Software Engineering

Part (V)- Requirement Engineering

By: Mehran Alidoost Nia

Shahid Beheshti University, Fall 2023

What is requirement Engineering?

- Requirements engineering (RE) is the term for the broad spectrum of tasks and techniques that lead to an understanding of requirements.
- It is a major software engineering action that begins during the communication activity and continues into the modeling activity.

Requirements engineering builds a bridge to design and construction.

Seven Tasks of RE

- Inception
  - Basic understanding of the problem.
  - The people who want a solution.
  - The nature of the solution that is desired.
- Elicitation
  - Ask the customer, the users, and others what the objectives for the system or product are

Seven Tasks of RE

- Elaboration
  - Developing a requirements model that identifies software function, behavior, and information.
- Negotiation
  - Customers or users to propose conflicting requirements.
-

Seven Tasks of RE

- ➢ Specification
  - ○ A written document, a set of graphical models, a formal mathematical model, a collection of usage scenarios, a prototype, or any combination of these.
- ➢ Validation (consistency)
  - ○ E.g., the software should be user friendly.
- ➢ Management (Change)

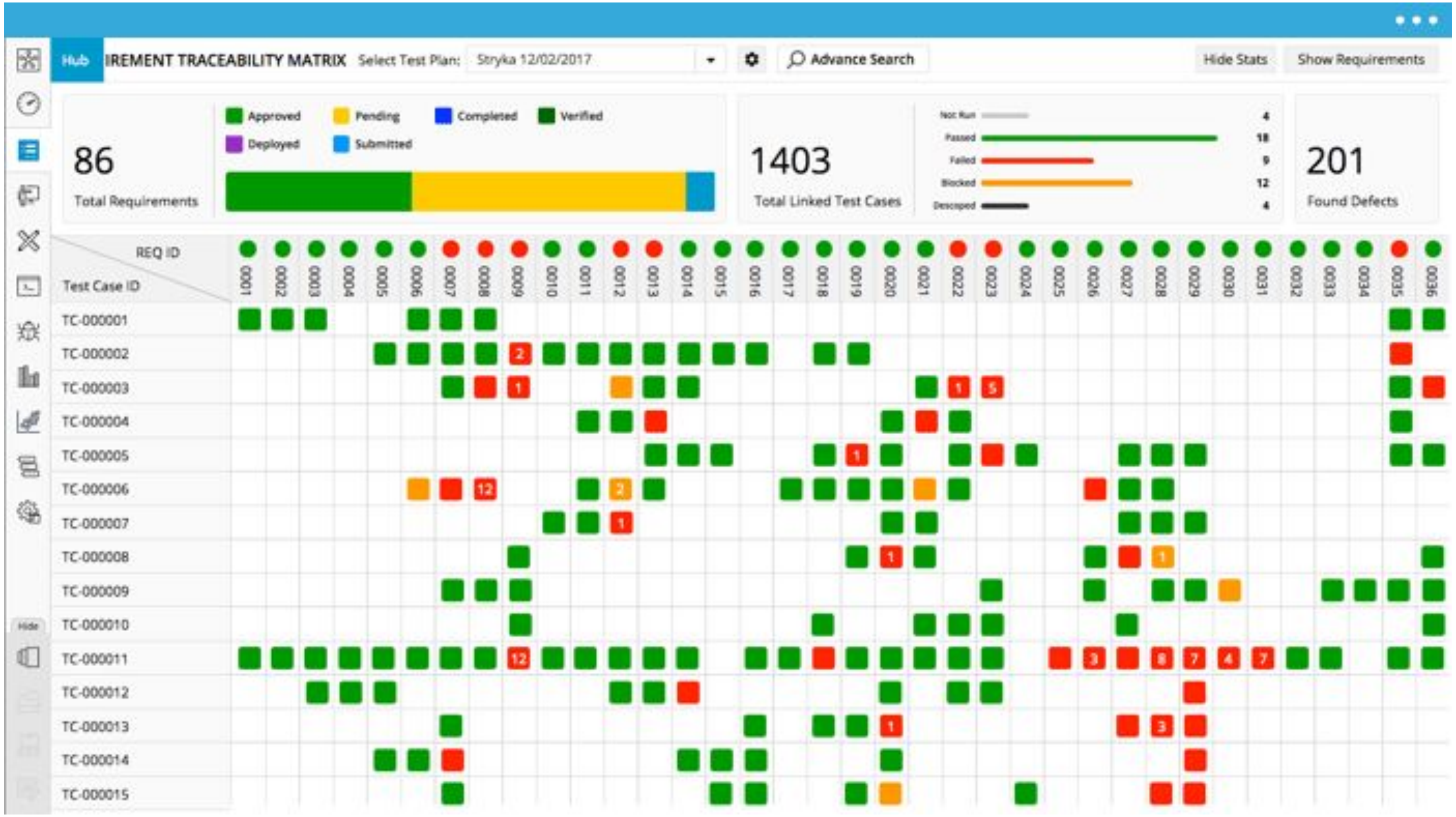
Groundwork of RE

- Identifying Stakeholders
- Recognizing Multiple Viewpoints
- Working Toward Collaboration
- Asking the First Questions
- Nonfunctional Requirements (NFR)
- Traceability

A Few Questions

- Who is behind the request for this work?
- Who will use the solution?
- What will be the economic benefit of a successful solution?
- Is there another source for the solution that you need?

