Міністерство освіти і науки України

Державний університет «Одеська політехніка»

Інститут комп’ютерних систем  
Кафедра інформаційних систем

**КУРСОВА РОБОТА**

з дисципліни «Технології створення програмних продуктів»

за темою

«Digital Write»

Пояснювальна записка до етапів визначення вимог до програмного продукту та планування процесів розробки програмного продукту

Виконали:

студенти 3-го курсу

групи АІ-192

Таужнянський А.В.,

Макаренко А.О.

Перевірив:

Блажко О. А.

Одеса-2021

**Анотація**

В курсовій роботі розглядається процес створення програмного продукту «Digital Write» на етапах визначення вимог до програмного продукту та планування процесів розробки.

Робота виконувалась в команді з декількох учасників: Таужнянський А.В., Макаренко А.О.

В робота пов`язана з такими матеріальними потребами споживача як Актуальність . Аналіз вказаних потреб визначив інформаційну потребу - Повільне проходження аутентифікації.

При визначені ступеня готовності існуючих програмних продуктів до вирішення інформаційної потреби проаналізовано наступні програмні продукти: Google Authentificator, Cyberarms IDDS.

Поточну версію пояснювальної записки до результатів роботи розміщено на *GitHub*-репозиторії за адресою: https://github.com/TauzhnyanskiyArtiom/digital-write.git

**Перелік скорочень**

ОС – операційна система

ІС – інформаційна система

БД – база даних

СКБД – система керування базами даних

ПЗ – програмне забезпечення

ПП– програмний продукт

UML – уніфікована мова моделювання

ІА - ідентифікатори / аутентифікатори

**Зміст**

|  |  |
| --- | --- |
|  | стор. |
| 1 Вимоги до програмного продукту | 6 |
| 1.1 Визначення потреб споживача | 6 |
| 1.1.1 Ієрархія потреб споживача | 6 |
| 1.1.2 Деталізація матеріальної потреби | 7 |
| 1.2 Бізнес-вимоги до програмного продукту | 7 |
| 1.2.1 Опис проблеми споживача | 7 |
| 1.2.1.1 Концептуальний опис проблеми споживача | 7 |
| 1.2.1.2 Опис цільової групи споживача | 7 |
| 1.2.1.3 Метричний опис проблеми споживача | 7 |
| 1.2.2 Мета створення програмного продукту | 10 |
| 1.2.2.1 Проблемний аналіз існуючих програмних продуктів | 10 |
| 1.2.2.2 Мета створення програмного продукту | 10 |
| 1.2.3 Назва програмного продукту | 10 |
| 1.2.3.1 Гасло програмного продукту | 10 |
| 1.2.3.2 Логотип програмного продукту | 10 |
| 1.3 Вимоги користувача до програмного продукту | 11 |
| 1.3.1 Історія користувача програмного продукту | 12 |
| 1.3.2 Діаграма прецедентів програмного продукту | 12 |
| 1.3.3 Сценаріїв використання прецедентів програмного продукту | 13 |
| 1.4 Функціональні вимоги до програмного продукту | 16 |
| 1.4.1. Багаторівнева класифікація функціональних вимог | 16 |
| 1.4.2 Функціональний аналіз існуючих програмних продуктів | 18 |
| 1.5 Нефункціональні вимоги до програмного продукту | 18 |
| 1.5.1 Опис зовнішніх інтерфейсів | 18 |
| 1.5.1.1 Опис інтерфейсів користувача | 18 |
| 1.5.1.1.1 Опис INPUT-інтерфейсів користувача | 18 |
| 1.5.1.1.2 Опис OUTPUT-інтерфейсів користувача | 18 |
| 1.5.1.2 Опис інтерфейсу із зовнішніми пристроями | 20 |
| 1.5.1.3 Опис програмних інтерфейсів | 20 |
| 1.5.1.4 Опис інтерфейсів передачі інформації | 21 |
| 1.5.1.5 Опис атрибутів продуктивності | 21 |
| 2 Планування процесу розробки програмного продукту | 22 |
| 2.1 Планування ітерацій розробки програмного продукту | 22 |
| 2.2 Концептуальний опис архітектури програмного продукту | 22 |
| 2.3 План розробки програмного продукту | 23 |
| 2.3.1 Оцінка трудомісткості розробки програмного продукту | 23 |
| 2.3.2 Визначення дерева робіт з розробки програмного продукту | 25 |
| 2.3.3 Графік робіт з розробки програмного продукту | 27 |
| 2.3.3.1 Таблиця з графіком робіт | 27 |
| 2.3.3.2 Діаграма Ганта | 27 |
| Висновки до курсової роботи | 28 |

# 1 Вимоги до програмного продукту

**1.1 Визначення потреб споживача**

* + 1. **Ієрархія потреб споживача**

Відомо, що в теорії маркетингу потреби людини можуть бути представлені у вигляді ієрархії потреб ідей американського психолога Абрахама Маслоу включають рівні:

* фізіологія (вода, їжа, житло, сон);
* безпека (особиста, здоров'я, стабільність),
* приналежність (спілкування, дружба, любов),
* визнання (повага оточуючих, самооцінка),
* самовираження (вдосконалення, персональний розвиток).

