

# IT-314 Software Engineering

GUI Testing Using selenium



1.Home page testing :

Functionality	Description of Tests Performed	Comments
Primary Navigation	Verified top navigation links: Features , How It Works , FAQs , Home logo	All primary nav links route correctly
Call-to-Action (CTA) Buttons	Tested “Get Started”, “Login”, “Back to Home”, “Sign Up Now”	All CTA buttons navigate properly;
Routing Stability	Multiple forward/back routes between: Home ↔ Auth, Home ↔ Signup, Home ↔ Login	No broken routes or crashes; navigation state consistent
Features Section	Opened each feature card (1–4), tested expand/collapse, validated “See Less”	Expand/Collapse logic works without overlap

<b>Premium Features Cards</b>	Opened “Premium Feature 1” and “Premium Feature 2” cards	UI elements respond correctly
<b>FAQ Accordion</b>	Expanded/collapsed questions 1–5, tested single click & double click	Accordion behaves correctly, no glitches
<b>Scrolling Behavior</b>	Programmatically scrolled to feature sections	Scroll loading and animation work properly
<b>Footer Navigation Links</b>	Verified footer links (Features, How It Works, Privacy Policy, etc.)	Footer routing stable & correctly linked
<b>Mobile Menu / Hamburger Menu</b>	Tested menu toggle, navigation items inside menu, overlay interactions	Mobile menu opens/closes properly and items route correctly
<b>Form Entry Points</b>	Verified all entry points to signup/login from navbar, CTA, footer, and mobile menu	All routes consistent and functional

## Selenium test cases & output :

The screenshot displays the Selenium IDE interface within a Mozilla Firefox browser window. The main area shows a list of test cases on the left, including 'auth\_test\*', 'home\_page\_testing\*', and 'home\_page\_testing\*'. The central pane shows the details of the selected test case, 'home\_page\_testing\*', with a table of commands and targets. The commands include 'click' for various links like 'linkText=How It Works?', 'linkText=Privacy Policy', and 'linkText=Terms Of Service'. The bottom pane shows a log of test execution results, with a final status message: 'home\_page\_testing\* completed successfully'.

Command	Target	Value
click	linkText=How It Works?	
click	linkText=Privacy Policy	
click	linkText=Terms Of Service	

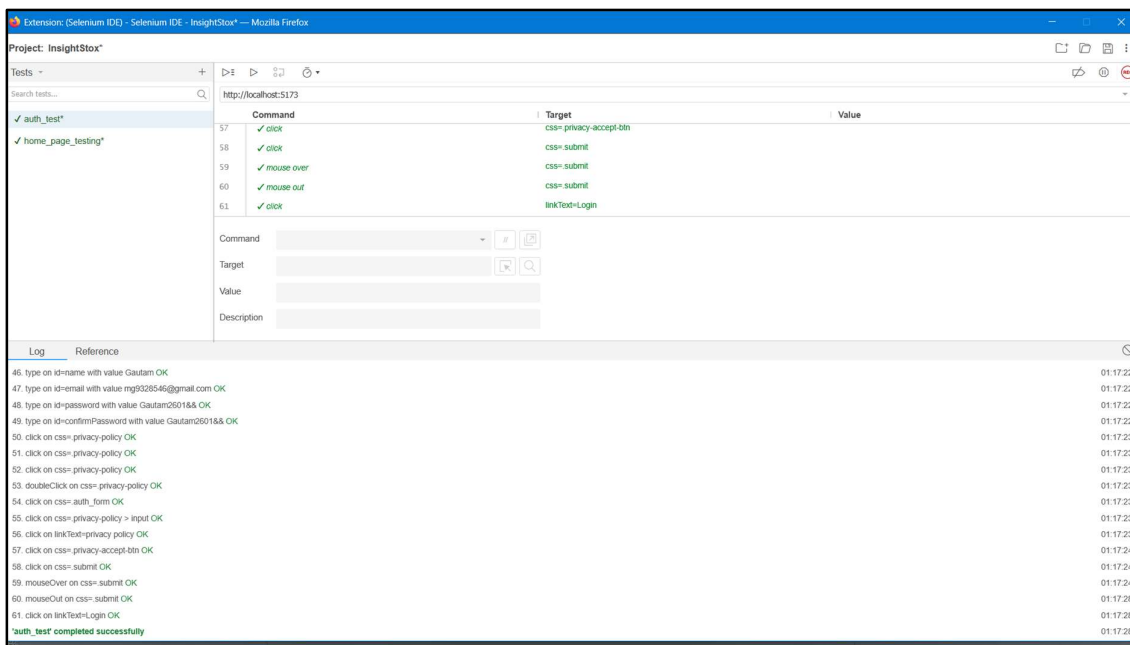
Log:

