**HOMEWORK WEEK 5-6**

(handout for students)

**TASK 1 (Agile Techniques)**

**Question 1**

**Complete definitions for Scrum related key terminology provided below**.

SCRUM CEREMONIES

Product backlog refinement – is when the product owner and the development team work together where they add detail, implement time estimates and order the items within the backlog in terms of levels of difficulty before starting the tasks. The benefit of doing this is that it helps the development teams to have a better understanding of the tasks at hand and to meet the goals for each sprint whilst planning more efficient sprint planning meetings. The tasks in the backlog are reviewed can be updated at any time by the product owner and the scrum team and development members whilst working under the product owner. The scrum team will normally decide and at what point the refinement of the items will take place.

Sprint planning - this is the first event that takes place in scrum to start the sprint. Within the sprint planning the goals of what tasks should be delivered before the next sprint which is usually done by the product owner. How the tasks will be achieved and the process of what needs to be done complete it is usually discussed by the scrum team and which backlog tasks need to be worked on to help achieve this goal. Sprint planning is done in collaboration with the rest of the team members and is used to help create a positive environment to help motivate people to want to get tasks completed. The development team helps to plan the work that is needed to reach the goal and explains their thoughts with the product owner. The better the product backlog refinement the more efficient and successful the sprint planning will go. Sprint planning should be at a maximum of eight hours with roughly two hours of sprint planning for every one week of sprint.

Daily scrum – this is approximately a 15-minute meeting in the morning where the developers are asked what was achieved yesterday to the development team to reach the goals associated with the tasks. What do you aim to complete today to help the development team meet their overall sprint goals. Lastly are there any blockers that are potentially hindering from reaching your sprint goals. This meeting usually takes place at the same time every day during the sprint and the product owner as well the scrum master could also be working as developers trying to complete the tasks within the sprint backlog. This time as well as additional time throughout the day accounts for the developers to adapt their way of finishing a task and have detailed conversations with other members of the scrum team about this. Overall, a daily scrum helps to improve self-management of tasks, communication between different people and problem solving.

Sprint review- is the second last meeting to take place within the sprint and the duration of this is a usually up to about 4 hours for a month sprint and can be shorter depending on the length of the sprint. The product owner can invite the scrum team and any other people he sees valuable which allows the scrum team to display what tasks within the Backlog have been completed and which has not been completed. The key factors that went well in the sprint, the problems that occurred and how these problems were overcome will also be discussed by the developers. The developers will also demonstrate the tasks that they have managed to complete successfully and answer any questions that people have. The product owner can then discuss the remaining tasks in the product backlog and helps to estimate what the potential target dates are for the delivery of the project which is dependent on the current progress being made. The whole scrum will input on the following that need to be achieved which can be seen as useful for the next sprint plan so the result of this will help to provide an idea of what the potential product backlog tasks could be for the next sprint. Lastly having an overall review on the budget, change in the marketplace and when the next tasks of the product will be completed by.

Sprint retrospective – this is a meeting that takes place at the end of the sprint to talk about what went well during the sprint and what could be improved for the next sprint as well how to go about it is discussed. The time frame of the retrospective is usually about three to four hours for a one-month sprint and can be shorter for shorter sprints. The benefits of this are to help increase the quality and success of the project as well as help to develop, deliver and manage it. It is the role of the scrum master is to help make the rest of the team in a positive mindset to improve how they have carried out their process of work during the sprints to allow the scrum team to help increase the quality pf the product that is being produced. As a result of this there should be areas of improvements identified that can be applied within the next sprint which can be taken place at any during the following sprints. The sprint retrospective mostly focuses on the team whereas the sprint review will mostly focus on the product.

SCRUM ROLES

ScrumMaster – helps to implement the use of scrum as well as coaching, planning, and advising the rest of the scrum team to ensure that the following guidelines are being implemented. This is done by helping manage the teams, removing any blockers, understanding the tasks that need to be completed in the product backlog and ensuring all the scrum events take place within the given time frame. They also help to find techniques that are effective to help with the management of the product backlog.

