

Національний технічний університет України
«Київський політехнічний інститут ім. Сікорського»

Кафедра обчислювальної техніки

Проект «Система управління проектами»
з дисципліни « Організація баз даних»

Виконали:
студенти 2 курсу
ФІОТ гр. ІО-82

Бригада:
Білич М.В
Галацин З.Б
Кузенний П.В
Роман О.Т
Самутін О.І
Шевчук О.Б
Ющенко А.В

Київ
2019

Зміст

1. Вступ. Система Управління Проектами	3
1.1 Мета управління проектами.....	3
1.2 Групи оцінок успішності проекту	3
1.3 Ролі в проекті.....	4
1.4 Методології управління проектами	4
1.4.1 Традиційна (Каскадна) методологія управління проектами.....	4
1.4.2 Методологія управління проектами PRINCE2	5
1.4.3 Гнучка методологія управління проектом (Agile Project Management)	7
1.4.4 Методологія швидкої розробки додатків (Rapid Application Development - RAD)	8
1.5 Порівняння програмних засобів.	8
2. Короткий огляд продукту	9
2.1. Основна інформація	9
2.2. Призначення системи управління	10
2.3. Політика взаємодії з користувачем	11
2.4. Характеристика ділового процесу	11
2.5. Призначення системи	11
2.6. Надійність	11
2.7. Функціональність.....	11
3. Ділові правила та прописи.....	12
3.1 Діаграма юзкейсів	12
3.7 Діаграми послідовностей	19
4. Висновок	26
1. Список використаної літератури.....	26

1. Вступ. Система Управління Проектами

Управління проектами - область діяльності, в ході якої визначаються та досягаються чіткі цілі проекту при балансуванні між обсягом робіт, ресурсами (такими як гроші, працю, матеріали, енергія, простір і іншими), часом, якістю та ризиками.

1.1 Мета управління проектами

«Метою управління проектом (ами) є досягнення заздалегідь визначених цілей при заздалегідь відомих обмеженнях і доцільному використанні можливостей, реагуванні на ризики.»

Навіть при досягненні поставлених цілей і доцільності змін, проект може не відповідати очікуванням зацікавлених сторін. У проектах з високим рівнем змін потрібно керувати очікуваннями.

1.2 Групи оцінок успішності проекту

Успішність проекту різним чином оцінюється в різних методиках. Успішність може порізному оцінюватися різними учасниками проекту.

Групи оцінок успішності:

- Орієнтовані на контракт з жорсткою фіксацією вимог і мінімізацією змін в ході проекту, наприклад традиційні методології, в тому числі PMBOK: «Проект успішний, якщо виконаний відповідно до затверджених критеріїв: обсягу, терміну, якості». Тобто проект успішний, якщо виконаний і закритий договір між Замовником та Виконавцем (незалежно від того, чи був він юридичним документом в разі зовнішніх проектів або визначався як-то інакше в разі внутрішніх проектів). При цьому оцінка успішності єдина як для замовника так і для виконавця.
- Орієнтовані на задоволеність замовника з гнучким управлінням вимогами, наприклад гнучкі методології SCRUM: «Проект успішний, якщо замовник задоволений»
- Орієнтовані на тривалий взаємодія з Замовником: управління програмами, Спрямоване на тривалий взаємодія, а не на один проект / контракт. Тут робиться акцент на продовження співпраці Виконавця з Замовником в рамках наступних проектів і іншої взаємодії.
- Збалансовані, наприклад PRINCE2: «Проект успішний при збалансованості принаймні за трьома категоріями - бізнесу, орієнтації на користувача і технологічної зрілості». Тут робиться акцент на фінансовій успішності проекту, задоволеності користувачів і розвитку технологій. Оцінка успішності може відрізнятися з точки зору бізнесу,

користувача і виконавця. Такі методики оцінки частіше використовуються для внутрішніх проектів, коли замовник і виконавець знаходяться в одній організації.

Так, наприклад, проект, укладіть в узгоджені терміни і витрати, але не окупить за результатами проекту (витрати великі, результат неактуальний наприкінці проекту, замовник не може скористатися результатом і т. П.) Буде успішним за традиційною методологією, але не успішний по методології, орієнтованої на замовника. Відповідальність за неуспішність такого проекту несе замовник і, в деяких випадках, проектний офіс або служба замовника.

1.3 Ролі в проекті

У багатьох випадках у проекті виділяють ролі замовника, виконавця (і іноді інвестора або спонсора). Такі ролі майже завжди є для зовнішніх проектів. Для внутрішніх проектів такий поділ ролей також бажано з метою підвищення ефективності при поділі праці і для усунення конфлікт інтересів при прийманні результатів, визначення зон відповідальності.

Замовник визначає мету і обмеження проекту і його фінансування. Виконавець виконує проект згідно із затвердженим планом.

Замовник несе відповідальність за постановку і актуальність цілей та пріоритетів, ефективність експлуатації результатів проектів. Централізацією функцій замовника і управлінням портфеля проектів займається проектний комітет. У будівельних організаціях для цього виділяють спеціальну службу єдиного замовника.

У разі чіткого поділу ролей замовник-виконавець метою управління проектом є стабілізація робіт і мінімізація відхилень від затвердженого замовником плану.

Якщо замовник і виконавець знаходяться в різних організаціях, то складається договір на виконання проекту. При зміні вимог замовника може бути підписана додаткова угода до договору в рамках обмежень сумарного бюджету програми проектів, Обумовлених основним договором.

Для пов'язання проекту з інтересами бізнесу часто вводять ролі куратора (зазвичай від виконавця) і іноді спонсора (куратора від замовника), які мають найбільшу обізнаність про інтереси бізнесу, мають право затверджувати ключові зміни в проекті.

1.4 Методології управління проектами

1.4.1 Традиційна (Каскадна) методологія управління проектами

Традиційна методологія управління проектами може бути використана в усіх галузях, але найбільш поширена в будівництві. Вона також носить назву каскадної або Водоспадної моделі, внаслідок того, що пропонується нею послідовність фаз нагадує потік. Методологія виділяє сім послідовних етапів проектного управління:

1. визначення вимог
2. проектування
3. Реалізація (будівництво, виробництво ...)
4. впровадження
5. Тестування та налагодження
6. установка
7. Експлуатація та супровід

Перехід до наступної фази проекту можливий тільки в тому випадку, якщо попередній етап завершений і прийнятий замовником. Застосування даної методології найпереважніше в проектах, результатом виконання яких є матеріальний продукт (наприклад, будівельні проекти, установка обладнання та ін.) І для реалізації яких потрібна конкретна послідовність дій. Крім того, розроблені плани можливо використовувати повторно для аналогічних проектів в майбутньому.

Однак з іншого боку, каскадна модель управління проектом вимагає інвестицій в планування. Зазвичай перші дві фази займають від 20 до 40% від усього часу виконання проекту по даній методології. Внаслідок структурованого підходу, внесення змін в зміст робіт відбувається повільно, що робить методологію не гнучкої і непотрібної в випадках, якщо клієнт не впевнений, який результат він хоче отримати.