- 36. click on css=ques img OK
- 37. click on css=ques img OK
- 38. click on css=logotext > img OK
- 39. click on css=github\_logo OK
- 40. click on css=footer\_first\_column OK
- 41. click on css=footer\_first\_column OK
- 42. click on css=footer\_second\_column OK
- 43. click on css=footer\_second\_column OK
- 44. click on linkText=How It Works? OK
- 45. click on css=question\_div OK
- 46. doubleClick on css=question\_div OK
- 47. click on css=footer\_second\_column > .nav\_link:nth-child(2) > a OK
- 48. click on css=footer\_second\_column > .nav\_link:nth-child(3) > a OK
- 49. click on linkText=How It Works? OK
- 50. click on linkText=Privacy Policy OK
- 51. click on linkText=Terms Of Service OK
- home\_page\_testing\* completed successfully

## 2.Authentication page testing :

Functionality	Generalized Actions Performed	Comments
Page Load & Basic UI	Opened /auth, validated screen size, checked visibility of Login / Signup UI elements	Page and UI rendered as expected
Login Form Inputs	Interacted with Email field, Password field, typed multiple values, toggled password visibility	Input fields responsive; visibility toggle works
Login Button Interactions	Click, mouseOver, mouseOut on .submit button	Button state transitions work correctly
Password Visibility Toggle	Clicked .pi-eye, .pi-eye-slash, toggled states repeatedly	Visibility toggle stable under rapid actions
Forgot Password Flow	Clicked “Forgot?”, entered email, submitted, entered OTP, resubmit tested	End-to-end forgot password logic working
OTP Field Testing	Entered values, double-click, mouse interactions, re-typing OTP	OTP input stable to various interactions
Multiple Window Handling	New window opened (e.g., privacy policy), window switching and closing tested	External links open correctly; window switching stable
Signup Page Navigation	Clicked “Sign Up” links from login page	Navigation to signup successful
Signup Form Validations	Typed name, email, password, confirm password, tested mismatched passwords, corrected them	Validation flow works; error handling verified
Password Validation in Signup	Entered invalid and valid passwords, toggled visibility multiple times	Signup password rules properly enforced

<b>Privacy Policy Flow</b>	Opened privacy policy, accepted policy	Redirects and user acceptance logic work correctly
<b>Signup Submission</b>	Completed form, submitted, OTP process tested (123, then correct OTP)	Signup + OTP verification flow validated
<b>Reattempting Login &amp; Signup</b>	Repeated login attempts with various credentials	Auth routes stable under repeated submissions
<b>Return to Login / Home Navigation</b>	Navigated between Login ↔ Forgot ↔ Signup ↔ Privacy Policy ↔ Home	All internal routes functional
<b>Mobile Menu Interactions</b>	Toggled menu → clicked first menu item	Mobile navigation functioning
<b>Extra UI Interactions</b>	Clicked areas like .email-error, .auth_form to test error states	Error popups and focus behavior consistent
<b>Overall System Stability</b>	Full flow executed: Login → Forgot → OTP → Signup → Privacy Policy → OTP → Login	Auth module passed all GUI tests



### 3.chatbot page testing :

Functionality	Generalized Actions Performed	Comments
Navigation to Chatbot	Clicked “AI Insights” from menu	Chatbot page routing working correctly
Sidebar Toggle	Opened/closed chatbot sidebar multiple times using .bot-toggle-sidebar-btn	Sidebar state stable under repeated toggles
Chatbot Feature Cards	Clicked feature cards: chatbot-feature f1 ,f2, f3	Feature sections expand/respond correctly
Profile Menu Interactions	Opened profile menu, interacted with avatar, title, settings button → hover and unhover tested	Profile section animations & interactions OK
Overlay Handling	Clicked on` .profileoverlay multiple times to close menus	Overlay behavior consistent
Send Button Behavior	Single click, double click, repeated clicks on .send-btn	Button works even with rapid interactions
Chat Input Testing	Clicked input, typed text (“hy”), pressed Enter, clicked send button	Message input & submission working
Chat Message Area	Clicked chat-messages & welcome message area	UI responsiveness validated
Menu Toggle (Global Header)	Opened & closed .menu_toggle repeatedly	Mobile menu system stable within chatbot page
General UI Interactions	Clicks on layout elements like chat input area, message area	No UI freeze or misbehavior