На рисунку 1.1 представлено одну ієрархію потреби споживача, яку хотілося б задовольнити, використовуючи майбутній програмний продукт.



Рис. 1.1 – Приклад ієрархії потреби споживача

1.1.2 Деталізація матеріальної потреби

На рисунку 1.2 представлено Mindmap-карту деталізації ієрархії потреби споживача.

Рис. 1.2 - Mindmap-карта

1.2 Бізнес-вимоги до програмного продукту

1.2.1 Опис проблеми споживача

1.2.1.1 Концептуальний опис проблеми споживача

Умова задоволення потреб переведення даних в інформацію: Актуальність.

Загальний опис проблеми: Повільне проходження аутентифікації користувача в системі.

1.2.1.2 Опис цільової групи споживачів

Будь-який користувач, який хоче пройти аутентифікацію в системі.

1.2.1.3 Метричний опис проблеми споживача

Довге очікування на підтвердження першого та другого фактору аутентифікації 3-7 секунди, в залежності від стану зв’язку.

На рисунку 1.3 представлена анкета для визначення метрики проблеми споживача*.*

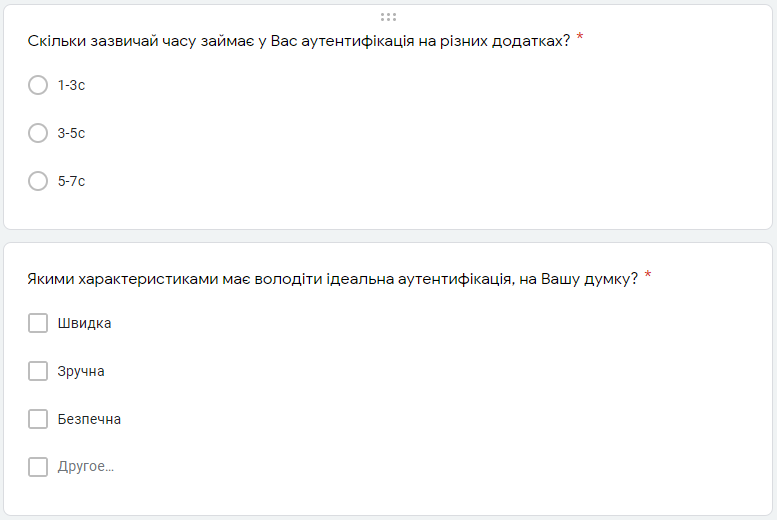
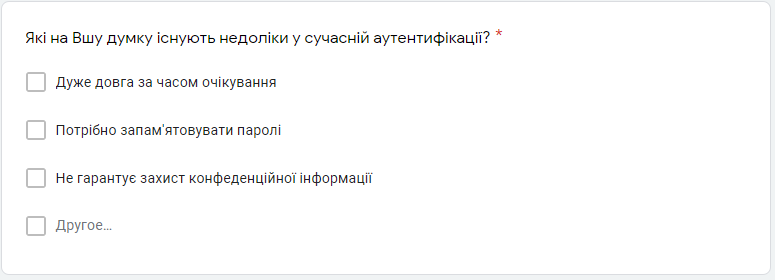


Рис 1.3 – Анкета для споживача

На рисунку 1.4 представлено результати анкетування.



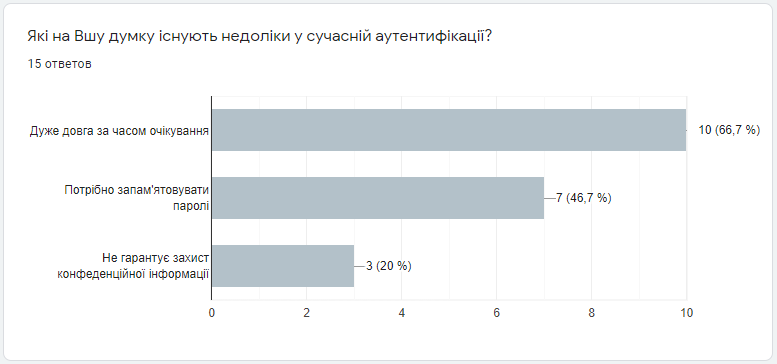


Рис 1.3 – Результати анкетування

Можна зробити висновок, що головним недоліком є довге очікування від двофакторної аутентифікації. Та аутентифікація має бути зручною, швидкою та безпечною.

