

Mod Updates on the *Fallout 4* Main Menu

emilio@neurostyle.net

Need: Downloadable game modifications can affect (and effect) stability and performance, as well as add new features, so users should stay up-to-date.

Problem: Users only know when mods are updated after they've opened the mod browser.

Solution: Display a mod update notification outside the mod menu.

User story

Linda loves mods, and she carefully picks through the categories to groom a collection of several dozen mods for her personal wasteland experience. There are a couple armor and weather mods that she knows can harm performance on her Xbox One, but they are just too great to pass up. Now that everything is to her liking and her playthrough is underway, she has no need to open the mod menu; whenever she opens the game she just selects "CONTINUE" on the main menu as quickly as she can.

"It sucks that the game chugs when the awesome-looking storms roll through, and it's even crashed a couple times. But, whatever, it mostly works. That's how mods are, right?"

Linda has accepted a mediocre-quality experience because she thinks of *Fallout 4* and its mod system as a static game, rather than a platform for continual and arbitrary customization. She needs a reminder that the platform exists and changes for the better - she needs mod update notifications.

Feature

Visual notification of updated mods on the game's main menu.

Requirements: Be easy to implement (few releases are left, if any). Do not be ambiguous; do not look like an error or warning. Do not visually disrupt the main menu screen. Disappear when unnecessary.

Development question: When can connections be made? How early can the check start, and how long does it take? Ideally, it would start before the intro video and finish before the main menu is displayed.

Possible issue: They've probably already considered this. I imagine that if they could have done it by now, they would have. Still, in-game Bethesda.net integration should (very, very carefully) grow to underscore the platform experience.

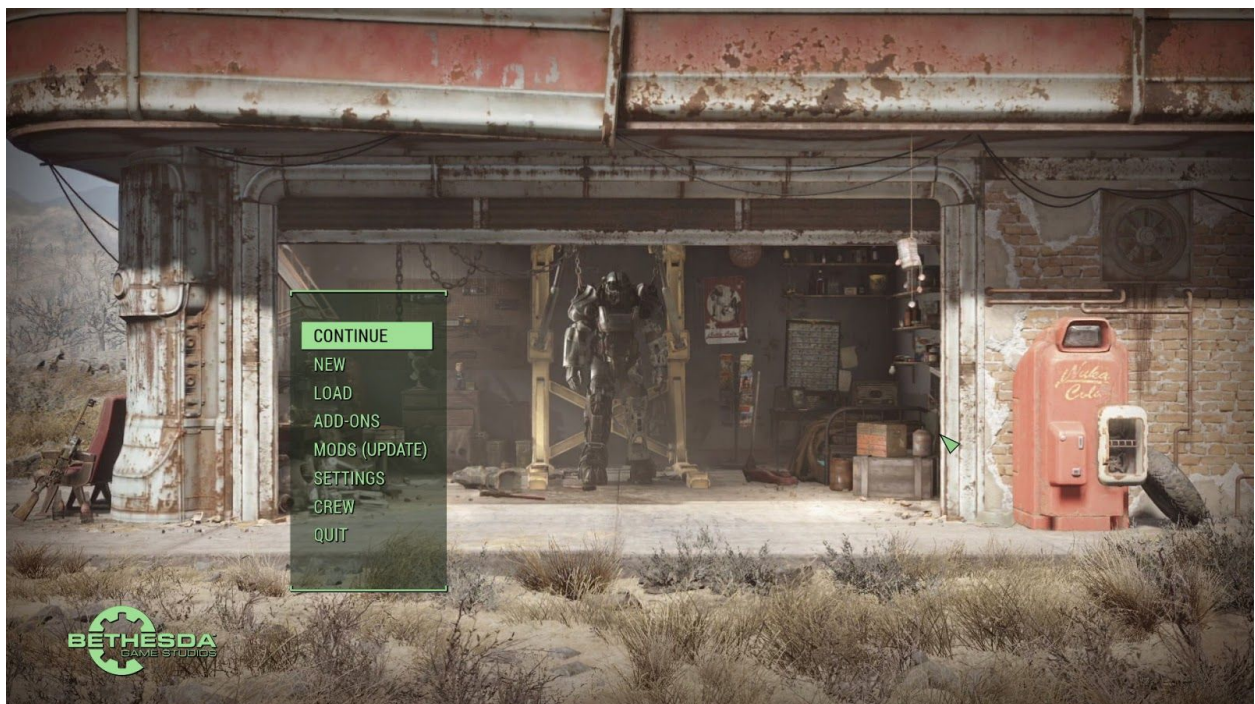
Simple implementation: Text-only

Append a string to the Mods menu item.

