Lecture 4

Iterative Development, Unified Process, & Usage Case

- Waterfall Development Model: In software development, it tends to be among the less iterative and flexible approaches, as progress flows in largely one direction ("downwards" like a waterfall) through the phases of conception, initiation, analysis, design, construction, testing, deployment, and maintenance.

- Iterative Development Model: In software development, it is a flexible and iterative approach to software design, where the software development of a large application is broken down into smaller chunks. The "Unified Process" is a very popular example of the iterative model.

- Unified Process: It organizes software development over 4 main phases: Inception, elaboration, construction, and transition.

- There are 7 parts in the inception phase:

1- Vision: Defining the most important goals and creating an executive summary.
2- Usage Case Modeling: Defining names for most usage cases and analyzing almost 10% of them which are the riskiest.
3- Supplemental Specification: Defining other requirements that are needed for our system.
4- Risk-Management Plan: Defining the biggest risks and how they can be solved.
5- Proof of Concepts: Defining the vision of the system.
6- Iteration Plan: Defining what to do in the first elaboration iteration.
7- Software Development Plan: Defining the time needed for each iteration.

- The "Unified Process" requirements are structured based on the "FURPS+" model: Functional, Usability, Reliability, Performance, Supportability, Implementation, Interface, Operations, & Licensing.

- Usage Case: It provides a simple way to discover and record system requirements. It consists of actors and scenarios. There are three basic types of actors: Primary user (the person the system is designed for), supporting actors (e.g. external computer systems, verifiers, people, etc.), and offstage actors (outside elements that have interest in the system but are not key to the system primary use). A scenario documents the series of interactions between the actors and the system, each actor could have numerous scenarios while interacting with the system.

- There are 3 usage-case formats:
  (1) Brief: It consists of one paragraph that describes the most successful scenario.
  (2) Casual: It consists of many paragraphs that cover many scenarios.

(3) Full Dress: It lists each possible step and variation in the process. The “Alistair Cockburn Format” is a very popular fully dressed usage case.

- Example on the “Alistair Cockburn Format” for usage cases:

1- Use/Usage Case: Provide ATM User with Money from Checking Account

2- Scope: ATM Software Application

3- Level: User Goal

4- Primary Actor: Bank Customer

5- Stake Holders and Interests
  a- Bank Customer
    1- Access to All Funds in their Checking and Savings Accounts.
    2- Deposit Checks
    3- Balance Checking
    4- Receive a Receipt
  b- ATM
    1- Providing Available Funds to Customers
    2- Automatic and Instant Update of Fund Changes
    3- Protecting the Bank & Customers from False Cards or PINs.
  c- Bank Home Office Computer
    1- Providing Accurate Account Funds Information
    2- Immediate System Update upon the Occurrence of Any Transaction
  d- Fraud Department
    1- Verifying Whether a Card is Stolen

6- Preconditions
  a- A valid card is entered.
  b- A valid PIN is entered that matches the card.

7- Postconditions
  a- Customer Satisfaction
  b- The transaction is disseminated to all bank systems.
  c- A photo of the customer is taken.
  d- A receipt is generated.
  e- Card is provided back to customer.

8- Main Success Scenario

a- The ATM displays a message on screen identifying itself to customers.
b- The customer inserts their card.
c- The ATM verifies the legitimacy of the card.

9- Extensions
  a- The ATM machine runs out of funds:
    1- Display a closed message.
    2- Shutdown the machine.
    3- Send a message that a technician needs to fill up the ATM with funds.
  b- The ATM runs out of receipt paper:
    1- Display a closed message.
    2- Shutdown the machine.
    3- Send a message that a technician needs to fill up the ATM with receipt paper.

10- Special Requirements
  a- The text must be readable by color-blind people.
  b- The text must be readable by people who speak other languages.

11- Data Variations List
  a- All data uses 256 bit AES encryption.

12- Frequency Occurrence
  Could be constantly.

13- Miscellaneous
  a- Is there anything we could do to improve user experience for the blind?
  b- Can polarized screens help improve security?