1.4.2 Методологія управління проектами PRINCE2

PRINCE2 (Projects in Controlled Environments) так само є структурованою методологією до проектного управління. Це одна з найпопулярніших методологій управління проектами, широко використовувана в Великобританії в управлінні як в бізнесі, так в органах влади. PRINCE2 - це процесно-орієнтована проектна методологія, яка фокусується на процесах верхнього рівня (управління, організація, контроль), а не на нижчих завданнях (декомпозиція робіт, розробка графіків). Методологія PRINCE2 базується на семи принципах, семи темах і семи процесах. Принципи є центральним елементом методології: якщо хоча б один з них не виконується, то не можна говорити про те, що проект виконується в рамках PRINCE2.

Принципи методології PRINCE2:

1. Постійна оцінка економічної необхідності - залишається чи незмінною економічний зиск від проекту протягом усього життєвого циклу проекту
2. Навчання на досвіді - команда проекту повинна постійно шукати і вивчати досвід попередніх проектів

3. Визначення рольової моделі - команда проекту повинна мати ясну організаційну структуру і залучати відповідних людей для вирішення потрібних завдань
4. Управління по етапах - необхідно, щоб проекти були сплановані, а також піддавалися моніторингу та контролю на кожному етапі виконання;
5. Управління за відхиленнями - слід чітко позначити допустимі межі відхилень в проекті, щоб встановити межі відповідальності
6. Фокус на продуктах - необхідно концентруватися на визначенні та досягненні якості продуктів (результати проекту)
7. Адаптація до проектної середовищі - слід адаптувати процеси і інструменти управління проектом до вимог проектного середовища, а також до масштабу робіт, їх складності, важливості, кваліфікаційним вимогам і ступеня ризику

Аспекти є напрями проектного управління, на які слід звертати увагу протягом тривалості всього проекту.

Аспекти методології управління проектами PRINCE2:

1. **Обґрунтування проекту:** яку цінність проект принесе організації?
2. **організація:** Яким чином необхідно розподілити ролі і відповідальність між членами проектної команди для того, щоб ефективно управляти проектом
3. **якість:** Які є вимоги і критерії до якості і яким чином можна їх забезпечити
4. **плани:** Кроки, необхідні для розробки плану, і інструменти PRINCE2, необхідні для використання
5. **ризики:** Яким чином менеджмент проекту буде вирішувати проблему наявності невизначеностей в плані проекту і в зовнішньому середовищі
6. **зміна:** Як керівництво проекту буде оцінювати вплив непередбачених завдань і змін і реагувати на них
7. **прогрес:** Реалізація проекту, виконання планів і подальший розвиток проекту

Нарешті, сім процесів розбивають життєвий цикл проекту на різні фази, для кожної з яких є свої рекомендовані до прийняття дії, одержувані продукти та зони відповідальності.

PRINCE2 має на увазі наступні процеси управління проектом:

1. запуск проекту
2. керівництво проектом
3. ініціація проекту

4. контроль етапів
5. управління створенням продукту
6. управління кордонами етапів
7. закриття проекту

PRINCE2 дозволяє стандартизувати процедури управління проектами, поліпшити координацію діяльності, а також допомагає зрозуміти, яким чином слід планувати проект і здійснювати моніторинг його виконання, що слід робити, якщо план проекту не виконується. Однак методологія PRINCE2 не є кращим вибором для проектів невеликого масштабу або для проектів з більшим ступенем імовірності змін обсягу робіт і вимог до них.

1.4.3 Гнучка методологія управління проектом (Agile Project Management)

Гнучке управління проектом являє собою поступальну і ітеративну проектну методологію. Її головною особливістю є те, що на початку виконання проекту точно невідомо, яким повинен бути кінцевий продукт і яким буде життєвий цикл проекту. Замість цього, проектна діяльність розбивається на кілька ітеративних фаз, званих «Спринт». Кожен спринт складається з безлічі завдань і має свій кінцевий продукт і результат. Методологія Agile дозволяє менеджерам проектів постійно отримувати зворотний зв'язок і покращувати продукт після кожної ітерації.

Відповідно до даної методологією управління проектами, відповідальність за результат ділиться між трьома ролями:

- **власник продукту** - визначає проектні цілі, розробляє оптимальний графік при заданих проектних параметрах, адаптує процес виконання проекту до змінами вимогам і встановлює пріоритети в характеристиках продукту
- **Scrum майстер** - встановлює пріоритети у виконанні завдань командою проекту і усуває виникаючі труднощі, що перешкоджають цьому
- **члени команди** - виконують більшість поставлених завдань, здійснюють щоденний менеджмент, створюють звіти про хід виконання проекту, контролюють якість продукту

Методологія Agile є гнучкою і дозволяє легко змінити параметри проекту, що є значущим для таких сервісно-орієнтованих проектів, як розробка програмного забезпечення або графічний дизайн. Але це методологія не підходить для проектів зі строго заданими параметрами та вимогами.

1.4.4 Методологія швидкої розробки додатків (Rapid Application Development - RAD)

Швидка розробка додатків (RAD) - це проектна методологія, найчастіше використовувана в проектах по розробці ПЗ, основною метою яких є швидке і якісне створення програми. Дана методологія управління проектами виділяє 4 стадії проекту:

- планування
- користування проектування
- швидке конструювання
- перемикання

Методологія швидкої розробки додатків, з одного боку, допомагає поліпшити показники результативності проекту та підвищити якість ризик-менеджменту. Але з іншого боку, дана методологія не підходить для масштабних ІТ проектів, може привести до низької якості коду і вимагає постійного залучення клієнта в процес виконання всього проекту.

1.5 Порівняння програмних засобів.

Назва	опис	Складання звітів	створювати пріоритет	графічний інтерфейс	створення груп	Визначати робочих днів	Призначення прав доступу членам	Відстежує успіхи проекту	Створювати to-до листи	Відслідковувати помилки	Виконання одного завдання групою людей
Jira	онлайн сервіс, що дозволяє командам-розробникам планувати проекти, призначати виконавців завдань, виставляти пріоритети і терміни.	+	-	+	+	+	-	+	-	+	-
Slack	web-сервіс для створення чатів і окремих груп для ведення обговорень з командою	-	-	+	+	-	-	-	-	-	-
GanttPro	дозволяє планувати і управляти проектами онлайн, візуалізувати	+	-	+	-	+	+	+	-	+	+

	процеси, створювати завдання і призначати їх учасникам, виставляти терміни виконання										
Redbooth	web-сервіс, який дозволяє відстежувати помилки, планувати проекти і активності, управляти ресурсами і ставити завдання учасникам проекту	+	+	+	+	+	+	-	-	+	-
Basecamp	онлайн-сервісів для спільної роботи над проектами.	+	-	+	+	-	-	-	+	-	-
Asana	web-додаток,, що дозволяє призначати завдання, виставляти пріоритети і дедлайни.	-	+	+	+	+	-	-	+	-	-
Trello	популярний онлайн-сервіс	-	+	-	+	+	-	-	+	-	-
Gemini	додаток для управління проектами, що включає систему стеження за вадами	-	+	+	-	+	-	-	-	+	-

2. Короткий огляд продукту

2.1. Основна інформація

Система Управління Проектами складається з двох розділів програмного забезпечення:

1. Централізована база даних на головному сервері

2. Локальний програмний продукт на боці користувача, який міститиме інструменти для взаємодії між різними всередині проекту, різні можливості для управляючого проектом по розподілу завдань між його підлеглими, а також функціонал який дозволяв би ефективно відслідковувати процес роботи над проектом.

