Homework: Lesson 10 – Agile Techniques

# Question 1

Complete definitions for Scrum related key terminology provided below.

SCRUM CEREMONIES

# Product backlog refinement

# This is the process of adding detail or effort estimations and ensuring that all who are involved understand the details within the product backlog, and any refinements or amendments needed to it. An example of the items found within the backlog could include a list of features, changes to existing features, bug fixes, or other actions that need to be completed within that sprint, i.e. tasks that meet the collected requirements.

* **Sprint planning.**

This is an event that kicks of the new sprint. It is where the team will get together to initially plan the product backlog items that they will be work on during this sprint and an initial plan for the project overall for completing them.

* **Daily scrum.**

A daily scrum, also referred to as a standup meeting, is a typically 15 minute meeting whereby the team of developers, and sometimes the product owner or scrum master if needed will also participate, in order to communicate and discus the progress of activities, make changes if need be, and to make quick decisions regarding the workload. It is usually set at the same time and place every day to allow for consistency.

* **Sprint review.**

This event takes place at the end of a sprint and is designed to be able to allow all the participants to reflect and give actionable feedback for the next sprint. They will describe the work they have done during the sprint and to mainly demo the product to key stakeholders.

* **Sprint retrospective**

The sprint retrospective is a recurring meeting that is held at the end of a spring whereby everyone discusses what went well, what could be improved, what they would commit to doing in the next sprint – whereby actionable commitments are made. It is essentially a tool to identify improvements in order to make the next sprint better.

SCRUM ROLES

# ScrumMaster

# The scrum master is the person responsible for ensuring that the team functions within an agile environment, and follows the process and practices within it, making sure to clear any obstacles they face, addressing team dynamics, ensuring good relations and communications between the team and the product owner. It is someone who is a scrum master and can lead the team in an agile way of working.

* **Product Owner**

The product owner is the person who is accountable for maximizing the value of the product. They are in charge of developing the product goal, creating and clearly communicating the requirements, user stories and the product backlog, as well as organising the items within the backlog, and making sure that everything in it is clear and well understood. They are important in retaining and maintaining the conceptual vision and technical integrity of the features that need to be built.

* **Development Team**

The development team is the team that will build the product needed to be built within the sprint. It is typically built up out of 3 to 9 member teams which doesn’t include the scrum master or the product owner. Teams of more that 9 people have been found to have communication issues, therefore the preferred number is 5-7 people. This team is usually self-organised and cross functional in that it has people with varied skillset in order to accomplish the various tasks within the backlog.

**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, schedule etc should be stored in a SQL database.
* There is ‘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.

**User personas:**

***Administrator: Karen***

* Owns the yoga studio and manages the yoga session schedules as the managing administrator
* 45 years old, uses computers daily as she manages her yoga studio, and her yoga instructors as well as client inquiries or issues that arise.
* She comes from a fitness obsessed family, and her 2 children are both enrolled into various sports program after school
  + She’s very passionate about making sports accessible to all and getting people into fitness

***Yoga participant: Yasmin***

* She is 26 years old and works at a local law firm
* As an associate lawyer, she sometimes works irregular hours and late nights so sometimes her schedule isn’t as regular as it normally would be
* Did yoga during her university days and found this studio by her workplace and so wants to be able to continue it around her work schedule
  + wants to join a yoga class that fits around her timetable and the description fits the type of workout she is looking for in terms of activity level or ability, as she has experience in yoga beforehand

***Yoga instructor: River***

* She is 31 years old and works as a yoga instructor
* She has been certified for the last 8 years, and been teaching at Karen’s studio for the past 3
* As an avid yoga enthusiast, her aim is to make sure that her sessions and space is a place to disconnect from the busy world and to focus on oneself and their yoga journey
* She likes to eat full and nutritious foods, often favouring a large bowl of salad for lunches in between teaching with continuous large bottles of water of refreshment
* She is single and enjoys her free time outside of work in going on hikes, meeting with friends, and walking her dog

**Epics**

* **Scheduling System:** As an administrator, I would like to add or cancel classes on a system, and to add or delete the instructor accounts and user accounts.
* **Yoga Session Booking:** As a fitness enthusiast, I would like an easy way to quickly book and/or cancel my yoga session from the palm of my hand, around my busy schedule.
* **Schedule Viewer:** As an instructor, I would like to see my classes and how many people are booked into my classes from the palm of my hand.

**User stories:**

As an administrative user, I would like:

* + to be able to schedule and/or rearrange classes on the system, so that everyone has access to the up-to-date classes schedule
  + to assign or re-assign instructors to classes, so that everyone is aware of who is teaching which class
  + to ping booked participants of any updates, e.g., if class is cancelled last minute or instructor changed, so that the relevant participants are easily notified of any changes in the yoga studio
  + to be able to create accounts for new instructors, so that they have access to their teaching schedules and joining participants within their classes

As an instructor, I would like:

* + to be able to have an account on the system, so that I can have access to the instructor area on it
  + to be able to request to cancel a class, so that is an emergency occurs where I cannot teach, then the class would be cancelled by the administrator
  + to be able to see who is booked into my classes

As a fit young adult, I would like:

