**Software Requirements Specification for**

**<** **Програмна система для планування та моніторингу виконання особистих задач і досягнень. Серверна частина>**

**Version 1.1 approved**

**Prepared by <Анастасія Червенко>**

**<ПЗПІ-21-1>**

**<03.06.2025>**

**ЗМІСТ**

[ВСТУП 3](#_Toc200110967)

[1 Вимоги предметної галузі 4](#_Toc200110968)

[1.1 Актуальність теми 4](#_Toc200110969)

[1.2 Формування завдання з урахуванням потреб користувачів 5](#_Toc200110970)

[2 Формування вимог до програмної системи 6](#_Toc200110971)

[2.1 Постановка мети 6](#_Toc200110972)

[2.2 Загальний опис 6](#_Toc200110973)

[2.3 Загальні обмеження 8](#_Toc200110974)

[2.4 Припущення та залежності 8](#_Toc200110975)

# **ВСТУП**

У сучасному світі, що стрімко розвивається завдяки технологічному прогресу та соціальним трансформаціям, дедалі більше людей прагнуть до саморозвитку й особистісного зростання. Це дозволяє їм залишатися конкурентоспроможними, адаптуватися до викликів сьогодення та досягати гармонії між особистими цілями й темпом життя. У динамічному ритмі сучасності планування стало необхідним етапом на шляху до досягнення будь-якої мети. Водночас після її постановки вкрай важливо систематично відстежувати рівень власного прогресу, щоб підтримувати мотивацію та впевнено рухатись до результату.

Однак збереження стабільної мотивації, ефективне планування задач і контроль за власним розвитком є складним аналітичним процесом, що потребує якісних інструментів підтримки. У разі їх відсутності процес досягнення мети може бути суттєво ускладнений або навіть виявитися неможливим.

Основна мета програмної системи «Naviria» – надати користувачам інструмент для структурованого формування особистих цілей, розбиття їх на досяжні задачі, розвитку корисних звичок, відстеження досягнень і збереження мотивації у довгостроковій перспективі. Система поєднує логіку конструктора різних типів задач і підзадач, моніторинг прогресу у виконанні завдань, досягненні цілей, розвитку навичок, а також персоналізовану статистику, що дозволяє краще розуміти динаміку власного зростання.

З урахуванням важливості емоційної підтримки та залученості, «Naviria» впроваджує елементи гейміфікації, що включають систему рівнів користувачів, особисті досягнення, підтримку з боку інших учасників через соціальні механіки. Додатково вбудований персональний AI-асистент надає допомогу у формуванні задач, структуризації планів та відповідає на супутні запити користувача, що робить роботу із системою ще ефективнішою та зручнішою.

# **АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ**

* 1. Актуальність теми

У сучасному інформаційно насиченому середовищі, де високий темп життя поєднується з постійною потребою до адаптації, цифрові платформи відіграють дедалі важливішу роль у підтримці особистісного зростання. Люди все частіше звертаються до технологій не лише як до інструментів продуктивності, а як до засобів для досягнення глибших цілей – формування звичок, розвиток навичок і збереження внутрішньої мотивації. У цьому контексті особливо актуальними стають системи, що інтегрують гейміфікацію, персоналізацію, планування й моніторинг діяльності.

Гейміфіковані платформи, що включають елементи винагород, рівнів, досягнень і викликів, не лише підвищують мотивацію користувачів, а й сприяють розвитку когнітивних і соціальних компетенцій. Завдяки вбудованим механізмам зворотного зв’язку та статистики, користувачі отримують чітке уявлення про свій прогрес, що стимулює їх до подальших дій. Не менш важливою є персоналізація: адаптація платформи до стилю мислення, звичок і ритму життя користувача значно підвищує її ефективність і гнучкість.

