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

  The Product Backlog describe the work to be done for a product placed in the order of importance determinate by the Product Owner and based on the input of a customer. The Product Backlog Refinement is an ongoing process where the Product Owner and the Development Team collaborate, this process review and add to the product backlog details, estimates to ensure the backlog have the appropriate items, that they are prioritized, and that the items at the top of the backlog are ready for delivery.
  Some activities of product backlog:

  - Remove user stories that aren't relevant.
  - Creating new user stories when new needs are discovery.
  - Re-assessing the relative priority of stories
  - Assigning estimates to stories which have yet to receive one
  - Correcting estimates when new information is discovered.
  - Splitting user stories

- Sprint planning

  The team determines the product backlog items to work on during that sprint and discusses the initial plan for completing the items also the team determinate sprint goals. In the sprint planning is involve the Product Owner, the scrum master and the development team, it is held usually for 2 hours per week of a sprint.

Some activities:

- The scrum master is a coach that typically facilitates sprint planning
- Product Backlog Review, the product Owner prioritized and present a list of backlog items.
- The team members determine how many of the product backlog items they will be able to complete and deliver.
- Sprint Backlog Creation
- Task Identification
- Reassesses the Sprint Backlog: split user stories, etc.

· Sprint review.

The sprint review is the name given to the meeting held for the purpose of evaluating the results obtained by the Scrum team after a sprint. It is important to give transparency to the increment (the sum of all the elements of the product backlog completed during a Sprint and the value of the increments of all the previous Sprints) in front of the stakeholders and to be able to inspect the product. In this process the Development Team, Scrum Master, and Product Owner and the other Stakeholders participate and it is facilitated by the Scrum Master. The duration is of 1 hour per week. The team reviews the work completed and show demos to the stakeholders also discuss what the work of the next sprint is.

· Sprint retrospective

The sprint retrospective are the recurring meetings to discussing about workflow, continuous improvement of the process and the work and the interaction between the participants at the end of a sprint to go forward to the next one. It is important because each member of the team brings a different perspective and also to recognize each other achievement. The scrum team meets with the guidance of the Scrum master. It can be done in five phases: Opening, information gathering, brainstorming, action plan, and closure. There is also three question to answer: What went well?, What did not go well during the sprint?, What could be improved for better productivity in the next sprint?

SCRUM ROLES

- ScrumMaster

  The Scrum Master leads the Scrum team and keeps members focused on the principles of the framework. In addition, he helps Product Owners and their organizations by sharing the practices of Scrum and Agile methodologies with others in the organization. The scrum master is a person with experience in the framework and processes who has the responsibility to maximize the efficiency of the team that they become more effective and autonomous and work at the highest level. The Scrum Master is also a guide that encourages the team and solve obstacles in the sprint.

- Product Owner

  Product Owner is a standard role in Scrum teams that focuses on delivering the best product possible. It connects the Scrum team with the stakeholders and focuses on the needs of the end users, so that everyone understands what the product is trying to achieve and why. The product owner represents the business, customer or users to guide the team to develop the right product. Prioritize and maintain the backlog, create a Product Vision, participate in Sprint Meetings

- Development Team.

  It is a group of people with technical skills who jointly develop the product of the project. The Development Teams are structured and empowered by the organization so that they can decide how to organize and manage their own work. This should create synergies that allow the Development Team to improve its efficiency and effectiveness. This team is conceived under the concept of "Different people with the same purpose", they have a common objective and share the responsibility of the work they carry out, adding the quality of the project in each iteration The Development Team builds high-quality products that consists of 4 to 9 members.

