

МІНІСТЕРСТВО ОСВІТИ І НАУКИУКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ

імені ІГОРЯ СІКОРСЬКОГО”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**Лабораторна робота**

з курсу: «Проєктування ПЗ технології ЦД»

Виконав

студент 2 курсу

групи КП-21ф

Лук’янець Михайло

Київ – 2024

**ЗАВДАННЯ З КОМПЛЕКСНОЇ ЛАБОРАТОРНОЇ РОБОТИ**  
1. Обрати об'єкт або процес - фізичний двійник (погодити з керівником роботи).  
2. Розробити програмний генератор потокових даних або (за бажання) використати апаратне забезпечення.  
3. Створити 3D модель фізичного двійника.  
4. Обрати платформу для створення цифрового двійника.  
5. Розробити цифровий двійник.  
6. Протестувати цифровий двійник.  
7. Підготувати демонстрацію та записати відео.  
8. Підготувати звіт.  
9. Завантажити на GitHub або аналогічний ресурс розроблений проєкт.  
10. Захистити комплексну лабораторну роботу.

Хід роботи:

1. **Обрати об'єкт або процес - фізичний двійник (погодити з керівником роботи).**

Обраний об’єкт - Розумний будинок: контролювання температури та вологості та управляння пристроями

2. **Розробити програмний генератор потокових даних або (за бажання) використати апаратне забезпечення.**

Розроблено генератор, що симулює роботу системи контролю клімату та освітлення. Контроль клімату автоматичний, в створеному скрипті відображено декілька сценаріїв роботи, таких як есктремальний нагрів, помірний нагрів, екстремальне охолодження та помірне охолодження. Одночасно з цим змінюється і вологість. Контроль освітлення відбувається за рівнем шуму (хлопками), що також симулюється.

3. **Створити 3D модель фізичного двійника.**

Модель будинку створена на основі існуючих моделей (рис. 1).

4. **Обрати платформу для створення цифрового двійника**

Було обрано платформу AWS та використано відповідні можливості, які вона надає.

Изображение выглядит как текст, снимок экрана, Графическое программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рис. 1. Створена модель будинку

5. Розробити цифровий двійник.  
Розроблено наступний цифровий двійник (рис. 2).

Изображение выглядит как текст, снимок экрана, Графическое программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рис.2. Цифровий двійник розумного будинку

На цифровому двійнику відображені графіки вологості, температури, та гучності за останні 5 хвилин (інтервал можливо змінити). Для пришвидшення завантаження на модель додано лише основні об’єкти, а саме лампу та кондиціонер. При роботі кондиціонер змінює колір (синій при охолодженні та червоний при нагріванні) (рис. 3-4), а при переохолодженні чи перенагріванні змінюється індикатор над кондиціонером. Робота лампи відображенна зниканням або відображенням лампочки, оскільки регулювання роботою освітлення на даній платформі виявилося неможливим (рис. 5).

Изображение выглядит как текст, визитная карточка, снимок экрана, дизайн

Автоматически созданное описание

Рис. 3. При нагріванні колір кондиціонеру червоний, при температурі нижче 20 градусів індикатор синій

Изображение выглядит как снимок экрана, визитная карточка, текст, дизайн

Автоматически созданное описание

Рис. 4. При охолодженні колір кондиціонеру синій, при температурі в нормі індикатор білий

Изображение выглядит как красный, пол, на открытом воздухе, деревянный

Автоматически созданное описание

Рис. 5. Освітлення увімкнене – лампа при цьому присутня

Изображение выглядит как на открытом воздухе, красный, деревянный

Автоматически созданное описание

Рис. 6. Освітлення вимкнене – лампа відсутня

**Висновки.** Було розроблено цифровий двійник розумного будинку, що демонструє поточний стан клімату та освітлення будинку. Розроблено на платформі AWS. Серед переваг платформи варто відзначити можливість генерації даних, створення двійника та окреме редагування відображення даних та сцени з моделлю двійника. Серед недоліків варто відзначити відстутність деяких функцій, що обмежують відображення змін на моделі (зміна освітлення сцени) та багатократну зміну відображення при оновленні даних замість однократного після фінального отримання даних.