Question 1

SCRUM CEREMONIES

· Product backlog refinement : A process where teams/ induvials reflect on progress of their work and identify barriers and priorities that must be faced. This process can entail the adding of details and estimates into the development of a product, and during this process items are reviewed for feedback and improvements. Product backlog development aims to refocus the team and the current development of a product.

· Sprint planning: Projects in scrum are divided up into sprints, this is a measure of time for targets that must be completed to stay on track of the larger project completion. Sprint planning meetings are done in collaboration with product managers and team members where they effectively discuss what targets should be delivered in the next upcoming sprint. During this meeting, meeting members will discuss backlog items and plans on how to deliver those.

· Daily scrum: This is a daily review meeting for a team during a sprint. This meeting is lead by the Developers within the team and use this time to update peers on progress and main focuses, this reduces complexities and misunderstandings within the team. Also, the daily scrum will cover specific actions that will be taking place on the day or teh next day, this aligns the team so they are on the same page. Developers leading the daily scrum have the freedom to structure this time how they like, as long as it is inline within SCRUM guides.

· **Sprint review:** This is a type of meeting that is completed once the sprint is over, in this meeting the team come together with stakeholders for the project to review and give feedback. During this meeting discussions on what the next steps should be are confirmed and deliberated, this will be informed by the product back log where the team may be able to adjust priorities of some items and could also potentially add new ideas and add plans on how to meet targets.

· Sprint retrospective: This meeting is planned on a recurring basis at the end of each print, during this meeting the team will discuss what went well and what needs improving for future sprints. This is a core part of agile team development and project planning.

Question 2- task 1

Plan

Research tasks

- How does the yoga group already connect with customers, is there anything form this process that we can keep. (eg do they have social media platform where customers communicate through).
    - If so, find a way this can be integrated to the new booking system. Eg portal on website from social media
- Speak to clientele, gain feedback on what they find most helpful when booking. What information do they need and what other information if just helpful to know.

- The aim is to build a streamlined booking system that yoga customers are able to easily book onto. Find out how many clicks are necessary to book onto sessions.

    - How much information should be displayed to user?

        Must:

        - Calendar overview of classes – upcoming 7 days
        - Different times of classes

Extra:

- Yoga teacher leading the class (data can be stored on popular yoga teachers, feedback provided for not so popular ones)

- Do customers have accounts for booking on ? We can store this date on a SQL DB, this will help scalable projects. How often is customer data updated?
- Can we keep this data for marketing ?

Requirements

- The client wants a simple system: MUST HAVE: display of classes, and allow a user to input their choice of class.
    - Action (from planning): What other information is necessary? Yoga teacher info(specialisations), different type of yoga classes
- What is on the user interface? What should be prioritised(Trainer, location or class preference/ availability) ?
    - User interface should be clear and easy to use: User to input two things, location and date they want to use and then select trainer from user availability.
- As users will have an account- they can chose a main location for their classes (if the yoga business has more than one location). This will make the process of booking in easier
    - Extra:? DB can collect info on popular styles of yoga, business insight into producing more of these classes. Software can create reccomendations of classes to user based on patterns. Software to suggest a class if user has attended <3 of a class
- GDPR to collect customer data

**Planning and requirements are both tasks that can be run alongside each other, this should be the firs step. Following the planning we should create designs, as below, and confirm with yoga business that this meet their needs.

Design

- Software design, from front to back end:

Web based app, greeting yoga client, they will log in through their account

Display: date (7 days from current date)

customer choses date display: yoga sessions ordered by time with details of what type of yoga class is happening and name of trainer.

Info presented on User Interface from data pulled from Sql data base

Back end logic to interpret trends in classes

Back end logic to prevent double bookings and last min cancelations/ no shows

Back end logic to make recommendations based observed patterns of user for a quick booking

DB to store data on customer (contact info, birthdays, NOK for emergencies), Yoga schedule and for yoga teachers (their availability, and specialisations)

Once design is complete and confirmed then testing can begin

Testing

- Tests completed on the following:

- test correct characters are put in for emails

-test correct card details for payments

-test booking system: eg booking into available classes and not out of hours

Deployment

When tests have been ran and there are no issues this web app should be moved to deployment. This web app should be live and ready to use.

Maintenance

Depending on what is agreed between to yoga business we will decide who will own the app. We will be able to have responsibility of maintenance, and with this we can support the web app during times of scalability for customers and also more classes. This will need an increase in database storage and how well the front end will be able to talk to the back end logic.

Task 2 SQL CINEMA BOOKING SYSTEM

Requirements:

- Clear and simple booking system making it easy for customers to book into a desired film
- User interface to allow customers to select location, date, film and seats.
- Accessibility for users who may have impairments
- Be able to take payments
- Data on each cinema (directions, features and facilities, seats, parking, opening and closing times), Data on each film (Age rating, film length, actors?) These can be stored in python dictionaries or SQL database
- Store Trailers for films
- Updates on availability when a new customer buys seats

Considerations:

- Keeping customer data, such as payment details. If the user has an account how can we safely store this
- Flow of user interface, what must the user select first- this should be the cinema> date> available films
- 'Search' feature for specific film
- How often would the data need to be updated and refreshed to accurately represent number of seats for a screening of a film.
- May be useful to display when a last screening is so customers don't miss out
- DB storing all sales online/ in store

What would be the biggest problems?

- This software requires constant updating and modification, this is for seats and also for upcoming films
- Backing up data (there will be a lot of it), connecting to SQL to store this

Components/ tools used:

- create a web page/ web app with url using flask
- SQL store:
    - Data base of cinema's, screenings and screening schedule
    - Data base of sales for customers and cinema locations and payment ID's
- Python programming to process what cinema/ date/ film to present to the user