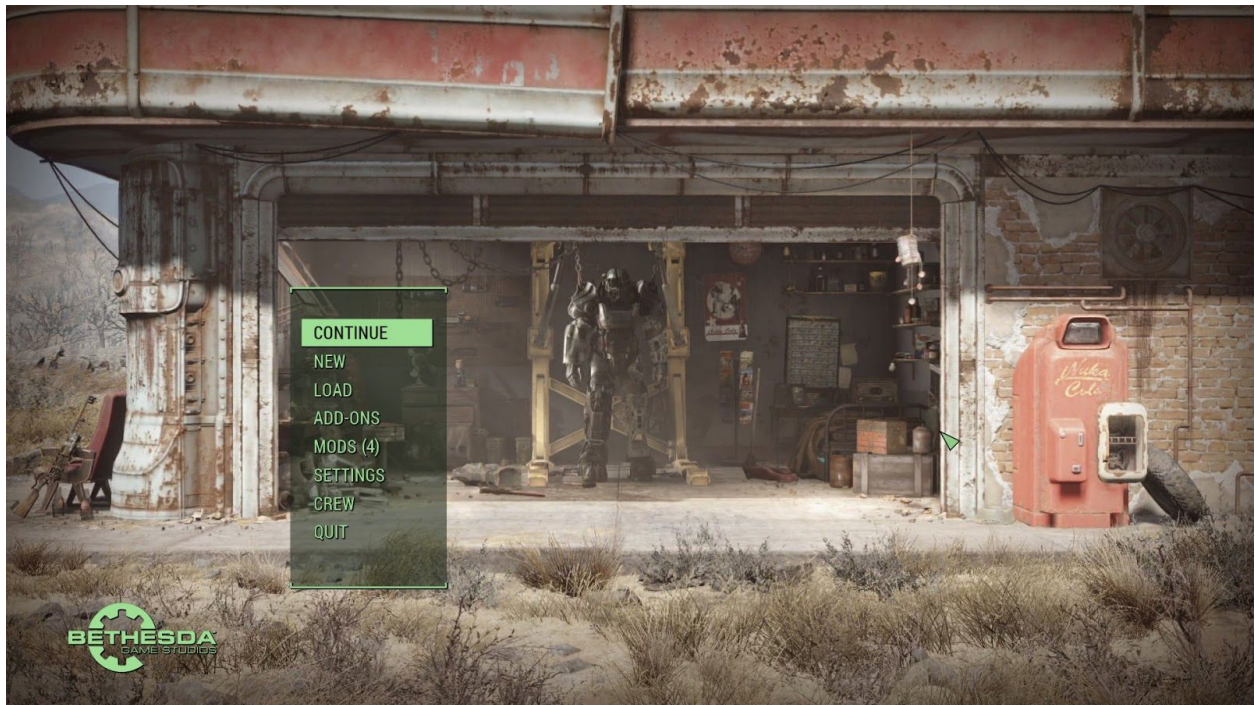
Pros: Trivial UI implementation.

Cons: Not a significant visual change to easily alert the user. The "update" string would be a localization nightmare for over-running that line. The number-only state may be ambiguous.

