Quiz 02: User Stories

Due Feb 4 at 11pm **Points** 50 **Questions** 11 **Available** until May 15 at 11:59pm **Time Limit** None

Instructions

You **may** use the slides from the lecture and other sources to answer these questions. Please be sure to cite any references but be sure to answer the following questions in your own words. Do NOT simply cut and paste the information from the slides. You will receive a score of 0 if you copy the prose from the slides.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	207 minutes	50 out of 50

Score for this quiz: **50** out of 50

Submitted Feb 2 at 7:35pm

This attempt took 207 minutes.

Question 1

5 / 5 pts

Compare and contrast the role of **developers** in Agile and plan driven cultures.

Your Answer:

In Agile developer assign themselves to the team. Teams are self-organizing. Achievements are earned by how much the team has worked. Team members select tasks. Since Individuals work for the team, team is responsible for reflection and continuous improvement.

In Plan driven culture manager is the one who assigns the teams and what roles everyone will play. Here individual developer work for the manager and not for the team. Developers earn their achievements solely based on their individual strength and task assigned to them. Manager assigns tasks. Since everything is handled by managers improvement in projects the responsibilities of managers as well.

Plan Driven Cultures	**Agile Cultures**
Managers assign teams	Self organizing teams
Individuals work for the manager	Individuals work for the team
Individuals measured on individual achievements	Individuals measured on team achievements
Manager assigns tasks	Team members select tasks

Question 2

5 / 5 pts

Compare and contrast the role of **managers** in Agile and plan driven cultures.

Your Answer:

Managers in Agile runs according to the team which is self-organized and works as the requirement of the software. Whereas in plan driven method the manager sets up the team and assigns tasks accordingly to the members of the team.

In plan driven method manager defines the solution and asks the team members to implement it for the software. Whereas in the agile method manager asks for feedback and questions about the project and waits for the team to find a solution for a particular task.

Plan driven method has a leader as manager of the project guiding the team to the success. While in agile method the manager makes the path much easier for the team to find solutions and guiding towards the success.

The trust of the manager in agile methods is more compared to plan driven method. The managers depend on the team while in plan driven the manager order and controls whole project until it is delivered.

If there are any errors or failures in the system or the project while building the process all the blame is owned by the manager in plan drive method. Whereas in agile method it is the responsibility of the team in such kind of situations.

Plan Driven Cultures	**Agile Cultures**
Define solutions to be implemented by the team	Asking questions while allowing the team to create the solution
Define roles and responsibilities	Help team to self organize
Leading the effort	Enabling the team while clearing roadblocks to success
Command and Control of the team	Trusting the team
Manager owns the problem	Team owns the problem

Question 3

5 / 5 pts

Describe two examples of what can go wrong with teams transitioning from plan-driven to agile methods.

Your Answer:

The transition would lead to teams, fault but rather working in agile environment they were working in plan driven which is usually controlled by the manager. The working in agile environment without any task assigning process would be more confused and would face lack of knowledge of the software or project. Since the team was trained in a plan driven system it would fail to adapt quickly in Agile environment. This may lead to many errors and create problems in software development situations. It would fail in developing the software as lack of training towards the iterative method used in Agile environment.

Continuous testing and customer feedback is needed in agile methods while the team is again not trained for the same.

Since the team is from plan driven methods it may face huge problems like taking early decisions implying it without customer involvement. This may make situation very difficult when the team must adapt a new thing or make a change in the software. Employees might fail in changing and updating the systems due to their nature in plan driven system.

Failure may be attributed to:

Lack of clarity across the team

Using only the ***worst*** of Waterfall and Agile together:

Forcing frequent short deliverables without the Agile advantages

Inadequate training or support for Agile Methods

Yesterday: product manager; today: scrum master

Failing to use automated testing and/or continuous integration

Continuing to plan everything in advance and not allowing change

Failing to change employee performance metrics

Failing to inspect and adapt – Andy Hunt

Question 4

5 / 5 pts

You've been hired by Driverless Software to gather requirements for new "hands-free" parking hardware/software solution.

Identify two features that the "hands-free" parking solution should include.

