1. Вступ
   1. Призначення

Ця специфікація описує функціональні та нефункціональні вимоги до програмної системи для пошуку серверів популярної комп’ютерної гри Minecraft. Документ орієнтований на розробників, тестувальників, адміністраторів системи, а також інших зацікавлених осіб, що беруть участь у розгортанні та підтримці продукту. Метою документу є формалізація вимог до програмного забезпечення задля забезпечення його коректної реалізації, перевірки та подальшої підтримки.

* 1. Сфера застосування

Програмна система розробляється як засіб для пошуку, налаштування та запуску збірок модифікацій Minecraft, сумісних між собою та з відповідними ігровими серверами. Рішення орієнтоване як на новачків, що не мають технічних навичок, так і на досвідчених гравців, які хочуть створювати та поширювати власні збірки. Система також забезпечує синхронізацію клієнтської та серверної частин гри для багатокористувацького середовища.

У застосуванні програми передбачається участь таких категорій користувачів

* гравці-початківці;
* досвідчені користувачі;
* адміністратори приватних серверів;
* користувачі, що змінюють пристрої;
* соціально активні гравці.

Система включає веб-інтерфейс (на Razor Pages) для керування обліковими записами та перегляду контенту, серверну частину (на ASP.NET) з REST API та настільний лаунчер (на Avalonia UI) для встановлення та запуску збірок.

* 1. Визначення, абревіатури та скорочення

API – Application Programming Interface (інтерфейс прикладного програмування)

JWT – JSON Web Token, формат маркерів для авторизації

Modpack (збірка) – структурований набір сумісних модифікацій для Minecraft

Mod (модифікація) – сторонній компонент, що розширює або змінює функціонал гри

Launcher (лаунчер) – окремий настільний застосунок для запуску модифікованої гри

Forge, Fabric – найпоширеніші ядра для роботи з модами у Minecraft

REST – архітектурний стиль для створення веб-сервісів, які працюють через HTTP

EF6 – Entity Framework 6, ORM для .NET

Razor Pages – підхід до розробки вебінтерфейсів у ASP.NET

Avalonia UI – кросплатформенний фреймворк для створення настільних застосунків

* 1. Огляд документу

Цей документ структуровано відповідно до стандарту IEEE 830–1993. У розділі 2 подано загальний опис системи, її контекст, функціональність та обмеження. У розділі 3 деталізуються специфічні вимоги – функціональні, інтерфейсні та нефункціональні. Додатки містять графічні моделі, структури даних, приклади сценаріїв взаємодії користувача із системою, а також інші допоміжні матеріали.

* 1. Конвенції читання та посилань

Для опису взаємодії користувача з системою використовується стандартна UML-нотація. Всі приклади структур бази даних, API або форм взаємодії наведені у спрощеному вигляді для фокусування на логіці роботи системи.

1. Загальний опис
   1. Перспективи продукту

Програмна система є незалежним продуктом, що доповнює ігровий клієнт Minecraft, забезпечуючи зручне керування модифікаціями та ігровими збірками. Вона не замінює офіційний лаунчер гри, а розширює його функціональність у напрямку гнучкого налаштування середовища гри та організації мережевої взаємодії.

Основними перевагами продукту є:

* зниження технічного бар’єру для новачків;
* централізоване зберігання і перевірка збірок;
* інтеграція з серверною частиною гри;
* підтримка персоналізації та багаторазового використання профілів.

Програмна система передбачає інтеграцію з зовнішніми платформами (CurseForge, Modrinth), що дозволяє імпортувати наявні збірки користувача без втрати прогресу.

* 1. Функціональність продукту

Система реалізує наступну функціональність:

* реєстрація та автентифікація користувача;
* перегляд, фільтрація та пошук збірок;
* завантаження та встановлення збірок через лаунчер;
* створення, редагування та поширення власних збірок;
* перевірка сумісності модифікацій;
* прив’язка збірок до серверів;
* імпорт збірок із зовнішніх джерел;
* збереження даних користувача (історія, налаштування, улюблене);
* автоматичне оновлення клієнта та модифікацій.

