

# «Тенденції розвитку технологій в автоматизації, приладобудуванні та робототехніці»

## СЕКЦІЯ 4. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА ТЕЛЕКОМУНІКАЦІЇ

УДК 004

О.С. Свістельник, студент, гр. ПЗ-19-2

Т.М. Локтікова, ст. викладач

Державний університет «Житомирська політехніка»

### ОНЛАЙН-СЕРВІС ПРОХОДЖЕННЯ КУРСІВ УКРАЇНСЬКОЇ МОВИ

У сучасному світі важко уявити людину, яка не використовує Інтернет. Щодня ми чуємо нові слова, значення яких нам невідомі, і ми одразу користуємося можливостями Всесвітньої мережі для отримання експертної думки з цього приводу. Ми можемо використовувати потужності Інтернету як нам заманеться, але чому б не спрямувати їх у правильне русло для нашої країни. На жаль, історично склалося так, що істотна кількість населення України не розмовляє державною мовою, надаючи перевагу спілкуватися мовою окупанта, який століттями намагався нас завоювати. Проте повномасштабне вторгнення російської федерації у лютому 2022 року докорінно змінило думку більшості свідомих громадян України про важливість мови для самоідентифікації держави в світовому суспільстві. Тому й було вирішено розробити вебдодаток, який може полегшити процес «лагідної українізації» для мешканців України, які невпевнені у своїх знаннях державної мови.

Кожен проєкт починається з ідеї та загального бачення кінцевого продукту. Тому спочатку формується призначення системи, що розроблюється. Вебзастосунок призначений для покращення володіння державною мовою громадян України завдяки проходженню відповідних курсів з обов'язковою перевіркою отриманих знань за допомогою виконання тестових завдань. Курс поділяється на розділи. В кожному розділі користувач має можливість переглянути відеоматеріал та опрацювати опорний конспект. Для побудови додатку обрано клієнт-серверну архітектуру з використанням вебтехнологій, які не вимагають додаткового ліцензування. Серверна частина застосунку базується на вебсервері Apache2 та системі керування базами даних MySQL. При проєктуванні програмного додатку застосовано архітектуру SPA (Single-page application), з метою забезпечення користувачу досвіду, близького до використання настільної програми.

Клієнтська частина побудована за сучасними стандартами створення односторінкового застосунку, тобто весь необхідний код завантажується разом зі сторінкою, або динамічно довантажується за потребою, зазвичай у відповідь на дії користувача. Для реалізації вказаного продукту було спроєктовано базу даних. ER-діаграма створеної бази даних представлена на рис. 1.

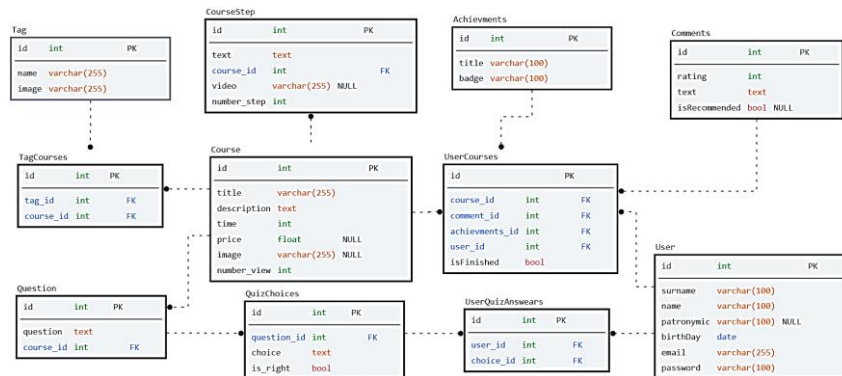


Рис. 1. ER-діаграма створеної бази даних

Для створення сучасного та якісного додатку необхідно обрати інструментарій, доречний саме для цього випадку. В ході проєктування було вирішено використовувати фреймворк Laravel, побудований за допомогою мови програмування PHP, для створення приватного API, тобто бекенд частини додатку. Вибір пав на дану технологію, бо вона має велику спільноту і продуману файловою структурою. Для реалізації клієнтської частини використовується бібліотека мови програмування JavaScript – React. На даний момент часу, це – найпопулярніше рішення для SPA архітектури.

Створення будь-якого програмного продукту передбачає припущення певних помилок у ході розробки, тому без тестування програмного забезпечення неможливо отримати додаток високої якості.

Наразі розроблений додаток проходить як функціональне, так і нефункціональне тестування. Це допоможе отримати додаток, який відповідатиме всім вказаним вимогам.

#### Список використаних джерел:

1. Парасочкін В.В. (2020). Розробка веб-додатку для вивчення англійської мови. Архів кваліфікаційних робіт.

## АНАЛІЗ ПРОГРАМНИХ ІНСТРУМЕНТІВ ДЛЯ ПЛАНУВАННЯ ЗАВДАНЬ

Для кожної людини важливо вміти планувати свій час, щоб мати повноцінне життя, розвиваючи всі його сфери. Розробити робочий проект, зустрітися з друзями, провести час з сім'єю – на кожен день у Вас є свої завдання та цілі, про які не можна забувати. Для полегшення планування задач існує велика кількість застосунків, таких як Evernote, Google Docs, Todoist, Trello та Notion. Аналіз програмних інструментів для планування завдань вказує на зручність у використанні застосунку Notion [1].

Notion – це зручна програма для планування робочих та особистих задач. Застосунок містить в собі багато корисних функцій, які значно спрощують життя. У програмі можна створювати документи та замітки, стилізувати їх на власний смак, використовуючи широкий вибір інструментів: від різних видів шрифтів та маркерів списків до наборів наліпок. У Notion дані зберігаються в окремих блоках, тому за необхідності їх можна швидко та просто переміщати та міняти місцями. Є можливість використовувати спеціальні блоки для програмного коду, що є зручною функцією для програмістів.

Однією з переваг Notion є можливість налаштування зовнішнього вигляду сторінок, додавання їх у розділи і розташовування на головній сторінці програми для швидкого перегляду та зручного користування ними (рис. 1). За допомогою календаря можна планувати події та встановлювати нагадування для них.

У Notion є можливість створювати чек-листи, ставити завдання та цілі на день, місяць або рік. У застосунку, за потреби, можна додавати таблиці, формули, різні типи списків та працювати з медіафайлами.

Важливою функцією є створення своєї бази знань. Для цього потрібно створити пусту сторінку та додати посилання на інші сторінки, – так вся необхідна інформація буде зберігатися в одному місці у вигляді каталогу. У застосунку є інтеграція з іншими платформами: сховище даних Google Drive, вебсервіс для спільної розробки програмного забезпечення GitHub тощо.

Для новачків Notion пропонує понад 20 різноманітних шаблонів, які можна використовувати як основу для оформлення свого ідеального планувальника. Наприклад: шаблон для ведення сімейного бюджету, куди можна записувати доходи та витрати; шаблон для планування подорожей, в який можна збирати фотографії пам'яток, які хотілося б відвідати, корисні посилання і навіть додавати маршрути Google Maps.

У вищевказаному застосунку можна поширювати власні сторінки або надавати доступ до них іншим користувачам для спільного опрацювання, що є дуже зручною функцією при роботі дистанційно.

