|  |  |  |  |
| --- | --- | --- | --- |
| МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені Тараса Шевченка ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  **Кафедра програмних систем і технологій**  Дисципліна  **«**Моделювання інформаційних систем та бізнес систем**»**  **Лабораторна робота № 4**  «Проектування рівня взаємодії людина–комп’ютер» | | | |
| **Виконав:** | Гоша Давід | **Перевірив**: |  |
| Група | ІПЗ-43 | Дата перевірки |  |
| Форма навчання | денна | Оцінка |  |
| Спеціальність | 121 |
| 2022 | | | |

**Завдання:**

1. Використовуючи communication diagrams та матрицю CRUDE, створіть діаграму пакетів рівня проблемної області (problem domain layer);
2. Виходячи з analysis models, які були створені, і вашого поточного розуміння інформаційної системи, яку стратегію проєктування (design strategy) ви б порекомендували? Чому?;
3. Виберіть один із класів і створіть набір інваріантів для атрибутів і зв’язків, додайте їх до картки CRC для класу;
4. Виберіть один із методів у вибраному класі та створіть для нього контракт. Використовуйте Object Constraint Language, щоб вказати будь-яку pre- або postcondition;
5. Розробіть два користувацьких сценарії (use case scenarios). Сценарії обирайте ті, для яких було створено комунікаційні діаграми;
6. Намалюйте діаграму навігації вікон (windows navigation diagram, WND). WND використовується, щоб показати, як пов’язані всі екрани, форми та звіти, які використовує система, і як користувач переходить від одного до іншого. WND базується на behavioral state machine;
7. Спроектуйте розкадровку (storyboard).

**Варіант “Віртуальний Тур Факультетом”**

Інформаційна система "Віртуальний Тур Факультетом" призначена для організації та проведення онлайн-екскурсій по факультету для майбутніх студентів та їхніх батьків. Система дозволяє абітурієнтам краще пізнати факультет, вивчити його інфраструктуру та отримати відповіді на свої питання.

Основні функції системи:

* Проведення віртуальних турів по факультету.
* Надання інформації про інфраструктуру факультету.
* Організація онлайн-консультацій з викладачами та студентами факультету.
* Можливість задавати питання та отримувати на них відповіді.
* Реєстрація та вхід в систему для збереження індивідуального прогресу.

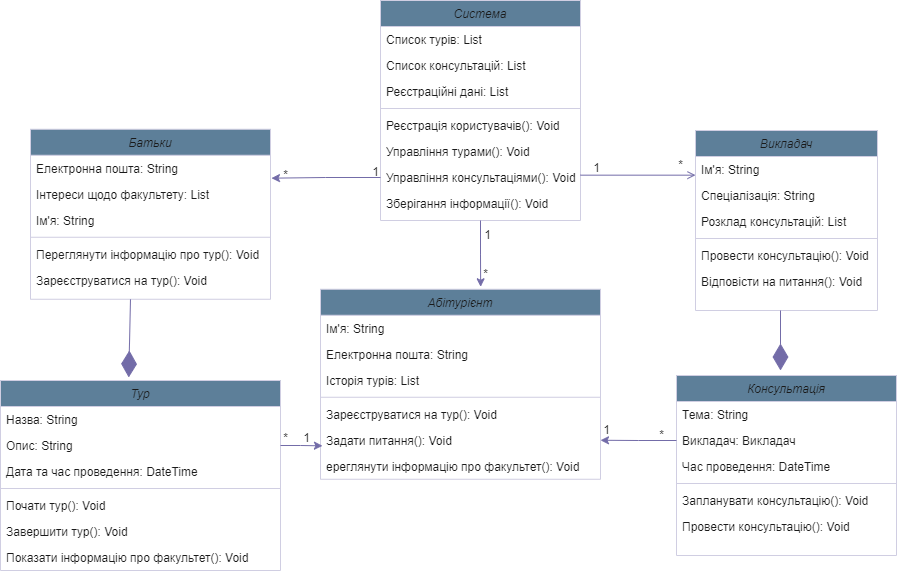
Актори системи:

* Абітурієнт - основний користувач системи, який може взаємодіяти з усіма її функціями.
* Батьки абітурієнта - користувачі, які можуть вивчати інфраструктуру факультету та брати участь у віртуальних турах.
* Система - забезпечує функціонування всіх сервісів та взаємодію з користувачами.
* Викладач - бере участь в онлайн-консультаціях, відповідає на питання користувачів.