Водночас, відсутність належних інструментів цифрового самоконтролю, призводить до частих відволікань, втрати концентрації й зниження продуктивності. Натомість навіть базові інструменти візуалізації прогресу та блокування відволікаючих факторів здатні істотно підвищити якість виконання завдань. Це особливо актуально для систем, що мають на меті формування звичок та підтримку стабільної динаміки у досягненні цілей. Ключову роль у цьому процесі відіграють саме компоненти планування та моніторингу. Залучення до планування власної діяльності, фіксації цілей та саморефлексії значно покращує навчальні результати. Отже здатність до попереднього планування та моніторингу є визначальною при вирішенні складних завдань.

Таким чином поєднання гейміфікації (через досягнення, рівні, винагороди), персоналізації (через адаптацію задач до вимог користувача), планування (структуроване розбиття цілей на задачі) та моніторингу (візуалізація прогресу, аналітика, нагадування) дозволить створити платформу, яка не просто фіксує дії користувача, а формує в нього сталі навички саморегуляції, надає мотивацію та забезпечує сталий прогрес

* 1. Формування завдання з урахуванням потреб користувачів

Для задоволення потреб цільової аудиторії (користувачів, які прагнуть до особистісного розвитку, досягнення цілей та підтримки мотивації), необхідно розробити базу даних та серверну частину програмної системи «Naviria». Back-end частина повинна реалізовувати взаємодію з базою даних, забезпечити ефективну взаємодію з клієнтською частиною через REST API, підтримувати гнучке управління цілями та завданнями, зберігати і обробляти прогрес користувачів, формувати аналітичні звіти, реалізовувати механізми гейміфікації та підтримувати елементи соціальної взаємодії.

Результатом проєкту має стати розроблена база даних та серверна частина застосунку «Naviria», яка ефективно реалізовує бізнес-логіку системи, відповідає вимогам продуктивності, безпеки й гнучкості, та закладає основу для подальшого розширення функціоналу. Очікується, що система буде конкурентоспроможною, зручною для інтеграції, придатною для навантаження з боку великої кількості користувачів і здатною забезпечити сучасний мотиваційний досвід.

# **ФОРМУВАННЯ ВИМОГ ДО ПРОГРАМНОЇ СИСТЕМИ**

* 1. Постановка мети

Метою виконання роботи є розробка серверної частини програмної системи «Naviria» для відстеження своїх досягнень або задач.

У серверній частині системи повинна бути реалізована ключова функціональність, що забезпечує персоналізоване управління цілями та задачами користувачів. До основних механік належать створення ієрархічних цілей із підзадачами, планування дедлайнів, категоризація завдань, а також формування гейміфікованих елементів – таких як рівні, досягнення, таблиці лідерів і віртуальні нагороди. Важливою частиною є реалізація аналітичного модуля, який дозволяє генерувати статистичні звіти про прогрес користувача, а також підтримка інтерактивної взаємодії з іншими користувачами через систему друзів, спільних цілей і мотиваційного чату. Також система має підтримувати взаємодію з персональним AI-асистентом для генерації задач та допомоги користувачам з особистими питаннями. Особливу увагу необхідно приділити масштабованості, стабільності та безпеці серверної частини, щоб забезпечити безперервну роботу системи в умовах високого навантаження.

* 1. Загальний опис

У програмній системі мають бути реалізовані наступні функціональні вимоги:

* + - реєстрація нових користувачів системи;
    - авторизація та автентифікація зареєстрованих користувачів системи;
    - перегляд та редагування персонального профілю користувача;
    - додавання, редагування та видалення персональних задач користувача;
    - категоризація та систематизація ієрархії завдань по папкам та типізація підзадач для кожної задачі;
    - графічна демонстрація аналітичних даних з відстеження процесу виконання завдань та результатів досягнених цілей;
    - гейміфікація процесу виконання завдань (система рівнів, балів, таблиця лідерів);
    - сповіщення про дедлайни, досягнення рівнів, отримання досягнень, підтримку від друзів, тощо;
    - перегляд акаунтів інших користувачів, додавання їх у друзі, підтримка одне-одного мотивуючими цитатами;
    - чат з персональним асистентом (штучним інтелектом).

Окрему увагу під час проєктування було приділено нефункціональним вимогам, що охоплюють аспекти масштабованості, безпеки, сумісності з різними пристроями та загальної якості роботи системи.

