Специфікація програмного забезпечення

1 ВСТУП

1.1 Огляд продукту

У сучасному кіберспортивному середовищі гри Zenless Zone Zero відсутні спеціалізовані інструменти для автоматизації драфту персонажів у режимі Shiyu-defense. Існуючі універсальні платформи не враховують унікальні особливості цього режиму: систему вартості персонажів, дворівневу структуру команд, складні правила заборон та обмеження на склад команд.

Автоматизована система драфту персонажів "Shiyu Nexus" вирішує цю проблему, надаючи гравцям та організаторам турнірів комплексний веб-інструмент для проведення змагань з автоматичним контролем правил, real-time синхронізацією та інтуїтивним інтерфейсом.

1.2 Мета

Метою створення є розробка автоматизованої системи драфту персонажів, яка забезпечує ефективне проведення змагань у режимі Shiyu-defense гри Zenless Zone Zero, автоматизує контроль дотримання правил, надає інструменти для створення та управління пресетами налаштувань, а також забезпечує синхронну взаємодію між учасниками в реальному часі.

1.3 Межі

Функціональні межі:

* система дозволяє лише авторизованим користувачам створювати та редагувати пресети драфту;
* участь у драфт-сесіях можлива тільки після автентифікації через Discord OAuth;
* неавторизовані користувачі можуть переглядати публічні сесії драфту в режимі спостерігача.

Нефункціональні межі:

* система не підтримує інтеграцію з офіційним API гри Zenless Zone Zero (за його відсутності);
* відсутня підтримка голосового чату між учасниками драфту;
* немає можливості прямого імпорту даних персонажів з ігрового клієнта;
* система не включає функціональність стримінгу або запису відео драфт-сесій.

1.4 Посилання

Даний документ складено відповідно до вимог стандарту IEEE Std 830-1998 (IEEE Recommended Practice for Software Requirements Specifications), а також на основі положень національного стандарту ДСТУ ISO/IEC 15910:2002 «Процес створення документації користувача для програмного застосунку». Структура специфікації вимог до програмної системи відповідає загальноприйнятим міжнародним практикам з розробки технічної документації.

1.5 Означення та абревіатури

API – Application Programming Interface

CORS – Cross-Origin Resource Sharing

DTO – Data Transfer Object

JWT – JSON Web Token

OAuth – Open Authorization

REST –Representational State Transfer

SPA –Single Page Application

SSR – Server-Side Rendering

TTL – Time To Live

WebSocket – протокол повнодуплексного зв'язку поверх TCP

Агент – ігровий персонаж у грі Zenless Zone Zero

Драфт – процес вибору та заборони персонажів перед початком матчу

Shiyu-defense – змагальний режим гри з двокімнатною структурою боїв

2 ЗАГАЛЬНИЙ ОПИС

2.1 Перспективи продукту

Автоматизована система драфту "Shiyu Nexus" має значний потенціал розвитку в екосистемі кіберспорту гри Zenless Zone Zero. Система орієнтована на зростаючу спільноту конкурентних гравців, організаторів турнірів та контент-креаторів, які потребують надійних інструментів для проведення якісних змагань.

У перспективі система може стати стандартом для організації офіційних турнірів з Zenless Zone Zero, інтегруватися з платформами стримінгу для трансляції драфтів та розширитися підтримкою інших ігрових режимів. Модульна архітектура дозволяє адаптувати функціональність для подібних ігор з механікою драфту.

2.2 Функції продукту

Функціональні можливості продукту охоплюють повний життєвий цикл організації та проведення драфт-сесій:

FE-1: Автентифікація користувачів через Discord OAuth 2.0 з підтримкою ролей (користувач, адміністратор).

FE-2: Перегляд бази даних персонажів (агентів) з детальною інформацією про характеристики, фракції та вартість у різних системах.

FE-3: Створення та редагування пресетів налаштувань драфту з комплексною валідацією правил та обмежень.

FE-4: Ініціація нових драфт-сесій на основі існуючих пресетів або з ручним налаштуванням параметрів.

FE-5: Приєднання до активних драфт-сесій у ролі учасника або спостерігача.