6

#### 4.Preference page testing :

Functionality	Generalized Actions Performed	Comments
Page Load & Navigation	Opened <b>My Profile</b> page and navigated to <b>Preferences</b> via Sidebar	Page loads correctly, sidebar navigation works without errors
Theme Dropdown Interaction	Clicked on <b>Theme Select</b> dropdown and tested option visibility	Dropdown opens and responds properly
Dark Theme Selection	Selected <b>Dark</b> theme from dropdown	Theme updated correctly and alert displayed
Theme Update Alert	Verified alert message " <b>Theme updated successfully!</b> " after Dark theme selection	Alert working correctly
Light Theme Selection	Re-opened dropdown and selected <b>Light</b> theme	Theme switched successfully
Simple Layout Selection	Selected <b>Simple (Essential Metrics)</b> layout	Layout switched back correctly
Layout Re-Update Alert	Verified alert after switching to Simple layout	Alert displays correct confirmation
UI Responsiveness	Checked page stability during multiple dropdown interactions	No crashes, UI stays consistent
Overall Preference Flow	Completed full preference change cycle (Theme + Layout)	Full flow successful without errors

Selenium test cases & output :

Project: GUI\_TESTING

Tests

Search tests...

✓ HelpSupport\_Page

✓ Preference\_Page

http://localhost:5173/my-profile

	Command	Target	Value
11	assert alert	Layout updated successfully!	
12	click	id=layout-select	
13	select	id=layout-select	label=Simple (Essential Metrics)
14	assert alert	Layout updated successfully!	

Command

open

Target

http://localhost:5173/my-profile

Value

Description

Log

Reference

2. click on css= Sidebar-item:nth-child(4) OK

3. click on id=theme-select OK

4. select on id=theme-select with value label=Dark OK

5. assertAlert on Theme updated successfully! OK

6. click on id=theme-select OK

7. select on id=theme-select with value label=Light OK

8. assertAlert on Theme updated successfully! OK

9. click on id=layout-select OK

10. select on id=layout-select with value label=Detailed (Advanced Insights) OK

11. assertAlert on Layout updated successfully! OK

12. click on id=layout-select OK

13. select on id=layout-select with value label=Simple (Essential Metrics) OK

14. assertAlert on Layout updated successfully! OK

✓Preference\_Page completed successfully

14:00:26

14:00:26

14:00:27

14:00:27

14:00:44

14:00:45

14:00:45

14:00:45

14:00:46

14:00:46

14:00:47

14:00:47

14:00:47

14:00:48





5.HelpSupport page testing :

Functionality	Generalized Actions Performed	Comments
<b>Page Load &amp; Navigation</b>	Opened <b>My Profile</b> page and navigated to <b>HelpSupport</b> via Sidebar	Page loads correctly, sidebar navigation works without errors
<b>Quick Help – Getting Started Modal</b>	Opened <b>Getting Started Guide</b> quick help modal and clicked Explore my Dashboard button	Modal opens successfully and button correctly routes to the Dashboard Page
<b>Quick Help – Portfolio Guide Modal</b>	Opened <b>How to Build Portfolio</b> modal and clicked Go to Portfolio button	Modal opens successfully and button correctly routes to the Portfolio Page
<b>Quick Help – AI Suggestions Modal</b>	Opened <b>Understanding AI Suggestions</b> modal and clicked on view my AI Insights button	Modal opens successfully and button correctly routes to the Insights page
<b>Quick Help – Troubleshooting Modal</b>	Opened <b>Troubleshooting</b> modal and closed using overlay	Modal overlay close works properly
<b>Quick Help Buttons Interaction</b>	Clicked all <b>Quick Help</b> buttons sequentially (1–4)	All buttons responded correctly without any UI glitches, and the modal closed successfully using the Escape key or by clicking outside the modal.
<b>Contact Form Field</b>	Located and clicked <b>Message input</b> field	Field is accessible and editable
<b>Contact Form Submission</b>	Clicked <b>Submit Ticket</b> button	Form submits successfully
<b>Contact Form Alert</b>	Verified alert <b>“Support ticket submitted successfully!”</b>	Success alert is displayed correctly
<b>Feedback Form Field</b>	Clicked <b>Feedback input</b> field	Input box responds correctly

