**Програмна інженерія 2022-2023**

**Проект**

Загальні правила:

1. Всі завдання повинні бути виконані до вказаного терміну і у повному обсязі. Тоді вони можуть бути оцінені до 10б, в іншому випадку - не більше ніж 5 б
2. Для кожного завдання кожен учасник команди повинен створити принаймні один специфічний (конкретний) таск на дошці гітхабу і після завершення завдання перемістити в колонку done
3. Кожного тижня проводяться статус мітинги, результати яких (відповіді на питання: що робив? що планую робити? і які проблеми?) записуються у файл зі статусами
4. Кожен учасник команди повинен здійснювати регулярні коміти на гітхаб згідно зі своїми завданнями

|  |  |  |
| --- | --- | --- |
| Завдання | Термін виконання | Лінки |
| Завдання 1. Команда та початкові налаштування | | |
| 1. Створити команди 3-4 студенти в кожній 2. Створити відкритий GitHub репозиторій з назвою проекту та учасниками - лінк додати в табличку 3. Створити дошку на гітхабі, для трекання тасок з стовпчиками Planned, InProgress, ReadyForReview, Done. Кожен студент повинен мати таску на кожне завдання, яку він буде представляти на занятті 4. Обрати проект – десктопна аплікація, технології .Net: ADO.Net, Entity FW, WPF’ 5. Сформувати ціль аплікації, які проблеми вона вирішує, типи користувачів аплікації, основні вимоги до аплікації 6. Провести аналіз подібних аплікацій, які є на ринку, їх переваги і недоліки та їх основний функціонал (2-3 аплікації) 7. Оформи всі ці 3-5 пункти в презентацію і представити її на занятті. Залити презентацію на гітхаб в папку Документи | 12 вересня |  |
| Завдання 2: Збір і оформлення вимог, юзкейс діаграми, графічний дизайн програми | | |
| 1. Описати основні функціональні та нефункціональні вимоги до проекту.  Функціональні вимоги включають опис вікон програми, що на них буде розміщено і яку вони будуть мати функціональність, переходи між вікнами, а також треба описати типових юзерів аплікації, по ролях, якщо будуть.  Нефункціональні вимоги - це загальні характеристики програми: мова інтерфейсу, вимоги до ПЗ для запуску програми, яка база даних, вимоги по секюриті 2. По визначених для вашого проекту ролях намалювати usecase діаграми, що може робити кожен тип юзера з вашою програмою 3. Використовуючи один з інструментів (типу balsamiq.com ) для кожноі сторінки аплікації створити wireframes - схематична розмітка вікон аплікації 4. Оформити це все у вигляді документу, який потрібно залити на гітхаб | 23 вересня |  |
| Завдання 3. Розробка архітектури проекту та DB | | |
| 1. Визначити які дані будуть зберігатися в базі даних, розробити схему бази даних з таблицями у вигляді відповідної UML діаграми. Для кожної таблиці вказати дані, які там будуть зберігатися, типи, а також залежності між таблицями. Вибрати провайдера баз даних. Створити базу даних, використовуючи написаний скрипт 2. Створити Console App (.NET Core)  проект, Використовуючи ADO.Net підключити до проекту вашу базу даних. На консоль вивести дані з табличок. Створити метод в цій консольній апп, для заповнення бази даних тестовими рандомними даними через ADO.Net по 30-50 записів в кожній табличці 3. Створити скрипт для генерації бази даних. Створити ER діаграму | 30 вересня |  |
| Завдання 4. | | |
| 1. WPF. Трьохрівнева архітектура проекту   У солюшині вашого проекту додати папки-проекти, для трьохрівневої архітектури: DAL (доступ до бази даних через Entity Framework Core)<-> BLL (рівень бізнес логіки, де буде основна функціональність вашого проекту+Unit tests)<->Presentation(UI проекту). Рознести логіку вашої програми у ці рівні   1. Використовуючи WPF розробити графічний інтерфейс вікон програми. Повинно бути меню, контекстне меню, валідація введення даних. Presentation рівень 2. Підключення до бази даних через Entity Framework - DAL рівень | 7 жовтня | 1)https://docs.microsoft.com/uk-ua/ef/core/get-started/index?tabs=visual-studio |
| Завдання 5. StyleCop, Юніт тести | | |
| 1. Доставити до студії StyleCop і прогнати ваш проект через код аналіз цього інструмента. Створити звіт - скільки і яких зауважень зробив StyleCop, також враховуючи і бест практики Clean code, виправляєте всі ці помилки, Звіт по отриманому чистому коду 2. Реалізувати всі юзкейс з вашої діаграми у двох напрямках: дані вводяться юзером, обробляються і записуються в базу і навпаки юзер шукає щось, запит обробляється і повертається на форму 3. Покрити Юніт тестами проект.(в основному юніт тести можна писати до рівня бізнес логіки) | 14 жовтня |  |
| Завдання 6. Логування подій. Анімація |  |  |
| 1. Додати можливість логування подій, зауважень та помилок у файлову систему (logger) | 21 жовтня |  |
| 1. На одному з вікон повинна бути реалізована анімація |  |  |
| Захист проектів | 28 жовтня |  |
|  |  |  |
|  |  |  |
|  |  |  |