Update string:



Update number:



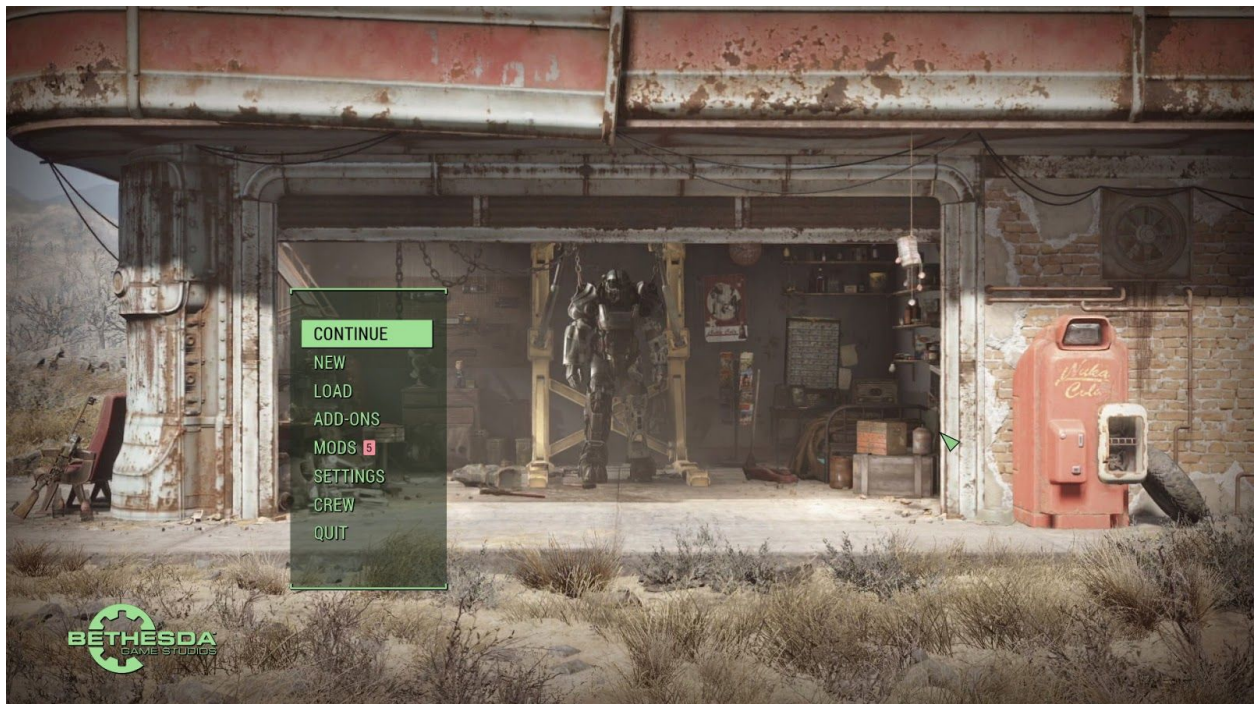
Mid-complexity implementation: Alert badge

Add a badge to the Mods menu item.

Pros: Common modern design language. Compact. Very noticeable in a different color (the red & blue are triadic colors calculated from the Pip-Boy color preference). A little more friendly for localization.

Cons: More than just a text string. The "update" text version may still be too long in some languages.

Four updates:



444 updates:



Update strings & other colors:



Fancy implementation: Icon notifications

Combine the mod "box art" icons with the badge.