1.2.2 Мета створення програмного продукту

1.2.2.1 Проблемний аналіз існуючих програмних продуктів

Таблиця 1.1 – Проблемний аналіз існуючих програмних продуктів.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Назва продукту | Вартість | Ступінь готовності | Примітка |
| 1 | Google Authentificator | Безкоштовно | 1 | Довге очікування від другого фактору аутентифікації. |
| 2 | Cyberarms IDDS | Безкоштовно | 1 | Попереднє блокування доступу від атак перебором паролю |

1.2.2.2 Мета створення програмного продукту

Прискорення ідентифікації користувача на основі розпізнавання цифрового почерку через створення Rest API сервісу.

1.2.3 Назва програмного продукту

1.2.3.1 Гасло програмного продукту

Digital Write – швидка аутентифікація на основі розпізнавання почерку користувача.

1.2.3.2 Логотип програмного продукту

На рисунку 1.5 представлено логотип Digital Write.

Рис. 1.5 – логотип Digital Write

1.3 Вимоги користувача до програмного продукту

1.3.1 Пригодницька історія користувача програмного продукту (за бажанням)

Завдання захисту інформації від несанкціонованого доступу стає все більш актуальною. найбільш достовірні результати дає використання біометричних методів аутентифікації. До них відносяться наступні методи:

* розпізнавання голосу;
* розпізнавання особи;
* розпізнавання райдужної оболонки ока;
* розпізнавання відбитків пальців;

Крім того, для кожної людини характерний власний клавіатурний почерк. При цьому використання розпізнавання клавіатурного почерку має низку переваг перед іншими методами аутентифікації. Справа в тому, що для розпізнавання клавіатурного почерку НЕ потребує додаткового обладнання, внаслідок чого ціна впровадження такої системи невисока.

За допомогою нашого сервісу можна здійснити швидку аутентифікацію користувача на основі передачі основних характеристик клавіатурного почерку: швидкість набору тексту, точність набору тексту, час утримання клавіш. Також наш сервіс дозволяє керувати користувачами, а саме: реєструвати нових користувачів, керувати ролями користувача, видаляти користувача.

1.3.2 Історія користувача програмного продукту

Наш користувач – це зовнішня система, яка взаємодіє з нашим Rest API сервісом.

1. Зовнішня система реєструє нового користувача через передачу ІА-параметрів(швидкість набору тексту, точність набору тексту, частота натискань клавіш) та ідентифікатора системи, та отримує унікальний токен користувача.
2. Зовнішня система авторизує користувача через передачу ІА-параметрів та токена користувача, та отримує унікальний ідентифікатор сесії.
3. Зовнішня система авторизує користувача через передачу ідентифікатора сесії.
4. Зовнішня система видаляє користувача через передачу токену користувача.
5. Зовнішня система змінює ролі користувача через передачу токену та нових ролей користувача.

1.3.3 Діаграма прецедентів програмного продукту

На рисунку 1.6 представлено Use Case UML-діаграму для ПП*.*

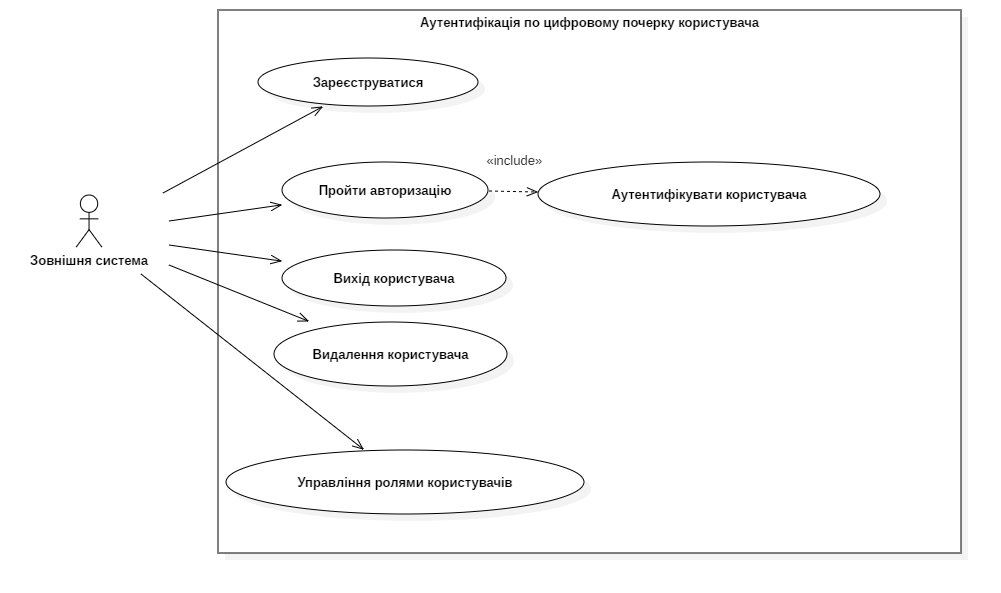


Рис. 1.6 - Use Case діаграма

1.3.4 Сценаріїв використання прецедентів програмного продукту

Прецедент «Зареєструватися користувача»:

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: користувач успішно зареєструється в системі.