FE-6: Покроковий процес драфту з автоматичним керуванням фазами (pre-ban, pick, ban, post-pick).

FE-7: Real-time синхронізація стану драфту між усіма учасниками через WebSocket з'єднання.

FE-8: Автоматична валідація виборів персонажів з контролем обмежень вартості та правил складу команд.

FE-9: Система таймерів ходів з автоматичною обробкою таймаутів.

FE-10: Збереження історії драфт-сесій з можливістю перегляду детальної статистики.

FE-11: Адміністративні інструменти для управління базою даних персонажів, систем вартості та користувачів.

FE-12: Аналітичні дашборди з метриками використання персонажів та ефективності драфтів.

2.3 Характеристики користувачів

Цільова аудиторія системи включає декілька категорій користувачів з різними рівнями технічної підготовки:

Конкурентні гравці: досвідчені гравці Zenless Zone Zero, які беруть участь у змаганнях. Вони мають глибокі знання ігрової механіки, метагейму та стратегій, але можуть не мати технічного досвіду роботи зі спеціалізованим програмним забезпеченням. Для них критично важлива швидкість та точність інтерфейсу під час драфту.

Організатори турнірів: адміністратори спільнот, модератори та eSports менеджери, відповідальні за проведення змагань. Вони потребують розширених інструментів для створення пресетів, модерації сесій та аналізу результатів. Технічний рівень може варіюватися від базового до просунутого.

Контент-креатори та стримери: особи, які створюють розважальний контент навколо гри та можуть використовувати систему для організації публічних драфтів. Вони потребують візуально привабливого інтерфейсу та можливості демонстрації процесу аудиторії.

Спостерігачі та аналітики: особи, зацікавлені у відстеженні трендів метагейму, статистики використання персонажів та аналізі стратегій. Вони потребують доступу до історичних даних та аналітичних інструментів.

2.4 Загальні обмеження

Розроблена система має низку обмежень, зумовлених поточним етапом розвитку, технічними можливостями та специфікою предметної галузі:

Технологічні обмеження:

* Система розроблена як веб-застосунок, що вимагає стабільного Інтернет-з'єднання для роботи. Offline режим не підтримується, що може створювати труднощі при проведенні LAN турнірів або в умовах нестабільного зв'язку.
* Продуктивність системи обмежена потужністю серверної інфраструктури. При значному зростанні кількості одночасних користувачів може знадобитися горизонтальне масштабування, що потребує додаткових налаштувань.

Інтеграційні обмеження:

* Відсутність офіційного API гри Zenless Zone Zero обмежує можливості автоматичного оновлення інформації про персонажів та їхній баланс. Оновлення даних здійснюється вручну адміністраторами системи.
* Система не інтегрована з ігровим клієнтом, що вимагає ручного перенесення результатів драфту в гру.

Правові та етичні обмеження:

* Система використовує зображення та назви персонажів з гри Zenless Zone Zero виключно в освітніх та неккомерційних цілях. Права на інтелектуальну власність належать компанії miHoYo.
* Обробка персональних даних користувачів здійснюється відповідно до політики конфіденційності Discord при використанні OAuth автентифікації.

2.5 Припущення і залежності

Функціонування системи базується на низці припущень щодо технічного середовища та поведінки користувачів:

Технічні припущення:

* Передбачається, що користувачі мають доступ до сучасних веб-браузерів з підтримкою JavaScript ES6+, WebSocket API та localStorage. Система тестувалася в Chrome 90+, Firefox 88+, Safari 14+ та Edge 90+.
* Очікується стабільна робота MongoDB та Redis серверів з налаштованими репліками для забезпечення відмовостійності. Система розрахована на середнє навантаження до 100 одночасних драфт-сесій.

Користувацькі припущення:

* Передбачається, що користувачі мають базові знання ігрової механіки Zenless Zone Zero та розуміють концепцію драфту персонажів. Новим користувачам може знадобитися додаткове навчання правил режиму Shiyu-defense.
* Очікується, що організатори турнірів мають досвід проведення онлайн змагань та розуміють необхідність дотримання встановлених правил.