Ключовими поняттями в даній системі є:

- Керівник - особа, яка має розширені права для взаємодії з даними, що знаходяться у базі, через розширений інтерфейс користувача, та займається розподілом завдань серед інших користувачів системи.
- Працівник - суб'єкт, що звертається до системи або посередника за одержанням потрібної йому інформації стосовного його ролі у проекті, та завдань що йому необхідно виконати.
- Спостерігач – особа, що має можливість вести спостереження за ходом виконання проекту, та різними його аспектами.
- Виключні ситуації - можливі проблеми, які можуть виникнути при виконанні програми.
- Синхронізація - узгодження наявних у різних системах персональних даних з даною системою за для полегшення роботи адміністраторів системи, та складання більш повної бази даних певного Викладача.
- Реєстрація - процес введення особистих даних, для того, щоб стати повноцінним користувачем системи.
- Верифікація - процес підтвердження справжності внесених у базу даних.
- Статичні поля - поля, які є однаковими (або обов'язковими) для будь-якого облікового запису Викладача.
- Обліковий запис - одиниця даних, що узагальнює у собі усю наявну в базі інформацію про одного Викладача.
- Поля подій - поля, що відображають у собі інформацію про зміну інших полів (таку як підвищення, зміна посади, догани і т.д.)

2.2. Призначення системи управління

Для швидкого і зручного управління діяльністю команди що залучена до проекту.

Головним сценарієм є надання зацікавленій особі (управляючий проектом) потрібної інформації щодо управління проектом.

2.3. Політика взаємодії з користувачем

Користувачем системи є управляючий проектом, що безпосередньо керує розподілом різних частин проекту серед членів команди що займається проектом, а також працівники, які будуть виконувати завдання які були поставлені їм управляючим.

Користувач системи може, згідно встановлених адміністраторами правил, отримувати інформацію з бази даних, а також редагувати і доповнювати її за необхідності.

2.4. Характеристика ділового процесу

Кожен працівник системи виконує свою роботу. Менеджер першого рівня керує працю менеджерів другого рівня. В свою чергу менеджери другого рівня відповідають за працю своїх підлеглих (воркерів). Кожен воркер виконує своє певне завдання, а також має певні ресурси для його виконання. За працездатність системи відповідають як воркери (програмісти, дизайнери та інші), так і менеджери усіх рівнів.

2.5. Призначення системи

Система призначена для полегшення управління проектами, що полягає в покращенні ефективності різних учасників проекту, та оптимізації часу виконання проекту, за допомогою певної зручних інструментів комунікації, відстеження загального стану виконання проекту, та системи розподілу задач між учасниками проекту.

2.6. Надійність

1. Захист особистих даних виконується у відповідності до закону України про захист персональних даних **(Відомості Верховної Ради України (ВВР), 2010, № 34, ст. 481)**
2. Резервне копіювання повинно проводитись для забезпечення цілісності системи та збереження даних після неочікуваних збоїв та інших обставин, що можуть призвести до їх втрати.
3. Система повинна бути добре захищена від різного роду зловмисних атак із метою заволодіння інформації або її невірифікованої зміни.
4. Система повинна витримувати великі навантаження, обслуговуючи значну кількість користувачів.

2.7. Функціональність

Основні вимоги до функціональності, що пред'являються зацікавленими

особами до предмета розробки, відносяться до трьох категорій:

- Керівник
- Працівник
- Спостерігач

3. Ділові правила та прописи

3.1 Діаграма юзкейсів

Сценарії:

Сценарії реєстрації користувача:

Ідентифікатор	UC-1.1
Назва	Реєстрація нового Користувача
Учасники	Користувач, Система
Передумова	Відсутня
Результат	Створено новий обліковий запис для користувача
Виключні випадки	Користувач не підтверджує ідентифікацію
Основний сценарій	<ol style="list-style-type: none">1) Користувач заходить на сайт та натискає кнопку «Реєстрація»2) Користувач заповнює поля Username, password та e-mail3) Користувач підтверджує вірність введених даних4) Система відсилає користувачу лист на пошту5) Користувач заходить на пошту та переходить за посиланням6) Система створює обліковий запис користувача

Сценарії входу у систему користувачем:

Ідентифікатор	UC-1.2
Назва	Вхід у систему
Учасники	Користувач, Система
Передумова	Користувач має Username, password

Результат	Надання користувачу доступу для користування/управління проектами
Виключні випадки	Username або password не вірні
Основний сценарій	<ol style="list-style-type: none"> 1) Користувач заходить на сайт та натискає кнопку «Вхід» 2) Користувач заповнює поля Username, password та натиска «Log in» 3) Система перевіряє вірність Username, password 4) При вірності введених Username та password користувачу надається доступ до користування/управління проектами

Сценарії для створення проекту:

Ідентифікатор	UC-1.3
Назва	Створення проекту
Учасники	Користувач(Підприємець), Система
Передумова	Користувач має Username, password
Результат	Створення проекту
Виключні випадки	Username або password не вірні
Основний сценарій	<ol style="list-style-type: none"> 1) Користувач натискає на кнопку «Створити Проект» 2) Користувач заповнює поля назва проекту та натискає кнопку «Підтвердити» 3) Система створює пустий проект, в якому є панель під назвою «Дошка проекту» 4) Система записує у даному проекті користувача як підприємця

Сценарії для створення Таблиці завдань:

Ідентифікатор	UC-1.4
Назва	Створення Таблиці завдань
Учасники	Підприємець, Система
Передумова	Існує вже створений проект
Результат	Створення Таблицю завдань
Виключні випадки	Не підтвердження створення Таблиці
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить у свій проект 2) Підприємець натискає на кнопку «Створити Таблицю завдань» 3) Підприємець заповнює поля назва кожної колони та натискає кнопку «Підтвердити» 4) Система створює пусту таблицю, Система закріплює створену таблицю за даним проектом

Сценарії для створення завдань у Таблиці для завдань:

Ідентифікатор	UC-1.5
Назва	Створення завдань у Таблиці для завдань
Учасники	Підприємець, Система
Передумова	Існує вже створений проект та створена Таблиця
Результат	Створення завдань у Таблиці для завдань

Виключні випадки	Не підтверджене завдання
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить у свій проект 2) Підприємець натискає у необхідній колоні Таблиці кнопку «Створити завдання» 3) Підприємець заповнює поля назва завдання та натискає кнопку «Підтвердити» 4) Система створює пусту Панель-завдання 5) Система додає створену пусту Панель-завдання до Таблиці у вибрану колонку

Сценарії для зміни колони завдання у Таблиці для завдань:

Ідентифікатор	UC-1.6
Назва	Зміни колони завдання у Таблиці для завдань
Учасники	Підприємець, Система
Передумова	Існує вже створений проект, створена Таблиця та створене завдання
Результат	Зміни положення завдання у Таблиці для завдань
Виключні випадки	-
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить у свій проект 2) Підприємець вибирає створене завдання 3) Система виділяє вибране Підприємцем завдання та копіює вибране завдання 4) Підприємець змінює положення вибраного завдання у Таблиці 5) Система видаляє вибране завдання 6) Система вставляє виділене завдання у вибрану Підприємцем колонку в Таблиці завдань

Сценарії додавання інших заінтересованих осіб до проекту:

Ідентифікатор	UC-1.7
Назва	Додавання заінтересованих осіб до проекту
Учасники	Підприємець, Система, заінтересовані особи
Передумова	Існує вже створений проект
Результат	Додано заінтересовану особу до проекту
Виключні випадки	Відмова користувачем примати запрошення, невірно ведений email
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить до проекту