Traceability Matrix



Select Test Plan: Stryka 12/02/2017

🟩 Approved 🟨 Pending 🟦 Completed 🟩 Verified 🟪 Deployed 🟦 Submitted

86 Total Requirements

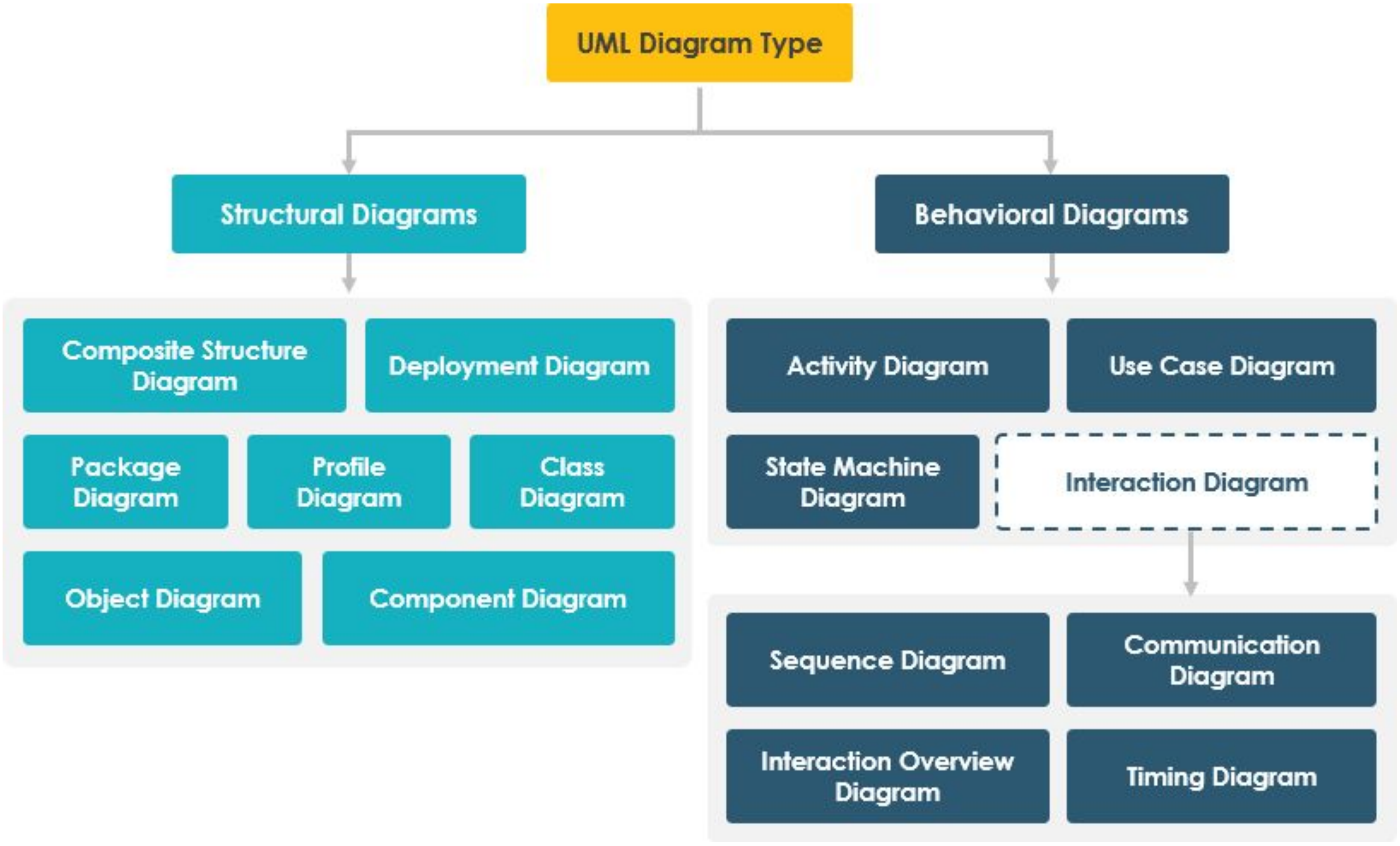
1403 Total Linked Test Cases

Not Run	4
Passed	18
Failed	9
Blocked	12
Descoped	4

201 Found Defects

REQ ID	🟢	🟢	🟢	🟢	🟢	🟢	🔴	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🔴	🟢
Test Case ID	0001	0002	0003	0004	0005	0006	0007	0008	0009	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	0030	0031	0032	0033	0034	0035	0036
TC-000001	🟩	🟩	🟩			🟩	🟩	🟩																											🟩	🟩
TC-000002					🟩	🟩	🟩	🟩	🟥 2	🟩	🟩	🟩	🟩	🟩	🟩	🟩		🟩	🟩																🟥	
TC-000003							🟩	🟥	🟥 1			🟧	🟩	🟩							🟩	🟥 1	🟥 5												🟩	🟥
TC-000004											🟩	🟩	🟥							🟩	🟥	🟩													🟩	
TC-000005													🟩	🟩	🟩			🟩	🟥 1	🟩		🟩	🟥	🟩			🟩	🟩	🟩						🟩	🟩
TC-000006						🟧	🟥	🟥 12			🟩	🟧 2	🟩				🟩	🟩	🟩	🟩	🟧	🟩				🟥	🟩	🟩								
TC-000007										🟩	🟩	🟥 1								🟩	🟩						🟩	🟩	🟩							
TC-000008									🟩										🟩	🟥 1	🟩					🟩	🟥	🟧 1								🟩
TC-000009							🟩	🟩	🟩														🟩			🟩		🟩	🟩	🟧			🟩	🟩	🟩	🟩
TC-000010									🟩									🟩			🟩	🟩	🟩				🟩									🟩
TC-000011	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥 12	🟩	🟩	🟩	🟩	🟩		🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩		🟥	🟥 3	🟥	🟥 8	🟥 7	🟥 4	🟥 7	🟩	🟩		🟩	🟩
TC-000012			🟩	🟩	🟩							🟩	🟩	🟥						🟩		🟩	🟩						🟥							
TC-000013							🟩									🟩		🟩	🟩	🟥 1							🟥	🟥 3	🟥							
TC-000014					🟩	🟩	🟥							🟩	🟩	🟩				🟩									🟥							
TC-000015							🟩								🟩	🟩			🟩	🟧				🟩				🟥	🟥							

UML Diagram



UML Diagram Type

- Structural Diagrams
  - Composite Structure Diagram
  - Deployment Diagram
  - Package Diagram
  - Profile Diagram
  - Class Diagram
  - Object Diagram
  - Component Diagram
- Behavioral Diagrams
  - Activity Diagram
  - Use Case Diagram
  - State Machine Diagram
  - Interaction Diagram
    - Sequence Diagram
    - Communication Diagram
    - Interaction Overview Diagram
    - Timing Diagram

Use Case

- A use case tells a stylized story about how an end user interacts with the system under a specific set of circumstances.
- The story may be narrative text (a user story), an outline of tasks or interactions, a template-based description, or a diagrammatic representation.
- Regardless of its form, a use case depicts the software or system from the end user's point of view.

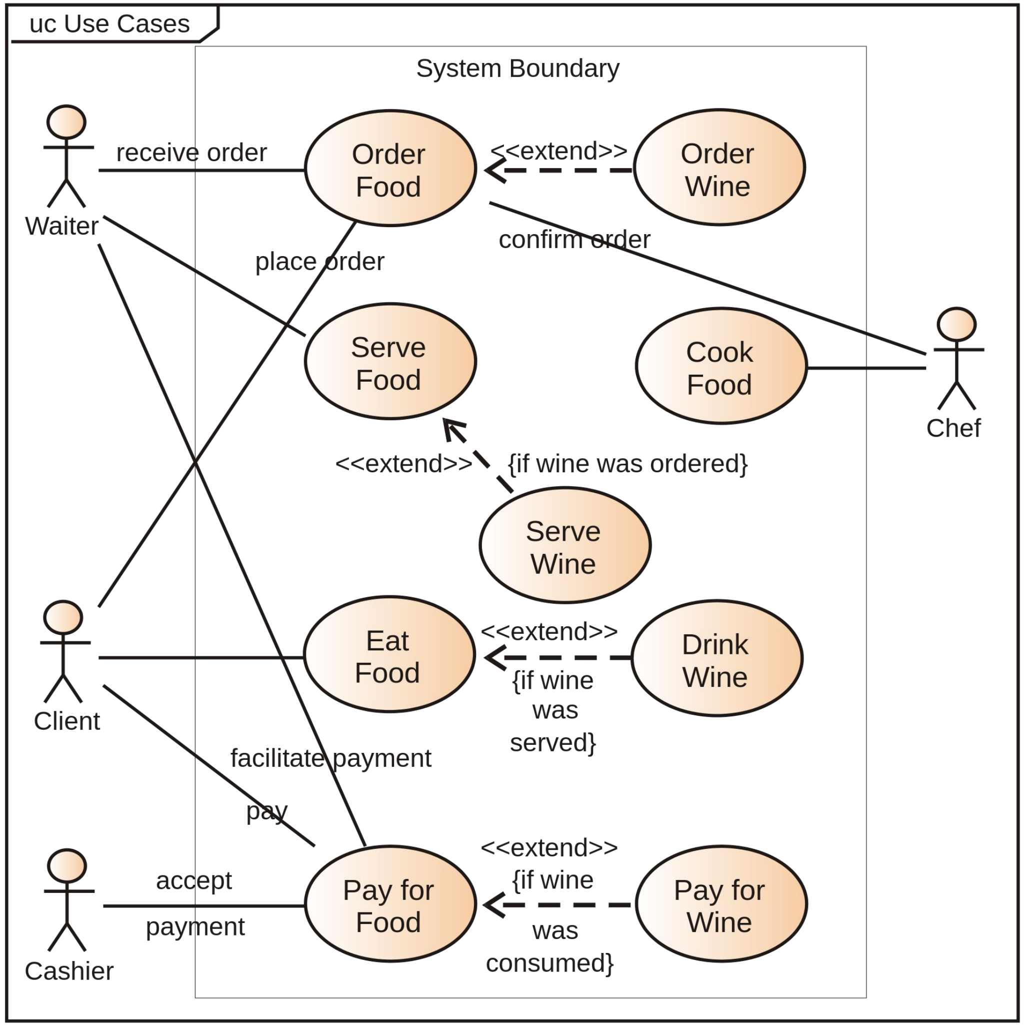
Use Case

- The first step in writing a use case is to define the set of “actors” that will be involved in the story.
- Actors are the different people (or devices) that use the system or product that is to be described.