Зовнішні залежності:

* Система залежить від стабільної роботи Discord API для автентифікації користувачів. Збої в роботі Discord можуть тимчасово обмежити доступ до системи.
* Актуальність інформації про персонажів залежить від своєчасного оновлення даних адміністраторами при випуску нових патчів гри.

3 КОНКРЕТНІ ВИМОГИ

3.1 Вимоги до зовнішніх інтерфейсів

3.1.1 Інтерфейс користувачів

Інтерфейс користувача реалізований як responsive веб-застосунок, оптимізований для роботи на настільних комп’ютерах із повною підтримкою мобільних пристроїв. Основні вимоги до UI/UX включають адаптивність, що забезпечується автоматичним підлаштуванням інтерфейсу до різних розмірів екранів — від 320px до 4K — з метою зручної навігації та оптимального відображення контенту. Значну увагу приділено доступності: інтерфейс відповідає стандартам WCAG 2.1 рівня AA, що гарантує підтримку екранних читачів, навігацію з клавіатури та зручність для користувачів з обмеженими можливостями. Продуктивність інтерфейсу передбачає завантаження початкової сторінки за час, не більший за 2 секунди при швидкості з’єднання 10 Мбіт/с, а всі інтерактивні елементи повинні реагувати на дії користувача протягом 100 мілісекунд. Також передбачено локалізацію: система підтримує англійську мову з можливістю подальшого розширення мовної підтримки.

3.1.2 Програмний інтерфейс

Програмний інтерфейс реалізовано через REST API, що забезпечує повноцінний доступ до функціоналу системи за допомогою стандартизованих HTTP-методів. Для автентифікації використовується механізм JWT токенів із підтримкою refresh token, що дозволяє безпечно отримувати доступ до закритих ендпоінтів. Усі запити та відповіді передаються у форматі JSON з кодуванням UTF-8. API підтримує версіонування через URL-префікси (наприклад, /api/v1/), що забезпечує зворотну сумісність і стабільність роботи клієнтських застосунків. Додатково для розробників надається автоматично згенерована документація у форматі OpenAPI 3.0, доступна через інтерфейс Swagger UI.

3.1.3 WebSocket інтерфейс

Для реалізації real-time комунікації використовується WebSocket-з’єднання, яке забезпечує синхронізацію стану драфт-сесій у режимі реального часу. Авторизація WebSocket-з’єднання здійснюється за допомогою JWT токенів, що гарантує безпечний доступ до конкретних сесій. Комунікація базується на структурованих JSON-повідомленнях із чітко визначеними типами подій, такими як join, leave, pick, ban та phase\_change. Система має механізми відмовостійкості: при втраті мережевого з’єднання відбувається автоматичне його відновлення з використанням алгоритму exponential backoff, що дозволяє зберігати стабільність взаємодії навіть за умов нестабільного інтернету.

3.2 Функціональні вимоги

3.2.1 Управління користувачами

FE-1.1 Автентифікація через Discord OAuth: Система надає можливість входу через Discord OAuth 2.0 з автоматичним створенням облікового запису при першому вході. Зберігається мінімально необхідна інформація: Discord ID, username, email, avatar URL.

FE-1.2 Управління ролями: Підтримка двох ролей користувачів: звичайний користувач та адміністратор. Адміністратори призначаються через прямі зміни в базі даних або існуючими адміністраторами.

FE-1.3 Профіль користувача: Відображення статистики участі в драфтах, створених пресетів та загальної активності в системі.

3.2.2 Управління базою даних персонажів

FE-2.1 Перегляд персонажів: Публічний доступ до повного каталогу персонажів з можливістю фільтрації за фракціями, спеціальностями, атрибутами та статусом релізу.

FE-2.2 Адміністрування персонажів (лише адміністратори): Повний CRUD функціонал для управління персонажами: додавання нових, редагування характеристик, завантаження зображень, встановлення статусу релізу.

FE-2.3 Системи вартості: Створення та управління різними системами розрахунку вартості персонажів з можливістю встановлення індивідуальних цін та винятків.

3.2.3 Система пресетів

FE-3.1 Створення пресетів: Авторизовані користувачі можуть створювати власні пресети налаштувань драфту з детальним налаштуванням правил, обмежень та дозволених персонажів.

