

## План роботи над командним проєктом

1. Студенти розподіляються за командами клієнт-менеджером.
2. Кожна команда отримує окремий канал у Slack.
3. Команді потрібно виконати завдання:
  - обрати назву команди;
  - обрати Team Lead;
  - обрати Scrum Master;
  - переглянути матеріали до проєкту.
4. Відвідати вступний вебінар на тему "Старт проєкту. Планування з ментором".
5. Scrum Mastery потрібно заповнити дошку Trello та запросити до неї всіх учасників команди.
6. Команді потрібно розподілити завдання між усіма учасниками проєкту.
7. Кожен працює над своїм завданням та своєчасно актуалізує інформацію про завдання в ході його виконання.
8. З технічними питаннями команда звертається до Team Lead'a.
9. Team Lead спілкується з ментором, якщо є питання, які не змогли вирішити командно.
10. Команда проводить щоденний Stand Up тривалістю 15-20 хвилин.
11. Кожен тестувальник на Stand Up відповідає на три запитання: Що він зробив вчора (сьогодні)? Що робитиме сьогодні (завтра)? У чому потрібна допомога?  
**!!!Важливо!!!** на Stand Up ми не починаємо розбирати ті моменти, де потрібна допомога, ми говоримо про них, щоб вся команда розуміла, як проходить процес виконання завдання. Після проведення Stand Up тестувальник, якому потрібна допомога, може залишитися і розібрати свої питання з членом команди, який знає відповідь або з Team Lead.

### **Проміжний дедлайн.**

У вказаний дедлайн, Team Lead надсилає посилання на готовий список баг-репортів та готові чек-листи від усієї команди проєктному ментору для перевірки. Після перевірки, ментор надсилає фідбек Team Leadу, після чого, за необхідності, кожен учасник проєкту обирає собі завдання для виправлення.

### **Презентація проєкту.**

1. Scrum Master готує слайди презентації проєкту згідно [шаблону](#). Тривалість презентації не повинна перевищувати 5 хвилин.
2. Проводиться репетиція. Команду представляє Team Lead або Scrum Master команди.

Студенти повинні структурувати свій виступ і показати/розповісти наступне:

- завдання, яке стояло перед командою;
- що було зроблено для вирішення завдання?
- які складнощі були? як їх вирішила команда? (сам процес, логіка мислення спеціалістів та швидкодія)

Після завершення всіх блоків представник команди (Team Lead, Scrum Master) завершує презентацію:

- які висновки зробили після командної роботи?
- що команда робитиме далі: проведе ретроспективу / візьме новий проєкт / спробує працювати в такій же команді над особистим проєктом. Подяка команді за виконаний проєкт та завершення презентації.

3. Проводиться презентація проєктів.
4. Презентація проєкту проводиться за участю клієнт-менеджера, проєктного ментора, всіх учасників команд та запрошених гостей.

### **Ролі**

**Team lead.** Стежить за процесом підготовки та виконання тестів, відповідає за якість чек-листів та баг-репортів, пише завдання щодо проєкту для учасників команди. Є власником файлів з чек-листами та баг-репортами від цілої командним. Безпосередньо спілкується з проєктним ментором з технічних питань, що виникають у команди.

**Scrum master.** Здійснює щоденний Stand Up для планування та звітності роботи команди. Створює та заповнює дошку Trello команди. Для проведення Stand Up можна

використати Google Meet, Zoom, Skype, Discord. Слідкує за наявністю завдань у всіх учасників команди та часом їх реалізації. Готує презентацію для захисту.

**Проектний ментор.** Є наставником під час роботи над проектом. Завданням проектного ментора є консультування з технічних питань щодо проекту, вибору способів проведення тестування. Також ментор проводить оцінку баг-репортів та надає зворотний зв'язок.

## ТЕСН-завдання проекту QA

1. Провести аналіз технічного завдання: дослідити технічне завдання та зрозуміти його вимоги до функціональності та якості програми. Важливо знати, що повинен робити програмний продукт, які функції він повинен мати та які вимоги повинні бути виконані.

2. Дослідити макет застосунку для розуміння того, як програма має виглядати та які функції повинні бути доступні для користувачів.

3. Провести аналіз Swagger документації: дослідити Swagger документацію для розуміння того, які функції повинні бути доступні та як вони повинні працювати.

4. Написати чек-листи: створення переліку тестів, які описують певні сценарії взаємодії з програмою та перевіряють її функціональність. Тести повинні бути зрозумілі, легко виконуватися і містити відповідні очікувані результати.

Тестування на негативні сценарії: Створіть тест-кейси для тестування негативних сценаріїв, таких як помилкові вхідні дані або помилкові взаємодії з користувачем.

5. Виконання тестів: виконання тестів для перевірки функціональності програми та виявлення будь-яких помилок та недоліків - (для трьох девайсів: мобілка, планшет, десктоп)

6. Запис результатів тестування: відстеження результатів виконання кейсів та запис виявлених помилок і проблем, включаючи їх опис, кроки відтворення та інші деталі - створення баг-репортів.

7. Підготовка звіту про тестування: складання звіту про виконані перевірки, знайдені помилки та рекомендації щодо вдосконалення програми.

8. Фінальна оцінка та підготовка презентації: оцінка якості програми та її готовності до випуску.

**Важливо відстежувати результати кожного кроку та вести звіти про кожен тест-кейс, щоб забезпечити якість тестування та виявлення всіх проблем програми.**