	<ul style="list-style-type: none"> 2) Підприємець натискає кнопку «Додати заінтересованих осіб» 3) Підприємець водить у поля email заінтересованої особи натискає кнопку «Підтвердити» 4) Система відправляє письмо на пошту до заінтересованої особи 5) Заінтересована особа переходить за посиланням та входить/реєструється у даній системі (см. UC-1.1/UC-1.2) 6) Система додає заінтересовану особи особу до проекту 7) Система записує заінтересовану особу як підприємця
--	---

Сценарії зміни прав доступу до проекту:

Ідентифікатор	UC-1.8
Назва	Зміни прав доступу до проекту
Учасники	Підприємець, Система, заінтересовані особи
Передумова	Існує вже створений проект та добавлені заінтересовані особи
Результат	Зміни прав доступу до проекту
Виключні випадки	Не має
Основний сценарій	<ul style="list-style-type: none"> 1) Підприємець переходить до проекту 2) Підприємець натискає кнопку «Налаштування проекту» 3) Підприємець вибирає розділ «Права доступу» 4) Підприємець встановлює права доступу стосовно до заінтересованої особи 5) Система змінює права доступу особи на вибране Підприємцем 6) Система відправляє письмо заінтересованій особі про зміну прав доступу у проекті

Сценарії об'єднання проектів:

Ідентифікатор	UC-2.0
Назва	Об'єднання проектів
Учасники	Підприємець, Система, Зацікавлені особи

Передумова	Існує вже хоча б два створених проекти за авторством одного і того ж Підприємця та добавлені Зацікавлені особи
Результат	Об'єднання проектів
Виключні випадки	Немає
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить до проекту 2) Підприємець натискає кнопку «Настройки проекту» 3) Підприємець вибирає розділ «Управління проектом» 4) Підприємець натискає кнопку «Об'єднати з іншим проектом» 5) Підприємець виділяє плюсами проект(и) з випадного списку, з яким(и) хоче об'єднати даний проект 6) Система об'єднує таблиці завдань всіх виділених проектів, завдання даного проекту будуть в колонках зліва 7) Система запрошує Зацікавлених осіб, листом на email, до участі в об'єднаному проекті 8) Система видаляє всі проекти, що об'єднуються, крім даного

Сценарії розбиття проекту на два окремих проекти:

Ідентифікатор	UC-2.1
Назва	Розбиття проекту на два окремих проекти
Учасники	Підприємець, Система, Зацікавлені особи
Передумова	Існує вже створений проект та добавлені Зацікавлені особи
Результат	Розбиття проекту на два окремих проекти
Виключні випадки	Якщо в проекті тільки одне завдання в Таблиці завдань
Основний сценарій	<ol style="list-style-type: none"> 1) Підприємець переходить до проекту 2) Підприємець натискає кнопку «Настройки проекту» 3) Підприємець вибирає розділ «Управління проектом» 4) Підприємець натискає кнопку «Розбиття проекту» 5) Система відкриває діалогове вікно з надписом «Виділіть завдання з Таблиці завдань, які Ви хочете перенести в новий проект» та кнопкою «ОК» 6) Підприємець натискає кнопку «ОК», виділяє лівою кнопкою миші завдання з Таблиці завдань, які будуть перенесені в новий проект, після чого натискає кнопку «Перейти до наступного кроку» під Таблицею завдань

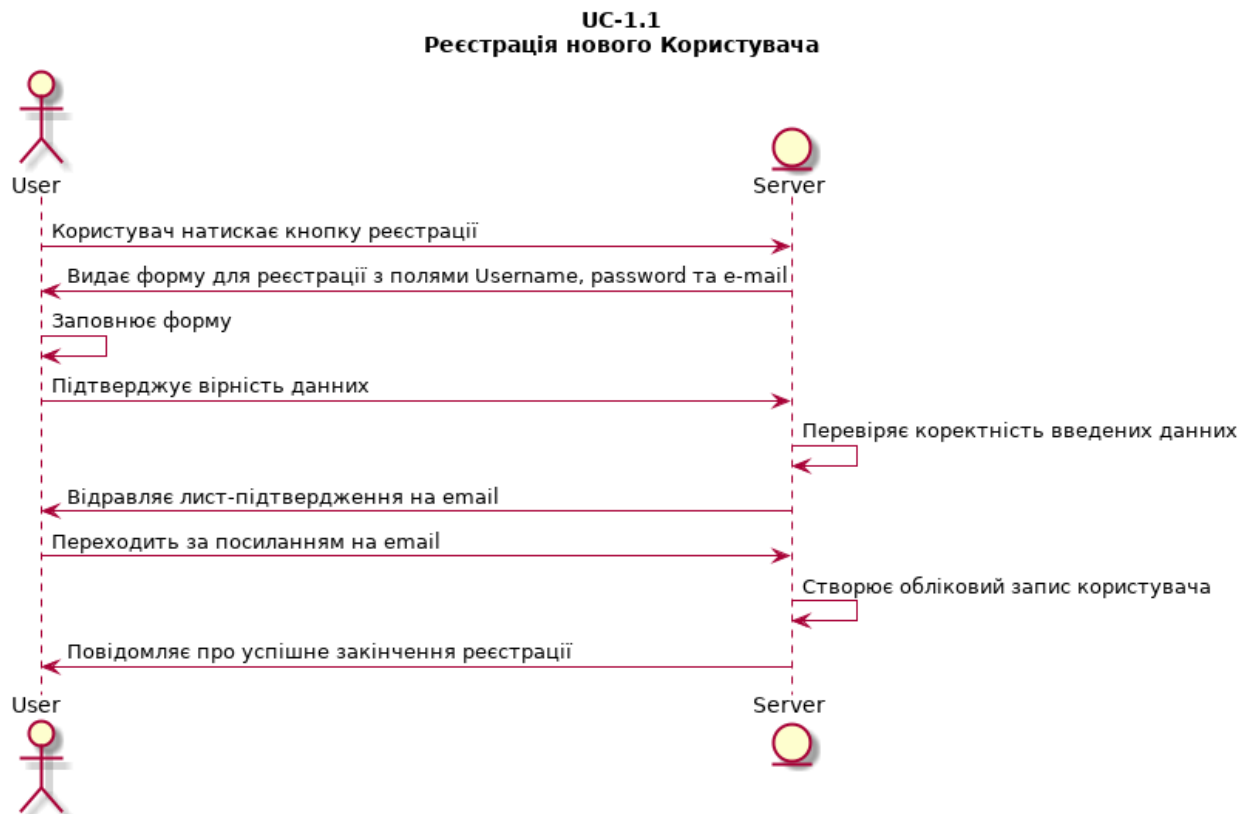
	<p>7) Система створює пустий проект без назви та видає діалогове вікно з проханням ввести назву нового проекту</p> <p>8) Підприємець вводить назву та нажимає «Наступний крок»</p> <p>9) Система присвоює проекту вказану назву</p> <p>10) Система видає діалогове вікно, де пропонується змінити Зацікавлених осіб, що будуть мати доступ до корегування нового проекту</p> <p>11) Підприємець вибирає, які Зацікавлені особи не будуть мати доступу до нового проекту, або зразу нажимає кнопку «Наступний крок», якщо хоче перенести всіх Зацікавлених осіб з минулого проекту</p> <p>12) Система створює Таблицю завдань в новому проекті, куди додає завдання, що були виділені в кроці 6) з початкового проекту та видаляє ці завдання з Таблиці завдань старого проекту</p> <p>13) Система запрошує вибраних Зацікавлених осіб, листом на email, до участі в новому проекті та сповіщає зацікавлених осіб про роздвоєння проекту</p>
--	---