Product Owner – they help to represent the customer to the development team and is known to help define stories which meet the needs of the user as well as organize and structure the team’s backlog. They can do this by understanding the needs of the current market and the customer. The product owner is known as the only person who can reorganize the order of the items within the backlog. They are also responsible for transferring the aims of the product manager into specific tasks that need to be worked on.

Development Team – this is team that works with agile methodology to produce a cross functional and self-organizing team that help to plan, test, perform analysis, write front-end and back-end code, and complete the tasks set to them from the product backlog. They are also responsible for creating a plan in the sprint and adapting this to endure they are meetings the goals set within the sprint as well as keeping themselves everyone else responsible.

**Question 2**

You are leading a development team that was given a task to create a new yoga booking system.

High level description of the system is as follows:

· It has a very simple interface to accept user input (bookings) and display classes information

· All bookings, appointments, schedules etc should be stored in a SQL database.

· There is a ‘backend’ system that should be written in Python to handle the logic and manage the data flow.

Your team has two weeks to build a simple prototype that will be shown to the client to seek their feedback and discuss further enhancements.

TASK

· Break this task into smaller stories (chunks of work) for the team to work on.

· Assume that one person works on one task.

· Mark tasks that can be worked on in parallel and perhaps those that need to be worked on in particular order.

**Solution**

Yoga booking system

1) The overall task will be to list the things that need to be done such as the booking form

webpage as well as the spaces available for each class. Then understand the features that need to be built followed by making the user interface connect it to the backend.

2) Tables in SQL create a database and connect the tables to have a database schema which contains the classes times and days and if the classes are junior, intermediate, difficult.

3) API endpoints would then be required to take the data and implement them within the table such as the number of available bookings left for each class.

4) Assign story points based on how difficult the task is and roughly how long each task should take where the more difficult the tasks the longer it will take.

5) The users should be about to login to the website page, they should be able to see the available slots for each class depending on the level of difficulty, they should be able to make a booking or cancel it if necessary and they should be able to fill out a form if they are struggling with anything on the website and can receive an email in return from the receptionist.

6) Have a daily stand-up to evaluate what went well, badly and any blockers.

7) Find out what people’s strengths and assign tasks accordingly.

8)Plan what can go in parallel where you can see which tasks depend on each other and do that accordingly.

9)Can make comparisons of the tasks where they have certain dependencies and assign people to them.

10)Tasks that have not been assigned yet go into the product backlog

11)Timeline and dependencies allow you to work out whether something will be parallel or not

**TASK 2 (SQL)**

***Question 1***

**Design a cinema booking system.**

Think how you would approach the problem and what are potential ways of solving it?

You do not need to write actual code, but describe the high-level approach:

· Draw a list of key requirements

· What are your main considerations?

· What would be your common or biggest problems?

· What components or tools would you potentially use?

· You are welcome to draw a diagram (a very simple one) for the process flow to explain how it is going to work.

**Solution**

Requirements

1. Film released
2. Dates and times
3. Seating area which room
4. When booked spaces and how to make those seats unavailable
5. If fully booked make sure it is clearly stated
6. Paying for tickets and when checking out make sure there is a code to use on the machines

Main considerations

1. For the user experience analysing how easy/difficult to use it.
2. The cost to build the booking system
3. How beneficial it will be to the cinema
4. Ensuring the right talent/expertise to build and maintain the system if there are potential errors
5. Text evidence of the current tasks and ensuring how easy it is to teach the booking system

Common or biggest problems

1. Overbooking system if you have self tickets and not stating if booked
2. When fully booked ensuring you don’t have the film still advertised
3. Code when given the tickets online could be incorrect
4. Too many people online and the server can’t work efficiently which means the server break down
5. No data type constraints
6. Backend and front end are not connected to each other properly for example when the data type is wrong but the front end does not have constraints that tell you to use a certain data type.

Components or tools you would potentially use

1) pycharm

2) mysql workbench

3) Flask Django to build the user interface

4) Django-> backend for building web applications -> works for python

5) Allow the system to export the data to excel

6) Trello, jira such as sprints using agile

GitHub, gitlab, git