Questions Answered by Use Cases

- Who is the primary actor, the secondary actor(s)?
- What are the actor's goals?
- What preconditions should exist before the story begins?
- What main tasks or functions are performed by the actor?
- What exceptions might be considered as the story is described?

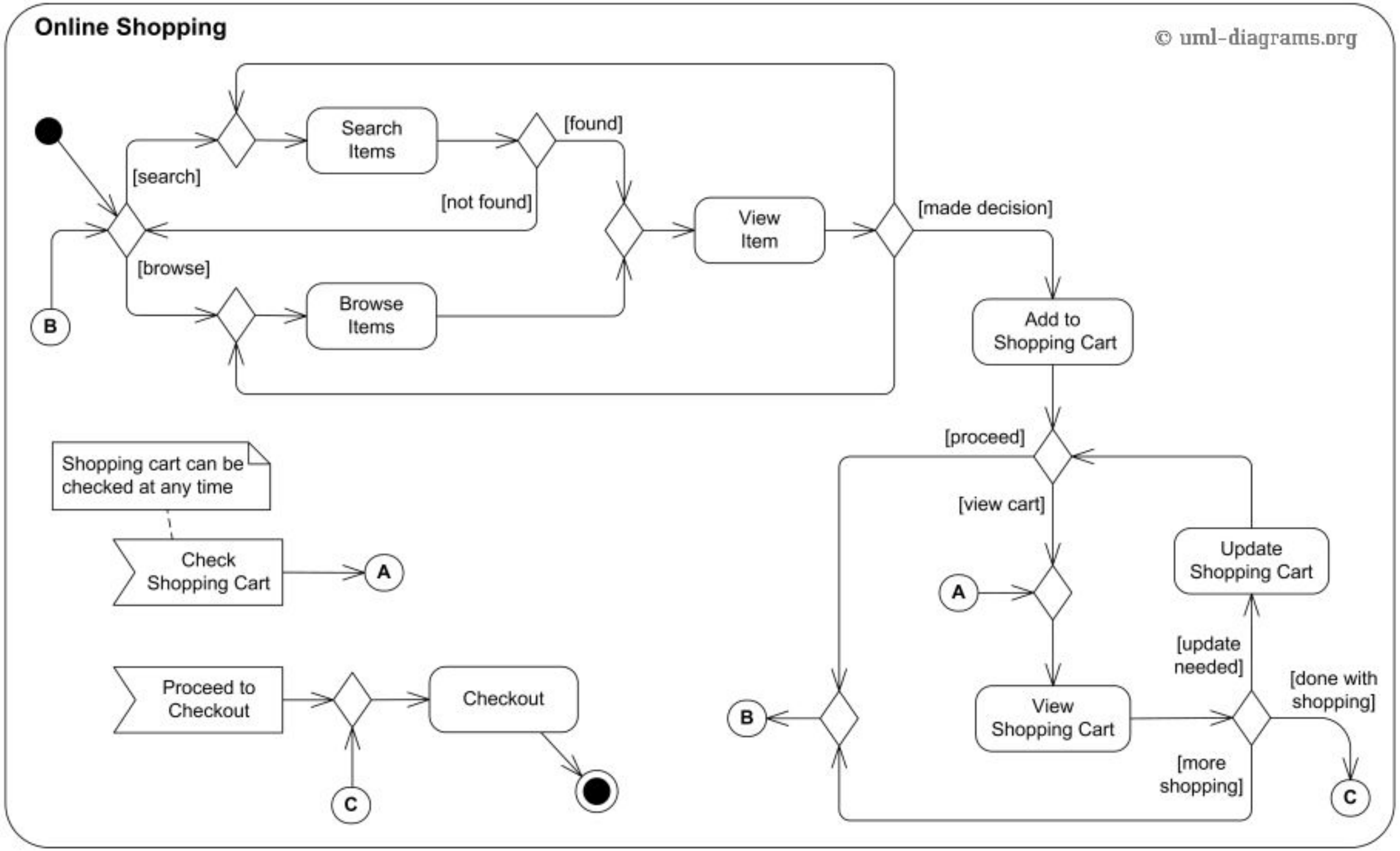
UML Use Case Diagram



Activity Diagram

- Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

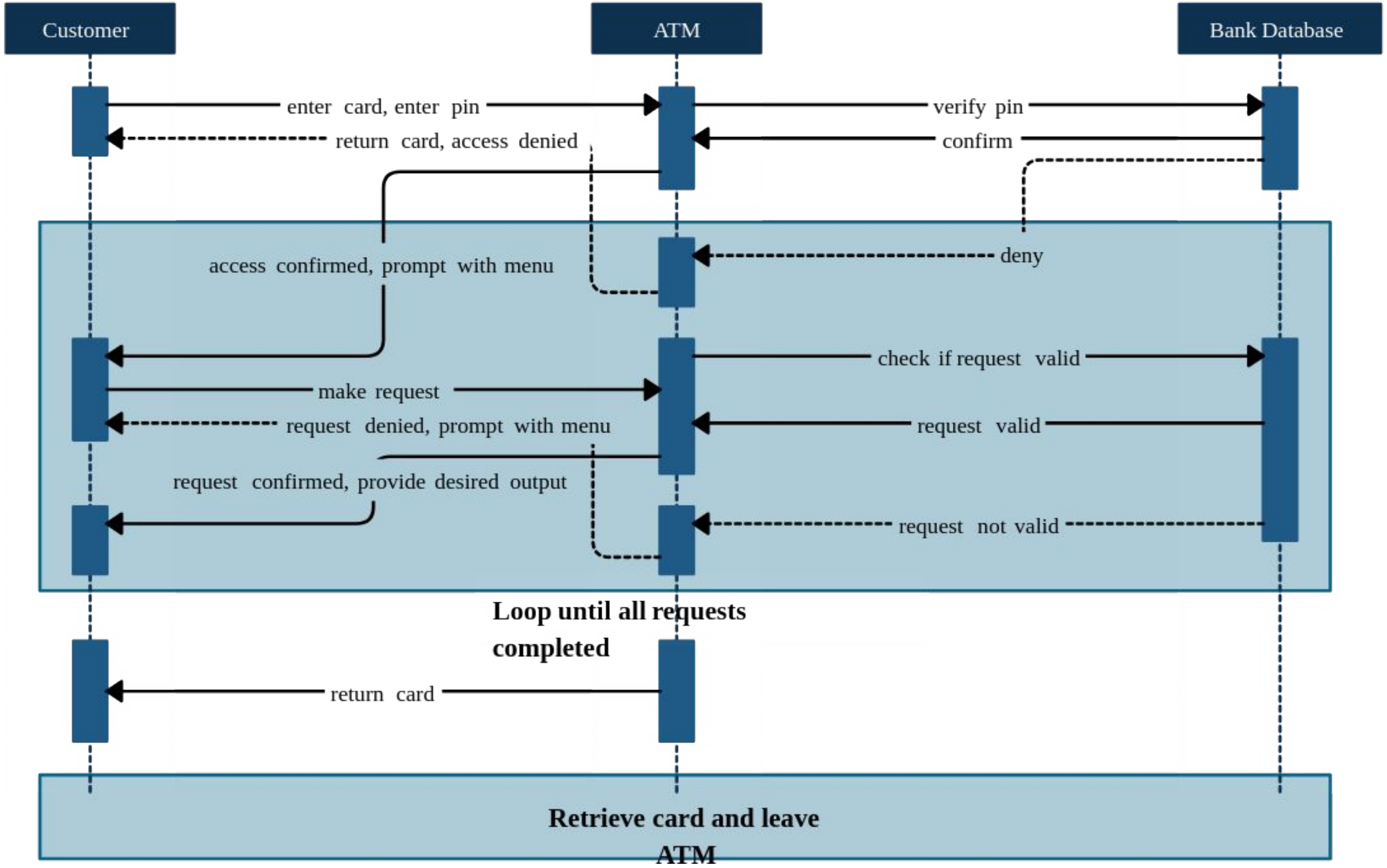
Activity Diagram



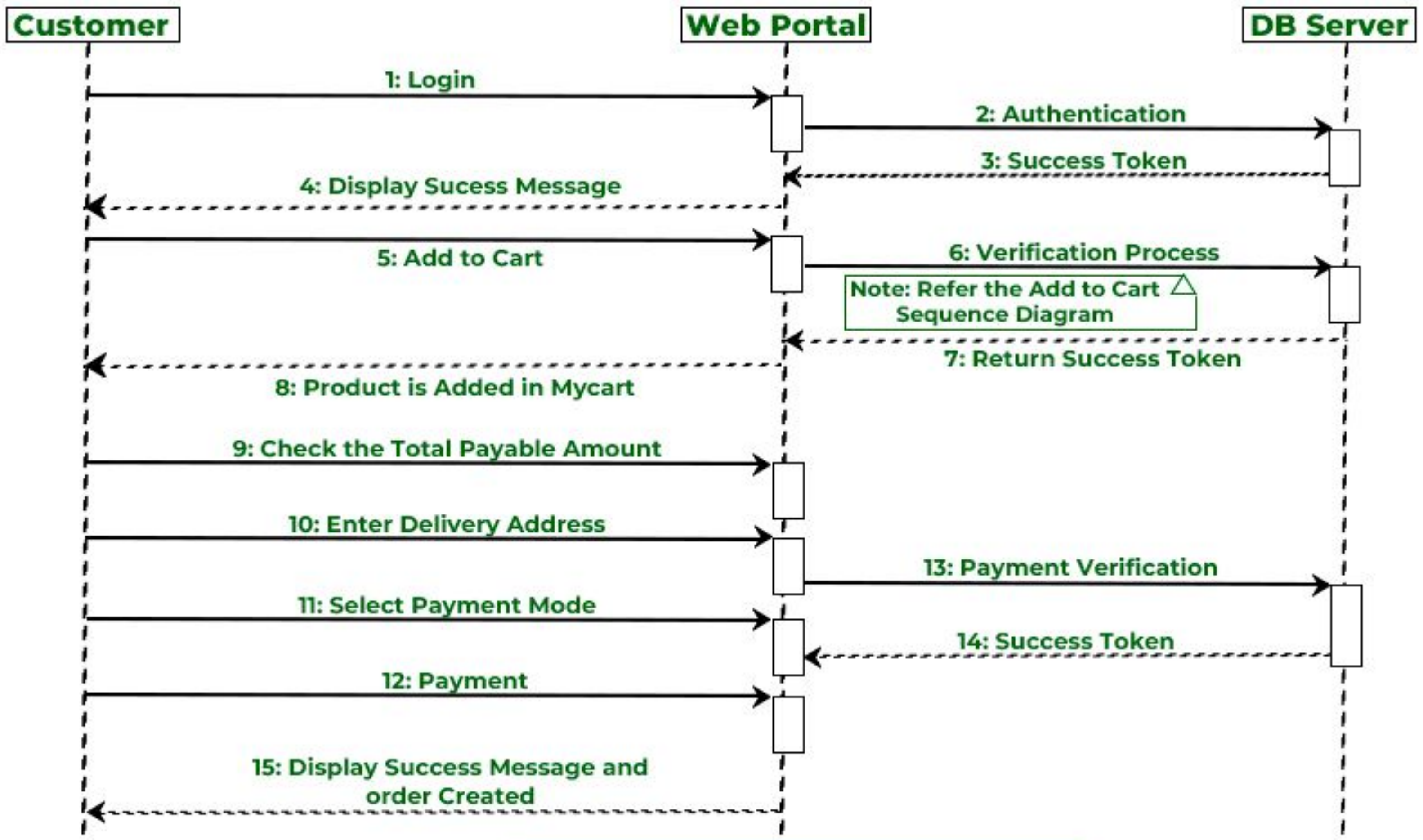
Sequence Diagram

- A sequence diagram shows process interactions arranged in a time sequence.
- The diagram depicts the processes and objects involved and the sequence of messages exchanged as needed to carry out the functionality.

Sequence Diagram (ATM)