Сценарії видалення проекту:

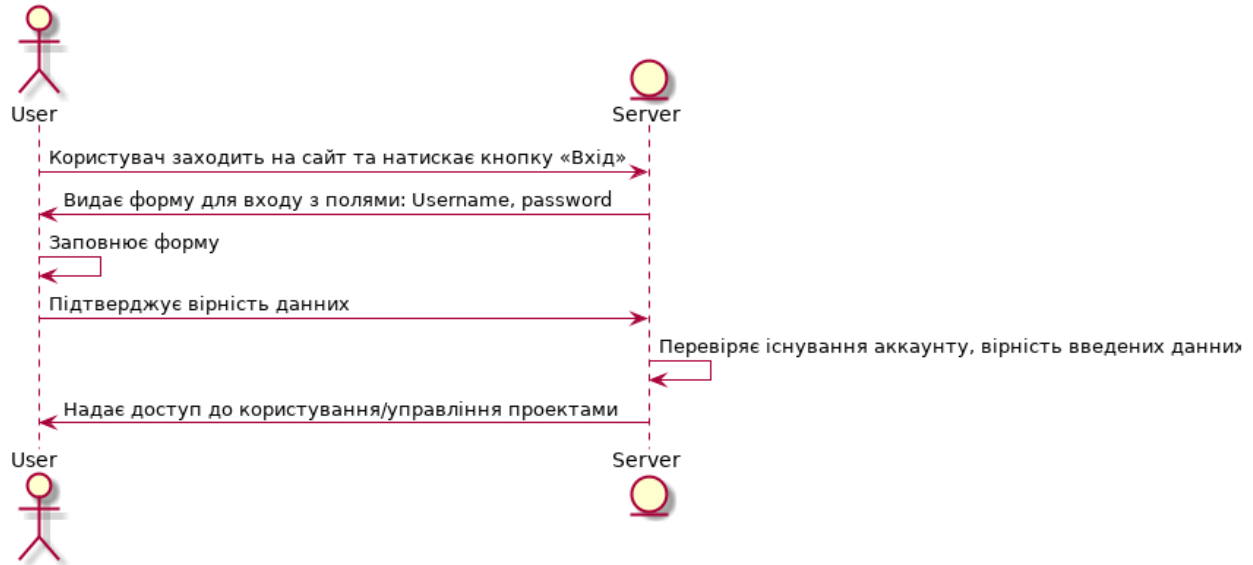
Ідентифікатор	UC-2.2
Назва	Видалення проекту
Учасники	Підприємець, Система, Зацікавлені особи
Передумова	Існує вже створений проект та добавлені Зацікавлені особи
Результат	Видалення проекту
Виключні випадки	-
Основний сценарій	<p>1) Підприємець переходить до проекту</p> <p>2) Підприємець натискає кнопку «Настройки проекту»</p> <p>3) Підприємець вибирає розділ «Управління проектом»</p> <p>4) Підприємець натискає кнопку «Видалити проект»</p> <p>5) Система попереджує Підприємця діалоговим вікном, що операція не може бути обернена та питає, чи впевнений Підприємець</p> <p>6) Підприємець натискає кнопку «Так» в цьому діалоговому вікні</p>

	7) Система видаляє проект та сповіщає всіх Зацікавлених осіб email листом
--	--

3.7 Діаграми послідовностей



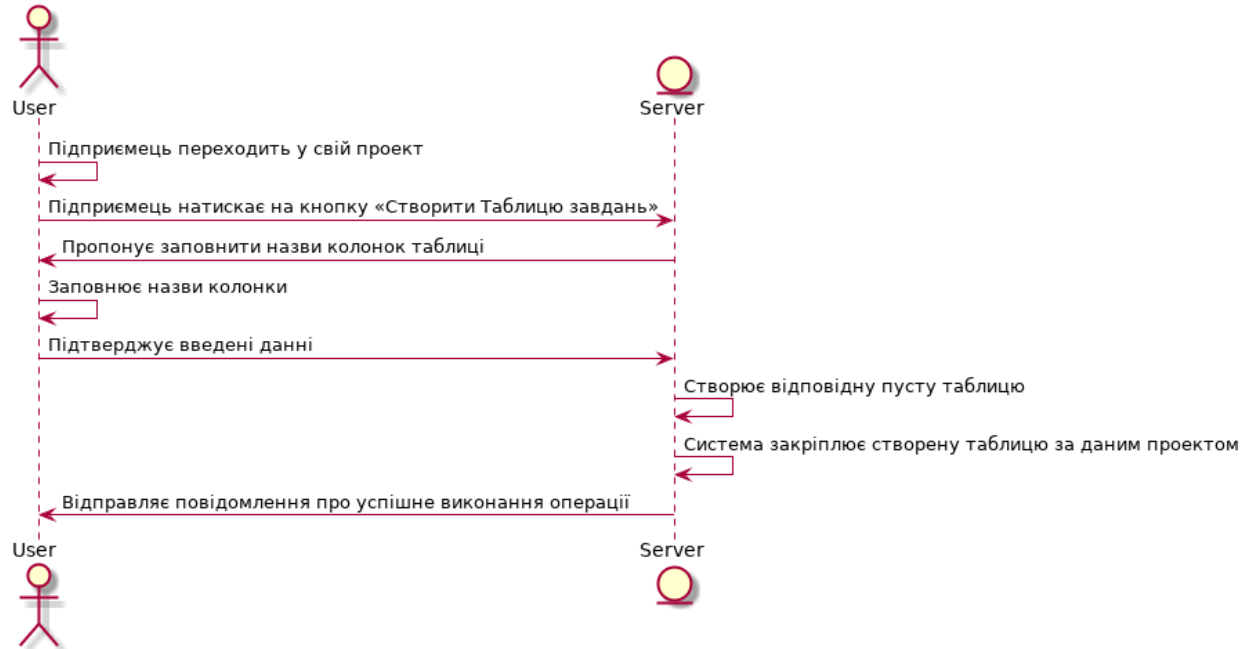
UC-1.2
Сценарії входу у систему користувачом:



