
PROJECT REQUIREMENTS

WEEK 4 – 09/16/2019



REVIEW QUIZ

- <https://play.kahoot.it/v2/?quizId=2acf7537-d594-4c7c-9092-b75428cda228>

PROJECT MILESTONE I

- <https://sreeshanath.github.io/Project%20Milestones/Project%20Milestone%20I/index.html>

COMPUTING AND SOFTWARE CAREER AND INTERNSHIP FAIR: FALL 2019

- September 17th, 2019 from 10am-3pm
- Arrow Touchdown Club at Folsom Field (enter on the north side of the stadium)/ Dal Ward Athletic Centre
- 40 organizations
- Students can go on Handshake to view the attending companies and the levels and skillsets those companies are seeking.
- Career Services will also be on site for resume review and information about their services.

SCOPE

- Easily understood as a sum of REQUIREMENTS
 - “If it can’t do....., I don’t want it.”
 - “It must be able to
 - “I want a system that will

TWO TYPES OF REQUIREMENTS

I. Functional Requirements - “what” –
(Users’ view)

2. Non-Functional Requirements - “how” –
(Developers’ view)

The definition of a functional requirement is:

“Any requirement which specifies what the system should do.”

User Requirements, Business Requirements

Documented by Use Case

The sum of the Use Cases = “User Requirements”

These requirements describe the users' expectations for what the system does for them

“WHAT I want the system to do for our organization”

A functional requirement for an everyday object like a cup would be:

“ability to contain tea or coffee without leaking”.

FUNCTIONAL REQUIREMENTS

NON-FUNCTIONAL REQUIREMENTS

The definition of a non-functional requirement is:

“Any requirement which specifies how the system performs a certain function.”

Ties Use Cases to Technical Implementation

Describes HOW the system should behave

A non-functional requirement for the cup mentioned previously would be:
“contain hot liquid without heating up to more than 45 °C”.

These requirements impact the users’ experience

For example, “The home page must load within 1.4 seconds.” Or, “If a document is flagged as private, only the user who created it can see it.”

THREE OPTIONS FOR REQUIREMENTS CAPTURE

Software Requirements Specification

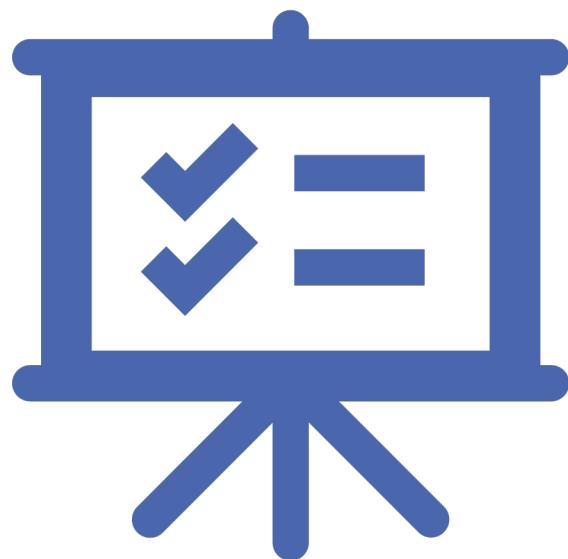
Use cases

User stories

For all 3 you need to be **SMART**

- **S**pecific – This is subjective but aim is to disambiguate what the requirement is to be accomplished
- **M**easurable – could you write a test for it?
- **A**chievable – can it realistically be implemented given constraints of available resources?
- **R**elevant – ask five dependent why questions – do they converge on the importance of the requirement to business goals
- **T**ime-Boxed – can a time-budget for developing be identified

THREE OPTIONS FOR REQUIREMENTS CAPTURE



- You'll choose to use one or more of these forms depending on the type of project and sponsor wishes:
 - Typically larger, long-ranging projects will look for the **Requirements Specification**
 - **Use Cases** are a lower overhead version of Requirements Specification used in large projects employing Object-Oriented approach
 - **User stories** are most common approach for Agile projects

