

User Journey Modelling

Introduction to User Journey's

- A user journey is a series of steps which represent a scenario in which a user might interact with a system. Typically, a user journey has a starting point, i.e. an entry point into the system, a series of transitions from one state to another and a set of triggers which cause the transitions.
- In the simplest terms a “user journey” is the path that a visitor takes through your site/application.
- a “user journey” generally refers to the path taken to achieve a specific goal (e.g. making a purchase of stock) or complete a specific purpose (e.g. view the stock value)
- End to end user journey depicts the path taken by a user from one state in the site/application to another end state with or without repeating states in the path.

PROBLEMS WITH EXECUTING ATOMIC TEST CASES

- Test cases do not model real world user experience (end to end testing)
- The number of test cases explode due to various combinations. Tracking and analyzing the coverage of these test cases is complex.
- Test completeness? There is no logical way to conclude that our test cases are complete.
- Prioritization of test cases, is challenging.
- No Traceability matrix

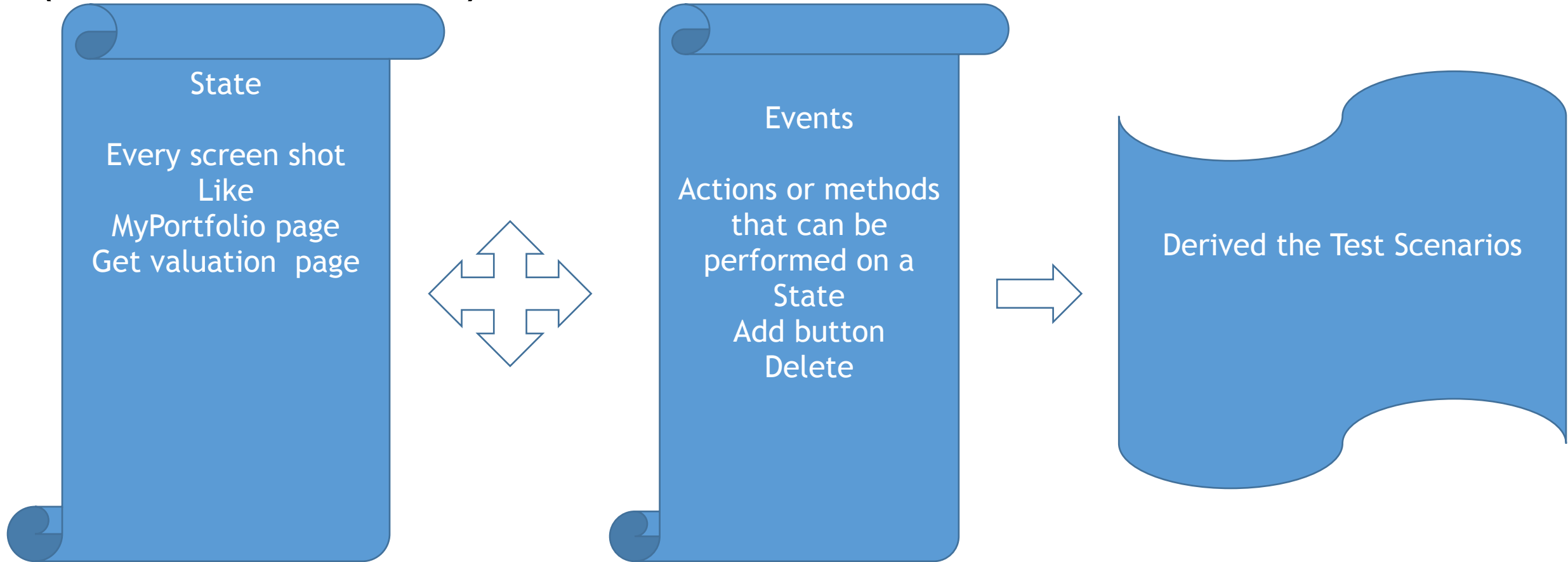
ADVANTAGES OF TESTING WITH USER JOURNEYS – (SOLUTION TO THE PROBLEMS)

- Create a unified model to derive test cases using state machine diagrams
- Create test cases which can model customer experience through transition of various states (user journeys)
- Model helps to achieve maximum coverage (test completeness)
 - covering all the states
- Derive traceability matrix, to capture detailed effort
- Logical way of prioritization -
- Automate user journey and scenarios
- Model can be used as a basis for Automation framework designs