Система "Віртуальний Тур Факультетом" дозволяє потенційним студентам та їхнім батькам отримати максимально повну інформацію про факультет, його програми та можливості, не виходячи з дому. Це сприяє підвищенню інформованості та залученню нових студентів.

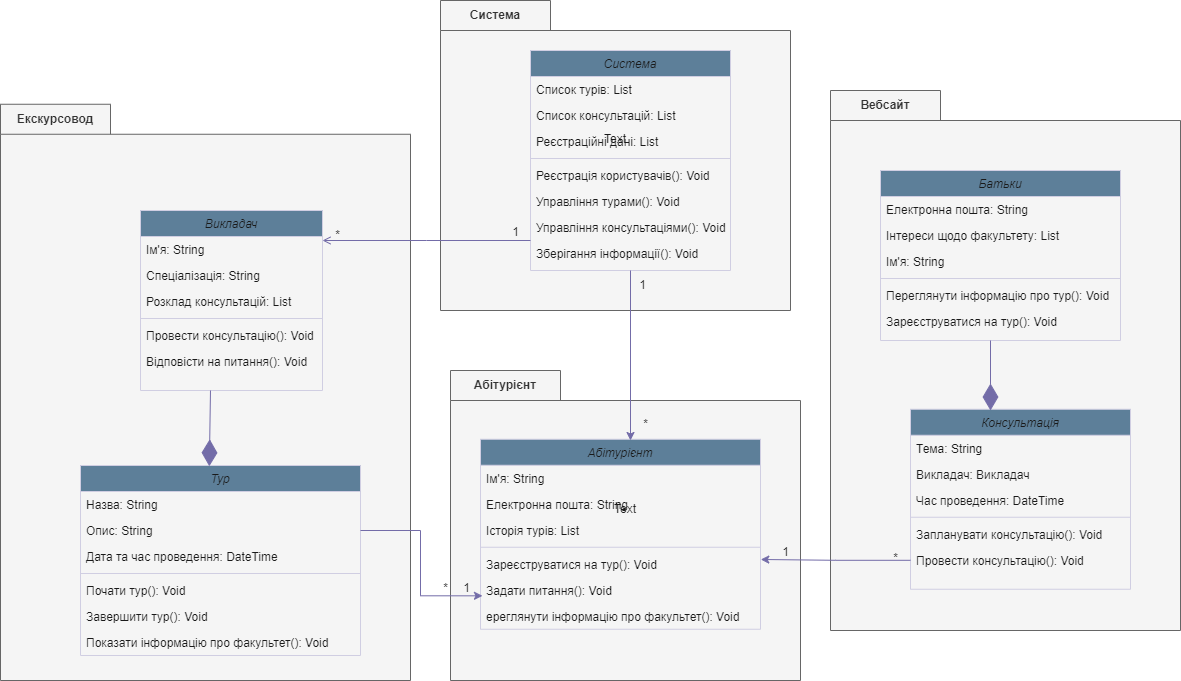
**Хід Роботи:**

**Діаграма класів:**

****

Перед тим як створити діаграму, давайте визначимо основні компоненти системи, які будуть представлені на діаграмі:

1. Пакет "Користувачі": Включає класи або компоненти, які представляють абітурієнтів, батьків абітурієнтів та викладачів.
2. Пакет "Тури": Включає компоненти для організації та проведення віртуальних турів.
3. Пакет "Консультації": Містить компоненти для організації онлайн-консультацій з викладачами.
4. Пакет "Інфраструктура Факультету": Включає інформацію про інфраструктуру факультету.
5. Пакет "Авторизація": Відповідає за реєстрацію та вхід в систему.
6. Пакет "Запитання та Відповіді": Для задавання питань та отримання відповідей.