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.**

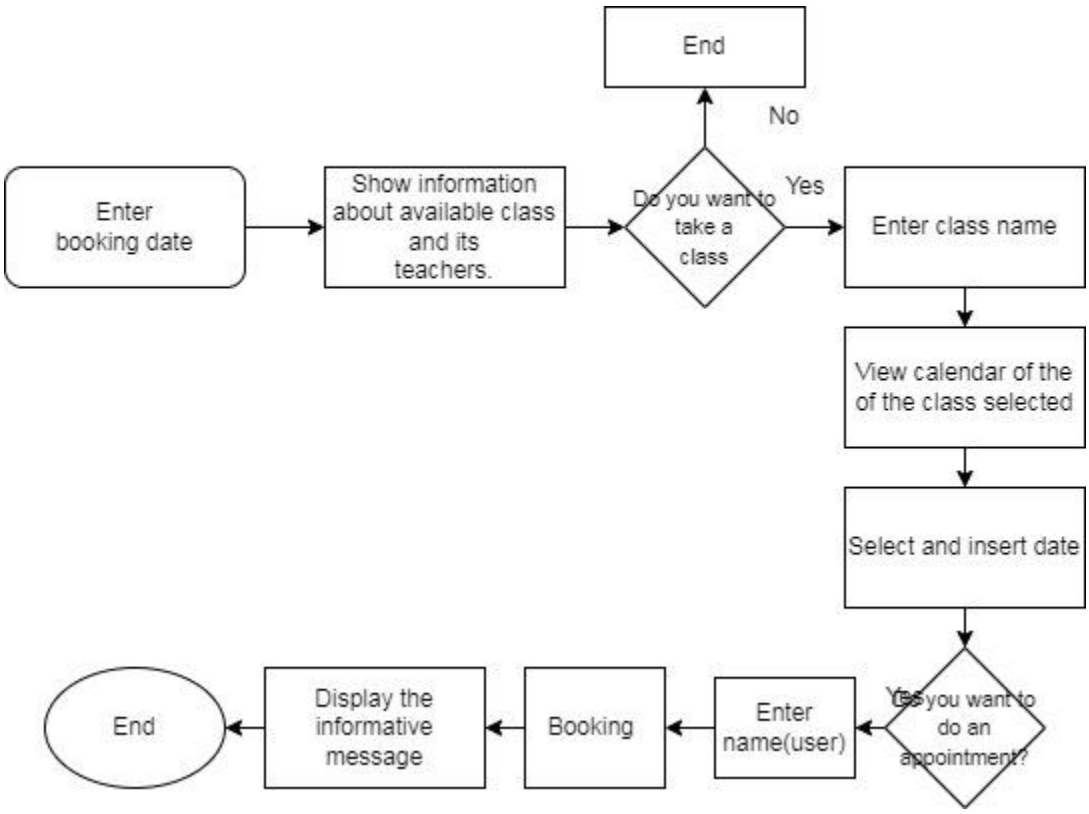
STORIES	TASKS	SUBTASKS
Story 1: As Yoga user system I want to watch all the schedules, classes and teacher available related to my booking date.	1. Design an SQL database. 2. Develop an input function to receive and store the booking date. 3. After the booking date input the system must display the yoga classes, the name and short	1. Design an SQL database with the tables and data that contain the teachers name, class information and description. 2. Store in a SQL database the information. We create a table of schedules, classes information, and teachers information.

	description of teachers close to the booking date. 4. Display a question, the user must selected one of the offered classes.	3. Ask for the user the date of the booking that he wants (crate an input). 4. Following the scheme when the user input the booking date that he/she wants it must be show the name of the available classes close to the booking date. 5. The user select a class of the interests and the information display also the name of the teachers, and their description. 6. Show (Select) the data from the SQL database. 7. Connect the database to the python system. 8. Design a python function to connect the database. 9. Design a python function to show the dates related to the input booking date. 10. 9. After of show the information, an input show a question about the class that the user want to choose.
Story 2: As yoga user app I want to see a calendar of available days of the class I selected.	1. Design and connect a calendar after select the yoga class. 2. Let the user select a date. 3. Display a confirmation message.	1. Connect the insert class to calendar of available days. 2. Create a table of available date of each class. Use SQL procedures. 3. Store the select date in SQL database. 4. 2. Show (Select) the data from the SQL database. 5. The dates must respect the booking initial date. 6. Design a python function to connect to the database. 7. Design a python function to show the data in SQL database related to the input booking date. 8. 6. Show the dates and create an input to store the selected input.
Story 3: As yoga user app I want to my appointments be	1. After display the question of confirmation, the user select an option.	1. Create a table of transactions. 2. Create an input with to store the name of the user to do the booking.