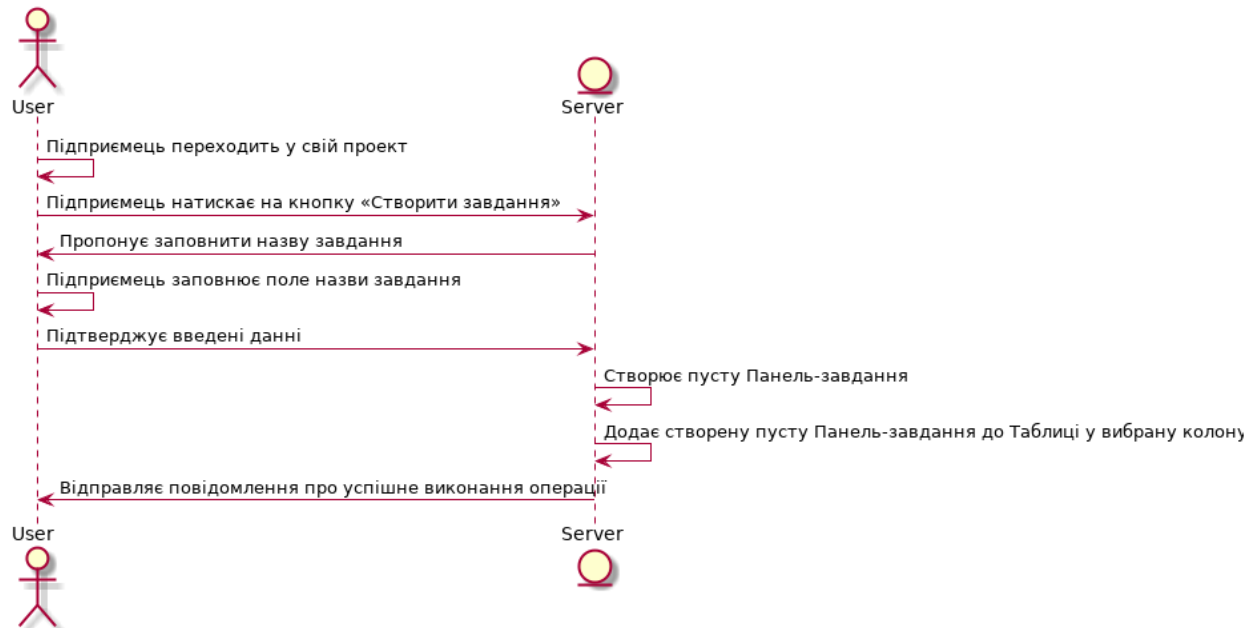
UC-1.3
Створення проекту



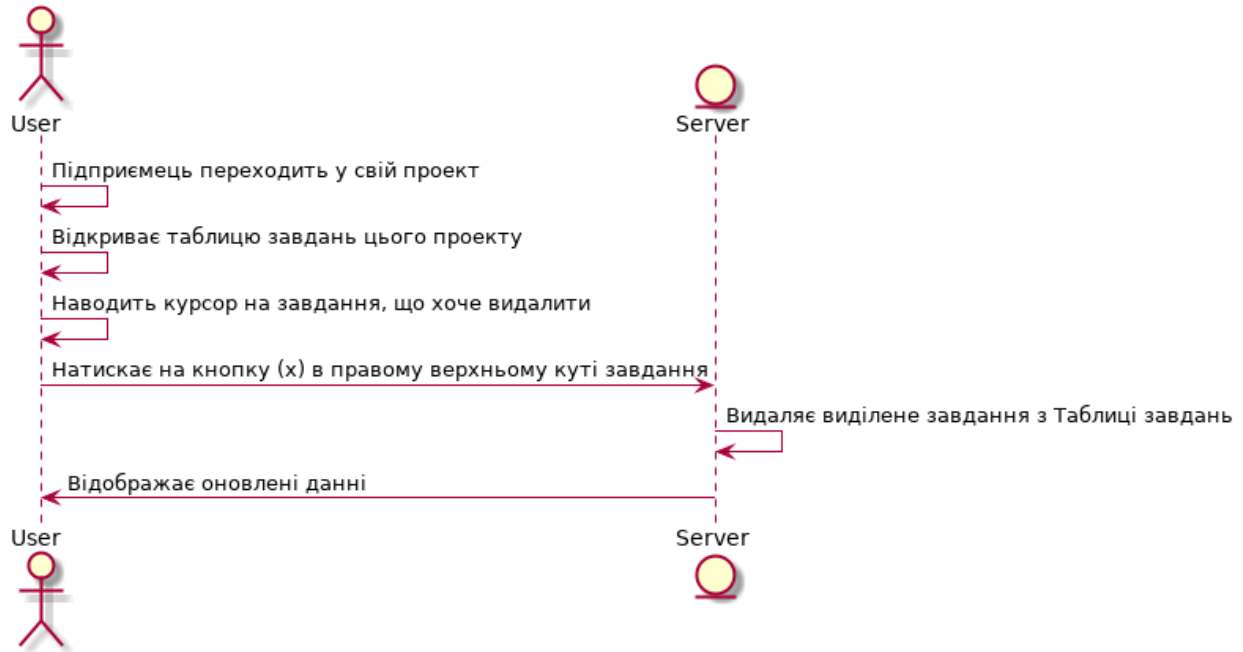
UC-1.4
Створення Таблиці завдань



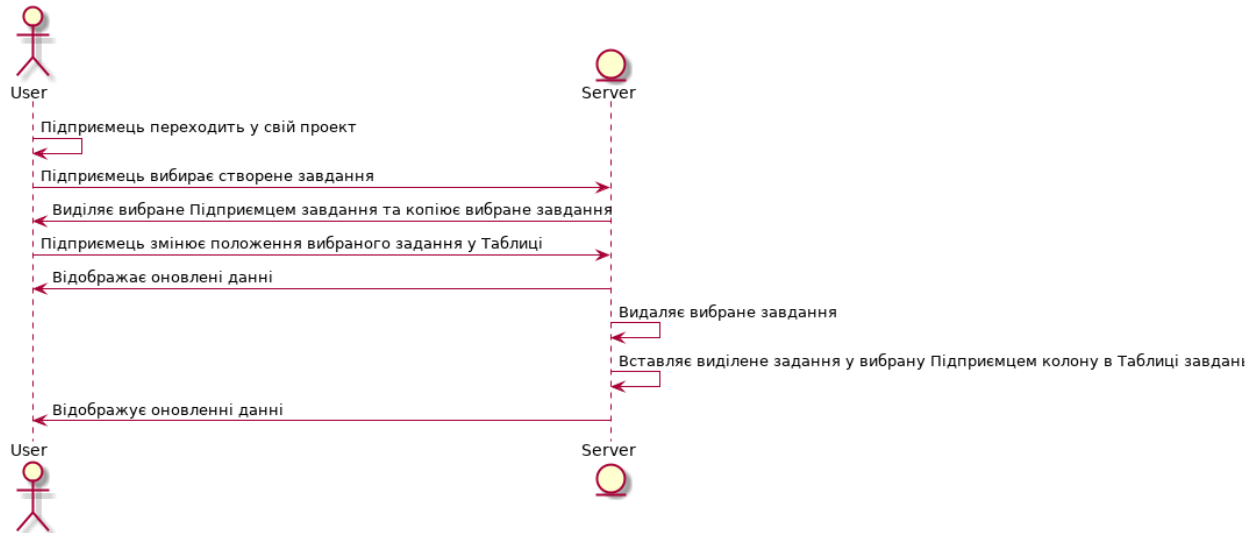
UC-1.5
Створення завдань у Таблиці для завдань