Засновуючи свій вибір на аналізі моделей та розумінні потреб інформаційної системи, я **вирішив вибрати стратегію кастомної розробки**. Ось мої основні міркування:

Унікальність Бізнес-Потреб

Моя система вимагає специфічних функціональностей, які не можуть бути повністю задоволені стандартними рішеннями на ринку. Кастомна розробка дозволить створити продукт, який точно відповідає цим унікальним вимогам, забезпечуючи необхідну гнучкість та масштабованість для майбутнього розвитку.

Внутрішні Ресурси та Навички

У моїй організації є достатньо функціонального та технічного досвіду для розробки і підтримки кастомної системи. Це знижує залежність від зовнішніх вендорів та сприяє розвитку внутрішніх компетенцій. Крім того, це надає можливість глибшого розуміння власних бізнес-процесів.

Гнучкість у Плануванні

Мій проект може дозволити собі гнучкість у плануванні часу. Це важливо, оскільки кастомна розробка часто вимагає більше часу на реалізацію, але такий підхід забезпечує кращу якість та точність відповідності продукту бізнес-вимогам.

Контроль над Проектом

Вибір кастомної розробки дає мені повний контроль над процесом розробки та кінцевим продуктом. Це дозволяє вносити зміни та адаптації в будь-який час, що є критично важливим для динамічного бізнес-середовища.

Враховуючи ці фактори, кастомна розробка є найбільш підходящою стратегією для моєї інформаційної системи. Вона дозволить створити унікальне рішення, яке точно відповідає потребам мого бізнесу та забезпечує необхідну гнучкість для майбутнього розвитку.

Для створення контракту методу в класі 'Абітурієнт' із використанням Object Constraint Language (OCL), спочатку виберемо метод, а потім визначимо для нього pre- та postconditions.

Припустимо, ми вибираємо метод registerForTour, який дозволяє абітурієнту реєструватися на віртуальний тур. Ось приклад контракту для цього методу:

**Метод: registerForTour(tourID: String)**

Precondition:

* Абітурієнт повинен бути зареєстрований у системі.
  + self.registrationStatus = 'registered'
* ID туру, на який абітурієнт хоче зареєструватися, повинен існувати.
  + Tour.allInstances()->exists(t | t.tourID = tourID)

Postcondition:

* Абітурієнт повинен бути доданий до списку учасників обраного туру.
  + Tour.allInstances()->any(t | t.tourID = tourID).participants->includes(self)
* Статус реєстрації на тур повинен бути оновлений.
  + self.registeredTours->includes(Tour.allInstances()->any(t | t.tourID = tourID))