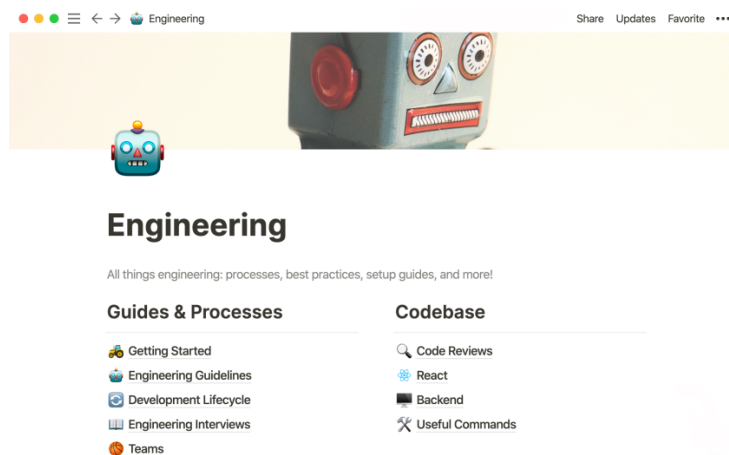


Рис. 1. Приклад організації роботи в Notion [2]

**Висновки:** Notion є універсальним інструментом для планування завдань. Цей застосунок ідеально підходить для використання у всіх сферах життя. Він дозволяє зберігати інформацію про проект та працювати над ним з колегами разом у реальному часі; створювати списки завдань; планувати свій розпорядок дня та зберігати медіафайли і корисні посилання.

### Список використаних джерел:

1. Guides & tutorials. Notion. URL: <https://www.notion.so/help/guides>. (дата звернення: 03.05.2023).

### АНАЛІЗ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ МОБІЛЬНОГО ДОДАТКУ ДЛЯ ВИВЧЕННЯ ІНОЗЕМНИХ МОВ

На сьогоднішній день, з розвитком людства та появою інноваційних технологій, знання іноземних мов стає все більш затребуваним, адже зараз немає такої сфери людської діяльності, де б не використовувались іноземні мови. Вони мають велике значення, для людей, які хочуть побудувати вдалу кар'єру та досягти успіху в житті [1]. Окрім цього, є й інші причини, чому вивчення іноземних мов є важливим:

- стає важливою конкурентною перевагою на ринку праці, особливо в сферах пов'язаних з бізнесом, міжнародними відносинами, туризмом, програмуванням тощо;
- дає можливість спілкуватися з іноземцями та знаходити друзів з усіх куточків світу;
- дозволяє отримати освіту за кордоном, що відкриває нові можливості не тільки в кар'єрі, але і в особистому житті;
- дозволяє комфортно та впевнено подорожувати по всьому світу, вільно спілкуватися з місцевими жителями та краще орієнтуватися в незнайомих місцях;
- розвиває мозок та чудово тренує пам'ять, адже запам'ятовування нових слів, віршів, текстів є відмінним тренажером клітин головного мозку, завдяки чому збільшується мозкова активність, що сприяє не тільки поліпшенню пам'яті, а й також збільшенню рівня інтелекту.

У зв'язку з прогресом технологій і змінах в методології навчання мови, такі традиційні методи вивчення іноземних мов, як запам'ятовування списків лексики та граматичних правил, застаріли, адже ці методи часто зосереджуються на запам'ятовуванні великого обсягу інформації напам'ять і цим самим не дають людям достатньо можливостей практикуватися, використовуючи мову в реальних життєвих ситуаціях. Сучасні методи вивчення іноземної мови більш динамічні, адже наголошують на спілкуванні та взаємодії, використовуючи різні мультимедійні ресурси та автентичні матеріали для створення захоплюючої атмосфери навчання. Для швидкого, зручного, доступного, а найголовніше, ефективного вивчення та практики іноземної мови у власному темпі, виникла необхідність створення мобільних додатків для вивчення іноземних мов. Такі програми пропонують низку інтерактивних функцій та надають можливість отримувати доступ до ресурсів для вивчення мови в будь-який час і місці, забезпечуючи більшу гнучкість і персоналізоване навчання.

Мобільні додатки для вивчення іноземних мов забезпечують персоналізований досвід навчання, який відповідає конкретним потребам і стилям навчання. Окрім того, мобільні програми часто включають елементи гейміфікації, які роблять процес навчання більш захоплюючим і веселим. Одними з найбільш популярних за рейтингом і кількістю завантажень користувачами, за даними сервісу «Google Play Market», є додатки «Qlango», «Babbel», «Mondly», «Falou», «Duolingo». Для визначення найкращого мобільного додатку для вивчення іноземних мов було проведено аналіз наукових статей про ефективні методи вивчення іноземних мов, а також аналіз функціональних можливостей даних мобільних додатків. Узагальнені результати приведені в таблиці 1.

Проведений аналіз показує, що ці мобільні додатки мають багато спільних функцій, наприклад: вправи та короткі уроки, для тренування нової лексики; тлумачний словник, аудіо-вимова тощо.

Таблиця 1

Аналіз функціональних можливостей існуючих мобільних додатків

Функціональні можливості	Словниковий запас	Грамматика	Аудіо-вимова	Аудіювання	Говоріння	Виконання вправ	Інтерфейс
«Qlango»	4	4	3	4	2	3	4
«Babbel»	5	5	4	4	4	4	4
«Mondly»	5	5	4	4	4	4	5
«Falou»	4	4	5	4	5	4	5
«Duolingo»	5	4	5	4	5	5	5

**Висновок.** Проведений аналіз показав, що на даний момент, найкращим безкоштовним мобільним додатком для вивчення іноземних мов є «Duolingo», однак і його не можна віднести до ідеального додатку, тому залишається необхідність в розробці мобільного додатка для вивчення іноземних мов.

#### Список використаних джерел:

1. Багато методів навчання іноземних мов – комбінуйте їх. URL: <https://nus.org.ua/view/bagato-metodiv-navchannya-inozemnyh-mov-kombinujte-yih/>. (дата звернення: 03.05.2023).

**І.А. Кісільов, студент 3-го курсу**  
**Науковий керівник Т.І. Лумпова, к.е.н.**  
*Відокремлений структурний підрозділ «Оптико-механічний фаховий коледж КНУ імені Тараса Шевченка»*

## ПІДХОДИ ДО ОБРАННЯ МОДЕЛІ ЖИТТЄВОГО ЦИКЛУ ТА МЕТОДОЛОГІЇ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗАЦІЇ УЧБОВОГО ПРОЦЕСУ НАВЧАЛЬНОГО ЗАКЛАДУ

**Вступ.** Життєвий цикл програмного забезпечення (далі – ЖЦ ПЗ) є основним інструментом управління процесом розробки програмного забезпечення (далі –ПЗ). Він описує послідовність етапів та дій, що повинні бути виконані в процесі розробки ПЗ з метою забезпечення високої якості та ефективності продукту. Крім того, вибір моделі ЖЦ ПЗ є важливим кроком у проектуванні програмного забезпечення, оскільки він визначає, які процеси та методи будуть використані під час розробки ПЗ, які ресурси будуть витрачені, і які результати будуть досягнуті. Не менш важливим є обрання методології розробки ПЗ, оскільки вона визначає, які методи та підходи будуть використані під час розробки ПЗ, які ресурси будуть витрачені, і які результати будуть досягнуті. При виборі моделі ЖЦ ПЗ та методології розробки ПЗ необхідно враховувати ряд чинників, таких як тип проекту, його обсяг, часові рамки, вимоги до якості та ефективності, наявність ресурсів та технічних засобів. Ці задачі вирішувались для вирішення завдання створення проекту автоматизації обліку успішності студентів коледжу.

**Мета роботи.** Дослідити моделі ЖЦ ПЗ, методології розроблення ПЗ та обрати оптимальну для вирішення завдання створення проекту автоматизації обліку успішності студентів коледжу силами студентської спільноти в умовах військово стану.

**Матеріали та методи.** Для вирішення завдання створення проекту автоматизації обліку успішності студентів коледжу (далі – Проекту) потрібно розглянути основні моделі ЖЦ ПЗ, до яких в [1] віднесли каскадну («водоспаду»), інкрементну, спіральну, еволюційну, та методології, які застосовуються разом з цими моделями. Вибір потрібно зробити з погляду на сучасні обставини – воєнний стан і пов'язані з ним складнощі щодо організації командної роботи.

