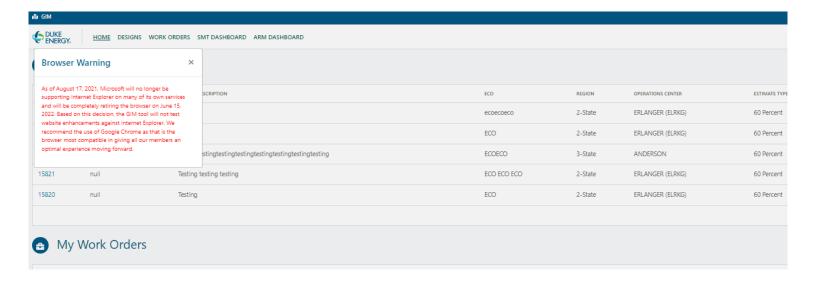
## CADT — 1642 — Add "Do not show again" option on the Browser Warning

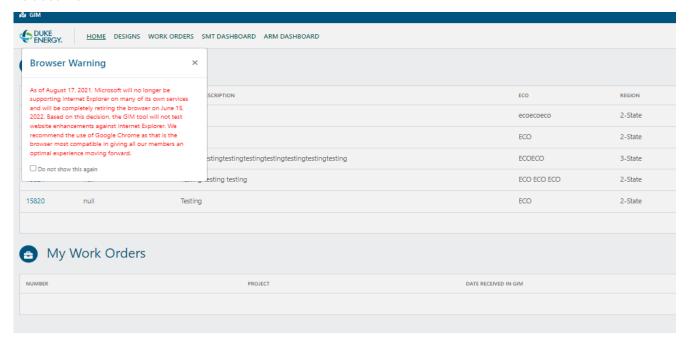
Add an option under the Browser Warning message in Microsoft Edge that allows a user to dismiss the message for all future logins.

This could be a check box and to the right of it we display "Do not show again"

**Analysis:** When the user logins using Microsoft Edge Browser window then this Message Warning Popup will come after successful login inside the application so now the requirement is like we have to remove this warning popup message for all the future logins for the user as screenshot attached below:



**Solution:** We must implement both Ui and backend change for this first need to implement checkbox into the warning modal box so that when the user click checkbox do not show again then when user closes the browser at that time when again user login inside the application at that time, he or she might not be able to view the document.



Here checking if the popup be shown for the logged-in user

```
// Check if the popup should be shown for the logged-in user
@GetMapping("/checkUserPopupStatus")
public ResponseEntity<?> checkUserPopupStatus(HttpSession session) {
    UserTim userTim = (UserTim) session.getAttribute("userTim");
    if (userTim != null) {
        boolean showPopup = userService.shouldShowPopup(userTim.getLoginId());
        return ResponseEntity.ok(Collections.singletonMap("showPopup", showPopup));
    } else {
        return ResponseEntity.status(HttpStatus.UNAUTHORIZED).body("User not logged in");
    }
}
```

## Updating the popup status for the logged in User

```
// Update the popup status in the database for the logged-in user
@PostMapping("/updatePopupStatus")
public ResponseEntity<?> updatePopupStatus(HttpSession session, @RequestBody PopupStatusRequest request) {
    UserTim userTim = (UserTim) session.getAttribute("userTim");
    if (userTim! = null) {
        userService.updatePopupStatus(userTim.getLoginId(), request.isHidePopup());
        return ResponseEntity.ok("Popup status updated");
    } else {
        return ResponseEntity.status(HttpStatus.UNAUTHORIZED).body("User not logged in");
    }
}
```

Here checking the popup based on the userId (isShowPopup in the entity class determine the Boolean value for True or False)

```
// Check if popup should be shown for a user
public boolean shouldShowPopup(String userId) {
    User user = userRepository.findById(userId).orElse(null);
    if (user != null) {
        return user.isShowPopup(); // Assuming you have this field in your User entity
    }
    return true; // Default to true if the user doesn't exist or no record found
}

// Update popup status in the database
public void updatePopupStatus(String userId, boolean hidePopup) {
    User user = userRepository.findById(userId).orElse(null);
    if (user != null) {
        user.setShowPopup(!hidePopup); // Set to false if user wants to hide the popup userRepository.save(user);
    }
}
```

Here using AJAX call we are updating the popup status inside the database with Post call.

This way we can now ensure this like if the user opted the checkbox (Do Not Show) then for the logged in user it will remain hidden for all his future logins.