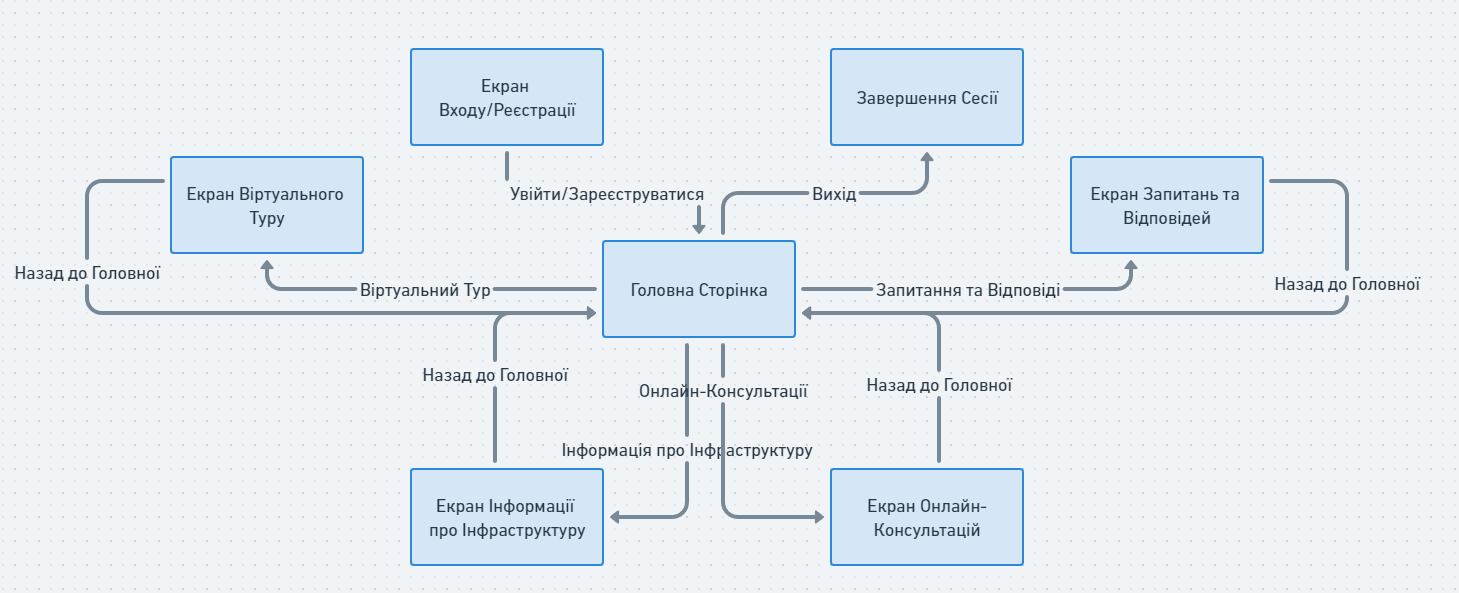
Цей контракт використовує OCL для визначення умов, які повинні бути виконані перед викликом методу registerForTour (preconditions), і стану, який повинен бути досягнутий після виконання методу (postconditions).

Для розробки користувацьких сценаріїв **(use case scenarios)** для інформаційної системи "Віртуальний Тур Факультетом", розглянемо два ключові процеси: "Реєстрація абітурієнта в системі" та "Участь в онлайн-екскурсії".

|  |
| --- |
| **Реєстрація абітурієнта в системі** |
| 1. Відкриття веб-сайту: Абітурієнт відкриває веб-сайт факультету і знаходить розділ для реєстрації. 2. Вибір опції реєстрації: Абітурієнт натискає на кнопку "Реєстрація" для створення нового акаунту. 3. Введення даних: Абітурієнт заповнює форму реєстрації, включаючи особисті дані (ім'я, прізвище, електронна пошта, тощо). 4. Підтвердження реєстрації: Система відправляє на електронну пошту абітурієнта лист з посиланням для підтвердження реєстрації. 5. Активація акаунту: Абітурієнт переходить за посиланням у листі, тим самим активуючи свій акаунт. 6. Вхід в систему: Абітурієнт входить в систему, використовуючи свої реєстраційні дані.   Альтернативні сценарії:   * Якщо абітурієнт вже зареєстрований, він може відразу перейти до входу в систему. * Якщо абітурієнт не отримав листа з підтвердженням, він може запросити повторну відправку. |