**Результати досліджень.** Спираючись на визначення в [1, 2] розглянемо основні моделі ЖЦ ПЗ .

1. Модель «водоспаду» (Waterfall model) – ця модель передбачає послідовне виконання етапів розробки ПЗ, починаючи з аналізу вимог і закінчуючи з випуску готового продукту. Ця модель має просту структуру та добре підходить для проектів з чіткими вимогами та обмеженим функціоналом. Розвитком цієї моделі є V-подібна модель життєвого циклу (V-model), в якій тестування розглядається як неперервний процес, інтегрований в процес розробки програмної системи і включає два взаємопов'язаних підпроцеси: планування тестування в рамках процесів розробки системи та проведення тестування відповідних об'єктів на відповідному етапі розробки. Такий підхід дозволяє забезпечити високу якість продукту.

2. Інкрементна модель (або ітераційна модель розробки) є однією з моделей ЖЦ ПЗ. Відрізняється від інших моделей тим, що розробка програмного продукту відбувається поетапно, причому кожен етап включає в себе певний набір функціональності. Ця модель надає можливість розробки продукту у короткі строки, а також здійснювати контроль замовником над процесом розробки, вносити корективи та зміни на кожному етапі.

3. Модель «спіралі» (Spiral model) – ця модель передбачає ітераційний підхід до розробки ПЗ, що означає, що розробка відбувається в серіях ітерацій, кожна з яких передбачає аналіз вимог, проектування, розробку та тестування. При використанні цієї моделі вимоги можуть бути не повністю визначеними.

4. Еволюційна модель розроблення ПЗ є ітеративною та інкрементальною моделлю, яка передбачає поступову розробку ПЗ, починаючи з мінімально необхідного функціоналу та поступовим додаванням нового функціоналу на кожній ітерації. Ця модель підходить для проектів, в яких важко визначити повний перелік вимог на початку проекту.

Вимоги до Проекту є визначеними, наведемо їх.

Функціональні вимоги: 1) реєстрація та авторизація користувачів (адміністраторів, викладачів та студентів) з використанням логіну та пароля; 2) можливість створювати та редагувати профілі користувачів з обов'язковими полями (ім'я, прізвище, поштова адреса, номер телефону, електронна адреса, номер студентського квитка тощо); 3) можливість додавання нових предметів, навчальних планів та груп студентів зі зазначенням кількості годин, викладача та дисциплін; 4) збереження даних про відвідуваність занять студентами; 5) облік оцінок та визначення середнього балу студента за семестр, рік, певний період; 6) перегляд інформації про студентів та їх успішність; 7) повідомлення користувачів (адміністраторів, викладачів та студентів) про нові події, зміни в графіку занять та інші важливі повідомлення.

Нефункціональні вимоги: 1) надійність: система повинна працювати стабільно та безперебійно якнайменш протягом робочого дня коледжу, забезпечуючи точність інформації та високу швидкість обробки даних; 2) безпека: система повинна забезпечувати конфіденційність, цілісність та доступність даних, а також має мати механізми захисту від зломів та несанкціонованого доступу до інформації; 3) ефективність: система повинна працювати швидко та ефективно, забезпечуючи зручний та простий інтерфейс для користувачів; 4) сумісність: система повинна бути сумісною з різними операційними системами та браузерами, наявними в коледжі, що дозволить користувачам використовувати її на різних пристроях та платформах.

З погляду на визначеність функціональних вимог доцільно обрати інкрементну модель, оскільки можна розробити базову функціональність і додавати окремі нові функції ітеративно. Але якщо ми застосуємо принципи V-подібної моделі, яка передбачає жорстку послідовність фаз, що повинні бути виконані перед тим, як перейти до наступної фази розробки ПЗ, то це дозволить нам і підвищити якість продукту, і отримати план тестування для приймальних випробувань разом із завершенням робіт, попередньо виконавши окремі етапи тестування на кожній ітерації. Це дозволить забезпечити високу якість програмного продукту та уникнути затримок у розробці. Важливо також врахувати таку обставину, що доопрацьовувати продукт за зауваженнями, отриманими в ході приймальних випробувань на платформі в коледжі може бути важко, тому доцільно застосувати прототипування на фазах розроблення та тестування.

Для Проекту з визначеними вимогами доцільно обрати V-модель ЖЦ ПЗ з використанням прототипування. Прототипування на фазах розроблення та тестування дозволить забезпечити взаємодію з користувачем та забезпечити відповідність програмного продукту вимогам користувачів. У процесі розробки прототипу буде поступово удосконалюватись та на основі зворотного зв'язку від користувачів будуть внесені необхідні зміни та доповнення.

Для розроблення Проекту найбільш прийнятною є гнучка методологія, принципи і підходи якої добре узгоджуються з V-моделлю. Проте в умовах військового часу, коли немає можливості організувати парне програмування та забезпечити постійну присутність «замовника», використання XP стає неможливим. Тут доцільним варіантом буде методологія Scrum, яка також відноситься до гнучких методологій (Agile), але запозичує ті елементи, які можна реалізувати в обставинах, що склалися. Одним з перших кроків буде визначення ролей розробників та спринт, який є згурпованою множиною робіт, що повинні бути виконані протягом короткого періоду часу (зазвичай, 1–4 тижні). Щодо щоденних зборів розробників, то доцільним буде скоротити їх до 1–2 разів на тиждень, розробивши графік з прив'язкою до плану виконання робіт, який відповідно до методології Scrum повинен представляти розбиття на невеликі підзадачі.

Важливим питанням є організація співпраці студентів в процесі розробки ПЗ. Необхідно пам'ятати про важливість правильної організації комунікації та співпраці між усіма учасниками процесу розробки. Тут можна запропонувати використання системи контролю версій Git та онлайн-дошки Канбан. Git дозволяє студентам працювати з одним і тим же кодом, зберігаючи всі зміни та історію роботи. Це дозволяє легко відслідковувати прогрес кожного з учасників проекту, вирішувати конфлікти та об'єднувати роботу в єдине ціле. Дошки Канбан є візуальним інструментом для організації робочого процесу та контролю прогресу. На дошці можна розміщувати картки з завданнями, які потрібно виконати, та переміщувати їх між колонками в залежності від стану виконання. Це дозволяє всім учасникам проекту бачити, які завдання наразі виконуються, які є в черзі та які вже завершені. Для використання онлайн-дошок Канбан можна скористатися, наприклад, Trello – візуальний інструмент, що дає змогу команді керувати різноманітними проектами й робочими процесами та відстежувати виконання завдань, або Asana – ПЗ для веб і мобільних пристроїв, призначене для спільної роботи над проектами без електронної пошти.

Таким чином, використання Git для контролю версій коду та онлайн-дошок Канбан для організації завдань та пріоритетів може значно полегшити організацію співпраці студентів в процесі розробки ПЗ, забезпечити контроль прогресу та ефективну комунікацію між учасниками проекту, а також може забезпечити успішне виконання задачі створення Проекту.

**Висновок.** З урахуванням вимог до розробки проекту, найбільш оптимальним підходом є використання V-моделі ЖЦ з прототипуванням, що дозволяє знизити ризики та витрати на внесення змін у вже готовий продукт. Таким чином, обрана комбінація методів та моделей дозволяє забезпечити більш ефективну та продуктивну розробку проекту. V-модель ЖЦ ПЗ дозволяє розбити проект на фази та забезпечити кожну з них необхідними тестами та верифікацією, що зменшує ризик виявлення помилок на пізніших етапах розробки. Загалом, обрана комбінація методів та моделей дозволяє забезпечити успішну та ефективну розробку проекту, що задовольнятиме вимоги замовника та користувачів.

#### Список використаних джерел:

1. Лавріщева К.М. Програмна інженерія : електронний підручник. URL: <http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf>.
2. Життєвий цикл програмного забезпечення. Навчальний посібник / Є.В. Левус, Т.А. Марусенкова, О.О. Нитребич. Львів : Видавництво Львівської політехніки, 2017. 208 с.