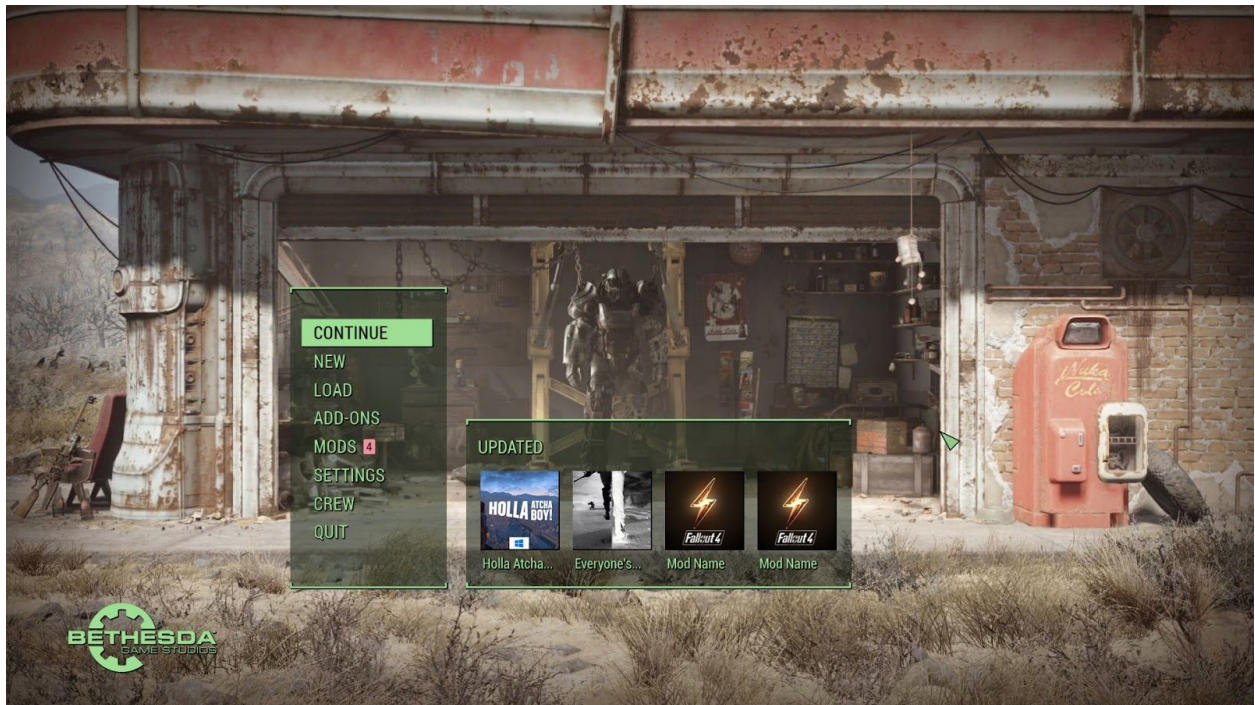
Pros: Immediately noticeable; visually engaging & attractive. Gives mods more visibility on the main screen. Flexible; could be re-used to display featured mods. Makes the main menu feel like a part of the platform.

Cons: A new piece of UI, not just an addition to the existing list. Mod icons might be ugly. "Updated" may be too long for the container in some languages (diff string? icon?).

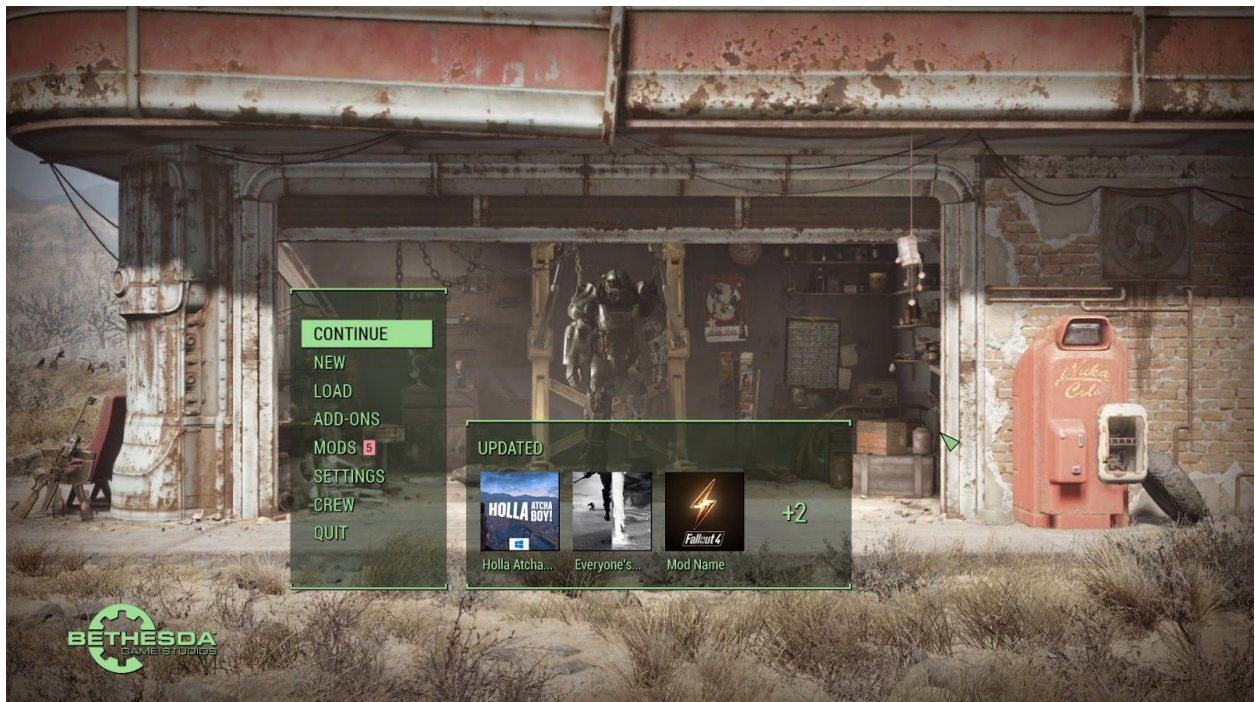
One update:



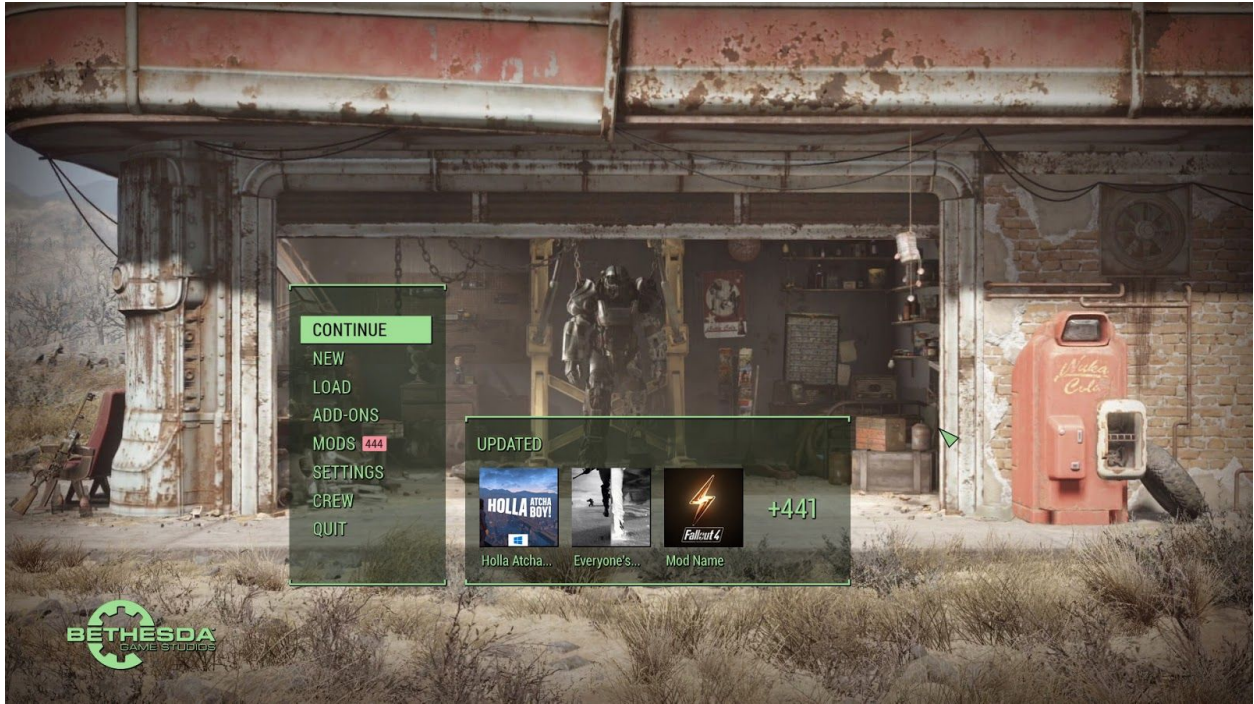
Four updates:



Five updates:



444 updates:



The box is one target:



Implementation issue: Connection status

Connection status should be subtly displayed by pulsing a text string in the corner. If there are no updates or errors, the string disappears and the main menu looks normal.