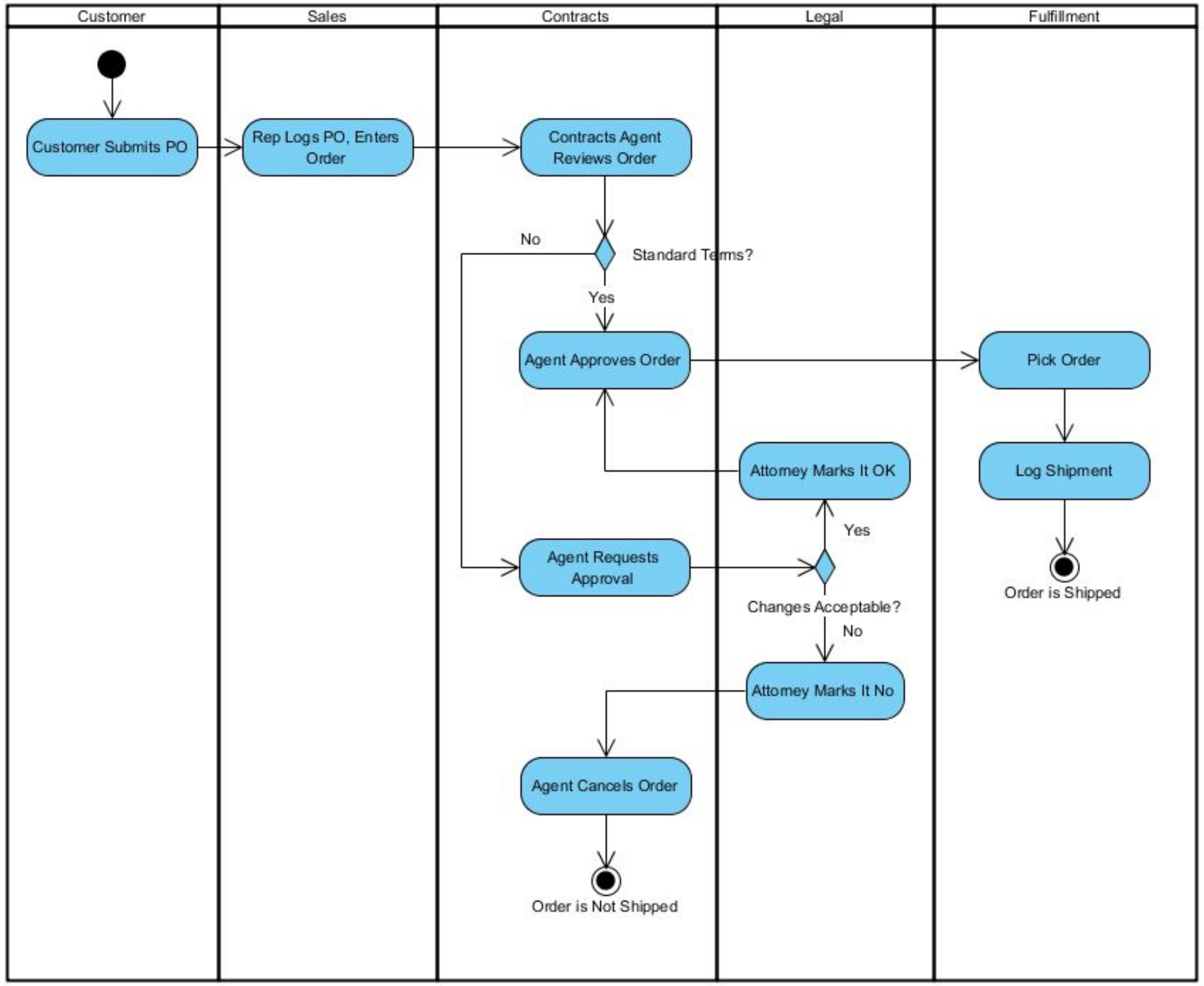

Customer
ATM
Bank Database
enter card, enter pin
verify pin
return card, access denied
confirm
deny
access confirmed, prompt with menu
check if request valid
make request
request denied, prompt with menu
request valid
request confirmed, provide desired output
request not valid
Loop until all requests completed
return card
Retrieve card and leave ATM

Sequence Diagram (Product Order)



Courtesy of https://www.geeksforgeeks.org/

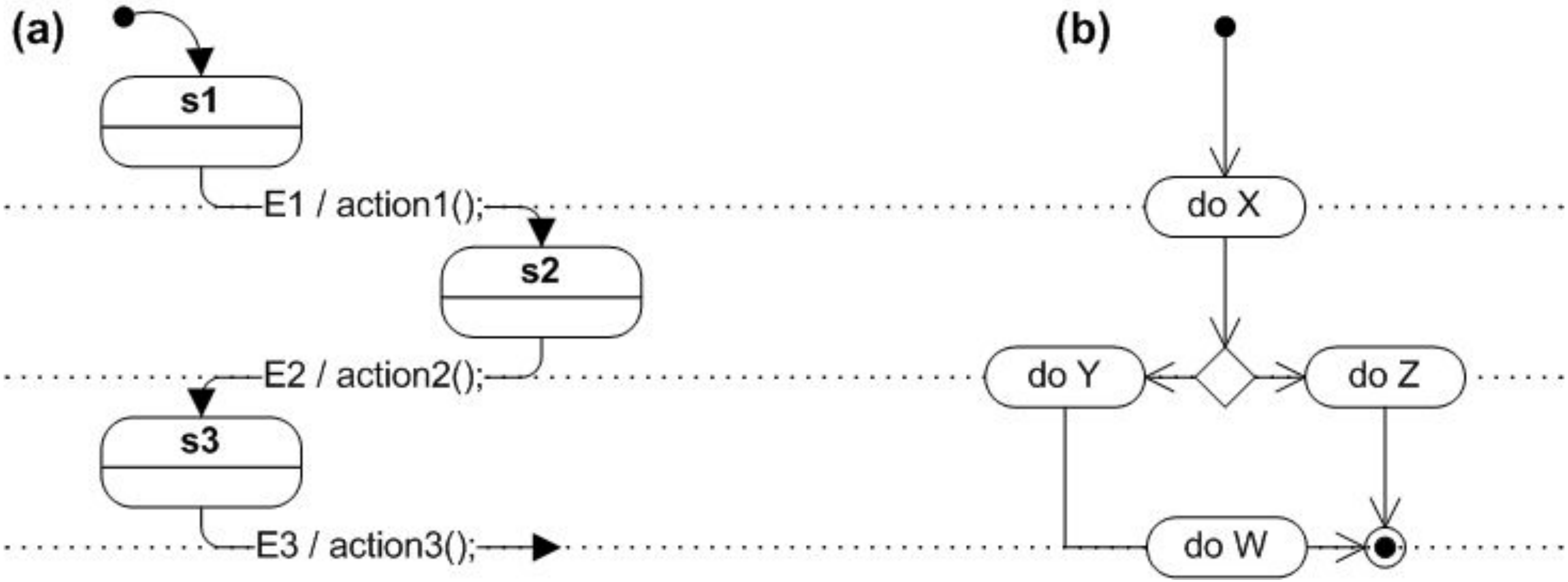
(Activity) Swimlane Diagram



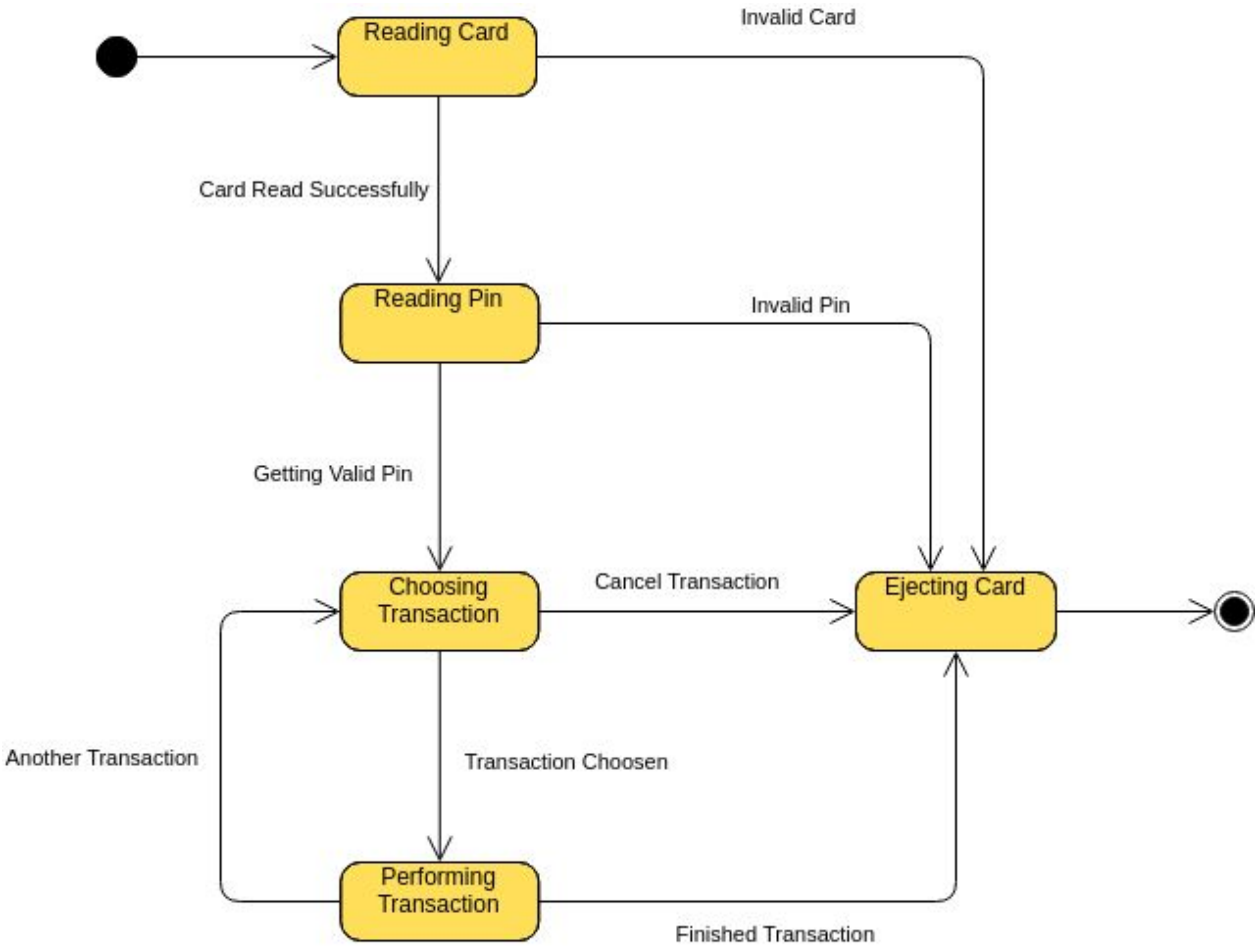
State Diagram

- A state diagram, (or state machine diagram), is a type of behavioral diagram that shows transitions between various objects.