Успішний сценарій:

1. Зовнішня система передає ІА-параметри: швидкість набору тексту, точність набору тексту, час утримання клавіш, ролі та ідентифікатора системи.
2. ПП перевіряє ІА-параметри, які передала зовнішня система.
3. ПП зберігає користувача в БД.
4. ПП передає повідомлення з приводу успішної реєстрації та унікальний токен користувача.

Альтернативний сценарій:

* 1. ПП виявляє, що зовнішня система передала їй некоректні значення ІА-параметрів.

1.а.1 ПП видає повідомлення про помилку.

Прецедент «Пройти авторизацію»:

Передумова: виконання прецедента «Аутентифікувати користувача»

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: користувач успішно авторизується в системі.

Успішний сценарій:

1. ПП надає користувачеві ролей на виконання певних дій з БД.
2. ПП створює нову сесію для користувача.
3. ПП передає повідомлення з приводу успішної авторизації та ідентифікатор сесії.

Альтернативний сценарій:

1.1 ПП виявляє, що користувач не має ролі в системі.

1.а.1 ПП видає повідомлення про помилку.

Прецедент «Аутентифікувати користувача»:

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: користувач успішно аутентифікується для подальшого використання системи.

Успішний сценарій:

1. Зовнішня система передає ІА-параметри: швидкість набору тексту, точність набору тексту, час утримання клавіш та ідентифікатор системи.
2. ПП перевіряє ІА-параметри.
3. ПП надає зовнішній системі доступ до прецеденту «Пройти авторизацію».

Альтернативний сценарій:

2.1 ПП виявляє, що зовнішня система передала їй значення ІА-параметрів, яких немає в БД.

1.а.1 ПП видає повідомлення про помилку.

Прецедент «Вихід користувача»:

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: користувач успішно виходить з системи.

Успішний сценарій:

1. Зовнішня система передає ідентифікатор сесії користувача.
2. ПП видаляє по ідентифікатору сесію з БД.
3. ПП передає повідомлення з приводу успішного виходу з системи користувача.

Альтернативний сценарій:

2.1 Немає доступу до БД.

2.а.1 ПП видає повідомлення про помилку.

Прецедент «Видалення користувача»:

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: користувача буде успішно видалено з системи.

Успішний сценарій:

1. Зовнішня система передає токен користувача.
2. ПП видаляє по токену користувача з БД.
3. ПП передає повідомлення з приводу успішного видалення з системи користувача.

Альтернативний сценарій:

2.1 Немає доступу до БД.

2.а.1 ПП видає повідомлення про помилку.

Прецедент «Управління ролями користувачів»:

Актор-основна зацікавлена особа: Зовнішня система

У разі успішного завершення прецеденту: буде успішно змінено ролі для користувача.

Успішний сценарій:

1. Зовнішня система передає токен користувача та список нових ролей.
2. ПП зберігає нові ролі користувача в БД.
3. ПП передає повідомлення з приводу успішного змінення ролей для користувача.

Альтернативний сценарій:

2.1 Зовнішня система передає невідомі ролі для ПП.

2.а.1 ПП видає повідомлення про помилку.

1.4 Функціональні вимоги до програмного продукту

1.4.1. Багаторівнева класифікація функціональних вимог

Таблиця 1.2 – опис функцій з наданням унікальних ієрархічних ідентифікаторів

|  |  |
| --- | --- |
| Ідентифікатор  функції | Назва функції |
| **FR1** | **Реєстрація користувача** |
| FR1.1 | Передача зовнішній системі довільного тексту для набору його. |
| FR1.2 | Передача від зовнішньої системи параметрів ідентифікації та  аутентифікації користувача (Швидкість набору тексту, точність набору тексту, час утримання клавіш) та ідентифікатор системи |
| FR1.3 | Збереження користувача в БД |
| FR1.4 | Передача користувачеві повідомлення з приводу успішної реєстрації. (унікальний токен та ролі користувача) |
| **FR2** | **Аутентифікація користувача** |
| FR2.1 | Передача зовнішній системі довільного тексту для набору його. |
| FR2.2 | Передача від зовнішньої системи параметрів ідентифікації та  аутентифікації користувача  (Швидкість набору тексту, точність набору тексту, час утримання клавіш) та ідентифікатор системи |
| FR2.3 | Перевірка параметрів ідентифікації та аутентифікації користувача |
| **FR3** | **Авторизація користувача** |
| FR3.1 | Виклик функції «Аутентифікація користувача» |
| FR3.2 | Надання користувачеві ролей на виконання певних дій. |
| FR3.3 | Створення нової сесії для користувача. |
| FR3.4 | Передача зовнішній системі даних успішної авторизації. (унікальний ідентифікатор сесії) |
| **FR4** | **Видалення користувача** |
| FR4.1 | Передача від зовнішньої системи токена користувача. |
| FR4.2 | Видалення користувача по токену з БД. |
| FR4.3 | Передача зовнішній системі повідомлення з приводу успішного видалення користувача. |
| **FR5** | **Вихід з системи користувача** |
| FR5.1 | Передача від зовнішньої системи ідентифікатора сесії користувача. |
| FR5.2 | Видалення по ідентифікатору сесії користувача з БД. |
| FR5.3 | Передача зовнішній системі повідомлення з приводу успішного виходу з системи. |
| **FR6** | **Управління ролями користувача** |
| FR6.1 | Передача від зовнішньої системи токену користувача та списку нових ролей. |
| FR6.2 | Збереження нових ролей користувача в БД. |
| FR6.3 | Передача зовнішній системі повідомлення з приводу успішної зміни ролей користувача. |

