Aaryaman Akhouri , Taylor Mannion, Uday Talasila , Thomas Powderly

Project Milestone 4

Process Deliverable III

Our team has opted to use the Incremental SE Process for our project. As such, we will be summarizing how our test cases map to our project scope, requirements, and design.

Our test cases map well to our project scope, directly testing the limits of the functionality of our project. For example, one of the more in-depth aspects of our project would be blocking certain website links while the pomodoro timer is active. While not a core requirement, it would be a great benefit to everyone involved, and the addBlockedWebsite/removeBlockedWebsite test cases serve as goals to fully encompass our project scope. Our test cases also map well to our requirements. Our core features of events, todo lists, and timer start/stop functionality all have tests present in the form of addEvent, addToDo startPomodoroTimer, removeEvent, removeToDo, and pausePomodoroTimer. Finally, our test cases map well to our project design. Our current project design uses a View-Model-Controller design. The test cases could operate on many different levels of our design, from testing the actual button presses in the View to testing the actual functionality in the Model to testing the connecting code in the Controller. For example, the addEvent test case could process when the user presses the "Add Event" button, and then could process when the Controller tells the Model to add an event to the event database.

Black Box Test Plan

Test ID	Description	Expected Results	Actual Results
addEvent	Preconditions: User is on FocusExtension Calendar, user has reminder preferences set Steps: User clicks on day in calendar, clicks "Add Event", fills out event title, chooses time, and picks a color Test Inputs: Normal event, "empty" event, event with extremely long title, event with	Test Outputs: Normal event creates event with automatic reminders at specified intervals before the event. Empty event (no event title written) prompts user to write a title. Each event title is capped at 100 characters, user is prompted to write a shorter title for the event if the title they gave is over 100 characters. Event set at impossible hours (e.g.	

	impossible times	0th hour or 25th hour on military time) prompts user to redo input for time.	
addBlockedWebsite	Preconditions: User is on FocusExtension Pomodoro timer Steps: User clicks "Add" on "Blocked Websites" section of Pomodoro Timer, types website URL Test Inputs: Valid URL, invalid URL, empty URL	Test outputs: Valid URL adds a website to the blocked list. Invalid URL (URL of a website that doesn't exist) prompts user to retype the URL. Empty URL prompts user to type a URL,	
startPomodoroTimer	Preconditions: User is on FocusExtension Pomodoro Timer, user has some websites configured in the blocked websites list Steps: User clicks on "Start" button for Pomodoro TImer Test inputs: starting timer with no blocked websites configured, starting timer with at least one blocked website configured	Test outputs: No blocked websites configured starts the timer but gives a message to the user telling them that there are no websites to block, allowing them to add websites if they want. Starting the timer with at least one blocked website configured starts the timer as normal and blocks the website(s) in the blocked list for 25 minutes.	
addToDo	Preconditions: User has FocusExtension set up, user is on FocusExtension To-Do section Steps: User clicks on "Add Task", fills out information Test inputs: Task with title and deadline, task with empty title and/or	Test outputs: Task with proper title and deadline adds it to the calendar. Task with empty title and/or deadline prompts the user to type in a title/deadline. Task with impossible deadline (e.g. 0 or 25th hour or 31st of a month that only has 30 days) prompts the user to	

	deadline, task with impossible deadline	retype the deadline.	
addNote	Preconditions: User is on FocusExtension Notes

Steps: User clicks on "+" button to add a note, is given option to make a title for the note, fills up a note

Test inputs: Empty note, note with specific title, note with default title | Test outputs: Empty note adds a note with no text, with either the title the user gave or the default title "Note [number]". Note with a specific title adds that note with its title. Note that is not given a title by the user is added with the default title "Note [number]". | |
| resetPomodoroTimer | Preconditions: User is on FocusExtension Pomodoro timer, there is a timer currently running

Steps: User clicks the reset button

Test inputs: Running pomodoro timer | Test outputs: Timer resets to the length of a work period, and the number of work and rest periods that have elapsed reset to 0 | |
| removeBlockedWebsite | Precondition: User is on FocusExtension Pomodoro timer, there is at least one website blocked currently

Steps: User right clicks on blocked website which opens a menu that has a read delete button. User clicks that button

Test inputs: Website url that is blocked currently | Test outputs: That website is removed from the list of blocked websites, and is no longer blocked | |
| pausePomodoroTimer | Preconditions: User is on FocusExtension Pomodoro timer, there | Test outputs: The timer stops at the current time, number of elapsed | |

	is a timer currently running Steps: User clicks the pause button Test inputs: Running pomodoro timer	work and rest periods stay the same	
removeEvent	Preconditions: User is on FocusExtension calendar, has successfully create an event Steps: User clicks on the event in the calendar, clicks the remove button, and confirms that they want to remove it Test inputs: Created event	Test outputs: Removes the event from the calendar	
removeToDo	Preconditions: User is on FocusExtension Todo Steps: user clicks on the task, clicks on the remove button and confirms Test inputs: created task	Test outputs: removes todo task	