Усі функції доступні через інтуїтивно зрозумілий інтерфейс, що адаптується під досвід користувача.

* 1. Користувацькі характеристики

Цільова аудиторія системи охоплює:

* новачків, що хочуть встановлювати модифікації без технічних знань;
* досвідчених гравців, які створюють і тестують власні збірки;
* адміністраторів приватних серверів, які потребують уніфікованого середовища;
* гравців, що працюють з кількох пристроїв;
* спільноти, які грають на спільних збірках.

Система має забезпечувати простоту інтерфейсу, зрозумілий процес налаштування ігрового середовища, мінімальну кількість кроків для запуску гри та можливість ділитися результатами з іншими користувачами.

* 1. Обмеження
* підтримується лише Java-видання гри Minecraft;
* платформи: Windows та Linux (x64); підтримка macOS у перспективі;
* установка модифікацій виконується лише через систему, сторонні методи не передбачено;
* веб-інтерфейс не реалізує повну адміністративну панель;
* система потребує стабільного підключення до Інтернету для авторизації, завантаження і синхронізації даних;
* усі збірки зберігаються централізовано й поширюються виключно через бекенд.
  1. Припущення та залежності
* користувач має стабільне з’єднання з Інтернетом;
* усі модифікації мають відкриту або умовно відкриту ліцензію;
* більшість користувачів працює в межах одного облікового запису;
* користувач володіє базовими навичками встановлення ігор та використання лаунчерів;
* автори збірок не вставляють шкідливий код.
  1. Майбутні вдосконалення:

У перспективі передбачається:

* підтримка macOS;
* реалізація адміністративної панелі в межах вебінтерфейсу;
* автоматичне створення серверної частини збірки;
* розширення соціальної взаємодії між користувачами (обговорення, групи, підписки);
* підтримка кастомних правил сумісності модифікацій.

1. Специфічні вимоги
   1. Функціональні вимоги
      1. Реєстрація та автентифікація користувача