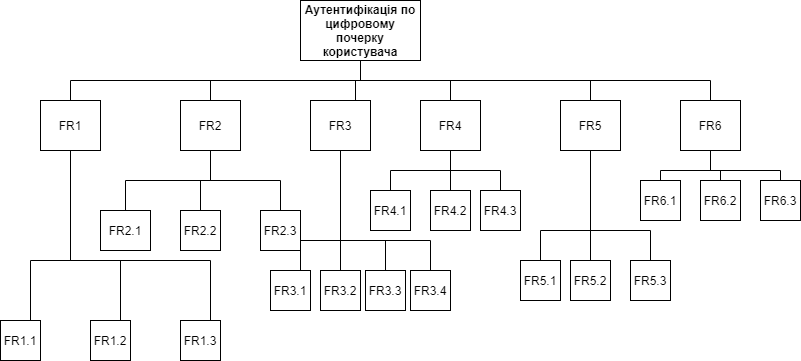
На рисунку 1.7 представлено опис ієрархічної WBS-структури багаторівневої класифікації функціональних вимог.

Рис. 1.7 - *WBS-структура*

1.4.2 Функціональний аналіз існуючих програмних продуктів

Таблиця 1.3 - функціональний аналіз існуючих програмних продуктів

|  |  |  |
| --- | --- | --- |
| Ідентифікатор  функції | Google Authentificator | Cyberarms  IDDS |
| FR1 | + | + |
| FR2 | + | + |
| FR3 | + | + |
| FR4 | - | - |
| FR5 | + | + |
| FR6 | - | - |

1.5 Нефункціональні вимоги до програмного продукту

1.5.1 Опис зовнішніх інтерфейсів

1.5.1.1 Опис інтерфейсів користувача

1.5.1.1.1 Опис INPUT-інтерфейсів користувача

Таблиця 1.4 - результати аналізу засобів INPUT-потоків

|  |  |
| --- | --- |
| Ідентифікатор  функції | Засіб INPUT-потоку |
| FR1 | Метод POST:   * typingSpeed: int (швидкість набору тексту) * accuracy: int (точність набору тексту) * holdTime: double (час утримання клавіш) * tokenSystem: String (ідентифікатор системи) * userRoles: String[] (ролі для користувача) |
| FR2, FR3 | Метод POST:   * typingSpeed: int (швидкість набору тексту) * accuracy: int (точність набору тексту) * holdTime: double (час утримання клавіш) * tokenSystem: String (ідентифікатор системи) |
| FR4 | Метод DELETE:   * tokenUser: String (ідентифікатор користувача) |
| FR5 | Метод PUT:   * sessionId: String (ідентифікатор сесії користувача) |
| FR6 | Метод PUT:   * tokenUser: String (ідентифікатор користувача) * userRoles: String[] (нові ролі для користувача) |

1.5.1.1.2 Опис OUTPUT-інтерфейсів користувача

Таблиця 1.5 - результати аналізу засобів OUTPUT -потоків

|  |  |  |
| --- | --- | --- |
| Ідентифікатор  функції | Засіб OUTPUT -  потоку | Особливості використання |
| FR1 | Успішна реєстрація.  Дані в форматі JSON |  |
| FR2, FR3 | Успішна авторизація.(токен для користувача)  Дані в форматі JSON |  |
| FR4 | Видалення користувача.  Дані в форматі JSON |  |
| FR5 | Вихід з системи.  Дані в форматі JSON |  |
| FR6 | Зміна прав доступу користувача.  Дані в форматі JSON |  |

1.5.1.2 Опис інтерфейсу із зовнішніми пристроями

Таблиця 1.6 - результат аналізу зовнішніх пристроїв

|  |  |
| --- | --- |
| Ідентифікатор  функції | Зовнішній пристрій |
| FR1, FR2, FR3, FR4, FR5, FR6 | Персональний комп`ютер, ноутбук, смартфон |

1.5.1.3 Опис програмних інтерфейсів

Для взаємодії з нашим Rest API сервісом не потрібні конкретні версії ОС та програмні бібліотеки.