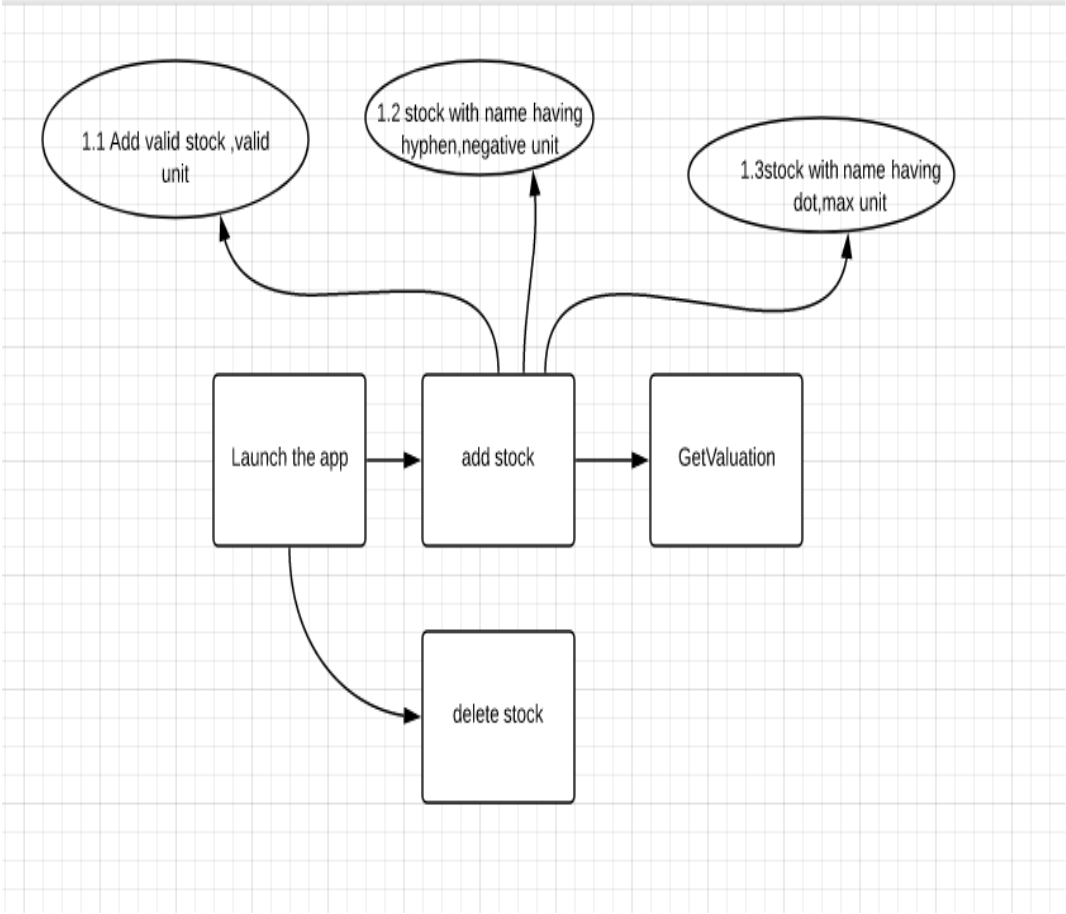
How to derive user journeys

- Model the all the states of the application as a state machine diagram
- Identify the actions possible on the different states
- On each action identify the transitions that happens from one state to another (Workflow)
- There can be multiple workflow's between state
- Each user journey is a combination of workflows
- Traceability can be achieved by mapping all the possible workflow between to end states
- Each user journey can include different workflows to minimize the number of user journeys and increase the coverage

STATE(S) + EVENT(S) = SCENARIOS (WORKFLOW)



User Journey for Stock Tracker



Feature	Test Scenario
Add	Scenario: User can add a new ticker with valid units
	Scenario: User can add a existing ticker to cover corner cases
	Scenario: User can add a different type ticker than the existing
	Scenario: User can add a different type ticker than the existing
	Scenario: User can update ticker with more ,less units
	Scenario: User can add a new ticker with invalid units - negative numbers
	Scenario: User can delete an existing ticker
	Scenario: User can delete a non existing ticker
	Scenario: User can delete all ticker
	Scenario: Verification of units with decimal numbers ,max number
	Scenario: Verification of possible ticker that can be added
	The user is able to view the stock value when stock is added
GetValuation	stock value when ticker is not added

TRACEABILITY MATRIX

A	B	C	D	E	F	G	H	I	J	K	L
Conditions to test					User journey						
User story	Feature	Components	Remarks	No. case	001	002	003	004	005	006	007
1	Add	stock with valid series		1	x						
		stock with hyphen	1	x							
		name with dot	1			x					
		name with small letter	0								
		name with invalid stock	1					x			
		valid unit	0								
		negative	1		x						
		max positive value the unit can take	1					x			
		can add same stock and same units	0								
		add different stock	1			x					
			0								
			0								
			0								
			0								

- Keep points of this Traceability is (this is just an example)
- This gives an visual picture of what sub component of feature are covered
- If the component is missing then it indicate 0
- Also give how many times this component is covered in test cases

FUTURE ROADMAP

- Automation tool to generate user journey
- All data will be entered into a database, with nodes (state), events(actions or methods) and transitions as a set
- A change in UX can be handled independently