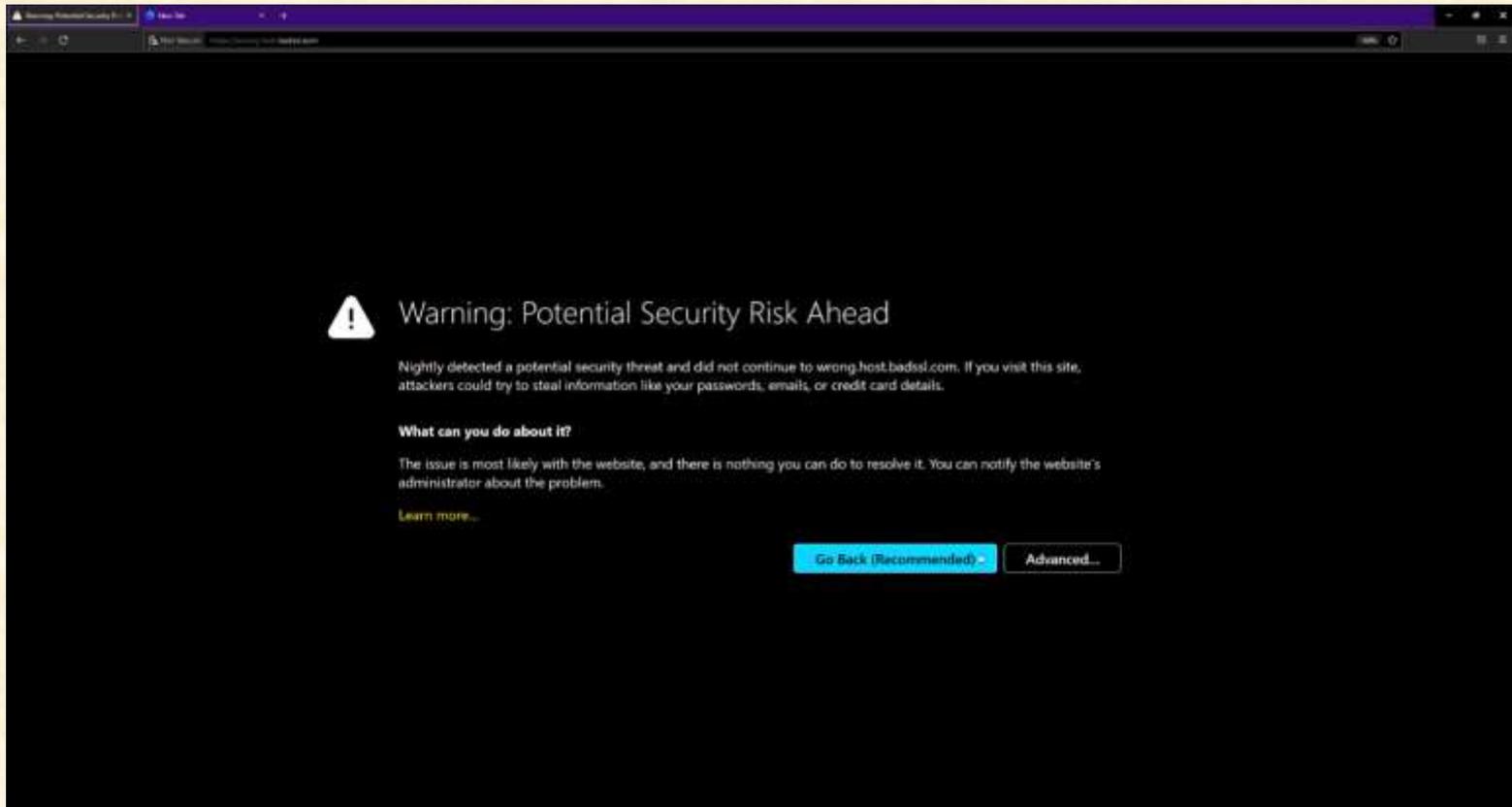
Screen Mockup: Fix accent color highlights in HCM