State Diagram



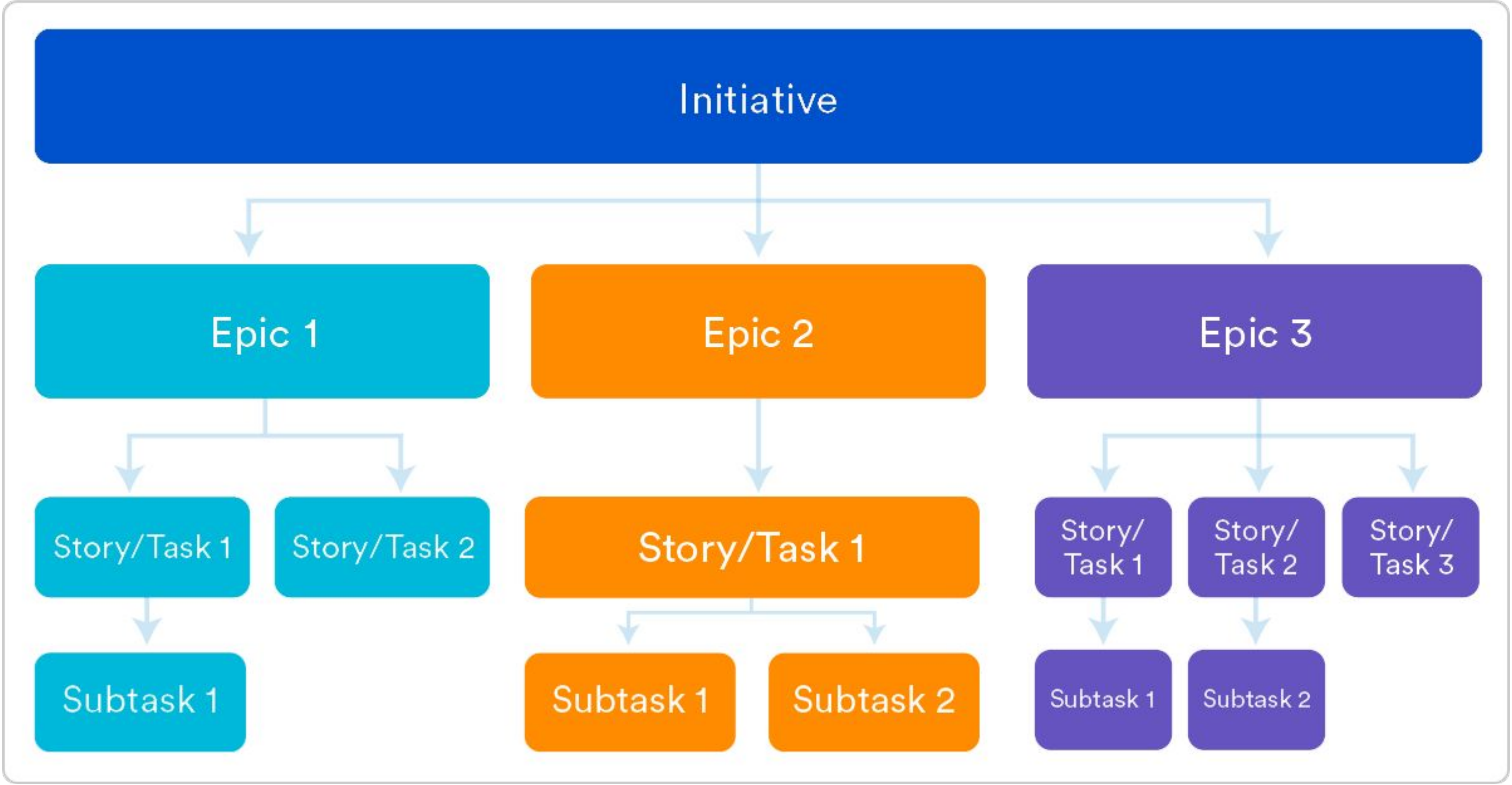
State Diagram (ATM)



Written Requirements

- Stories, also called “user stories,” are short requirements or requests written from the perspective of an end user.
- Epics are large bodies of work that can be broken down into a number of smaller tasks (called stories).
- Initiatives are collections of epics that drive toward a common goal.

Written Requirements



User Story Template and examples

- “As a [persona], I [want to], [so that].”
- Example:
  - As a driver, I want to see the list of potential passengers, so I can choose the best destination and price.
  - As a passenger, I want to pin the source and the destination on the map, so I can request an online taxi.

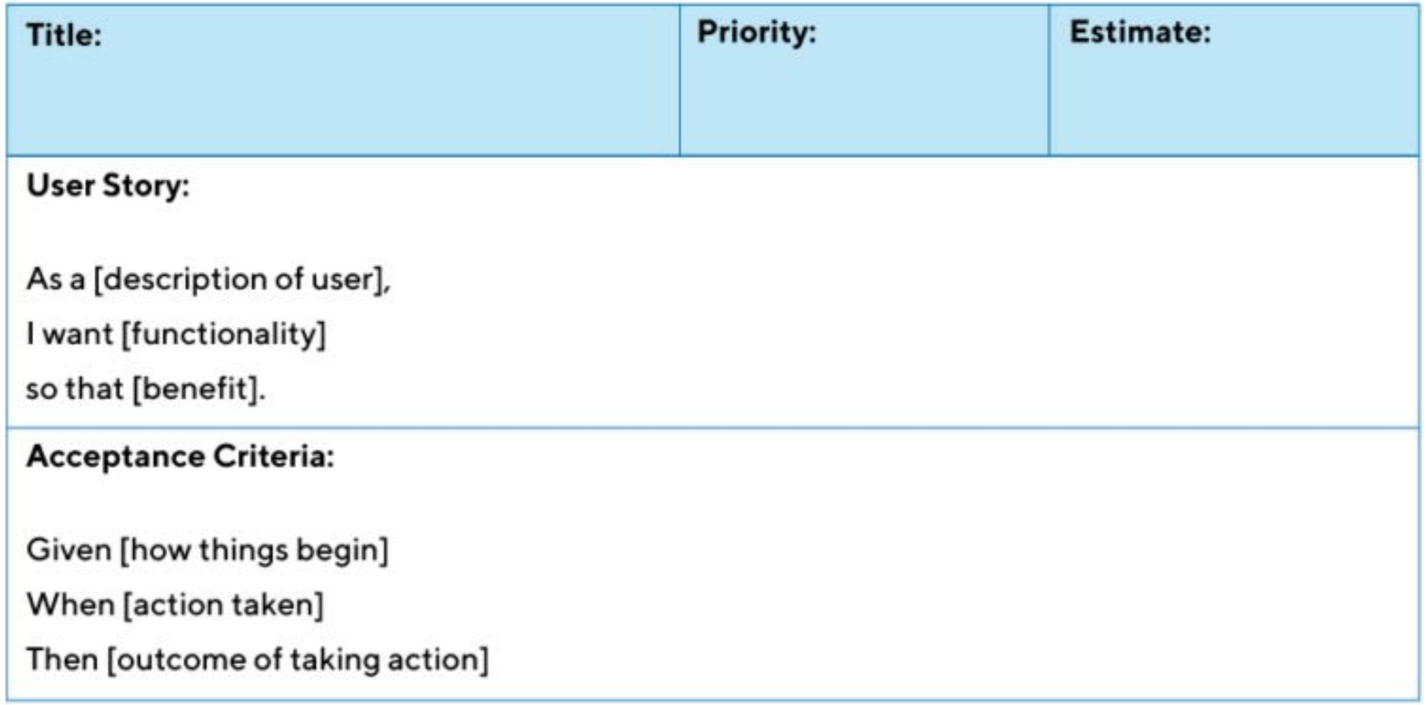
How to Write User Stories?