Your Answer:

Two features which are most important in hands-free parking solution should include:

1. Proximity sensors- There are many companies which have adapted parallel parking or hands-free parking in their cars. While still many systems require some assistance by the driver while the automated systems park the car. sensors are most important feature without which the system cannot complete the process successfully. These sensors can locate the parallel parking upon activation of such proximity sensors. These sensors are located on the front and backside of the car. These help to detect nearby objects follow the grid while parking in the parking spot. This feature helps the car to determine the signal which indicates what side of the road it should look for a space. When this space is located it gives directions to the driver to follow few steps so that the car park without any assistance required from the driver. These sensors detects when the car shifts to the reverse gear. Hands-free parking system will be activated and then it will itself operate the gas and brakes until the car gets parked by itself. These sensors can also provide steering assistance for leaving the spot.

2.UI and mapping- Along with proximity sensors the system requires one more important feature that is UI and software help for mapping the spot where the car must be parked and the UI which will help the driver to understand the hands-free system and requirements which can be needed while parking the car. There can also be failures or errors which the system must notify to the driver that there's an issue with a particular sensor camera or the system itself. This UI can help the driver to determine or set few boundaries when needed. Mapping is an important feature that sensors cannot complete. Proximity sensors does not help the car to plot and make grids which car can follow to park in the parking spot

Question 5

5 / 5 pts

Write **use cases** to represent the two features you chose for the "hands-free" parking solution. Be sure to follow the format specified by Ivar Jacobson. (You do not need to include UML diagrams)

Your Answer:

1)Name: Proximity sensors

Brief description: These sensors are located on the front and backside of the car. These help to detect nearby objects follow the grid while parking in the parking spot. This feature helps the car to determine the signal which indicates what side of the road it should look for a space. The car can complete the process of parking without any failures and crashes.

Actors: Driver, software, and sensors.

Basic flow:

1. the driver switches to reverse gear

2)the proximity sensors detect it.

3)hands-free Parking process starts.

4)Follows the grid and guides car to the parking spot.

5)Handles break, gas and steering to have a successful completion.

6)Everything is notified to the driver in the screen.

Alternate flow:

1)User may pause the process and change the desired spot.

2)User may change the direction of the car which was originally followed.

3. User may abort the process due to some error.

4)It may be not possible for the hands-free system to park in the spot chosen by the user.

2)Name: UI and mapping

Brief Description: The system requires one more important feature that is UI and software help for mapping the spot where the car has to be parked and the UI which will help the driver to understand the hands-free system and requirements which can be needed while parking the car.

Actors: Driver, Software.

Basic flow:

1. The system notifies user that the sensors are on.
2. System then performs mapping of the parking spot.

3)Software then develops grids and map that should be followed by the car for hands-free parking.

4)UI notifies the user and shows the user grids and guides the driver if anything is needed.

Alternate flow:

1. UI may fail to draw grids and map.

2)There can be any obstruction that can lead to errors and failure.

3. Driver may change the grid or map and software has to adapt accordingly.

Each use case should include:

1.Name

2.Brief description

3.Actors

4.Basic flow

5.Alternate flows

Question 6

5 / 5 pts

Write **user stories** to represent the same two features you chose for the "hands-free" parking solution. Be sure to follow the format used by agile methods.

Your Answer:

User Story 1:

Title: UI and Mapping

Acceptance test: Designing grids and path for parking spot.

Priority: 1

It is the most important part of the software which includes building the UI and having a hardware which will help building grids and mapping route for the parking spot. This may require more 10 people and more than 15-20 days which will include software development and hardware setup.

Story points: 15

User story 2:

Title: Proximity sensor

Acceptance test: These help to detect nearby objects follow the grid while parking in the parking spot.

It is next important part which completes this hands-free parking system. These sensors are important and located in front and back of the car. This includes both hardware and connection to the software which can keep notifying status and service provided by the sensors. This may include 5-6 people and approximately 3-4 days for connection and installation of the proximity sensors.

Story points: 3

Should include:

Title – a short handle for the story. Present tense verb in active voice is desirable

Acceptance test –the name of a method to test the story

How to determine if the functionality is provided?

Acceptance test helps to flesh out the details of the user story

Priority – decided by the customer

Story points – estimated time to implement expressed in relative units

Description – one to three sentences describing the story

Priority is mentioned in the first user story but missing in the second user story . both of them should have a priority.

Question 7

5 / 5 pts

Would you recommend use cases or user stories for this application? Why?

Your Answer:

User cases and user stories both focus on building a software, but both have a different approach. User stories is a goal-oriented approach which is written from the perspective of a user, uses the natural language of the business, and follows its method internally not having all the steps mentioned. While use case is a granular approach which defines the internal system and how it will act when it will be ready.