* + to be able to create or delete my account on the system, so I can either use or stop using the system
  + to be able to see the full schedule of classes for the week, so that I can plan which classes I can attend against my own schedule
  + to see which classes are fully booked and which classes are still accepting bookings, so that I can book into classes still accepting participants
  + to be able to filter between classes level ability and/or specific instructor classes, and/or morning, afternoon or evening classes, so I can quickly and easily find a specific class for my needs
  + to be able to click multiple classes to book for, so I can book them all at once rather than one at a time
  + to see own bookings on own user dashboard, so I can check which classes I’m booked into
  + to have the option to cancel the booking, so that if I can’t attend it anymore

**Tasks**

* Create **backend** system space
* Create the database tables to include a **user’s tables**, a **staff table** (for all staff including instructors within the yoga studio), and a **yoga classes table**, and a **bookings table** that contains the yoga session, participants - from the users tables - of that yoga session using the yoga session ID.
* Create logic to add, update and delete classes from database within backend, administrator use only
* Create logic to assign, update (i.e. re-assign) instructors to specific yoga class based on its class ID within the backend
* Create an alert function to that is run whenever there is an update to a class or a cancellation of a class, i.e. when the update or delete functions are run within the backend
* Create new account classes
  + One for administrator
  + One for instructors
  + One for regular users
* Create functions to update and to delete user accounts
  + For instructor accounts
  + For regular user accounts
* Create request cancellation function to be used by instructor accounts only, for administrator account to verify
* Create verify and delete class function to be run only when an instructor requests a cancellation of a class, to the administrator user only to confirm
  + Once confirmed, the delete class is then run
* Create an add booking function for regular users
  + This function can accept multiple checked selection of classes to book that user into
* Create a cancel booking function for regular users
* Create a view currently booked classes to return all currently booked classes from the database to be viewed on frontend
* Create a view teaching classes to return all current classes to be taught by that specific instructor based on that instructor ID from the database to be displayed to frontend
* Create a delete account function for regular users
* Create a view all classes function to display all up-to-date classes from the database for the week to be displayed on the front end
* Create a filter function to filter the classes output result based on the 3 filter types:
  + Filter on instructor
  + Filter on Morning, Afternoon, Evening
  + Filter on Beginner, Intermediate, Advanced
* Create a function to return the number of booked participants for a specific class and how many places are left within it
* Create main **user interface**
* Create a new account user page
  + A form with a submit button to add new user to database
* Create a table view dashboard to see all the available classes for the week
  + Add a user dashboard button
  + Add a selected booking
    - Can accept multi checked classes
  + Add a log out button
* Create a table view for the instructor page to see their classes
  + Add a select classes checkbox
    - Add a cancel class button
  + Add a log out button
* Create an administrator dashboard
* Create a table view of the class schedule on the administrator dashboard page
  + Add an add class button
  + Add an update class button
  + Add a delete class button
  + Add a log out button

**Tasks prioritised**

|  |  |  |
| --- | --- | --- |
| **1st Set Tasks** | **2nd Set Tasks** | **3rd Set Tasks** |
| * Create **backend** system space | * Create an alert function to that is run whenever there is an update to a class or a cancellation of a class, i.e. when the update or delete functions are run within the backend | * Create a view teaching classes to return all current classes to be taught by that specific instructor based on that instructor ID from the database to be displayed to frontend |
| * Create new account classes   + One for administrator   + One for instructors   + One for regular users | * Create request cancellation function to be used by instructor accounts only, for administrator account to verify | * Create a filter function to filter the classes output result based on the 3 filter types:   + Filter on instructor   + Filter on Morning, Afternoon, Evening   + Filter on Beginner, Intermediate, Advanced |
| * Create the database tables to include a **user’s tables**, a **staff table** (for all staff including instructors within the yoga studio), and a **yoga classes table**, and a **bookings table** that contains the yoga session, participants - from the users tables - of that yoga session using the yoga session ID. | * Create verify and delete class function to be run only when an instructor requests a cancellation of a class, to the administrator user only to confirm   + Once confirmed, the delete class is then run | * Create a new account user page   + A form with a submit button to add new user to database |
| * Create functions to update and to delete user accounts   + For instructor accounts   + For regular user accounts | * Create an add booking function for regular users   + This function can accept multiple checked selection of classes to book that user into | * Create a table view dashboard to see all the available classes for the week   + Add a user dashboard button   + Add a selected booking     - Can accept multi checked classes   + Add a log out button |
| * Create logic to add, update and delete classes from database within backend, administrator use only | * Create a cancel booking function for regular users | * Create a table view for the instructor page to see their classes   + Add a select classes checkbox     - Add a cancel class button   + Add a log out button |
| * Create logic to assign, update (i.e. re-assign) instructors to specific yoga class based on its class ID within the backend | * Create a view currently booked classes to return all currently booked classes from the database to be viewed on frontend | * Create an administrator dashboard |
| * Create main **user interface** | * Create a delete account function for regular users | * Create a table view of the class schedule on the administrator dashboard page   + Add an add class button   + Add an update class button   + Add a delete class button   + Add a log out button |
|  | * Create a view all classes function to display all up-to-date classes from the database for the week to be displayed on the front end |  |
|  | * Create a function to return the number of booked participants for a specific class and how many places are left within it |  |