|  |
| --- |
| **Участь в онлайн-екскурсії** |
| 1. Вхід в систему: Абітурієнт входить в систему, використовуючи свої реєстраційні дані. 2. Перегляд розкладу екскурсій: Абітурієнт переглядає доступні онлайн-екскурсії та їх розклад. 3. Реєстрація на екскурсію: Абітурієнт обирає підходящу екскурсію та реєструється на неї. 4. Отримання посилання на екскурсію: Система надсилає абітурієнту посилання на онлайн-екскурсію. 5. Участь в екскурсії: Абітурієнт приєднується до онлайн-екскурсії в зазначений час. 6. Задання питань: Абітурієнт має можливість задавати питання під час екскурсії.   Альтернативні сценарії:   * Якщо абітурієнт не може приєднатися до екскурсії в зазначений час, він може переглянути запис пізніше. * Якщо виникають технічні проблеми під час екскурсії, абітурієнт може звернутися до технічної підтримки. |

Ці сценарії відображають ключові процеси взаємодії абітурієнтів з інформаційною системою "Віртуальний Тур Факультетом" та допомагають зрозуміти, як користувачі можуть використовувати систему для досягнення своїх цілей.



**Специфікація Діаграми Навігації Вікон "Віртуальний Тур Факультетом"**

1. Екран Входу/Реєстрації
2. Функція: Початковий екран для входу або реєстрації користувачів.
3. Переходи:
   1. Після успішного входу або реєстрації переходить на Головну Сторінку.
4. Головна Сторінка
5. Функція: Центральний вузол навігації, звідки доступні всі основні функції системи.
6. Переходи:
   1. До Екрану Віртуального Туру.
   2. До Екрану Інформації про Інфраструктуру.
   3. До Екрану Онлайн-Консультацій.
   4. До Екрану Запитань та Відповідей.
   5. Опція "Вихід" для завершення сесії.
7. Екран Віртуального Туру
8. Функція: Перегляд віртуальних турів по факультету.
9. Переходи:
   1. Назад до Головної Сторінки.
10. Екран Інформації про Інфраструктуру
11. Функція: Надання детальної інформації про факультет.
12. Переходи:
    1. Назад до Головної Сторінки.
13. Екран Онлайн-Консультацій
14. Функція: Взаємодія з викладачами та студентами, можливість задавати питання.
15. Переходи:
    1. Назад до Головної Сторінки.
16. Екран Запитань та Відповідей
17. Функція: Можливість задавати питання та переглядати відповіді.
18. Переходи:
    1. Назад до Головної Сторінки.

Кожен екран у специфікації має чітко визначену функцію та перелік можливих переходів до інших екранів. Це дозволяє розробникам та дизайнерам системи зрозуміти логіку навігації та взаємозв'язки між різними частинами інтерфейсу.

**Висновки:**

Під час розробки інформаційної системи "Віртуальний Тур Факультетом", я зіткнувся з рядом викликів та відкриттів. Однією з ключових проблем було точне визначення взаємодій між різними компонентами системи. Це вимагало не тільки глибокого розуміння функціональних аспектів системи, але й уваги до технічних деталей, особливо при створенні діаграми пакетів рівня проблемної області.

На відміну від цього, визначення основних акторів та їх ролей у системі було відносно простішим завданням. Це було безпосередньо пов'язано з основною метою системи та дозволило мені зосередитися на ключових потребах користувачів. Проте, створення контрактів для методів класів з використанням Object Constraint Language (OCL) виявилося значно складнішим. Це вимагало детального аналізу кожного методу та його впливу на стан системи, що було непростою задачею.

Процес моделювання був ітераційним, і кожна ітерація дозволяла мені уточнити та вдосконалити модель. Загалом, я провів близько 5-7 ітерацій для кожної діаграми, що включало перегляд взаємодій між об'єктами та коригування діаграм для кращого відображення структури та функціональності системи.

Цей досвід підкреслив важливість чіткого визначення вимог та взаємодій у складних інформаційних системах. Він також показав значення ітераційного підходу в процесі розробки, де кожна ітерація є можливістю для уточнення та вдосконалення. Це був навчальний досвід, який дозволив мені краще зрозуміти складність та важливість процесу моделювання в розробці інформаційних систем.