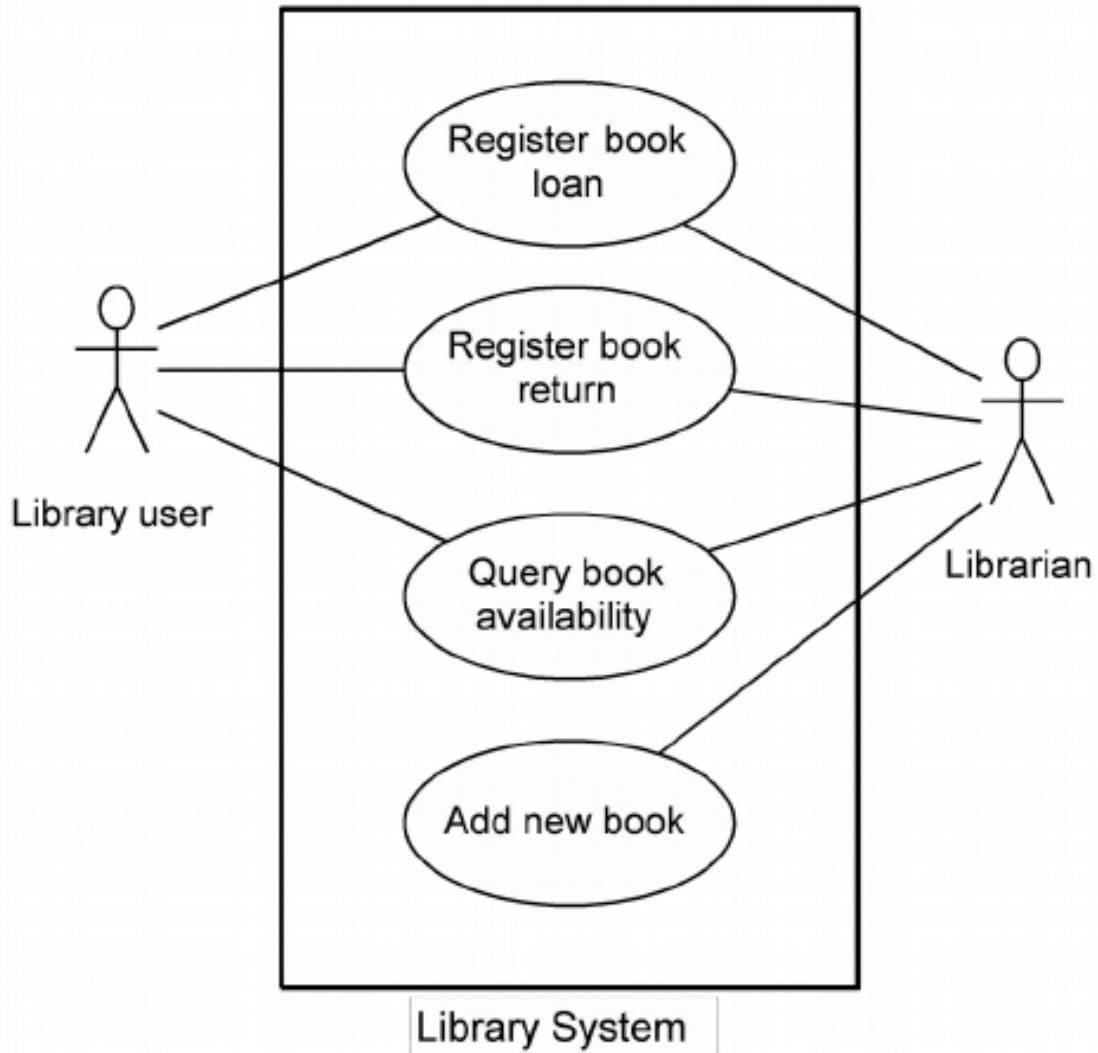
SYSTEM REQUIREMENTS SPECIFICATION

- A **System Requirements Specification (SRS)** (also known as a Software Requirements Specification) is a document or set of documentation that describes the features and behavior of a system or software application
- <https://dzone.com/articles/how-to-write-the-system-requirements-specification>

USE CASE

- A use case tells a stylized story about how an end user interacts with the system under a specific set of circumstances.
- A use case depicts the software or system from the end user's point of view.
- Define Actors
- Use case should answer a number of questions:
 - 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?
 - What variations in the actor's interaction are possible?
 - What system information will the actor acquire, produce, or change?
 - Will the actor have to inform the system about changes in the external environment?
 - What information does the actor desire from the system?
 - Does the actor wish to be informed about unexpected changes?