* користувач має змогу створити обліковий запис, вказавши нікнейм, пошту та пароль;
* підтвердження реєстрації здійснюється через електронну пошту;
* вхід у систему можливий лише після підтвердження пошти;
* у разі втрати доступу користувач може ініціювати процедуру скидання паролю з підтвердженням через email;
* сесія авторизації реалізується через JWT-токени зі строком дії.
  + 1. Перегляд і пошук збірок
* користувач має доступ до каталогу збірок;
* реалізовано фільтрацію за версією гри, кількістю модифікацій, рейтингом, датою оновлення;
* для кожної збірки відображається опис, скріншоти, список модів, автор, статистика завантажень;
* передбачено сортування за популярністю, новизною та алфавітом.
  + 1. Завантаження та встановлення збірок
* користувач може завантажити лаунчер з веб-інтерфейсу;
* через лаунчер користувач обирає збірку та запускає автоматичне встановлення;
* система перевіряє наявність необхідних файлів гри та сумісність версій;
* підтримується збереження кількох профілів зі збірками.
  + 1. Створення і публікація збірок
* авторизований користувач може створювати власні збірки;
* збірка містить назву, опис, список модифікацій, скріншоти;
* система перевіряє сумісність модифікацій у збірці перед публікацією;
* публікація можлива як у публічному, так і в приватному режимі;
* передбачено оновлення та версійність збірки.
  + 1. Інтеграція з серверною частиною
* користувач може прив’язати збірку до серверної версії;
* система перевіряє відповідність клієнта й сервера за версіями модифікацій;
* є можливість завантажити готову серверну частину відповідно до обраної збірки.
  + 1. Імпорт зовнішніх збірок
* підтримується імпорт збірок з CurseForge, Modrinth або локального архіву;
* користувач обирає джерело, система здійснює автоматичний аналіз та адаптацію структури.
  + 1. Управління профілем
* користувач має змогу редагувати дані облікового запису (нікнейм, опис);
* доступний список власних, встановлених і уподобаних збірок;
* підтримується перегляд статистики використання та взаємодії з контентом.
  + 1. Адміністрування
* адміністрація має доступ до модерації контенту: видалення збірок, блокування користувачів;
* адміністратор керує доступними тегами, версіями ядра, категоріями модифікацій.
  1. Інтерфейсні вимоги
     1. Інтерфейс користувача
* вебінтерфейс (Razor Pages): адаптивна верстка, фільтрація, перегляд збірок, керування профілем;
* настільний лаунчер (Avalonia UI): установка збірок, валідація, запуск гри, створення збірок;
* підтримка світлої і темної теми, багатомовність (українська, англійська);
* доступність: підтримка навігації з клавіатури, валідація форм у реальному часі, інструкції без технічного жаргону.
  + 1. Інтерфейс апаратного забезпечення
* система не взаємодіє безпосередньо з апаратними компонентами, окрім файлової системи;
* лаунчер взаємодіє з директорією Minecraft, не змінюючи конфігурацію ОС.
  + 1. Інтерфейс програмного забезпечення
* централізований REST API (JSON, ASP.NET Core);
* автентифікація через JWT;
* версіювання API;
* обмеження доступу за ролями (гравець, автор, адміністратор);
* кешування рідко змінюваних запитів (наприклад, публічних збірок);
* логування запитів для аналізу збоїв.
  + 1. Комунікаційний інтерфейс
* взаємодія між компонентами реалізована через HTTPS;
* клієнт (веб та лаунчер) взаємодіє з сервером через REST API;
* система надсилає email-повідомлення через SMTP (підтвердження реєстрації, скидання паролю тощо).
  1. Нефункціональні вимоги
     1. Продуктивність
* вебінтерфейс повинен реагувати на дії користувача не довше ніж 300 мс;
* середній час встановлення збірки – до 10 с при нормальному з’єднанні;
* запуск гри з лаунчера – не більше 5 кліків.
  + 1. Безпека
* шифрування паролів (хешування);
* передача даних лише через HTTPS;
* токени авторизації мають обмежений термін дії;
* захист від підбору паролів (ліміт спроб входу);
* відокремлення доступу до файлів і сервісів.
  + 1. Надійність та доступність
* система повинна працювати без збоїв упродовж тривалого часу;
* передбачена стійкість до втрати з’єднання та повторення критичних запитів;
* сервіс email-розсилок працює незалежно від основного API.
  + 1. Підтримуваність та масштабованість
* модульна архітектура (n-tier);
* слабкий зв’язок між компонентами;
* підтримка CI/CD;
* можливість додавання нових джерел модів без змін у клієнті.
  + 1. Портативність
* підтримка Windows та Linux (x64);
* у перспективі – підтримка macOS;
* інсталяція лаунчера не вимагає прав адміністратора.
  1. Вимоги до бази даних
* використовується реляційна БД, з підтримкою зв’язків один-до-багатьох і багато-до-багатьох;
* реалізація через Entity Framework 6;
* структура охоплює таблиці: Users, Roles, Mods, ModVersions, Modpacks, ModpackVersions, Tokens, Localization тощо;
* підтримка багатомовності через окремі таблиці локалізації;
* забезпечена валідація обов’язкових полів і правильності форматів.
  1. Вимоги до конфігурації та установки
* веб-інтерфейс розгортається як ASP.NET вебсайт на сервері;
* API і база даних – окремі процеси з можливістю хостингу на різних серверах;
* лаунчер постачається у вигляді готового інсталятора;
* конфігурація виконується через описові JSON-файли.
  1. Вимоги до локалізації
* інтерфейс користувача повністю локалізується (мінімум – українська та англійська);
* усі текстові елементи, включно з назвами модів та збірок, підтримують мультимовність;
* мова визначається автоматично або встановлюється вручну.
  1. Вимоги до доступності
* повна навігація з клавіатури;
* підтримка екранних рідерів;
* підказки для введення даних і попередження про помилки у реальному часі;
* уникнення технічних термінів у повідомленнях для користувача.