<b>Feedback Form Submission</b>	Clicked <b>Send Feedback</b> button	Feedback successfully submitted
<b>Feedback Success Alert</b>	Verified alert <b>“Thank you for your feedback!”</b>	Success alert appears correctly


## Selenium test cases & output :


Project: GUI\_TESTING

Tests +  

Search tests...

	Command	Target	Value
✓ HelpSupport_Page			
✓ Preference_Page	19 ✓ click	id=feedback-input	
	20 ✓ type	id=feedback-input	I am fine.
	21 ✓ click	css= help-section:nth-child(4) btn-primary	
	22 ✓ assert alert	Thank you for your feedback!	

Command  

Target  

Value

Description

Log	Reference
10. click on css= modal-overlay OK	14:36:04
11. click on css= quick-help-btn:nth-child(1) OK	14:36:04
12. click on css= quick-help-btn:nth-child(2) OK	14:36:05
13. click on css= quick-help-btn:nth-child(3) OK	14:36:06
14. click on css= quick-help-btn:nth-child(4) > quick-help-icon OK	14:36:07
15. Trying to find id=message-input... OK	14:36:08
16. type on id=message-input with value what are you doing? OK	14:36:11
17. click on css= submit-ticket-btn OK	14:36:12
18. assertAlert on Support ticket submitted successfully! OK	14:36:13
19. click on id=feedback-input OK	14:36:15
20. type on id=feedback-input with value I am fine. OK	14:36:16
21. click on css= help-section:nth-child(4) btn-primary OK	14:36:17
22. assertAlert on Thank you for your feedback! OK	14:36:18
<b>*HelpSupport_Page* completed successfully</b>	14:36:41

## 6.WatchList page with dashboard-header testing

Functionality	Description of Tests Performed	Comments
<b>Add Stock via Global Header</b>	Clicked global search bar (.search-input), typed "reliance", and clicked the add button (.pi) on the 2nd result item.	Global search integration works; stock added to watchlist successfully.
<b>Local Search Filtering</b>	Focused on Watchlist's local search input, typed "reliance" to filter the table rows.	Table filtered correctly to show specific stocks.
<b>Remove Stock Action</b>	Clicked the "Remove" button (.action-btn) on the specific stock row.	Stock successfully removed from the list; UI updated.
<b>Clear Local Search</b>	Double-clicked the local search input and cleared the text to reset the view.	Search input cleared; full watchlist restored.
<b>Filter Modal (Daily Change)</b>	Opened Filter Modal (.filter-btn), selected "Gainers" (#gainers), and clicked "Apply Filters".	Filter applied successfully; list updated to show gainers.
<b>Clear Filters &amp; Overlay</b>	Re-opened modal, clicked "Clear All", then closed the modal by clicking the background overlay (.filter-modal-overlay).	Filters reset correctly; overlay click successfully closed the modal.
<b>Filter Modal (Market Cap)</b>	Opened modal again, selected "Large Cap" (#large-cap), and applied filters.	"Large Cap" filter logic applied correctly.
<b>Stock Details Navigation (Table)</b>	Clicked on a specific stock row (e.g., TITAN.NS) within the table.	Successfully navigated to the Stock Details page.
<b>Stock Details Navigation (Header)</b>	Navigated back to Watchlist, searched for "titan" in global search, and clicked the result.	Global search successfully navigated to Stock Details page.