registered and indicates me that with a message.	2. Store the user name, user selection of yoga class, teacher and date. 3. Display a confirmation message to confirm the operation and also a message with all the information about the yoga class, the teacher and booking date.	3. Create a function in python to store the date, the name of the user, and the class of the booking in the SQL database. 4. In the function there must be a limit of students and must be reduced when the user make the appointment. 5. Create a SQL procedure 6. Ask to the user if is sure for the appointment. If the answer is yes continue. 7. The booking information must be stored in the SQL dataset. 8. When the transaction is finished and successful the system print a message that says that the booking is done and also prints the class, the date of appointment and teacher. 9. Update the data base.
Story 4: As user I want the yoga booking system don't mistake the date and information.	1. Validate all the input answers. 2. Store all the user booking data.	1. Verify the initial input that is the booking date, the class input. 2. The python code must have the exception code to avoid errors. 3. Validation of data before execute the functions in python.
Story 5: As yoga system I want the system indicates me if I am inserting correct information	1. Verify if the answer to the questions to input is right 2. Display messages if the answers are wrong.	1. The functions code has the exception parameters. 2. In the function insert messages when the option in not right. 3. The input of the function must describe the answer to insert. 4. Create Triggers in SQL to store the data correctly.

What has the system?

Workflow of the system:



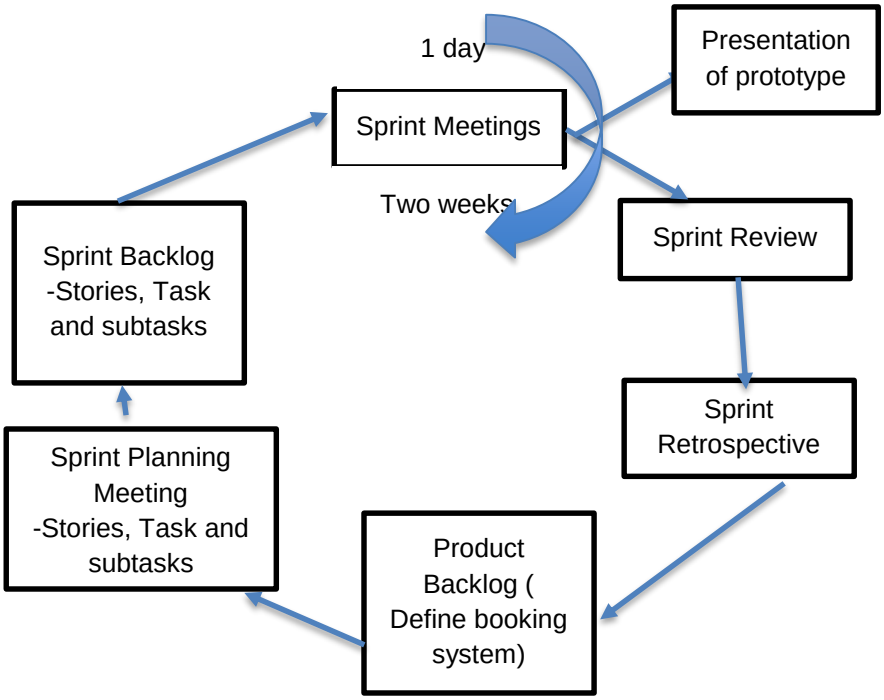
· Assume that one person works on one task.