1.5.1.4 Опис інтерфейсів передачі інформації

Інтерфейси передачі інформації:

* Ethernet
* Wi-Fi

1.5.1.5 Опис атрибутів продуктивності

Таблиця 1.6 - результат аналізу характеристик продуктивності

|  |  |
| --- | --- |
| Ідентифікатор  функції | Максимальний час реакції ПП на дії  користувачів, секунди |
| FR1 | 4 |
| FR2 | 2 |
| FR3 | 2 |
| FR4 | 3 |
| FR5 | 3 |
| FR6 | 3 |

**2 Планування процесу розробки програмного продукту**

**2.1 Планування ітерацій розробки програмного продукту**

З метою забезпечення вимог таких рекомендацій IEEE-стандарту, як необхідність, корисність при експлуатації, здійсненність функціональних вимог до ПП, визначено функціональні пріоритети, які будуть використані при плануванні ітерацій розробки ПП. Результати представлено в таблиці 2.1

Таблиця 2.1 – опис функціональних пріоритетів.

|  |  |  |  |
| --- | --- | --- | --- |
| Ідентифікатор  Функції | Функціональні  залежності | Вплив на  досягнення  мети, % | Пріоритет  функції |
| FR1 | - | 20% | M |
| FR2 | - | 20% | M |
| FR3 | FR2 | 20% | M |
| FR4 | - | 10% | S |
| FR5 | - | 20% | M |
| FR6 | - | 10% | S |

2.2 Концептуальний опис архітектури програмного продукту

Архітектурний тип ПП: Service Application (SA) – сервісний ПП, який виконує функції сервера застосунків, і представляє зовнішній інтерфейс за архітектурним стилем REST.

На рисунку 2.1 представлено UML-діаграму розгортання ПП на трьох рівнях (PL,BL,AL)*.*

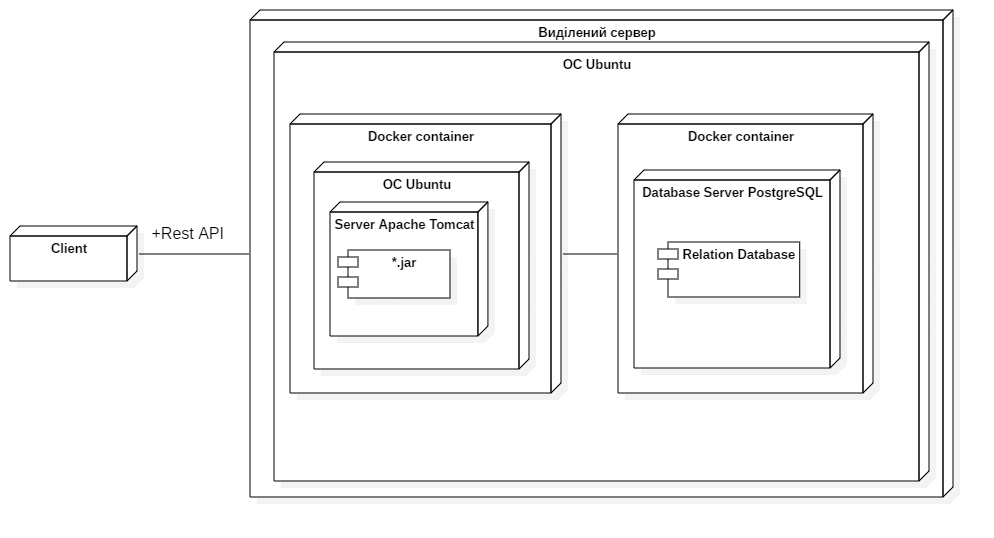


Рис. 2.1 - UML-діаграма розгортання

2.3 План розробки програмного продукту

2.3.1 Оцінка трудомісткості розробки програмного продукту

2.3.1.1 Визначення вагових показників акторів A

Таблиця 2.2 – вагові коефіцієнти акторів

|  |  |
| --- | --- |
| Тип актора | Ваговий коефіцієнт |
| Простий | 1 |
| Середній | 2 |
| Складний | 3 |

А = 2 (зовнішня система, яка взаємодіє з нашим Rest API сервісом)

2.3.1.2 Визначення вагових показників прецедентів UC

Таблиця 2.3 - Вагові коефіцієнти прецедентів

|  |  |  |
| --- | --- | --- |
| Тип прецеденту | Кількість кроків сценарію | Ваговий коефіцієнт |
| Простий | <= 3 | 5 |
| Середній | 4-7 | 10 |
| Складний | > 7 | 15 |