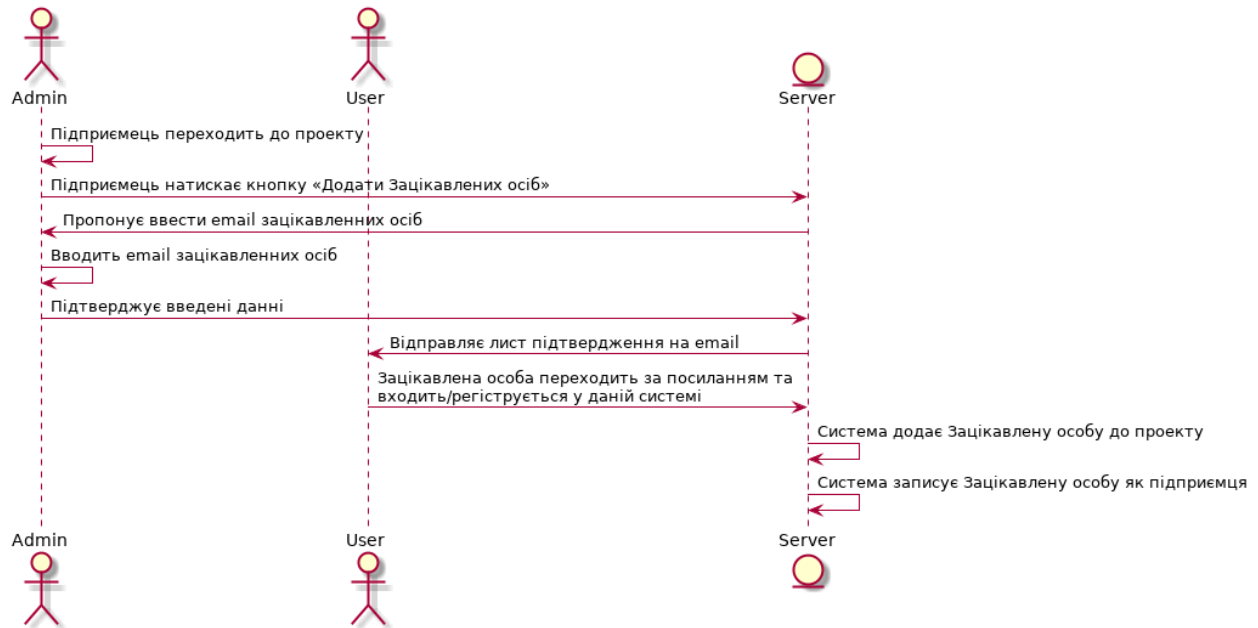
UC-1.6 Сценарії для видалення завдань у Таблиці для завдань



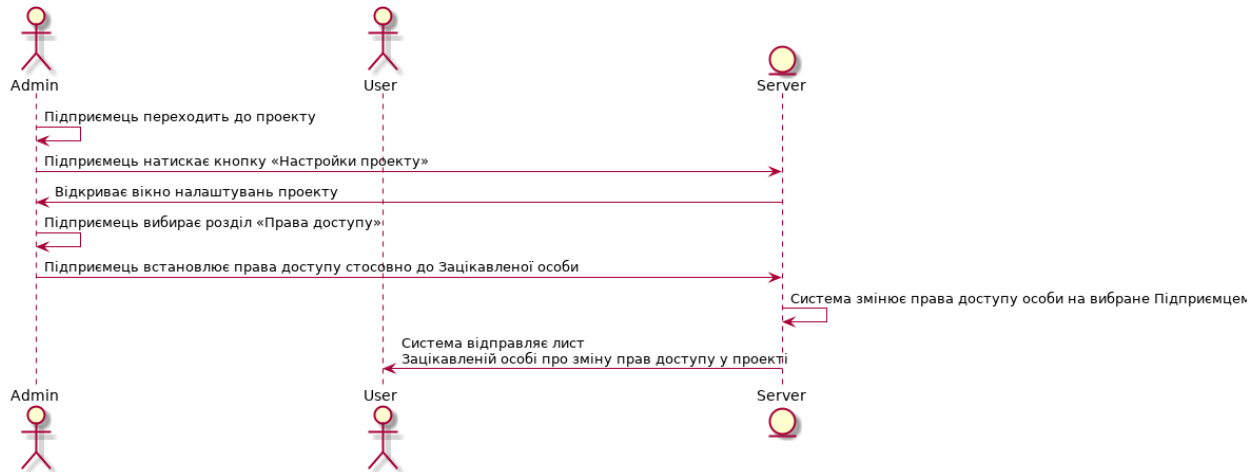
UC-1.7 Зміни колони завдання у Таблиці для завдань



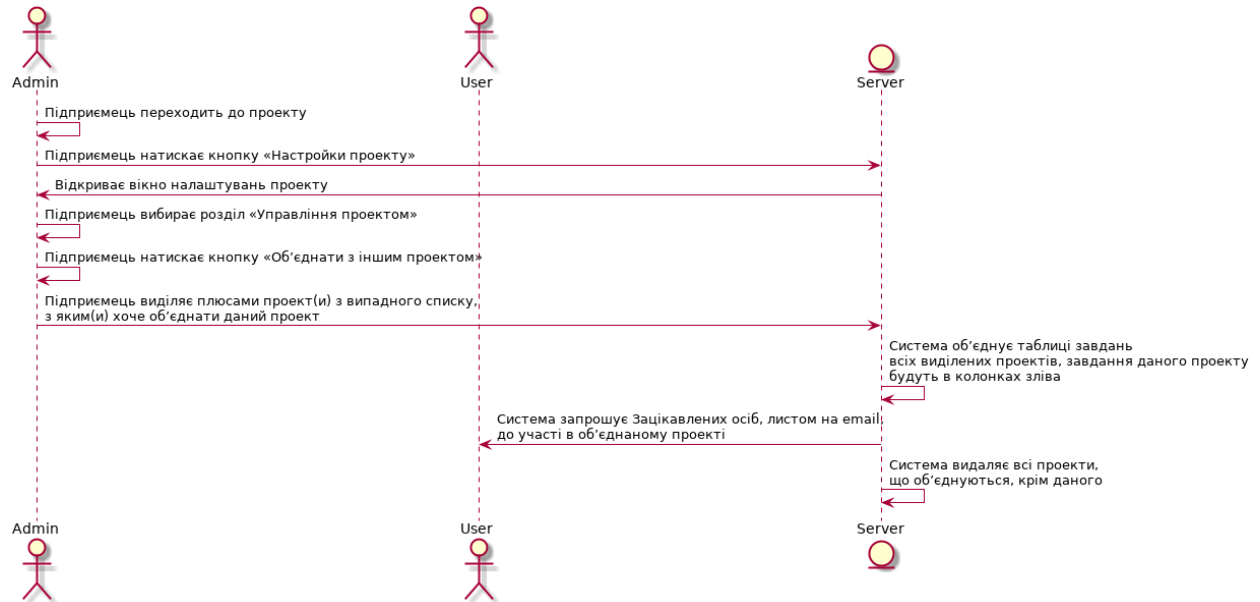
UC-1.8 Додавання Зацікавлених осіб до проекту