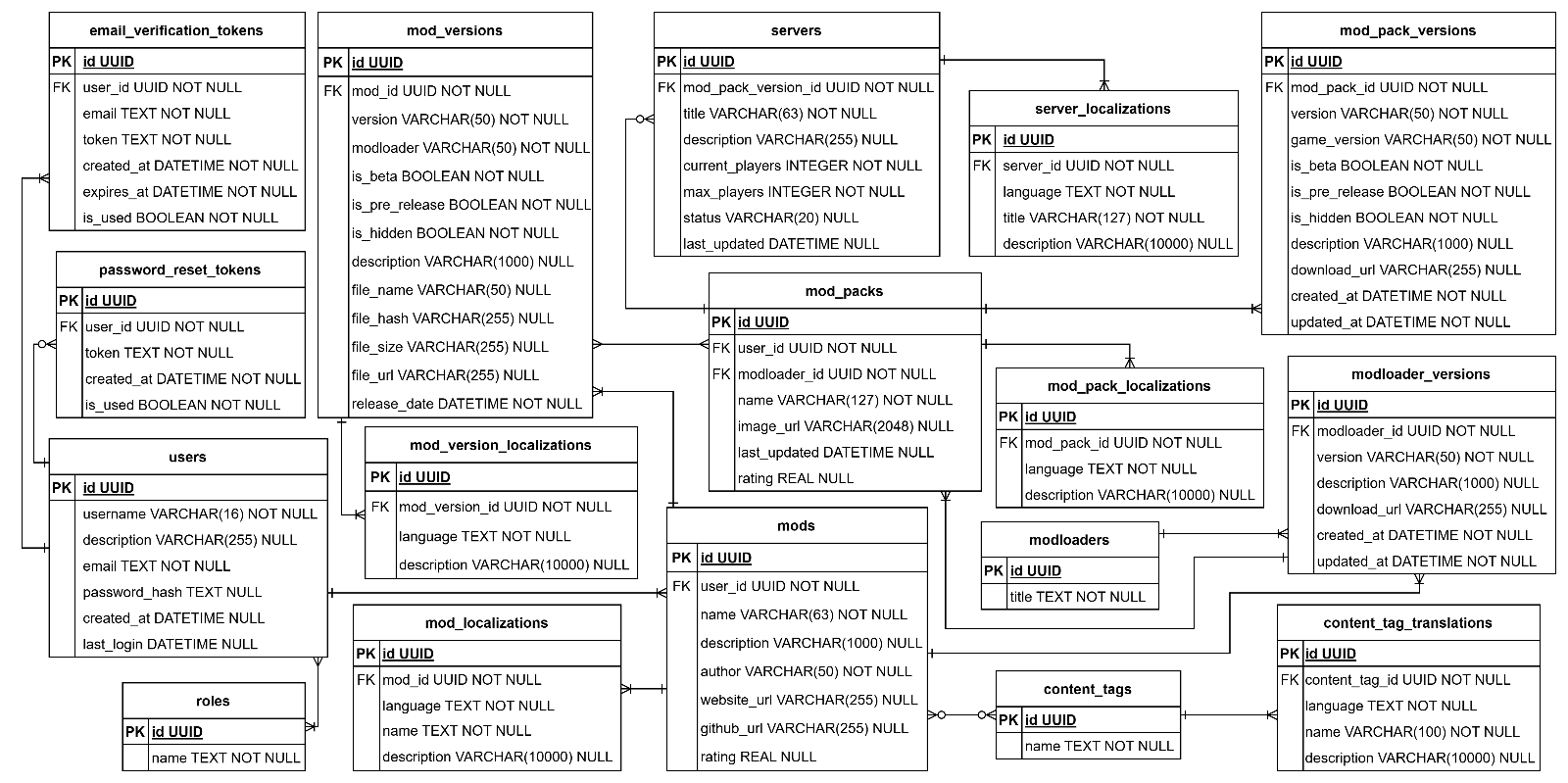
1. Додатки

Цей розділ містить допоміжні матеріали, які уточнюють або доповнюють основну частину специфікації. Графічні елементи (діаграми, приклади структури даних, фрагменти інтерфейсу) представлені у вигляді текстових вказівок.