**Є.С. Зінюк, гр. ТР-19-1**  
**І.Р. Гусаров, гр. ТР-19-1**  
**В.В. Ципоренко, к.т.н., доц.**  
**В.Г. Ципоренко, к.т.н., доц.**

*Державний університет «Житомирська політехніка»*

## **ГІБРИДНА СИСТЕМА ВІДЕОСПОСТЕРЕЖЕННЯ ОФІСНОГО ПРИМІЩЕННЯ**

Сучасні системи відеоспостереження являють собою комплекс апаратних та програмних засобів, що призначені для організації контролю на різних об'єктах. Системи відеоспостереження поділяють на аналогові, цифрові або комбіновані (одночасне використання цифрових та аналогових відеокамер). Зазвичай аналогові системи використовують на невеликих підприємствах. Перевагами аналогових систем відеоспостереження є: простота користування, відпрацьовані технології монтажу, доступна ціна, висока надійність, стабільна робота. Недоліками аналогових систем є: постійна потреба в обслуговуванні, обмеженість функцій.

Цифрову систему відеоспостереження також називають IP-системою. Вона підтримує кабельне та безпроводне підключення (передача по Wi-Fi або Bluetooth), відповідно використовує цифрові камери. Записуватись інформація може як на фізичний носій інформації так і на хмарне сховище. Перевагами цифрових систем відеоспостереження є: висока роздільна здатність, висока завадозахищеність, наявність функцій розпізнавання обличчя, номерів та інших об'єктів, висока якість масштабування, можливість кодування та стискання зображень у відеокамері. Недоліками цифрових систем є: перенавантаженість мережі при великій кількості камер, а також суттєва собівартість.

В роботі запропонована гібридна система відеоспостереження офісного приміщення з прилеглою вуличною територією та автопарковкою. Система вирішує наступні задачі: віддалений контроль роботи офісу в цілому та кожного співробітника окремо; мінімізація можливості втрати інформації, що становить комерційну таємницю; оперативний цілодобовий контроль автомобілів співробітників офісу на парковці; запобігання та розкриття випадків крадіжки або будь-яких інших протизаконних дій; запорука цілісності майна підприємства від пошкоджень та крадіжок; зростання продуктивності роботи персоналу за рахунок мінімізації непродуктивних часових витрат; контроль головного входу, прохідної; та коридорних приміщень; нічне спостереження території офісу; аналіз даних записів із камер відеоспостереження з метою покращення роботи офісу.

Цінною особливістю системи відео спостереження є можливість записувати відео навіть при вимкненій електриці, оскільки зловмисники, якщо спробують проникнути в приміщення, швидше за все вимкнуть електрику. Основні відеокамери повинні бути обладнані акумуляторами і картою пам'яті, щоб продовжити запис у разі відключення зовнішнього джерела живлення. В свою чергу, відеокамери повинні мати хорошу якість запису в темний час доби, оскільки швидше за все зловмисники діятимуть саме в цей час. Доцільно також використовувати допоміжні датчики руху, що забезпечить максимально ефективний і точний аналіз інформації в системі в нічний час доби. Додатково доцільно використання в системі інтелектуальної архівації відеозаписів. Ефективним алгоритмом, що забезпечує економію пам'яті з відеозаписами, є зберігання повного записів за останню добу, а також окремі фрагменти відео, в яких була помічена підозріла активність. При цьому користувачам надсилається повідомлення про підозрілі активності в реальному часі. Важливою задачею при виявленні підозрілої активності є пошук аномалій у відеоряді. Доцільно розглядати відео як послідовність кадрів і суттєві зміни від одного кадру до іншого зазвичай можуть вказувати на виникнення нових ситуацій. Така подія розглядається як вихід із стану стабільності, тобто як аномалія.

На практиці розглядають два типи задач, що пов'язані з пошуком аномалій. Перша задача – виявлення так званих викидів, які визначаються як спостереження, що розташовані далеко від інших. Алгоритми для детектування викидів здійснюють пошук областей, де зосереджена основна маса даних, ігноруючи аномальні спостереження. Друга задача – визначення незвичайного явища або ситуації, наприклад, аномальної поведінки, коли є спостереження, що описують різноманітні звичайні стани системи. В результаті при появі нових даних визначають, чиє вони аномальними.

Показана доцільність реалізації системи шляхом комбінації аналогових та цифрових відеокамер у вигляді мережі зі стаціонарним та безпроводовим доступом. Аналогові камери доцільно використовувати як оглядові засоби на вуличній території та автопарковці, а цифрові – у приміщеннях офісу та на автопарковці – з функцією розпізнавання обличчя персоналу та номерів автотранспорту. Також запропоновано підключення переносних відеокамер автопарковки з використанням каналу радіодоступу, що забезпечує ефективний цілодобовий контроль автотранспорту.

### **Список використаних джерел:**

1. Голдін О.М. (2022). Розробка телекомунікаційної мережі офісної будівлі з інтеграцією системи IP-відеоспостереження.

## ОБРОБКА JSON-СТРУКТУР ЗА ДОПОМОГОЮ PANDAS

Мова програмування Python стала однією із найпопулярніших мов програмування, які застосовуються для аналізу даних та машинного навчання [1].

Однією з найбільш поширених та популярних бібліотек для роботи з даними у Python є Pandas. Бібліотека надає простий спосіб для обробки, аналізу та роботи з різними типами, включаючи табличні дані та JSON. За допомогою Pandas можна виконувати велику кількість задач, таких як завантаження, очистку, фільтрацію та сортування даних, додавання нових стовпців, агрегацію та групування даних та ін. Бібліотека Pandas також легко інтегрується з іншими бібліотеками мови програмування Python, такими як NumPy, Matplotlib и Scikit-Learn [1].

З іншого боку, одним із дуже поширених на сьогодні форматів представлення даних є формат JSON (JavaScript Object Notation), який дуже активно використовується у веброзробці для передачі та зберігання даних [2]. Тому важливо при роботі з даними у Pandas мати можливість використовувати JSON-формат даних, враховуючи те, що він може представляти складні (вкладені) дані.

Припустимо, що у форматі JSON представлено певний набір об'єктів. Перетворення цього набору (списку) у формат DataFrame, який використовується у Pandas може бути складною операцією, враховуючи те, що дані можуть бути багаторівневими (мати вкладену структуру). Ці дані можна перетворити у зручний для аналізу вигляд за допомогою метода `explode`. Але при цьому не буде враховано вкладеність об'єктів JSON. Щоб перетворити вкладений об'єкт у табличний вигляд, знову потрібно застосувати метод `explode`.

Залишилось перетворити окремі об'єкти JSON, які відображаються як значення в окремі стовпці. Для цього рекомендуємо використовувати метод `json_normalize`. Але потрібно буде об'єднати вихідний DataFrame з нормалізованим DataFrame, створеним методом `exploded` за допомогою метода `pd.concat`. Виконуючи ці дії, потрібно не забути за допомогою методу `drop` видалити стовпець (або стовпці), що містить дані, які дублюються.

В результаті вийшов такий алгоритм та результат його роботи (рис. 1):

```
import pandas as pd # рядок з JSON
import json
str = '''[
  { "id": 1, "field1" : "one", "nested1": [ { "field111": "one-two", "field112": 12 },
    { "field111": "one-three", "field112": 13 } ] },
  { "id": 2, "field1" : "two", "nested1": [ { "field111": "two-two", "field112": 22 },
    { "field111": "two-three", "field112": 23 } ] } ]''';
data = json.loads(str)
df = pd.DataFrame(data)
exploded_df = df.explode("nested1", ignore_index = True)
pd.concat([exploded_df.drop('nested1', axis = 1),
pd.json_normalize(exploded_df['nested1']), axis = 1)
```

	id	field1	field111	field112
0	1	one	one-two	12
1	1	one	one-three	13
2	2	two	two-two	22
3	2	two	two-three	23

Рис. 1. Результат роботи алгоритму

### Список використаних джерел:

1. Yogesh Rana. Python: Simple though an Important Programming language (англ.) // International Research Journal of Engineering and Technology (IRJET). – 2019. – 2 February (vol. 06, iss. 2). – P. 1856–1858. – ISSN 2395-0056.
2. Crockford D. The application/json Media Type for JavaScript Object Notation (JSON) (англ.) – IETF, 2006. – 10 p. DOI: 10.17487/RFC4627.