Screen Mockup: Improve icon hover visibility in HCM



Screen Mockup: Add HCM hover color to buttons



Screen Mockup: Adding HCM tab buttons

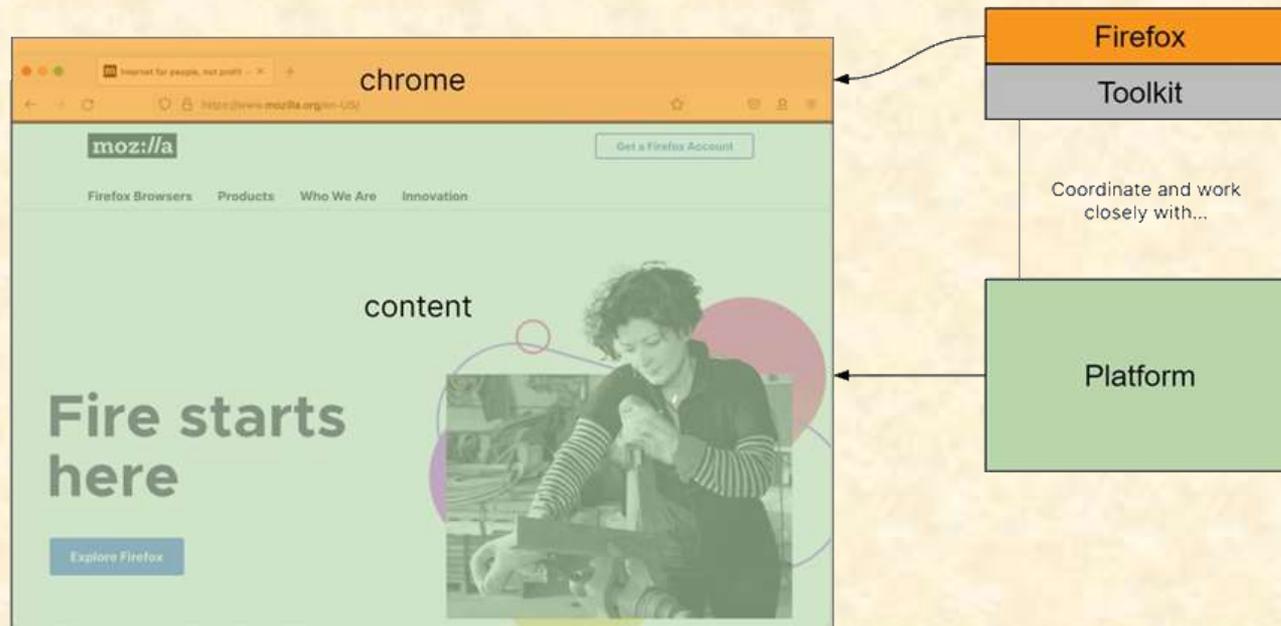


Technical Specifications

- High Contrast Mode (HCM) will be enabled in settings of OS before working on bugs assigned
- Need ability to replicate and fix bugs across Windows, macOS, and Linux
- Fixes to features focused on the frontend
 - HTML, CSS, and JavaScript



System Architecture



System Components

- Hardware Platforms
 - MacOS – One group member has a MacBook and the labs are equipped with iMacs.
 - Windows – Most of the members in our group have computers that run Windows 10.
 - Linux – Couple of our group members run Ubuntu for access to a Linux environment.
- Software Platforms / Technologies
 - HTML – core structure and content of the page.
 - CSS – "makeup" of the page. Adds styling rules to the HTML.
 - JavaScript – "Brains" of the page. Allows for more advanced features than just text and pictures.
 - Firefox Dev Tools – Tools used by developers to view HTML, CSS and JS within the browser
 - Mozilla Build Tools – Tools used to build and run the Firefox codebase.



Risks

- Bug Replication
 - Description: Some of the bugs can be hard to replicate as sometimes there is a very specific process to get the bugs to happen. We have also had issues with bugs only happening on certain operating systems.
 - Mitigation: To mitigate this risk we will keep in constant communication with the bug reporter and use them as a resource on how to replicate the bug. For the OS issues we mitigate this by testing all our fixes on all the operating systems.
- Fixes causing old features to break
 - Description: Due to the size of the code base being so large some of the fixes that we implement may cause some other feature that were previously working to break. It is important that our changes do not cause a regression in Firefox's features.
 - Mitigation: To mitigate this risk we will make sure to test our changes extensively. We will also make sure that all code gets reviewed and tested by a third party to make sure it doesn't break anything.
- Complications with cross OS code implementation.
 - CSS is handled differently depending on what operating system you use. This can be difficult to work around because a fix on one OS may not work on another.
 - Mitigation: Try and use generic or non-OS specific code to write fixes.
- Version Control Software (VCS)
 - Description: We are using Mercurial, we are not quite familiar with it, which means we are facing a whole new way of handling version controls. There have already been many issues requiring rebasing and solving merge conflicts.
 - Mitigation: Use the mercurial documentation and consult with client if there are issues.



Questions?

?

?

?

?

?

?

?

?

?