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

WEEK	TEAM MEMBER 1	TEAM MEMBER 2	TEAM MEMBER 3	TEAM MEMBER 4	TEAM MEMBER 5
	STORY 1	**STORY 2**	**STORY 3**	**STORY 4**	**STORY 5**
WEEK1	TASK 1: Design an SQL database.			TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answers are wrong

AND **WEEK** **2**	TASK 2: Develop an input function to receive and store the booking date.			TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answers are wrong
	TASK 3: After the booking date input the system must display the yoga classes, the name and short description of teachers close to the booking date.	TASK 1: Design and connect a calendar after select the yoga class.		TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answers are wrong
	TASK 4: Display a question, the user must selected one of the offered classes.	TASK 2: Let the user select a date.		TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answersare wrong
		TASK 3: Display a confirmation message.	TASK 1: After display the question of confirmation, the user select an option.	TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if

					the answer are wrong
			TASK 2: Store the user selection of yoga class, teacher and date.	TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answers are wrong
			TASK 3: Display a confirmation message to confirm the operation and also a message with all the information about the yoga class, the teacher and booking date	TASK 1: Validate all the input answers. TASK 2: Store all the user booking data.	TASK 1: Verify if the answer to the questions to input is right TASK 2: Display messages if the answers are wrong
WEEK 2	Test code	Test code	Test code	Test code	Test code
	Test prototype				
	Fix debugs and error	Fix debugs and error	Fix debugs and error	Fix debugs and error	Fix debugs and error
	Test prototype				
	Presentation the prototype				



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 <u>do not</u> need to write actual code, but describe the high-level approach:

- **Draw a list of key requirements**
  - The user enter to the booking system, the system must recognize if the user is an existent client or a new client. The system also have to ask the user if he want to register in the system or continue as guest user.
  - The user can go back to the last step until the payment confirmation step.
  - The system must show the cities and the cinemas.
  - The system show the movies of each cinema also the trailer, the Synopsis and

information of the movie(director, actors, duration, country, age limit).

- The user can select a movie and when the user does this the shows available must be shown.
- The user select the hour of the movie and then the system took the user to the seating selection.
- The seating are shown in an excellent graphic way and the user select them. The user can select many seats.
- The system show the available and non-available seats.
- The system ask for the user for food like popcorn, sausages, ice creams, soda, etc.
- If the user accept continue to the food selection, then go to the confirmation step.
- The system waits a time (5 min) for the users. The users can continue to the next operation if they want.
- When the user confirm the operation automatically the seats changes their status to not available and must be shown in this way for the future operations.
- If the user is a client offer promotions or gifts if it isn't the system continue with the next operation.
- The system took the user to the payment step. The system calculate the correct amount of the tickets.
- The payment has to be correct for every seat.
- The system have to be secure and protect the credit or debit card for the booking.
- The user confirm the payment process after this step it is impossible return to the last step.
- After the payment the tickets are generated. The system can send the tickets to an email or download it.
- Display the thanks message when all the process is finished.

- **What are your main considerations?**

  - The database model
  - Code functions
  - The security of the user

- The complexity of the system
- User account management
- The graphic, images, video to be display.
- The correct login.
- The seating available must be correct.
- Time limits to confirm the transaction.
- Store and connect the user with the answer.
- Store the data of the client
- The interface
- The payment system security.
- The system must be available the 24 hours
- Accessibility (for all users) and compatibility (telephone, internet, laptop, tablet)

- **What would be your common or biggest problems?**

  The transaction involves money, also the system has to be very understandable to the user to avoid complication in the booking process, the design has to be attractive, the store of information, the generation of the tickets, the system freeze, the sequence of steps.

- **What components or tools would you potentially use?**

  -Data cloud storage.

  -System payment validation and options (visa, credit card, debit card)

  -Python language, SQL, other languages.