## ВИКОРИСТАННЯ ІНТЕРНЕТУ РЕЧЕЙ ДЛЯ ЕКОНОМІЇ ЕНЕРГЕТИЧНИХ РЕСУРСІВ

У сучасному світі, зі сталим зростанням споживання енергії, виникає відчутна потреба в пошуку ефективних способів економії енергетичних ресурсів. Інтернет речей (IoT) набирає все більшої популярності та може стати одним з таких рішень.

Інтернет речей – це мережа взаємопов'язаних пристроїв, які автоматично збирають та обмінюються даними. Використання IoT у сфері енергетики дозволяє приєднати різноманітні датчики та пристрої, які вимірюють споживання енергії, а також дає можливість керувати цими даними. Це може забезпечити оптимальне використання енергії та сприяти економії ресурсів [1].

Використання IoT у практичних ситуаціях також дає можливість реалізації енергоефективних рішень. За допомогою IoT, розумні будинки можуть автоматично регулювати опалення, освітлення, електроприлади та інші системи на основі потреб користувачів та зовнішніх умов, забезпечуючи енергоефективність. Іншим прикладом може бути використання IoT у смартґрідах, що дає можливість управління розподілом та споживанням енергії, що приводить до зниження енергетичних втрат [2]. Крім того, IoT може сприяти розвитку відновлюваних джерел енергії, таких як сонячна та вітрова енергетика, шляхом моніторингу та контролю їх ефективності та адаптації до змінних умов [3].

Один з реальних прикладів використання IoT для економії енергетичних ресурсів – впровадженням систем на базі Інтернету речей (IoT) з метою створення та удосконалення електронних мереж. Компанія Leap сприяє розвитку електронних мереж у США, використовуючи IoT пристрої, такі як датчики та контролери, для автоматизації процесів та забезпечення діагностики. Вони впроваджують інтелектуальні лічильники, які дозволяють передавати дані про споживання енергії віддалено, а також розумні освітлювальні системи та системи управління споживанням, які можуть автоматично пристосовуватися до змінних умов. Завдяки цим IoT технологіям, Leap забезпечує оптимальне використання енергії та підвищення енергоефективності, що в свою чергу знижує навантаження на мережу та гарантує стабільність та надійність електропостачання. Таким чином, Leap сприяє впровадженню інтелектуалізації та більш екологічно сумісного споживання електроенергії у США, що сприяє сталому розвитку електроенергетичної інфраструктури [4].

Інший приклад використання IoT для економії ресурсів – це система управління водопостачанням у Барселоні, Іспанія. Завдяки використанню IoT-датчиків для відстеження тиску та якості води, а також регулювання розподілу води, місто змогло знизити водовідбір та забезпечити ефективне використання водних ресурсів. Застосування IoT технологій допомогло зекономити воду та підвищити ефективність водопостачальної системи [5].

Отже, Інтернет речей відіграє важливу роль в економії енергетичних ресурсів завдяки можливості автоматичного контролю і керування різними системами та пристроями. Розглянуті практичні приклади, такі як розумні будинки та смартґріди, демонструють вже досягнуті успіхи в оптимізації ресурсів, зниження втрат енергії, підвищення енергоефективності та стабільності електричних мереж. Однак для повної реалізації цих переваг потрібний подальший розвиток технологій IoT та їх широке впровадження в енергетичній галузі.

### Список використаних джерел:

1. Інтернет речей [Електронний ресурс] // Вікіпедія – Режим доступу: [https://uk.wikipedia.org/wiki/Інтернет\\_речей](https://uk.wikipedia.org/wiki/Інтернет_речей).
2. Siano P. (2014). Demand response and smart grids—A survey [Електронний ресурс] / P.Siano // *Renewable and Sustainable Energy Reviews*, 30, 461-478. – Режим доступу: <https://www.sciencedirect.com/science/article/abs/pii/S1364032113007211>.
3. The cognitive power meter: Looking beyond the smart meter / S. Makonin, F. Popowich, L. Bartram, B.Gill, I.V. Bajic // 2016 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). – 2016. – pp. 1-4. IEEE [Електронний ресурс]. – Режим доступу: <https://ieeexplore.ieee.org/document/7726766>.
4. *Thomas Folker Enabling Resilient Energy Grids / Thomas Folker // Impact Hustlers*. – 2022 [Електронний ресурс]. – Режим доступу: <https://www.impacthustlers.com/podcast-episodes/enabling-resilient-energy-grids-thomas-folker-of-leap>.
5. Redacción The New Now. Gestión integral del agua con IoT y Machine Learning / Redacción The New Now // *thenewnow*. – 2021 [Електронний ресурс]. – Режим доступу: <https://www.thenewnow.es/tecnologia/gestion-integral-agua-iot-machine-learning/>.



## AIRTABLE — ХМАРНА ПЛАТФОРМА ДЛЯ РОБОТИ З БАЗАМИ ДАНИХ

Хмарні послуги для роботи з базами даних є рішеннями, які дозволяють розгорнути і використовувати бази даних у хмарі. Основні переваги даного підходу: гнучкість і масштабованість, стійкість до відмов, швидкість розгортання, глобальний доступ: тощо.

Airtable – це гнучкий онлайн-сервіс для управління базами даних та проектами. Він є комбінацією таблиці, бази даних та інструментів управління проектами в одному додатку. Він обслуговує багато організацій завдяки надійному REST API та привабливому візуальному інтерфейсу користувача, який має власні поля для керування та представлення даних. Airtable повністю хмарний і має власну інформаційну панель, де можна створити робочий простір для організації даних і співпраці відповідно до потреб команди.

Airtable надає можливості для управління завданнями та проектами, включаючи календар, список завдань, графіки тощо. Це дозволяє користувачам планувати та відстежувати завдання, керувати термінами виконання проектів та керувати командними завданнями.

Airtable має інструмент Airtable Sync, що дозволяє розробляти складно організовані проекти з участю багатьох команд (рис. 1).

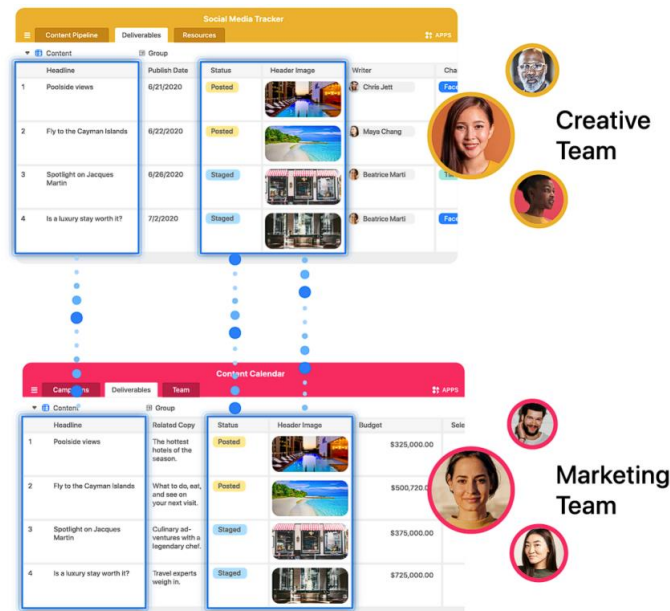


Рис. 1. Організація роботи декількох команд

Airtable Sync – дозволяє створювати та підтримувати автоматичні зв'язки між різними базами даних в Airtable. Це означає, що зміни, внесені до однієї бази даних, можуть автоматично відобразитися в інших базах даних.