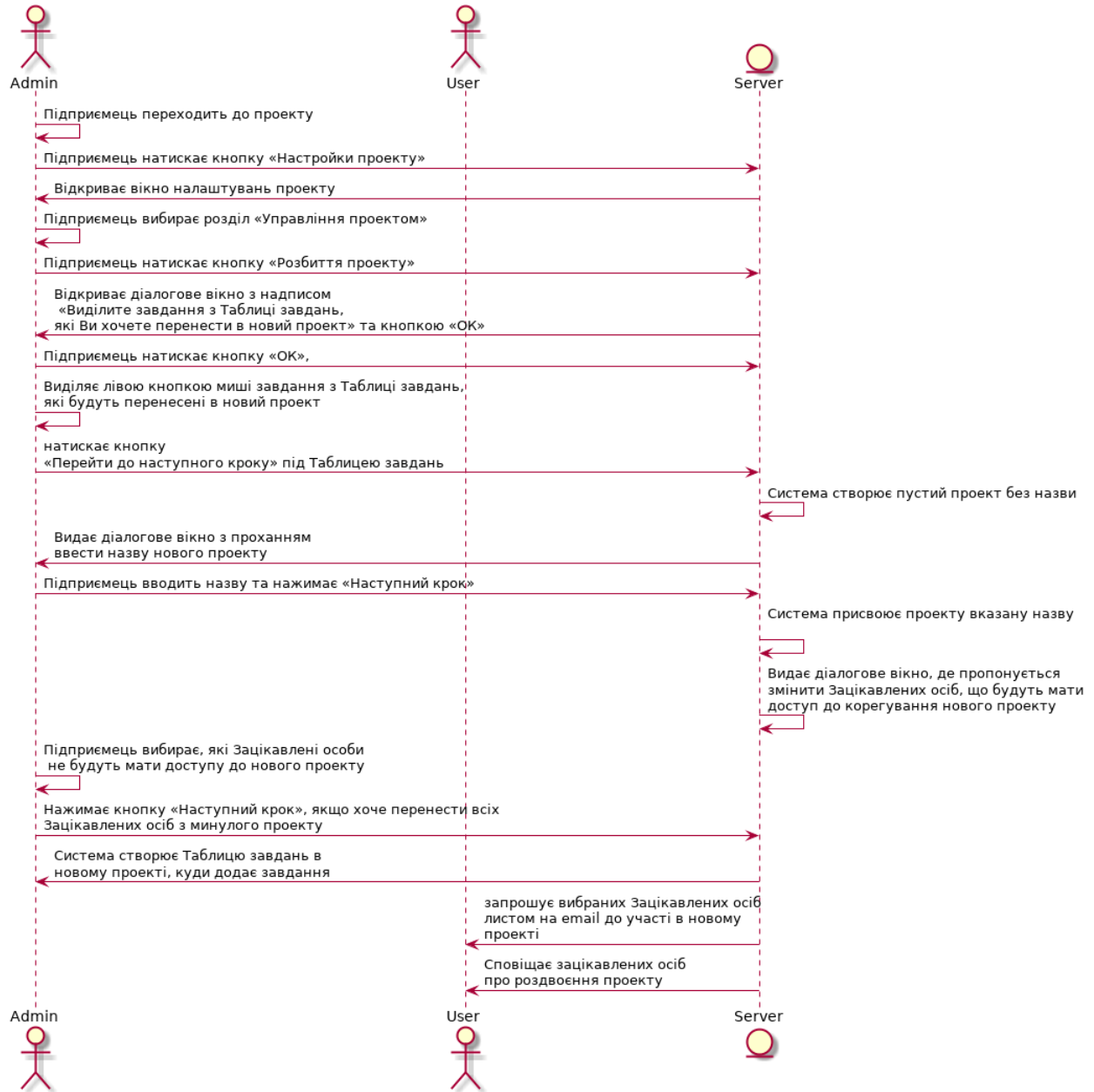
UC-1.9 Сценарій зміни прав доступу до проекту:

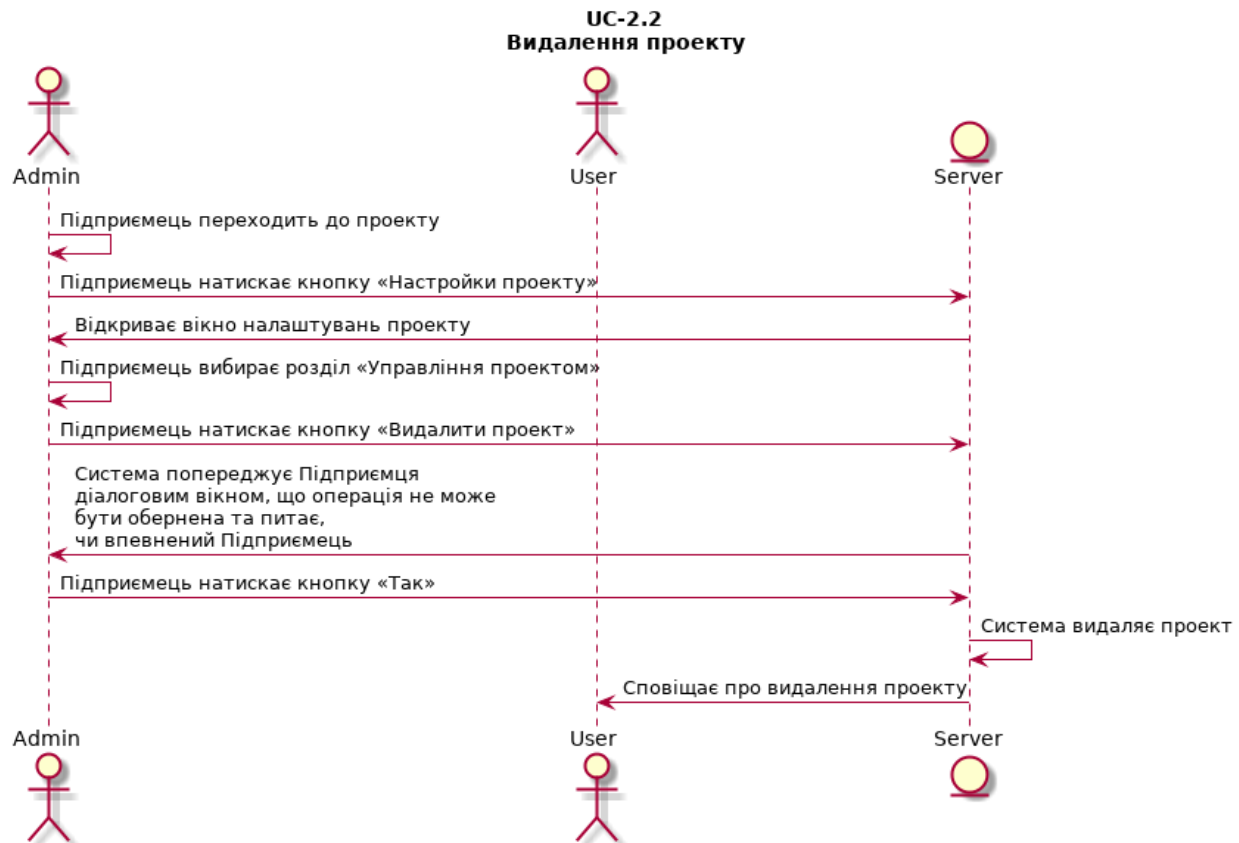


UC-2.0
Об'єднання проектів



UC-2.1
Розбиття проекту на два окремих проекти





4.Висновок

Був проведений аналіз ринку. При проведенні даного аналізу було виявлено, що представлені на ринку технології не можуть повністю задовольнити потребу швидкого і зручного управління діяльністю команди, що залучена до проекту. На основі цих факторів було прийнято рішення розробити систему управління проектами, яка б об'єднувала у собі існуючі у різних системах технології, а також була доповнена деяким функціоналом (див. П. 2.9)

1. Список використаної літератури

- https://ru.wikipedia.org/wiki/Управление_проектами
- <http://upr-proektom.ru/standarty-upravleniya-proektami>

- <https://dic.academic.ru/dic.nsf/ruwiki/48137>
- <https://www.pmservices.ru/project-management-news/top-4-metodologii-upravleniya-proektami/>
- <https://business.gov.kz/ru/business-development/tools/project-management-tools.php>
- <http://earchive.tpu.ru/bitstream/11683/40037/1/TPU392450.pdf>
- <http://pm.pacc.ru/stroydom/item/237-top4.html>