Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

ЗВІТ

з лабораторної роботи №1
з дисципліни «Архітектура програмного забезпечення»
на тему: «ДООПРАЦЮВАННЯ VISION & SCOPE»

Виконав:
ст. гр. ПЗПІ-22-5
Швець Микита Сергійович

Перевірив:
ас. каф. Дашенков Д.С.

Харків 2025

Vision & Scope

«Програмна система для автоматизації контролю доступу студентів та персоналу у кампусах»

Короткий опис системи

Програмна система для автоматизації контролю доступу студентів та персоналу на кампусі має на меті створити інтуїтивно зрозумілу та безпечну платформу, що дозволяє керувати правами доступу до навчальних корпусів, аудиторій, лабораторій, офісів та інших зон. Система включає серверну частину, web-інтерфейс, мобільний додаток і інтеграцію з IoT-пристроями (наприклад, смарт-замками, QR-сканерами), щоб забезпечити безконтактний і гнучкий механізм контролю доступу в режимі реального часу.

3 SCOPE AND LIMITATIONS

3.1 Scope of Initial Release

У першому (початковому) випуску система охоплює такі компоненти та функції:

1. Серверна частина (Backend)
    - Реєстрація та аутентифікація навчальних закладів (створення облікового запису адміністратора).
    - Реєстрація та авторизація користувачів (студентів, викладачів, співробітників) із розмежуванням ролей (Student, Staff, Admin).
    - Зберігання і обробка даних про користувачів, зокрема інформація про роль, особисті дані.
    - Моделювання структури кампуса: корпуси (Building), поверхи (Floor), приміщення (Room), точки доступу (AccessPoint).
    - Управління ролями та правами доступу: можливість створювати правила доступу (AccessRule) з урахуванням ролі користувача, типу зони (CampusZoneType), часу доби й датного інтервалу.
    - Збір і зберігання журналу доступу (AccessLog) із зазначенням користувача, часу спроби, назви кімнати та статусу (GRANTED/DENIED).
    - Інтеграція з IoT-пристроями:
        - Підключення до смарт-замків і контролерів через API для відкриття/закриття дверей.
        - Збір у режимі реального часу інформації про стан дверей (відкрито/закрито) та спроби несанкціонованого доступу.
        - Зберігання подій IoT у журналі.
2. Мобільний застосунок (Mobile Client)
    - Авторизація користувача через електронну пошту й пароль (отримання JWT).

- Відображення персонального QR-коду, що використовується для безконтактного проходу через сканери.
- Перегляд статистики відвідувань (з графіками й таблицями) та історії спроб доступу.
- Перегляд доступних дозволів (Permissions).

3. Web-застосунок (Web Client)
   - Інтерфейс користувача (User Web UI):
     - Авторизація через веб-форму (email/пароль).
     - Перегляд власного дашборда (Dashboard) з аналітикою, короткою статистикою відвідувань і графіками.
     - Перегляд журналу спроб доступу (History), фільтрація за датою й приміщенням.
     - Перегляд і заявка на зміну власних прав доступу (Permissions).
     - Зміна налаштувань облікового запису (Settings): мова, пароль.
   - Інтерфейс адміністратора (Admin Web UI):
     - Повний доступ до списку користувачів (CRUD для User). Створення/редагування/деактивація облікових записів.
     - Управління системними даними: корпуси (Building), поверхи (Floor), приміщення (Room), точки доступу (AccessPoint).
     - Управління правилами доступу (AccessRule) із можливістю призначення ролі, часу, дати й приміщень.
     - Перегляд і фільтрація журналу доступу (AccessLog) зі статусами, IP-адресами (якщо є).
     - Створення резервних копій налаштувань та даних: експорт/імпорт конфігурацій (Settings) у форматі JSON/CSV.

- Перегляд аналітики (Reports/Analytics): статистика за відвідуваністю зон, кількість спроб доступу, найактивніші користувачі.
- Налаштування багатомовності: додавання перекладів, форматування дат/часу за регіоном.

4. IoT-складова (IoT Devices)
   - Інтеграція з контролерами дверей (smart locks) та зчитувачами QR-кодів:
     - Відкриття/закриття дверей за авторизованим запитом (від мобільного чи web клієнта).
     - Детекція несанкціонованих спроб (запуск тривоги, сповіщення адміністратора).
   - Синхронізація стану пристроїв із сервером у реальному часі (через MQTT/WebSocket чи REST-запити).
   - Можливість дистанційного керування замком (через мобільний чи web UI).
5. Загальні сервіси й інфраструктура
   - База даних: PostgreSQL для зберігання даних про користувачів, приміщення, журнали.
   - Кешування: Redis для зберігання JWT-токена.

Таким чином, Scope of Initial Release включає сервер, мобільний та web-клієнти, а також інтеграцію з IoT. Усі компоненти працюють у єдиному екосистемному середовищі та забезпечують єдину безпечну платформу контролю доступу.