Airtable пропонує широкий спектр можливостей для імпорту даних. Він дозволяє імпортувати дані з Google Tables, Excel або CSV-файлів. Крім того, він має інтеграції з багатьма популярними програмами: Salesforce, Google Drive, Asana, Basecamp, Dropbox, Evernote, Facebook, GitHub, Gmail, Google Calendar, Google Contacts, LinkedIn, Pocket, Typeform та WordPress. Це дозволяє легко обмінюватися даними між різними програмами та підвищує продуктивність команди.

**Висновок.** Airtable – це потужний інструмент для організації та управління даними. Завдяки інтуїтивно зрозумілому інтерфейсу, кожен може легко розпочати роботу з базами даних, таблицями та іншими засобами організації даних. Також даний сервіс дає можливість спільної роботи в режимі реального часу декілька команд.

### Список використаних джерел:

1. Documentation. Airtable. URL: <https://support.airtable.com/docs> (дата звернення: 07.05.2023).

## АНАЛІЗ РОБОТИ CHATGPT НА ПРИКЛАДІ ГЕНЕРУВАННЯ CSS-КОДУ

Вебтехнології широко використовуються в різних сферах для створення сайтів та додатків. Деякі з основних переваг вебтехнологій містять: доступність, гнучкість, масштабованість, переносимість тощо.

Для спрощення застосування веб-технологій використовують різні допоміжні інструменти, наприклад, генератори CSS-стилів. Існує безліч подібних генераторів: генератори градієнтів для створення унікальних фонових зображень; генератори кнопок для створення стилізованих кнопок; генератори стилів для створення гнучкої верстки на базі технології Flexbox тощо. Основна проблема таких генераторів – це вузькоспрямованість, особливо якщо генератор не дозволяє врахувати всі параметри при формуванні CSS-стилів. Зовсім нещодавно на ринку високих технологій з'явився новий AI-бот – ChatGPT. Можливості для використання ChatGPT практично безмежні, а сам інструмент здатен спростити роботу та підвищити ефективність працівників майже в будь-якій галузі.

ChatGPT є потужним мовним модулем, який може генерувати текст, включаючи CSS-код, з мінімальною участю людини. Одна з головних переваг його використання є те, що він може генерувати код за текстовим описом. Це означає, що не потрібно мати багато генераторів та вивчати їх можливості і інтерфейс користувача, що значно зменшує час на розробку програмних продуктів. При використанні ChatGPT для генерації CSS, можна вирішити багато завдань, пов'язаних з генерацією CSS-коду, включаючи створення макетів, оформлення тексту, позиціонування елементів тощо.

ChatGPT використовує машинне навчання, щоб навчитися генерувати відповідний CSS-код. Чим більше даних вводиться в систему, тим краще код, який він генерує (рис. 1).

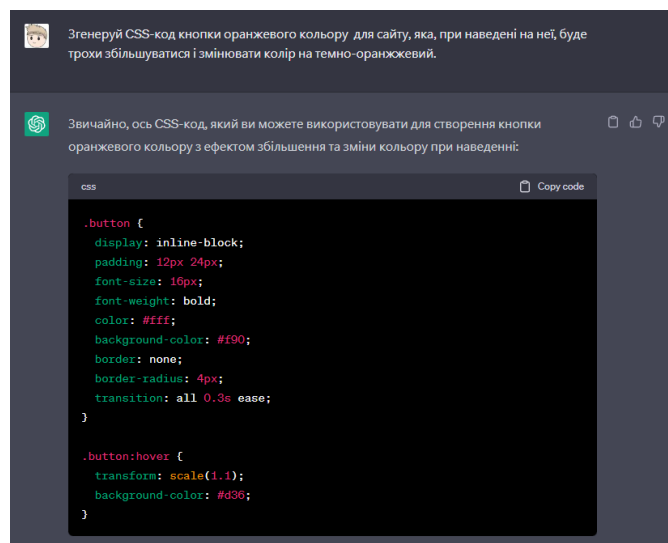


Рис. 1. Приклад генерації CSS-коду для кнопки

Хоча ChatGPT може бути дуже корисним інструментом для автоматичної генерації CSS, він не замінює фахівця. Людина все ж повинна вносити корективи до згенерованого коду, щоб досягнути більш точного та професійного результату, бо ChatGPT має свої обмеження. Як вже було сказано, ChatGPT є мовним модулем, тому він не вміє програмувати, а лише генерувати текст. Відповідно, якщо у нього є пробіли у «знаннях», то він не в змозі генерувати деякі конструкції CSS.

**Висновки:** використання ChatGPT для генерації CSS-коду може бути корисним інструментом для дизайнерів та програмістів, які хочуть збільшити продуктивність та знизити час, на написання CSS-коду. Однак, необхідно враховувати обмеження та користуватися додатковими програмними засобами також, щоб забезпечити якісний та професійний результат.

### Список використаних джерел:

1. A Top Collection Of CSS Code Generators. OpenReplay. URL: <https://blog.openreplay.com/a-top-collection-of-css-code-generators>. (дата звернення: 03.05.2023).
2. Documentation. OpenAI. URL: <https://platform.openai.com/docs> (дата звернення: 03.05.2023).

### АВТОМАТИЗАЦІЯ СТВОРЕННЯ АНІМАЦІЇ ДЛЯ ПІКСЕЛЬ-АРТА

Анімація в нашому світі існує ще з 1877 року, і за цей час вона пройшла великий шлях розвитку, змін і переосмислень. На сьогоднішній день в цій сфері є чимало різних стилів, методів і інструментів. І, наприклад, художникам для 2D-анімації вже не потрібно промальовувати кожен кадр вручну з нуля: достатньо намалювати персонажа один раз і накласти на нього скелет, за допомогою якої потім здійснити так звану скелетну анімацію скільки завгодно разів. Для 3D-графіки скелетна анімація є чи не найрозповсюдженішою технікою. Але досі є стилістика, де, враховуючи її особливості, не існує шляху застосувати скелетну анімацію. Ця стилістика – піксель-арт.

Яскравим прикладом піксельної анімації є комп'ютерні ігри. Як же реалізовували піксельну анімацію в них раніше та реалізують зараз? Художник просто малює кожен кадр за кадром, створюючи так звані спрайт-листки (спрайт-листки гри «The Legend of Zelda: A Link to the Past» [1]). Це потребує купу часу, але на фоні розповсюдженості анімації і звичності для нас цього явища, не вражає. Проте ця купа починає вражати, коли виникає потреба додати в гру багато варіацій зовнішнього вигляду персонажа або взагалі дати гравцям можливість налаштувати зовнішній вигляд за своїм бажанням. Тоді обсяги роботи і кількість кадрів, необхідних для малювання, стають просто колосальними. Виникає питання, чи можна автоматизувати процес? Один із можливих шляхів вирішення проблеми – скелетна анімація.



Рис. 1. Приклад спрайт-листа з гри The Legend of Zelda: A Link to the Past

Ідея виникла у розробника з псевдонімом aarthificial, про яку він розповів у своїй доповіді під назвою «Pixel Art Animation. Reinvented» [2].

Зазвичай, кожен піксель в зображенні зберігає чотири параметри: R, G, B та Alpha. Ці параметри репрезентують колір даного пікселя в системі RGBA. Якщо кожен піксель анімації буде зберігати не колір, а координати потрібного кольору на розгортці, тоді для реалізації анімації потрібно два графічних ресурси: анімація, в пікселях якої закодовані координати потрібного пікселя з текстури, та сама текстура (далі – розгортка).

Оскільки в більшості випадків текстури мають невеликий розмір, то і значення в пікселях анімації будуть теж невеликі, а невеликим значенням відповідають темні кольори. Для художника така анімація виглядатиме дуже темним, практично чорним прямокутником і працювати в таких умовах буде неможливо.

Щоб це виправляти, створимо колірну карту – копію розгортки, кожному пікселю якої присвоєний унікальний (і розбірливий для художника) колір. В самій анімації для опису кольору з розгортки художник використовує відповідний колір з колірної карти, а при завантаженні програми, запускується алгоритм, який «обійде» всі пікселі анімації та запише в них координати відповідного кольору з колірної карти. Таким чином, збережено переваги цього методу анімації і значно поліпшено роботу художникам.