Простий: 3

Середній: 3

UC = 3 \* 5 + 3 \* 10 = 45

UUCP=A+UC = 2 + 45 = 47

2.3.1.3 Визначення технічної складності проекту

Таблиця 2.4 - показники технічної складності проекту TCF та оцінки ST

|  |  |  |  |
| --- | --- | --- | --- |
| Показник | Опис показника | Вага | ST |
| Т1 | Розподілена система | 2 | 2 |
| Т2 | Висока продуктивність (пропускна здатність) | 1 | 3 |
| T3 | Робота користувачів в режимі онлайн | 1 | 5 |
| Т4 | Складна обробка даних | -1 | 2 |
| Т5 | Повторне використання коду | 1 | 1 |
| Т6 | Простота установки ПП | 0,5 | 5 |
| Т7 | Простота використання ПП | 0,5 | 5 |
| Т8 | Переносимість ПП | 2 | 5 |
| Т9 | Простота внесення змін до функціоналу | 1 | 5 |
| T10 | Паралелізм | 1 | 0 |
| T11 | Спеціальні вимоги до безпеки | 1 | 5 |
| T12 | Безпосередній доступ до системи з боку зовнішніх користувачів | 1 | 5 |
| T13 | Спеціальні вимоги до навчання користувачів | 1 | 0 |

TCF = 0,6 + (0,01 \* 43) = 1,03

2.3.1.4 Визначення рівня кваліфікації розробників

Таблиця 2.5 - показники рівня кваліфікації розробників та оцінки SF

|  |  |  |  |
| --- | --- | --- | --- |
| Показник | Опис показника | Вага | SF |
| F1 | Знайомство з технологією | 1,5 | 3 |
| F2 | Досвід розробки ПП | 0,5 | 0 |
| F3 | Досвід використання об'єктно-орієнтованого підходу 1 | 1 | 5 |
| F4 | Наявність провідного аналітика | 0,5 | 0 |
| F5 | Мотивація | 1 | 3 |
| F6 | Стабільність вимог до ПП | 2 | 5 |
| F7 | Часткова зайнятість | -1 | 3 |
| F8 | Складні мови програмування | -1 | 5 |

EF = 1,4 + (- 0,03 \* 14,5) = 1,11

2.3.1.5 Остаточне значення UCP (Use Case Points)

UCP = UUCP \* TCF \* EF = 47 \* 1,03 \* 1,11 = 49,52

2.3.1.6 Оцінка трудомісткості проекту

Розрахункове зусилля: 49 \* 28 = 1372 люд.-год

2.3.2 Визначення дерева робіт з розробки програмного продукту

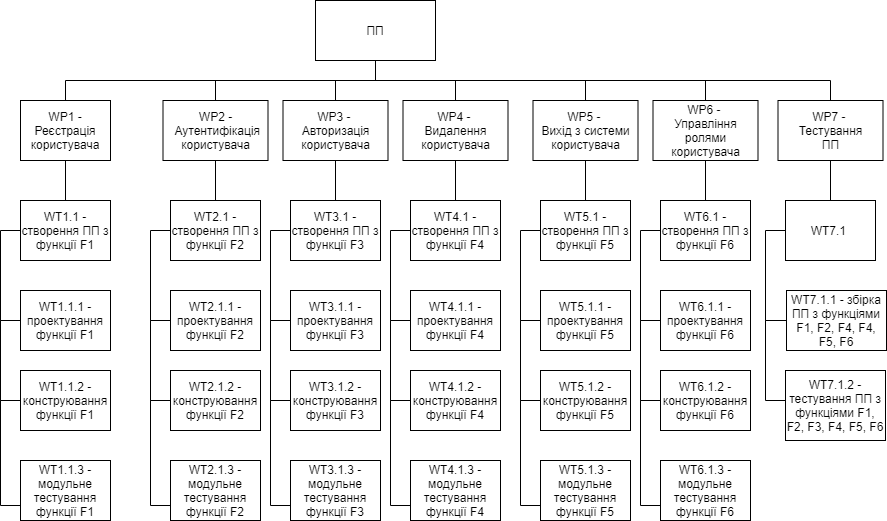
При створенні дерева робіт (Work BreakDown Structure- WBS) використовується дерево функцій, яке було створено раніше.

Кожна функція 1-го рівня ієрархії перетворюється в Work Package (WP)

Кожна функція 2-го рівня ієрархії перетворюється в Work Task (WT).

Для кожної задачі визначаються підзадачі - Work SubTask (WST) з урахуванням базових процесів розробки програмних модулів: проектування, конструювання, модульне тестування, збірка та системне тестування.

На рисунку 2.2 представлено опис ієрархічної WBS-структури дерева робіт.