3.2 Scope of Subsequent Releases

У наступних релізах передбачається еволюція системи з використанням технологій штучного інтелекту (ШІ). Нижче описано три ключові аспекти інтеграції ШІ.

3.2.1 Застосування ШІ

1. Аналіз поведінкових патернів: Автоматичне виявлення аномальних патернів використання замків (наприклад, якщо двері часто відчиняють поза навчальним часом чи не за звичним розкладом). Система зможе заздалегідь попереджати адміністрацію про потенційні ризики.
2. Оптимізація розкладу та доступу: На основі даних про відвідуваність аудиторій та завантаженість приміщень ШІ модуль зможе рекомендувати оптимальні розклади або сценарії доступу.
3. Персональні рекомендації: Для студентів та персоналу ШІ може пропонувати кращі часові вікна для відвідувань, підказувати вільні аудиторії для самостійної роботи тощо.

3.2.2 Збирання даних для навчання моделей

1. Логи активності: Усі події (відкриття, закриття, спроби доступу) з детальною позначкою часу, ролі користувача, типу дії.
2. Статистика відвідуваності: Кількість візитів у кожне приміщення, розподіл за часом дня та інші метадані, що можуть бути використані для навчання та побудови моделей.
3. Додаткові сенсори: Можливість підключення датчиків руху, камер спостереження (де це дозволено), збирання даних про поток людей у різних зонах кампусу.
4. Ці дані передаються та зберігаються на сервері, де попередньо обробляються, а потім можуть бути використані для навчання моделей ШІ.

3.2.3 Навчання та використання моделей

1. Готові API сервісів: Для завдань на кшталт розпізнавання облич або NLP (наприклад, обробка запитів користувачів) можна застосовувати

готові сервіси з відкритих платформ (Microsoft Azure Cognitive Services, Google Cloud AI тощо).

2. До-навчання існуючих моделей: Якщо є потреба адаптувати моделі до специфіки конкретного кампусу (наприклад, унікальні сценарії доступу або невеликі аномалії), доцільно до-навчати моделі на основі локальних даних.
3. Самостійне розгортання моделей: Для більш складних або чутливих кейсів (де безпека даних критична) можливо розгортати власні ML-моделі на внутрішніх серверах навчального закладу, забезпечуючи повний контроль над процесом навчання і зберіганням даних.

У рамках Subsequent Releases також планується розширення веб- та мобільного інтерфейсів (зручніші дашборди, детальні графіки в реальному часі, push-сповіщення про аномалії та рекомендації), а також інтеграція з різними модулями ШІ для підвищення безпеки та зручності.

3.3 Limitations and Exclusions

Нижче перелічено обмеження та функції, які з відповідних причин усвідомлено відкладено чи не ввійшли до функціоналу:

1. Неочевидні обмеження:
   - Залежність від інтернет-з’єднання: Хоча система використовує IoT-пристрої, всі операції (зокрема й взаємодія з ШІ-модулями) вимагають стабільного з’єднання із сервером. У випадку відсутності мережі смарт-замки можуть перейти в обмежений або аварійний режим роботи.
   - Якість даних для ШІ: Ефективність модельних прогнозів напряму залежить від якості та обсягу зібраних логів. Якщо дані надходять неповними або з багатьма пропусками, точність системи ШІ знижується.

- Вплив приватності та регуляторних вимог: Відстеження поведінкових патернів користувачів може викликати запитання щодо конфіденційності та захисту особистих даних. Необхідно чітко дотримуватися політик GDPR або місцевого законодавства щодо збирання та обробки персональних даних.

2. Функціонал, виключений з поточного та майбутніх релізів:
   - Повна автономія IoT-пристроїв: На даному етапі не передбачається сценарій, коли смарт-замки можуть працювати тривалий час без підключення до сервера. Це зроблено свідомо, аби мінімізувати ризики безпеки та ускладнення апаратної частини.
   - Інтеграція з розпізнаванням облич «на місці» (on-device): Хоча це потенційно можливо, у найближчих випусках основний фокус буде на використанні готових хмарних сервісів або до-навчанні моделей на сервері. Вбудована в самі IoT-пристрої система розпізнавання облич вимагає додаткових датчиків і значних обчислювальних ресурсів.
   - Розширене відеоспостереження: У рамках системи немає планів зберігати або обробляти великі обсяги відеоданих безпосередньо на сервері, оскільки це вимагатиме спеціалізованої інфраструктури та відповідного захисту даних.
   - Підтримка абсолютно всіх типів смарт-замків та сенсорів: Система тестується і сертифікується з обмеженим переліком IoT-пристроїв, щоб забезпечити якість і стабільність роботи.

Таким чином, для початкового та наступних випусків свідомо визначено перелік функцій і обмежень, які дають можливість реалізувати проєкт ефективно, зберігаючи гнучкість для подальшого розширення і використання новітніх інструментів штучного інтелекту.