Тепер анімацію достатньо намалювати один раз, а при потребі змінити зовнішній вигляд персонажа, художнику достатньо змінити лише розгортку. До того ж з'являється можливість маніпулювати виглядом персонажа програмно в реальному часі (наприклад, накладати ефекти бруду при переміщенні по певних поверхнях). Таким чином, якщо потрібно застосувати цю анімацію для нового персонажа, то достатньо підставити його розгортку.

В якості демонстрації було створено простий додаток [3], який реалізує весь описаний метод анімації. Практично весь простір вікна займає кінцевий результат – готова анімація. Під нею розміщені відповідно анімація колірної карти, сама колірна карта і розгортка, що застосовується до анімації.

Було розроблено клас Animation. При створенні його екземпляру відбувається циклічний обхід всіх пікселів колірної анімації, для кожного пікселя знаходиться ідентичний колір на колірній карті, а у відповідний піксель кінцевої анімації записуються координати цього кольору на колірній карті (R відповідає за перший індекс, G за другий, B і Alpha не використовуються). Для рендеру кадру анімації створено метод NextFrame в якому: створюється кадр, після чого кожному його пікселю присвоюється значення, що знаходиться в пікселі розгортки, на який вказують координати пікселя з анімації. На рис. 2 і 3 приведено частину вихідного коду та скрін додатка, які пояснюють ідею (докладніше можна переглянути за посиланням [3]).

```

Color target;
this.mapAnim = new Bitmap(animation.Width, animation.Height);
for (int i = 0; i < width; i++)
{
    for (int j = 0; j < height; j++)
    {
        target = animation.GetPixel(i, j);
        for (int mx = 0; mx < colormap.Width; mx++)
        {
            for (int my = 0; my < colormap.Height; my++)
            {
                if (colormap.GetPixel(mx, my) == target)
                {
                    mapAnim.SetPixel(i, j, Color.FromArgb(mx, my, 0));
                    mx = colormap.Width; my = colormap.Height;
                }
            }
        }
    }
}

public Bitmap NextFrame()
{
    Bitmap frame = new Bitmap(width, width);
    Color target;
    frameIndex = frameIndex * width + width >= height ? 0 : frameIndex + 1;

    for (int i = 0; i < width; i++)
    {
        for (int j = 0; j < width; j++)
        {
            target = mapAnim.GetPixel(i, (frameIndex * width) + j);
            frame.SetPixel(i, j, texture.GetPixel(target.R, target.G));
        }
    }
    return frame;
}

```

Рис. 2. Генерація кінцевої анімації і рендер кадру анімації з підстановкою розгортки

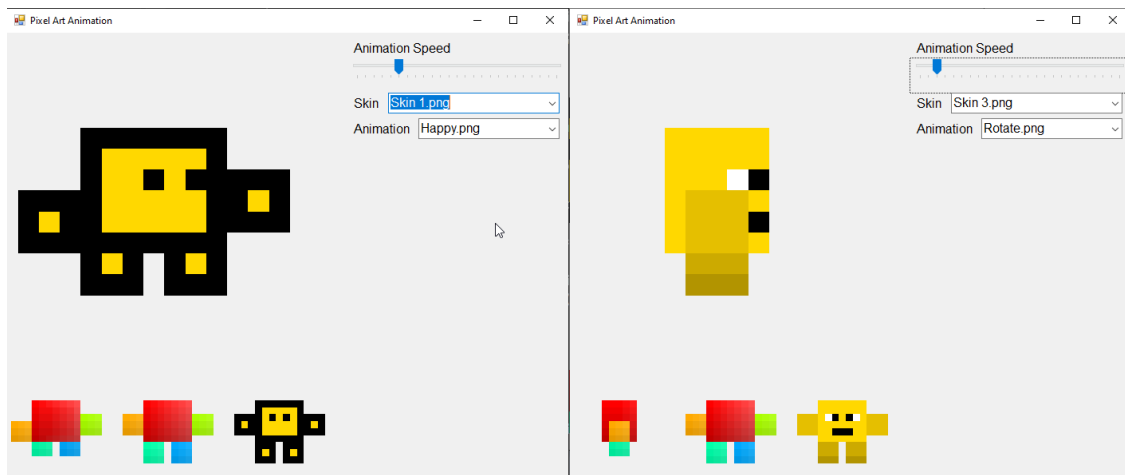


Рис. 3. Додаток для демонстрації описаного в тезі методу анімації піксель-арту

**Висновки:** Описано метод створення анімації для піксель-арт, яка суттєво зменшує обсяг роботи художників. Також вона дозволяє розробникам змінювати вигляд персонажа в реальному часі, змінюючи лише його розгортку, яка автоматично застосовується для всіх анімацій. Потрібно відмітити, що метод дозволяє скоротити обсяг обчислювальних ресурсів.

**Список використаних джерел:**

1. The VG Resource. URL: <https://www.sprites-resource.com/snes/legendofzeldaalinktothepast/sheet/7584/>.
2. Pixel Art Animation. Reinvented. YouTube. URL: <https://www.youtube.com/watch?v=HsOKwUwL1bE>
3. PixelArtAnimation. URL: <https://github.com/TimothySpaceman/PixelArtAnimation> (дата звернення: 04.05.2023).

## **ПЛАТФОРМА ОПРАЦЮВАННЯ ТА ПРОГНОЗУВАННЯ РЕЗУЛЬТАТІВ БІОМЕДИЧНИХ ПОКАЗНИКІВ**

Платформа опрацювання та прогнозування результатів біомедичних показників може бути побудована за допомогою різних методів та алгоритмів машинного навчання. Основна мета платформи полягає в тому, щоб автоматизувати процес обробки та аналізу біомедичних даних та забезпечити точне прогнозування результатів.

Процеси в медицині стали можливі для програмування вже наприкінці 50-х років ХХ сторіччя. З'явилися перші діагностичні алгоритми – програми, у яких було використано бальний принцип (підрахунок балів за отримані на запитання відповіді) кількісного оцінювання кожного симптому (запитанні!). Бальна діагностика – відомий метод у медицині. Певний Час діагностичні програми розроблялися в цьому напрямку. І всі програми мали консультативний характер, тому що не могли забезпечити високу ймовірність встановити правильний діагноз у зв'язку з відсутністю повної інформації про стан здоров'я пацієнта (наприклад, лабораторні й інструментальні дослідження). На ПК така програма представлена у формі тестового опитування.

Мета кожного запитання – отримати інформацію про симптоми, наявні або відсутні в обстежуваного. Модуль збирання інформації побудовано за принципом діалогу – на запитання треба відповідати тільки «так» чи «ні». Ця інформація формує базу даних та обробляється програмним модулем обробки й аналізу даних, після чого на екран подається діагностичне повідомлення. Сучасні вдосконалені бальні методики використовують у скринінгових системах.

Досконаліші програми-тести використовують інші методики діагностики (не бальні), включаючи в модуль обробки й аналізу даних імовірісно-статистичні методи (наприклад, імовірісний метод, що полягає в обчисленні імовірностей захворювання за формулою Баєса). Такі діагностичні програми називаються імовірісними. Імовірність діагностики в них вища, ніж у бальних програм. Вони так само є консультативними, тому що вирішальне слово залишається за лікарем.

Згодом з'являються діагностичні програми такого рівня, у яких алгоритм встановлення діагнозу наближений до мислення лікаря, тобто вони мають такий програмний логічний апарат, що дає змогу порівняти з існуючим медичним досвідом симптоми, виявлені при обстеженні хворого, а також швидко виконати складну статистичну обробку клінічного матеріалу. Такі системи називаються експертними.