FE-3.2 Валідація пресетів: Автоматична перевірка коректності налаштувань пресету, включаючи валідацію посилань на персонажів, логічність обмежень та правильність потоку драфту.

FE-3.3 Публічні та приватні пресети: Можливість створення публічних пресетів для загального використання та приватних для особистих потреб.

3.2.4 Драфт-сесії

FE-4.1 Створення сесій: Ініціація нових драфт-сесій з вибором пресету або ручним налаштуванням параметрів. Генерація унікального коду сесії для приєднання інших гравців.

FE-4.2 Приєднання до сесій: Можливість приєднання до активних сесій у ролі учасника (якщо є вільні місця) або спостерігача.

FE-4.3 Управління фазами драфту: Автоматичне керування послідовністю фаз драфту (pre-ban, pick phase 1, ban phase, pick phase 2) відповідно до налаштувань пресету.

FE-4.4 Валідація виборів: Real-time перевірка кожного вибору персонажа на відповідність правилам: доступність персонажа, дотримання ліміту вартості, унікальність вибору (якщо заборонено mirror picks).

FE-4.5 Таймери ходів: Відлік часу для кожного ходу з візуальними індикаторами та автоматичною обробкою таймаутів.

3.3 Нефункціональні вимоги

3.3.1 Продуктивність

* Час відгуку API: середній час відгуку основних ендпоінтів не повинен перевищувати 200 мілісекунд при нормальному навантаженні.
* WebSocket затримка: максимальна затримка синхронізації стану між клієнтами не повинна перевищувати 100 мілісекунд у локальній мережі та 300 мілісекунд через Інтернет.
* Пропускна здатність: система повинна підтримувати до 100 одночасних драфт-сесій без деградації продуктивності.
* Час завантаження: початкова сторінка повинна завантажуватися повністю протягом 3 секунд при швидкості з'єднання 5 Мбіт/с.

3.3.2 Надійність

* Доступність: система повинна забезпечувати 99.5% uptime в місяць з урахуванням планових технічних робіт.
* Відмовостійність: автоматичне відновлення WebSocket з'єднань при втраті мережі протягом 5 секунд.
* Цілісність даних: використання транзакцій MongoDB для забезпечення консистентності даних при одночасних операціях.
* Резервне копіювання: автоматичне створення backup даних щоденно з можливістю відновлення за останні 30 днів.

3.3.3 Безпека

* Автентифікація: всі API ендпоінти, що потребують авторизації, захищені JWT токенами з терміном дії 15 хвилин та refresh токенами на 7 днів.
* Авторизація: роль-базована система доступу з перевіркою прав на рівні API ендпоінтів та UI компонентів.
* Захист від атак: впровадження rate limiting, CORS політик, валідації вхідних даних та санітизації контенту.
* Конфіденційність: зберігання мінімально необхідних персональних даних з шифруванням sensitive інформації.

3.3.4 Супроводжуваність

* Логування: структуроване логування всіх важливих подій системи з різними рівнями деталізації.
* Моніторинг: метрики продуктивності, здоров'я системи та користувацької активності.
* Документація: повна API документація, архітектурні діаграми та інструкції з розгортання.
* Тестування: unit тести для критичної бізнес-логіки з покриттям не менше 80%.

3.3.5 Масштабованість

* Горизонтальне масштабування: stateless архітектура серверної частини дозволяє додавання нових інстансів без зміни коду.
* База даних: підтримка MongoDB replica sets та потенційного sharding для обробки великих обсягів даних.
* Кешування: ефективне використання Redis для кешування частовикористовуваних даних та сесій.
* CDN готовність: статичні ресурси оптимізовані для роздачі через CDN мережі.

3.3.6 Портабельність

* Браузерна сумісність: підтримка всіх сучасних браузерів (Chrome 90+, Firefox 88+, Safari 14+, Edge 90+).
* Операційні системи: серверна частина сумісна з Linux, Windows та macOS завдяки Node.js платформі.
* Контейнеризація: Docker контейнери для спрощення розгортання в різних середовищах.
* Хмарна готовність: архітектура оптимізована для розгортання в AWS, Google Cloud або Azure.