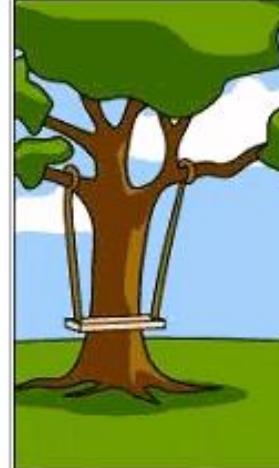
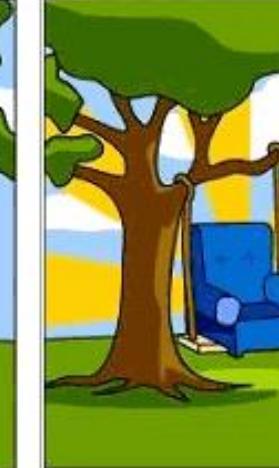
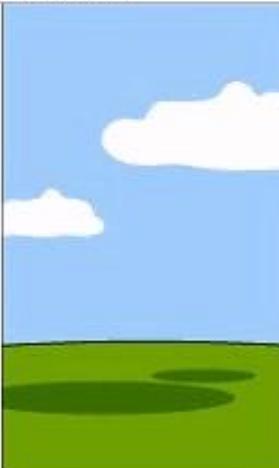
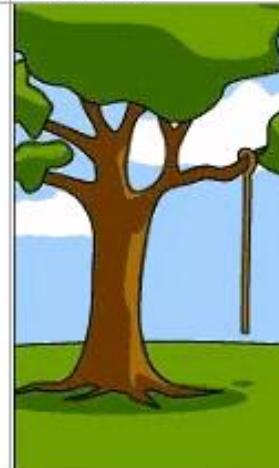
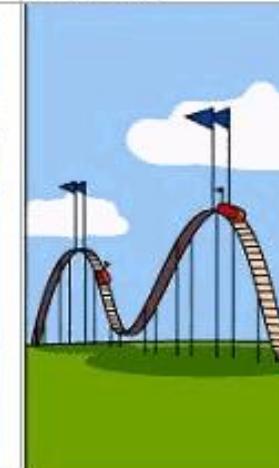
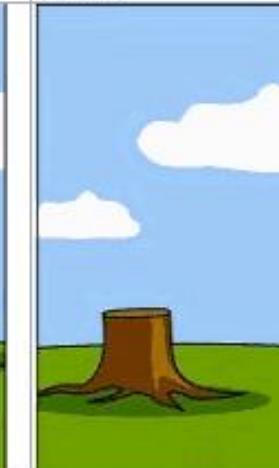
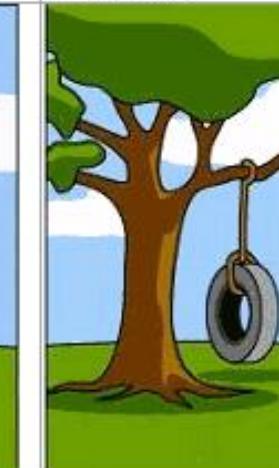
USE CASE EXAMPLE



USER STORIES

In Agile model, requirements are called user stories.

- Must be small enough that it can be completed within iteration.
- What is a user story?
 - **What** : Description of the feature
 - **Who**: Person who will use or benefit from the feature.
 - **Why**: “business value”
 - **When**: Depends on the priority
- Don't forget to add an **acceptance criteria**.
- An acceptance criteria is a set of conditions that are used to confirm when a Story is completed.

				
How the customer explained it	How the project leader understood it	How the analyst designed it	How the programmer wrote it	How the sales executive described it
				
How the project was documented	What operations installed	How the customer was billed	How the helpdesk supported it	What the customer really needed

ACTIVITY

- Lets assume we are required to create a website that helps students enroll in CSCI 3308
- Come up with:
 - 3 actors
 - 10 use cases
 - 5 user stories
- We'll build this website as we are navigating through concepts in the next few weeks