* + - система повинна підтримувати роботу з великою кількістю одночасно активних користувачів без помітного зниження продуктивності;
    - архітектура має бути легко масштабованою за потреби;
    - час відгуку сервера при обробці запитів не повинен перевищувати 2 секунд у більшості стандартних операцій;
    - автентифікація користувачів повинна базуватися на сучасних протоколах (JWT);
    - усі конфіденційні дані повинні зберігатися у базі даних лише у зашифрованому вигляді;
    - база даних має зберігатися у хмарному сховищі з автоматичним резервним копіюванням щонайменше раз на тиждень;
    - усі критичні помилки серверу, збої авторизації та системні винятки повинні логуватися для подальшої діагностики й аналізу;
    - застосунок повинен коректно функціонувати як у браузері, так і у мобільній версії;
    - фотографії профілів користувачів повинні зберігатися у зовнішньому хмарному сховищі (Cloudinary), без навантаження на сервер;
    - застосунок має бути готовим до розширення API для інтеграції з іншими сервісами.
  1. Загальні обмеження

Серверна частина програмної системи «Naviria» реалізується з використанням фреймворку ASP.NET Core (версія 8.0) мовою програмування C#. Для зберігання даних використовується MongoDB як нереляційна документно-орієнтована база даних, розміщена у хмарному середовищі (MongoDB Atlas). Для інтеграції AI-асистента використовується OpenAI GPT API, який дозволяє отримувати відповіді від штучного інтелекту. Виходячи з використання наведених технологій було визначено мінімальні системні вимоги:

* + - процесор: Intel Core i3-8100 або AMD Ryzen 3 3200G;
    - оперативна пам’ять: 2–4 GB RAM;
    - місце на диску: 10 GB SSD (для кешу, журналів, тимчасових файлів;
    - операційна система: Windows 10/11 x64 або Ubuntu Server 22.04 LTS;
    - .NET SDK: .NET 8.0 (ASP.NET Core);
    - стабільне підключення до Інтернету для обробки зовнішніх API-запитів.

Зазначені вимоги розраховані на підтримку до 500 активних користувачів. У разі зростання навантаження, система має бути адаптована шляхом збільшення ресурсів та масштабування.

* 1. Припущення та залежності

Для забезпечення коректної розробки та оцінки функціональності системи було сформульовано низку припущень, які відображають очікувані умови експлуатації та поведінку користувачів:

* + - система вимагає постійного підключення до Інтернету для коректної роботи більшості функцій, включаючи обмін із сервером, сповіщення та взаємодію з AI-асистентом;
    - користувач не може створювати або зберігати цілі, задачі та прогрес, якщо не зареєстрований та не автентифікований у системі;
    - AI-асистент не генерує завдання або поради, якщо користувач не надав базову інформацію про свої цілі та уподобання;
    - система не дозволяє отримувати досягнення або підвищення рівня, якщо задачі було виконано з порушенням дедлайнів;
    - у разі втрати підключення до Інтернету під час взаємодії з сервером, зміни не будуть збережені до моменту повторного підключення.

Окрім внутрішньої логіки системи, програмна система «Naviria» залежить від низки зовнішніх сервісів і технологій, без яких її повноцінне функціонування неможливе. Нижче наведено ключові зовнішні залежності, які враховуються під час проєктування та розгортання серверної частини.

* + - якщо користувач видаляє свій акаунт, усі пов’язані дані (досягнення, його задачі та підзадачі, історія взаємодій, тощо) видаляються без можливості відновлення;
    - якщо користувач перевищив встановлену кількість запитів до AI-асистента, чат оновлюється і попередні повідомлення автоматично видаляються;
    - у разі збоїв або перевантаження сервера можливе тимчасове обмеження в обробці сповіщень і оновлень у реальному часі;
    - якщо автентифікація через сторонній сервіс (Google OAuth) недоступна, користувач може пройти лише базову авторизацію через електрону адресу поштової скриньки та пароль.