Selenium test cases & output :

Selenium IDE - watchlistpage

Project: watchlistpage

Executing -

watchlist

http://localhost:5173

	Command	Target	Value
1	✓ open	/watchlist	
2	✓ set window size	1552x632	
3	✓ click	css= watchlist	
4	✓ click	css= search-input	
5	✓ type	css= popup-search-input	reliance

Command

Target

Value

Description

Runs: 1 Failures: 0

Log

Reference

20. click on css= filter-btn > img OK

21. click on id=large-cap OK

22. click on css= apply-filter-btn OK

23. click on css= filter-btn OK

24. click on css= clear-filter-btn OK

25. click on css= apply-filter-btn OK

26. click on css= watchlist OK

27. click on css= table-stock:nth-child(1) > td:nth-child(2) OK

28. click on linkText=Watchlist OK

29. click on css= search-input OK

30. type on css= popup-search-input with value titan OK

31. click on css= result-item:nth-child(1) OK

watchlist completed successfully

05:38:35

05:38:36

05:38:37

05:38:38

05:38:39

05:38:40

05:38:41

05:38:42

05:38:43

05:38:44

05:38:45

05:38:46

05:38:47

7. MyProfile page with testing

Extension: (Selenium IDE) - Selenium IDE - InsightStox - Group21\* — Mozilla Firefox

Project: InsightStox - Group21\*

Executing ▾

⏮ ⏪ ⏩ ⏭ ⌚ ▾

⏏ ⏸ ⏹

✓ My Profile Page\*

http://localhost:5173

	Command	Target	Value
51	✓ click	css=.myPage_InfRow1 .myPage_EditDetails	
52	✓ click	css=input:nth-child(1)	
53	✓ type	css=input:nth-child(1)	Ayush D
54	✓ click	css=.myPage_SaveBtn	

Command

Target

Value

Description

Runs: 1 Failures: 0

Log

Reference

50. click on css=.myPage\_CancelBtn OK

51. click on css=.myPage\_InfRow1 .myPage\_EditDetails OK

52. click on css=input:nth-child(1) OK

53. type on css=input:nth-child(1) with value Ayush D OK

54. click on css=.myPage\_SaveBtn OK

'My Profile Page' completed successfully

01:32:03

01:32:04

01:32:04

01:32:04

01:32:04

01:32:04

- There are a total of around 54 events that have been done as testing for the MyProfile page and all those events are triggering properly and working properly.

8. Portfolio Page with testing

Extension: (Selenium IDE) - Selenium IDE - InsightStox - Group21\* — Mozilla Firefox

Project: InsightStox - Group21\*

Tests

Search tests...

https://insight-stox-portfolio-analyzer-tra-gamma.vercel.app/portfolio

	Command	Target	Value
13	✓ click	css=.profile_btn > button	
14	✓ mouse over	css=button > img	
15	✓ mouse out	css=button > img	
16	✓ click	css=.profileoverlay	

Command

Target

Value

Description

Log

Reference

11. mouseOver on linkText=Portfolio OK

12. click on linkText=Portfolio OK

13. click on css=.profile\_btn > button OK

14. mouseOver on css=button > img OK

15. mouseOut on css=button > img OK

16. click on css=.profileoverlay OK

'Portfolio Page' completed successfully

01:53:55

01:53:59

01:53:59

01:54:00

01:54:00

01:54:01

01:54:01

- There are a total of around 16 events that have been done as testing for the Portfolio page and all those events are triggering properly and working properly.

9. StockDetails page with test

Extension: (Selenium IDE) - Selenium IDE - InsightStox - Group21\* — Mozilla Firefox

Project: InsightStox - Group21\*

Executing

Run all tests Ctrl+Shift+R .73/

	Command	Target	Value
26	✓ click	css=input:nth-child(1)	
27	✓ type	css=input:nth-child(1)	12
28	✓ click	css=.model-confirm-btn	
29	✓ click	css=button > img	
30	✓ click	css=.profileoverlay	

Command

Target

Value

Description

Runs: 1 Failures: 0

Log

Reference

25. click on css=.model-sell OK

26. click on css=input:nth-child(1) OK

27. type on css=input:nth-child(1) with value 12 OK

28. click on css=.model-confirm-btn OK

29. click on css=button > img OK

30. click on css=.profileoverlay OK

'Stock Details page' completed successfully

01:57:57

01:57:57

01:57:57

01:57:57

01:57:57

01:57:58

01:57:58

- There are a total of around 30 events that have been done as testing for the StockDetail page and all those events are triggering properly and working properly.