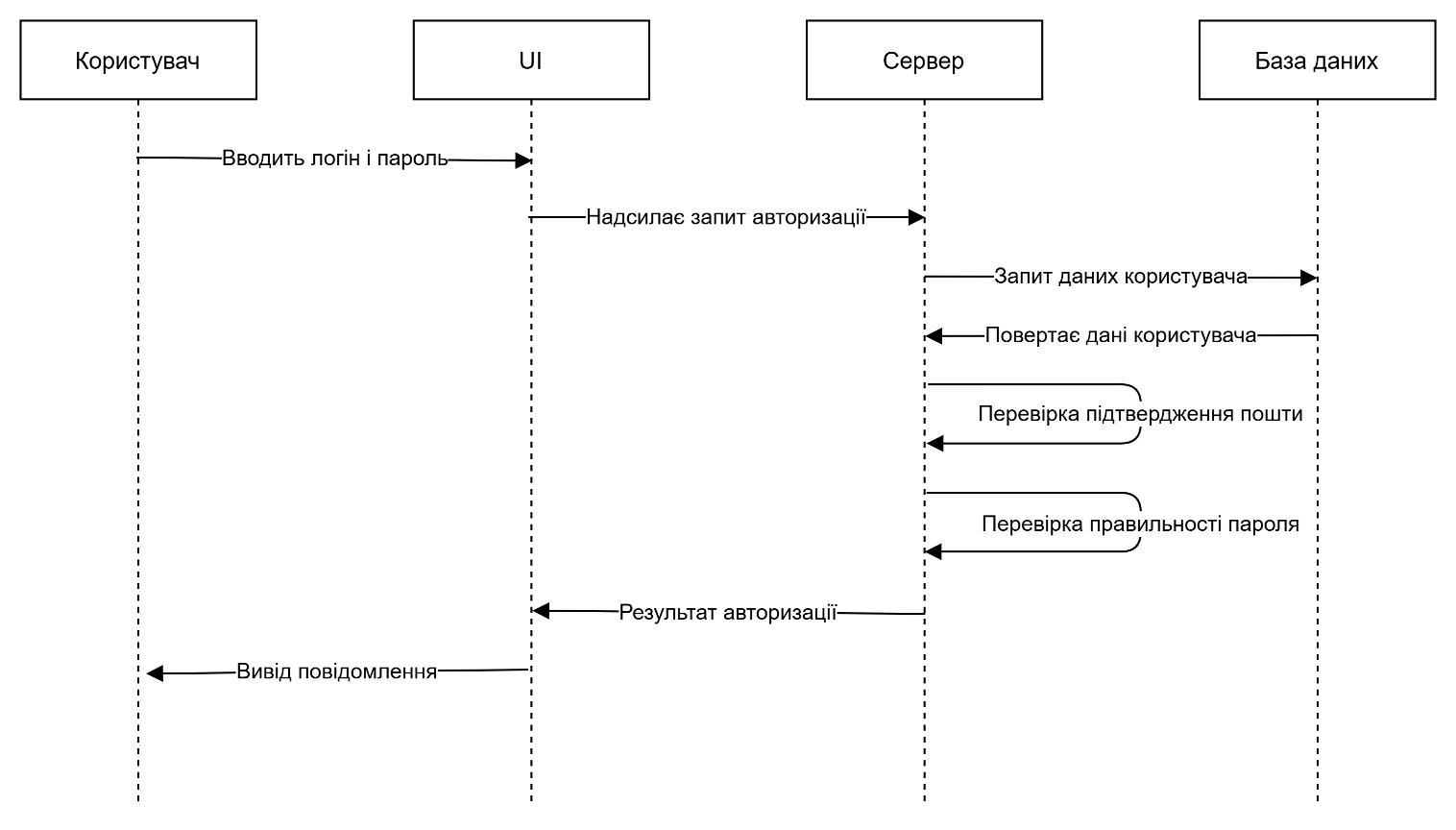
* 1. UML-діаграми
     1. Діаграма варіантів використання (Use Case Diagram)
* відображає основні сценарії взаємодії користувачів із системою: гість, зареєстрований гравець, адміністратор;
* демонструє доступні функції для кожної ролі;
* містить системні дії, що виконуються автоматично (наприклад, надсилання email-повідомлень).