Рис. 2.2 - *WBS-структура*

Для кожної підзадачі визначається виконавець, що фіксується у вигляді таблиці, що представлено в таблиці 2.6.

Таблиця 2.6 - опис підзадач із закріпленням виконавців

|  |  |
| --- | --- |
| Підзадача | Виконавець |
| WST1.1.1 | Таужнянський А.В. |
| WST1.1.2 | Таужнянський А.В. |
| WST1.1.3 | Таужнянський А.В. |
| WST2.1.1 | Макаренко А.О. |
| WST2.1.2 | Макаренко А.О. |
| WST2.1.3 | Макаренко А.О. |
| WST3.1.1 | Таужнянський А.В. |
| WST3.1.2 | Таужнянський А.В. |
| WST3.1.3 | Таужнянський А.В. |
| WST4.1.1 | Макаренко А.О. |
| WST4.1.2 | Макаренко А.О. |
| WST4.1.3 | Макаренко А.О. |
| WST5.1.1 | Таужнянський А.В. |
| WST5.1.2 | Таужнянський А.В. |
| WST5.1.3 | Таужнянський А.В. |
| WST6.1.1 | Макаренко А.О. |
| WST6.1.2 | Макаренко А.О. |
| WST6.1.3 | Макаренко А.О. |
| WST7.1.1 | Таужнянський А.В. |
| WST7.1.2 | Таужнянський А.В. |

2.3.3 Графік робіт з розробки програмного продукту

2.3.3.1 Таблиця з графіком робіт

На рисунку 2.3 представлено таблицю з графіком робіт.

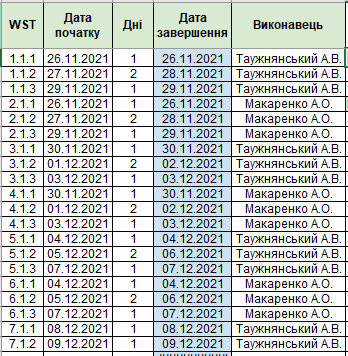


Рис. 2.3 – Таблиця графіка робіт

2.3.3.2 Діаграма Ганта

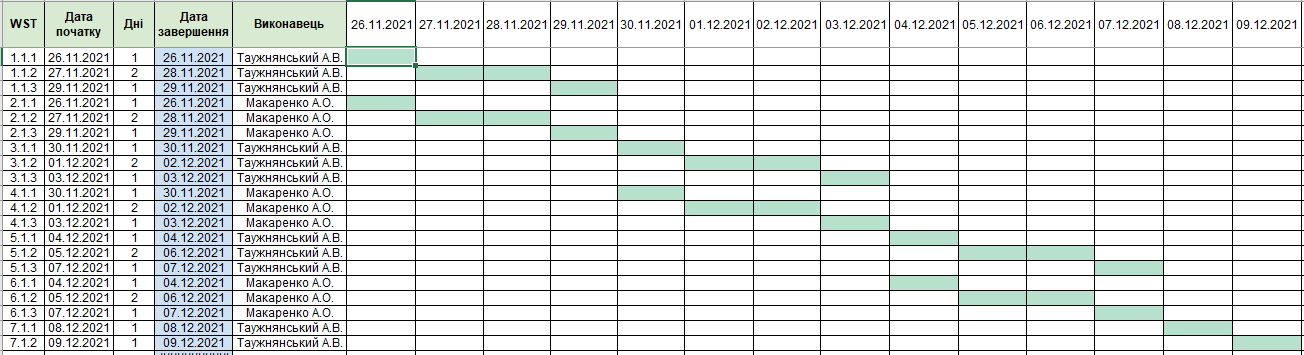
На рисунку 2.4 представлена Діаграма Ганта.

Рис. 2.5 – Діаграма Ганта

**Висновки**

З кожним днем концепція двофакторної аутентифікації втрачає свою популярність і релевантність. Багато досліджень, в тому числі і наша анкета показує незадовільні відгуки про проходження двофакторної аутентифікації. Багато користувачів незадоволені, що доводиться запам'ятовувати кілька різних паролів. Хтось скаржиться на сильно довгий час очікування процесу аутентифікації. Деякі не згодні, що треба ділитися своїми персональними даними, наприклад номером телефону або email-адресою.

Тому, задача цієї курсової роботи полягає у тому, щоб створити програмний продукт, який забезпечить безпечну і водночас швидку аутентифікацію.

Для реалізації зазначеної мети у першому розділі був проведений глибокий аналіз проблеми користувача. Були визначені потреби користувача та вимоги до програмного продукту. Також були описані інтерфейси та атрибути продукту.

У другому розділі був спланований процес розробки програмного продукту. Були виділені ітерації та архітектура продукту, далі був визначений план розробки, та розподілені обов’язки розробників. Також представлений графік робіт розробки програмного продукту.

Ми вважаємо, що мета цієї курсової роботи була досягнута нами на 100%.