LTAT.05.003
Software Engineering

Lecture 01.3:
Homework Assignment 1

Fall 2020
Lab 1 Assignment

Joostes Marss AS

Customer: Home Improvement International (HII)

POS System

To do:
- Interviews to identify roles and requirements (user stories)
- List of 20 functional user stories
- List of 10 non-functional user stories (5 performance & 5 usability)
Lab 1 Assignment

Joostes Marss AS

Customer: Home Improvement International (HII)

POS System

To do:
- Interviews to identify roles and requirements (user stories)
- List of 20 functional user stories
- List of 10 non-functional user stories (5 performance & 5 usability)

Have a look at the videos on the course wiki:
→ YouTube videos by Lars Bilde (Kaj Car Rental Service)
YouTube videos by Lars Bilde (Kaj Car Rental Service)

Lectures

Note: Information on how the lectures are conducted can be found here.
Also, the handouts and materials are still from last year’s lecture (2019). Updated slides, videos, and materials will be posted as we go. The overall structure and the topics will not change. According to the coronavirus rules for large classes in our institute, the lectures have to be conducted online.

- Lecture 1 (04-Sep-2020) - Introduction to Software Engineering
  - Course Organization: (pdf-2020-1) (video-2020-1)
  - Overview of Software Engineering: (pdf-2020-2) (video-2020-2)
  - Background / Motivation of Homework Assignment 1: (pdf-2020-3)
  - Zoom Lecture Recording:

- Lecture 2 (11-Sep-2020) - Requirements Engineering I
  - Slides: (pdf-2019)
  - Video: (video-2019)
  - Lean requirements (by Atlassian)
  - INVEST in good Stories, and SMART Tasks (by Bill Wake)
  - Hints on how to write testable requirements
  - Balsamiq - A tool for developing wireframes (Mockups)
  - A series of YouTube videos by Lars Bilde explaining the steps from first interviews with stakeholders to user stories, tasks, and sprint planning (note: these videos promote the use of the commercial tool ScrumWise - any other backlog management and sprint planning tool may be used instead):
    - Initial user stories extracted from interview
    - First backlog items (user stories)
    - Finding more User Stories from existing User Stories (e.g., by splitting up)
    - Splitting User Stories into Tasks?
    - General remarks about the right level of backlog planning detail
    - Planning the first (or next) sprint
    - Time tracking with the Task Board
    - Sprint Burndown Chart
A Requirement in NL

- The software should enable locking and unlocking of apartment doors.

Good / Bad requirement?
User Stories

As a tenant | I can lock the doors | to protect my apartment.

role (benefactor)  capability (action/means)  business-value (motivation/rationale/end)

- Similar to NL requirements, but focus on the user benefits, instead on system characteristics (alone).
- Unfortunately, third element (business-value) is often ommitted
- Preferred technique in agile methods.
User Story:

As a tenant | I can lock the doors | to protect my apartment.

• Similar to NL requirements, but focus on the user benefits, instead on system characteristics (alone).

• Unfortunately, third element (business-value) is often ommitted

• Preferred technique in *agile methods*. 
‘Normal’ User Story

<Actor/Role> As a logged-out user

... 

>Action> I want to be able to sign in to my app

... 

<Value> so I can access my personal profile

A good User Story is:

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

INVEST
‘Normal’ User Story

<Actor/Role> As a logged-out user
... 

<Action> I want to be able to
sign in to my app
... 

<Value> so I can access my
personal profile

Acceptance test:
Given I’m a logged-out system user
And I’m on the sign-In page
When I fill in the fields “Username” and “Password” with my credentials
And I click the sign-In button
Then the system signs me in
‘Non-Functional’ User Story

<Actor/Role> As an end user

... 

<Action> I want error messages be shown always at the same place of the screen 

... 

<Value> so I can easily detect them

A good User Story is:
- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable

INVEST
‘Device’ or ‘System’ User Story

<Actor/System> As a web crawler

... I need a URL dictionary without duplicates or dead links

... so that the crawling process is faster

A good User Story is:
- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable

INVEST
‘Technical’ User Story

<Actor/Role> N/A

<Action> Implement Smart Distance Algorithm for multi-paragraph web-pages

<Value> so that relevant and random connections between search entities could be distinguished

A good User Story is:
- Independent
- Negotiable
- Valuable
- Estimable
- Small
- Testable

INVEST
What is a good (or bad) User Story?

At the beginning of next lecture, I will
• explain INVEST and discuss quality characteristics of User Stories and
• present/discuss examples of good/bad User Stories

• To prepare, look at the following examples …
User Story Example – Wrong or Right?

I want to see an error when I cannot see recommendations after I updated an article.
User Story Example – Wrong or Right?

As a User, I am able to click a particular location from the map and thereby perform a search of landmarks associated with that latitude longitude combination.
User Story Example – Wrong or Right?

As a care professional I want to save a reimbursement—add save button on top right (never grayed out).
User Story Example – Wrong or Right?

As a User, I am able to edit the content that I added to a person’s profile page.
User Story Example – Wrong or Right?

As a care professional I want to see my route list for next/future days, so that I can prepare myself (for example I can see at what time I should start traveling).
User Story Example – Wrong or Right?

As an Administrator, I am able to add a new person to the database.

As a Visitor, I am able to view a person’s profile