- ➢ Definition of “done”
- ➢ Outline subtasks or tasks
- ➢ User personas
- ➢ Ordered Steps
- ➢ Acceptance Criteria

What Makes a Good User Story?

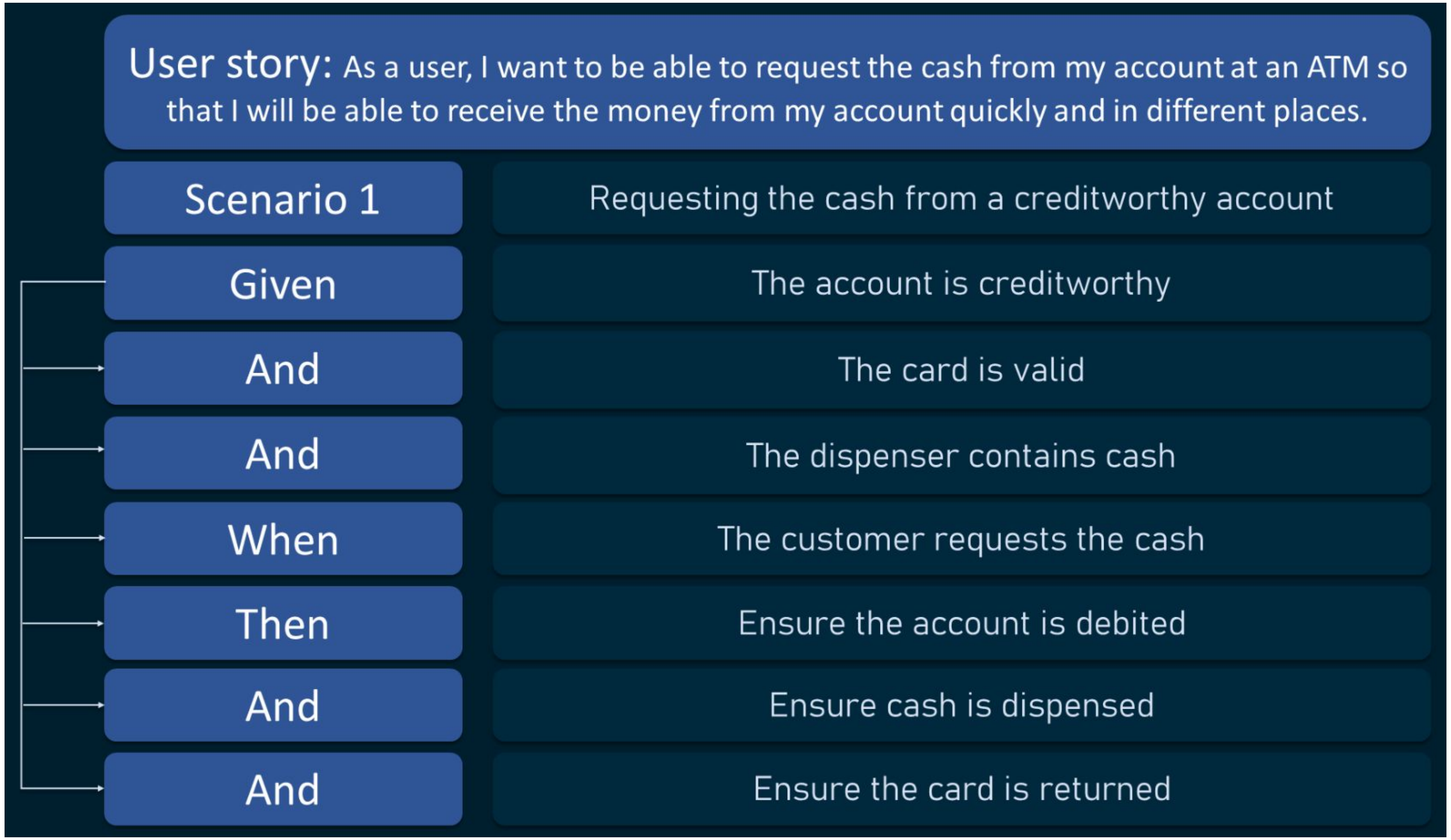
- ➢ Independent
- ➢ Negotiable
- ➢ Valuable
- ➢ Estimable
- ➢ Small
- ➢ Testable

User Story Template



Title:	**Priority:**	**Estimate:**
User Story: As a [description of user], I want [functionality] so that [benefit].		
Acceptance Criteria: Given [how things begin] When [action taken] Then [outcome of taking action]		

Scenario Oriented Acceptance Criteria



User story: As a user, I want to be able to request the cash from my account at an ATM so that I will be able to receive the money from my account quickly and in different places.

Scenario 1	Requesting the cash from a creditworthy account
Given	The account is creditworthy
And	The card is valid
And	The dispenser contains cash
When	The customer requests the cash
Then	Ensure the account is debited
And	Ensure cash is dispensed
And	Ensure the card is returned

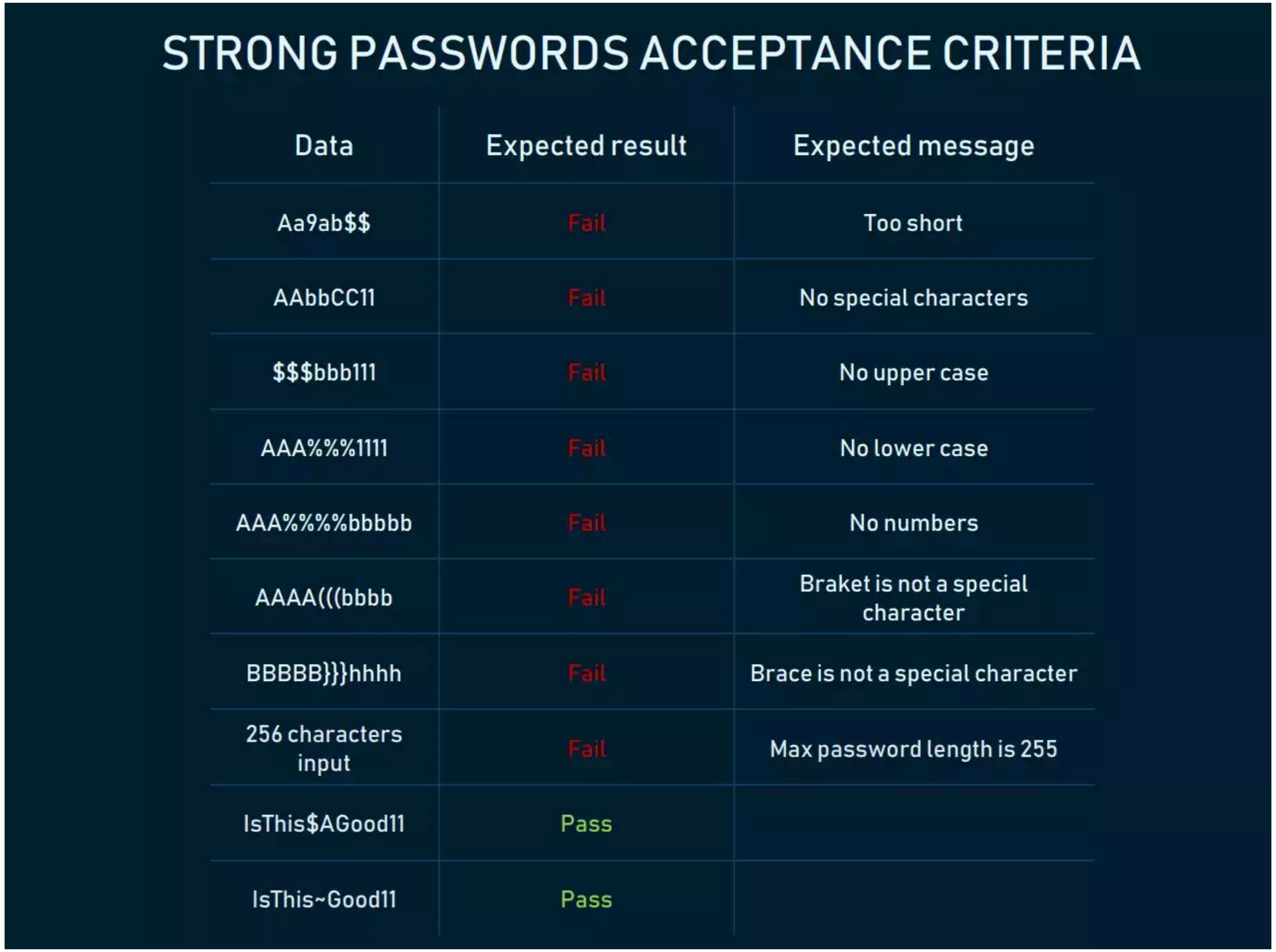
Courtesy of https://content.altexsoft.com/