  -Real-time booking

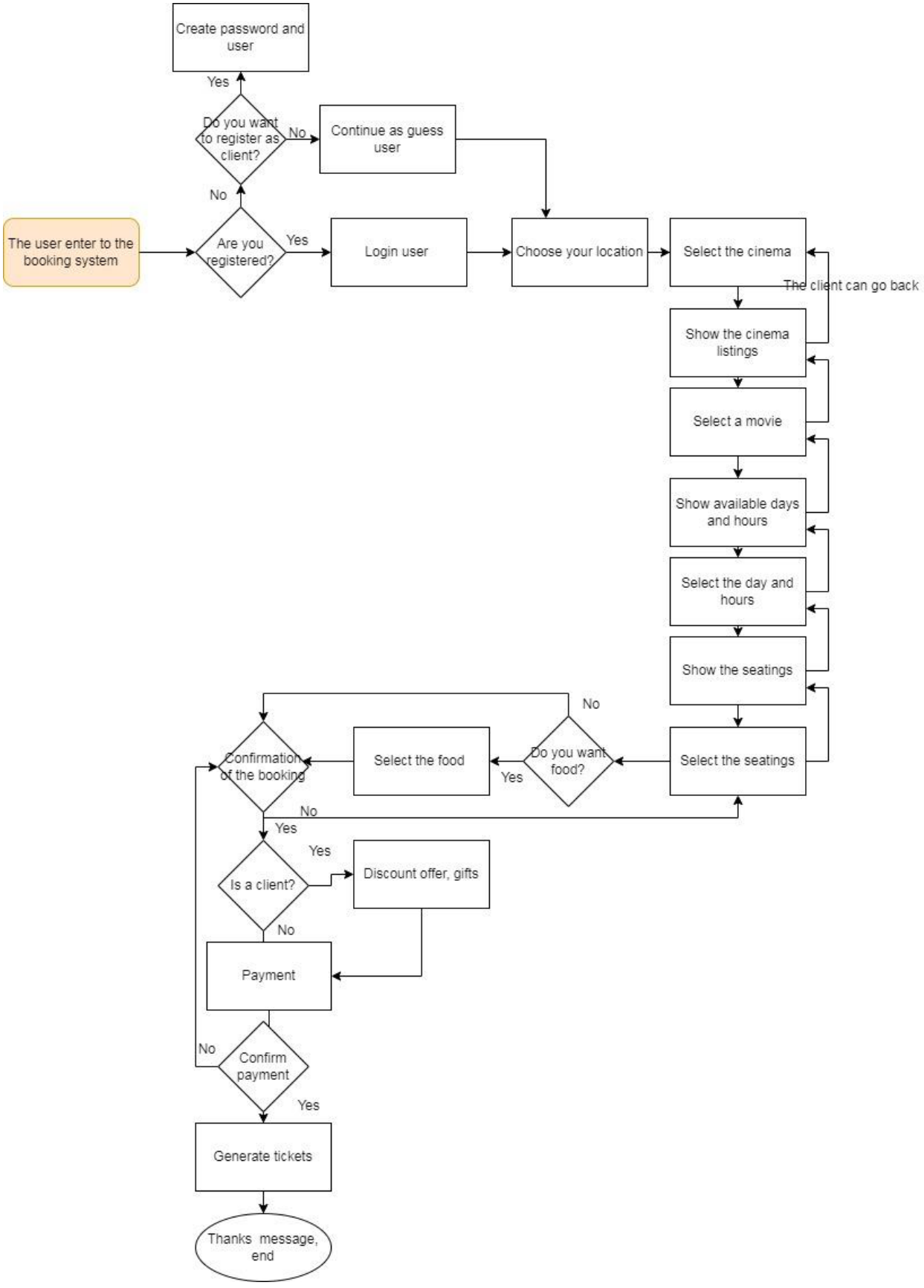
  -Design of the front and backend.

  -Voucher generation.

  -Integrations with an email

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

The workflow is shown below:


Create password and user
Yes
Do you want to register as client?
No
Continue as guess user
No
The user enter to the booking system
Are you registered?
Yes
Login user
Choose your location
Select the cinema
The client can go back
Show the cinema listings
Select a movie
Show available days and hours
Select the day and hours
Show the seatings
No
Select the seatings
Do you want food?
Yes
Select the food
Confirmation of the booking
No
Yes
Yes
Is a client?
Yes
Discount offer, gifts
No
Payment
No
Confirm payment
Yes
Generate tickets
Thanks message, end