According to me user stories is a better approach having its own benefits because it is a goal centered approach. In this method there are different stories explained accordingly and based on their priority respectively. Each stories have a brief description of tasks from user point of view. Unlike the old approach, User Story focuses on what the user needs instead of what the system should deliver. Having such efficient method, it improves flexibility of the system and communication with customers. This can also help solving customer's operational problems and most importantly adding value to the organization.

While it is an efficient and well-defined approach it still has problems which can affect the software and where we can again need use case method.

1)Lack of internal knowledge

2)No mechanism for further upgrade or changes that will be needed in the future.

3)Having goal centered approach team may miss error handling and other small iterations because of all focus is on the goal.

Question 8

5 / 5 pts

Describe two **criteria** for identifying valuable and useful user stories.

Your Answer:

Acceptance criteria:

The user describes the functionality needed by the software and the requirements to the team members. Team members use this feedback and build a much more efficient user story which can explain the feedback and system goal based on users feedback. It provides a detailed scope of the user story and what is needed so your team can understand what they’re up against.

1) Upgrades the communication and user experience with the team.

2) Speed is improved due to much better understanding of the system.

3) It also explains what the user story is intending to do.

Customers:

If the user is happy with the product, then only a system can be successful. For this there should be maximum interaction between the customer and team members building the software.

User stories should capture what the user wants to achieve in a simple sentence. It should be small, simple and understandable. It should have brief description of what user has requested for.

Conversation is very important between developers and customers. The user must not be an observer rather he should the part of the team building the system.

Each story should add value to the customer

Customers write user stories

(with help from developers if needed)

Stories need to be small enough that several can be completed per iteration

Replace big stories with several smaller stories

Stories should be independent (as much as possible)

Stories must be testable – like any requirement, if it cannot be tested, it’s not a requirement!

Include non-functional requirements as User Stories

Question 9

5 / 5 pts

Describe planning poker. What is the goal? What is the process?

Your Answer:

Planning poker is a consensus-based technique which helps the team in building the whole system. In agile environment teams usually use poker planning to estimate their product backlogs. Planning Poker can be used with story points, ideal days, or any other estimating unit. In a way it helps in estimating the amount of work that will needed to build the software and amount of time which will be required to complete the specified work.

This process makes sure all the team members get involved in the estimation process and share his or her knowledge which can help in building the system and further estimation. Poker planning meetings are usually attended by the team members, the project's owner. The team members in estimation process asks the product owner to answer few questions about the software and system which will be required.

All the members have card numbers from 0 to 100 and start estimating and assigning card numbers to specific task which corresponds to number of days needed to complete that task, stories, and other units. When discussion of task is complete each team member privately selects one card to represent his or her estimate for that task.

The middle cards are often numbers in the Fibonacci sequence. Each deck will also contain a card that represents the concept that work may take uncertain amount of time to complete that task. The process is repeated until the group achieves consensus or manager decides that consensus is not possible, and the story must be broken down into simpler parts before the project can move forward.

Question 10

5 / 5 pts

Your team has chosen to use planning poker to estimate effort for your user stories. Two of your best developers, Pam and Vincent, can't agree on an estimate. What are two different options to resolve the problem?

Your Answer:

Project owner and team comes up with different approaches most of the times and come up with plans, design, and a better approach towards the goal. During estimation, project owners describe his requirements and approach towards the planning and decides estimation for different tasks and completion of the project. There can be disagreements between the team members while deciding this estimation process. In this situation Pam and Vincent have disagreements in the estimation process. Let's assume that Pam is having estimation of 10 points and Vincent is disagreeing on both project owner and Pam's estimation.

In such situation, first option can be to use the INVEST acronym. The team can use N for negotiable, Both Pam and Vincent can settle on an estimation by having a conversation on the same task and agreeing on a particular point.

Second approach can be if they cannot negotiate or the worst case there is no agreement, according to greening the team should approach divide, defer, split, and take low estimates so everyone can have consensus and work incorporation.

Use the highest estimate

Use the lowest estimate

Break the task into multiple stories and estimate those.

Table the discussion for now and come back to it later.

Question 11

0 / 0 pts

“I pledge on my honor that I have not given or received any unauthorized assistance on this assignment/examination. I further pledge that I have not copied any material from a book, article, the Internet or any other source except where I have expressly cited the source.”

Correct!

● True

○ False

Quiz Score: **50** out of 50