<<діаграма варіантів використання>>

* + 1. Діаграма сутностей (ER Diagram)
* моделює реляційну базу даних системи: сутності (користувачі, моди, збірки, версії), атрибути, зв’язки;
* реалізовано таблиці зв’язків для підтримки багато-до-багатьох між модами та збірками;
* охоплює елементи безпеки (токени підтвердження, скидання пароля).

  
Рисунок 1 – діаграма сутностей

* + 1. Діаграма послідовностей (Sequence Diagram)
* демонструє процес автентифікації користувача: запит до API, перевірка логіну та пароля, видача токена;
* описує також обробку винятків (відсутність користувача, непідтверджена пошта, хибний пароль).

  
Рисунок 2 – діграма послідовностей процесу автентифікації

* 1. Типові сценарії використання

1. Гість заходить на вебсайт

→ переглядає доступні збірки

→ створює акаунт

→ підтверджує пошту

→ завантажує лаунчер

1. Зареєстрований гравець

→ входить у систему

→ переглядає каталог

→ фільтрує збірки

→ встановлює обрану

→ запускає гру

1. Автор збірки

→ створює нову збірку

→ додає модифікації

→ перевіряє сумісність

→ завантажує скріншоти

→ публікує або робить приватною

1. Адміністратор

→ переглядає всі збірки

→ змінює теги

→ блокує неприйнятний вміст

→ скидає пароль користувачу

* 1. Структура API

Метод: POST /api/email/send

Опис: надсилання електронного листа

Параметри:

* receiver (одержувач, ім'я або ідентифікатор);
* address (email-адреса одержувача);
* subject (тема листа);
* body (текст листа).

Відповідь: код успіху або повідомлення про помилку

Метод: GET /api/modpack

Опис: отримання списку всіх модпаків

Параметри: відсутні

Відповідь: список об'єктів ModPackResponse або повідомлення про помилку

Метод: GET /api/modpack/{id}

Опис: отримання інформації про конкретний модпак за його ідентифікатором

Параметри:

* id (Guid) — ідентифікатор модпаку

Відповідь: об'єкт ModPackResponse або повідомлення про помилку

* 1. Структура бази даних
* Users: ID, username, email, password\_hash, created\_at
* Roles: ID, name
* Mods: ID, name, description
* ModVersions: ID, mod\_id, version, supported\_cores
* Modpacks: ID, name, author\_id, description
* ModpackVersions: ID, modpack\_id, version, created\_at
* ModpackMod: зв’язок між версіями збірок та модів
* EmailVerificationTokens, PasswordResetTokens: для безпеки та верифікації
* LocalizationTables: для багатомовності назв і описів

1. Інструкція зі встановлення
   1. Завантажити лаунчер із вебсайту (посилання в особистому кабінеті).
   2. Запустити інсталятор; установка не вимагає прав адміністратора.
   3. Після входу – обрати збірку та натиснути “Встановити”.
   4. Дочекатися завантаження, натиснути “Запустити”.
2. Додаткові положення

* Резервне копіювання даних виконується щоденно;
* Журнал дій зберігається протягом 30 днів;
* У разі порушень політики використання – адміністрація має право обмежити доступ до акаунта.