Rule-Base Acceptance Criteria

User story: As a user, I want to use a search field to type a city, name, or street, so that I could find matching hotel options.

- ○ The search field is placed on the top bar
- ○ Search starts once the user clicks “Search”
- ○ The field contains a placeholder with a grey-colored text: “Where are you going?”
- ○ The placeholder disappears once the user starts typing
- ○ Search is performed if a user types in a city, hotel name, street, or all combined
- ○ Search is in English, French, German, and Ukrainian
- ○ The user can’t type more than 200 symbols
- ○ The search doesn’t support special symbols (characters). If the user has typed a special symbol, show the warning message: “Search input cannot contain special symbols.”

Other Formats (Acceptance Criteria)



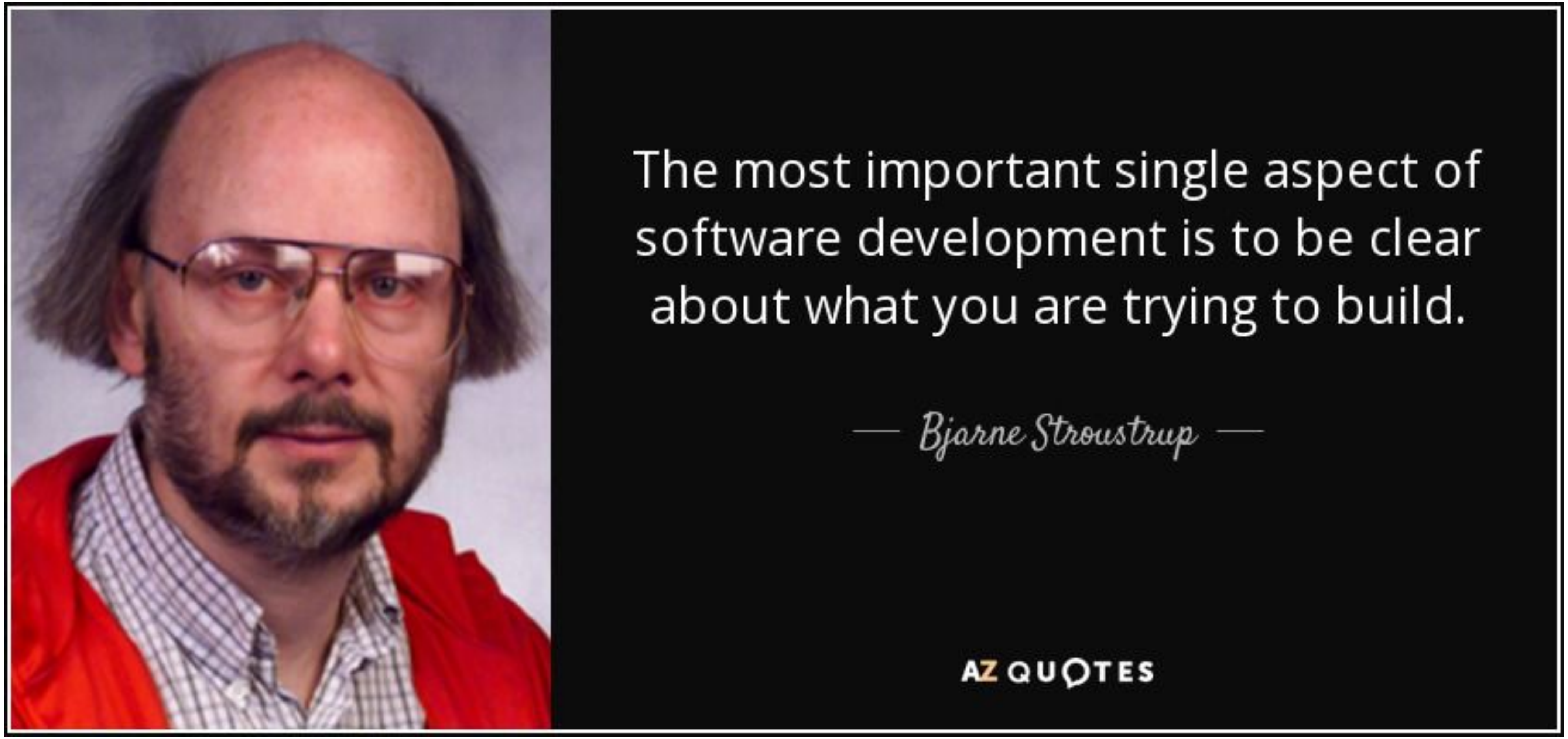
STRONG PASSWORDS ACCEPTANCE CRITERIA

Data	Expected result	Expected message
Aa9ab$$	Fail	Too short
AAbbCC11	Fail	No special characters
$$$bbb111	Fail	No upper case
AAA%%%1111	Fail	No lower case
AAA%%%bbbbb	Fail	No numbers
AAAA(((bbbb	Fail	Braket is not a special character
BBBBB}}}hhhh	Fail	Brace is not a special character
256 characters input	Fail	Max password length is 255
IsThis$AGood11	Pass	
IsThis~Good11	Pass	

Documentation

- ➢ Software Requirement Specification (SRS)
  - ○ https://www.perforce.com/blog/alm/how-write-software-requirements-specification-srs-document
- ➢ Product Requirements Document (PRD)
  - ○ https://www.perforce.com/blog/alm/how-write-product-requirements-document-prd

The Quote of the Day



The most important single aspect of software development is to be clear about what you are trying to build.

— Bjarne Stroustrup —

Readings

- ➢ Software Engineering: A Practitioner's Approach, Roger Pressman and Bruce Maxim, 9th Edition, September 2019, Chapters 7 and 8.
- ➢ Requirements Engineering for Software and Systems, Phillip A. Laplante, 3rd Edition, October 2017, Chapter 4.
- ➢ Scrum: The Art of Doing Twice the Work in Half the Time Hardcover, Jeff Sutherland, J.J. Sutherland, September 2014, Chapter 7.