Якщо в імовірісних діагностичних програмах дані оброблялися на основі бази даних, то в експертних уже присутні якісно нові бази даних, названі базами знань. Для обробки й аналізу даних база знань використовує досить складний математичний апарат. Таким чином, до початку 70-х років ХХ століття формуються різні типи напрямків автоматизованих систем діагностики захворювань і прогнозування результатів їх лікування, серед яких виділяють скринінгові та експертні системи.

Наразі персоналізована медицина з даними орієнтованими на пацієнта набуває все більшого значення з багатьох причин, серед яких: підвищення точності процесу обробки медичних даних для пошуку персоналізованих рішень, що впливає на забезпечення якісного та ефективного медичного обслуговування; можливість прогнозування зміни стану конкретного пацієнта тощо.

Один з можливих методів побудови такої платформи – це використання нейронних мереж. Нейронні мережі – це комп'ютерні системи, які можуть вчитися з даних, знаходити складні зв'язки та здійснювати прогнози на основі цих зв'язків. Для побудови платформи можна використовувати різні типи нейронних мереж, такі як глибокі нейронні мережі, рекурентні нейронні мережі, згорткові нейронні мережі та інші.

Над створенням продуктів з використанням штучного інтелекту для установ охорони області працюють розробники великих компаній, зокрема Microsoft, Apple, Google і IBM. За підрахунками аналітиків, таких фірм у світі вже 800. Незважаючи на те, що інноваційні технології тільки почали впроваджуватися у сферу медицини, згідно з дослідженням Frost & Sullivan стверджують, що цей ринок зростає до 6,16 мільярда доларів при складеному річному темпі зростання (CAGR) 68,55 % між 2018 і 2022 роками.

Одним з ключових етапів побудови платформи є збір та підготовка даних. Це може включати в себе збір даних зі сенсорів та медичних приладів, а також їх передачу до центральної системи для подальшого аналізу та обробки. Після збору даних необхідно їх підготувати, зокрема, очистити від шуму та артефактів, нормалізувати, зменшити розмірність та ін.

Наступним етапом є вибір та налаштування алгоритмів машинного навчання для аналізу даних та прогнозування результатів. Для цього можна використовувати різні алгоритми, такі як алгоритми класифікації, регресії, кластеризації та інші.

Після налаштування алгоритмів проводиться їх тренування на підготовлених даних. Цей процес полягає у тому, що модель машинного навчання вчитиметься з даних та буде знаходити закономірності, які допоможуть їй прогнозувати результати. Після тренування модель може бути використана для аналізу нових даних та прогнозування результатів.

Окрім того, платформа може включати в себе інші функції, такі як візуалізація даних, статистичний аналіз, інтеграція з іншими медичними системами та ін. Всі ці функції допомагають зробити платформу більш корисною та ефективною для лікарів та інших медичних фахівців.

Загалом, використання таких платформ може значно полегшити роботу медичних фахівців та покращити якість діагностики та лікування пацієнтів. Крім того, платформа може забезпечувати можливість використання різних типів моделей машинного навчання для різних завдань, таких як класифікація, регресія, кластеризація та інші. Це дозволяє підібрати оптимальний метод для кожного конкретного випадку.

Платформа опрацювання та прогнозування результатів біомедичних показників може бути використана в різних сферах медицини, включаючи дослідження, діагностику та лікування захворювань, моніторинг стану пацієнтів та багато іншого.

Які переваги платформи опрацювання та прогнозування результатів біомедичних показників?

- Швидка та точна обробка великої кількості даних.
- Використання різних типів моделей машинного навчання для різних завдань.
- Покращення рівня медичної допомоги та зменшення ризику розвитку серйозних захворювань.
- Моніторинг стану пацієнтів та покращення ефективності лікування.
- Можливість використання в різних сферах медицини, включаючи дослідження, діагностику та лікування захворювань.

Отже, розробка / вдосконалення такої платформи може допомогти покращити рівень медичної допомоги та зменшити ризик розвитку серйозних захворювань, забезпечивши обстеження на початкових етапах розвитку патологій.

#### **Список використаних джерел:**

1. Dzhebrailova T.D., Korobeinikova E. N. Autonomic correlates of individual differences in human intellectual activity / T. D. Dzhebrailov, I. I. Korobeinikov, E. N. Dudnik et al. *Human Physiology*. 2013. Vol. 39, Issue 1. P. 78-85.
2. Design and Evaluation of a Learner-Centric Immersive Virtual Learning Environment for Physics Education / J. Pirker, C. Gütl, J. W. Belcher, P. H. Bailey. *Human Factors in Computing and Informatics* / ed. : A. Holzinger et al. Vol. 7946. Berlin : Springer, 2013. P. 551–561.
3. Successful education for AEC professionals: case study of applying immersive game-like virtual reality interfaces / Pour Rahimian F., Arciszewski T., Goulding, J. S. *Vis. in Eng.* 2014. Vol.
4. Кальниш В. В., Левченко В. В. Комп'ютерне моделювання емоційних станів. *Мед. інформатика та інженерія*. 2019. № 3. С. 34– 39.
5. Good, B. M., Ainscough, B. J., McMichael, J. F., Su, A. I., & Griffith, O. L. (2014).
6. Organizing knowledge to enable personalization of medicine in cancer. *Genome biology*, 15(8), 1-9.
7. 'Personalized Medicine'. *Genome.Gov*, <https://www.genome.gov/geneticsglossary/Personalized-Medicine>.
8. 'Personalized Employee Experience Can Help with L&D Goals'. *Gartner*, <https://www.gartner.com/smarterwithgartner/3-ways-personalization-can-improvethe-employee-experience>.
9. Cambridge Dictionary | English Dictionary, Translations & Thesaurus. <https://dictionary.cambridge.org/>.
10. What Is Personalization and What Does It Mean to Digital Marketers? 29 Nov. 2018, <https://instapage.com/blog/what-is-personalization>.

**А.М. Концидайло, аспірант**  
**Д.Д. Плечистий, к.т.н., доц.**  
**Ю.О. Кубрак, к.т.н., доц.**  
**І.А. Толстой, ст. викладач**

*Державний університет «Житомирська політехніка»*

## **ВИКОРИСТАННЯ SELENIUM ДЛЯ АВТОМАТИЗАЦІЇ НАКЛАДАННЯ ЕЛЕКТРОННИХ ПІДПИСІВ**

В рамках цифрової трансформації в Україні у діяльності підприємств та організацій активно починають застосовуватися новітні технології [1]. Важливою складовою цифрової трансформації є застосування електронних підписів [2].

Для накладання електронних підписів в Україні існує декілька державних вебсервісів, серед яких найбільш популярними є <https://czo.gov.ua/sign> та <https://ca.djia.gov.ua/sign>. Використання даних сервісів передбачає вибір файла або носія електронного підпису, вибір формату електронного підпису, вибір файла, який має бути підписаний, скачування підписаного файла. Для того, щоб накласти окремі електронні підписи на множину файлів доводиться окремо підписувати кожний файл за допомогою перерахованих вище вебсервісів.

Для розв'язання проблеми підписання набору файлів електронними підписами пропонується програмний продукт, який дозволяє автоматизувати підписування файлів, які знаходяться у вказаному каталозі.

Для розробки програмного продукту використано Microsoft .NET (C#) та Selenium. Спочатку за допомогою Selenium IDE було записано послідовність дій для підпису одного файла за допомогою вебсервісу <https://ca.djia.gov.ua/sign> та експортовано його у вигляді C#-коду. Далі було написано основний код віконного додатку, що дозволяє вибирати папку на диску, яка містить файли, які потрібно підписати, вибрати файл та пароль до електронного підпису і запустити підписування документів.

Згенерований у Selenium IDE код було значно вдосконалено, оскільки він не передбачав, що деякі кроки виконуються не миттєво, а займають певний час. Тому для очікування появи окремих елементів інтерфейсу застосовувалися такі методи Selenium WebDriver:

1. Створення об'єкта та встановлення максимального інтервалу часу для очікування появи елемента на сторінці:

```
WebDriverWait wait = new WebDriverWait(webDriver, TimeSpan.FromSeconds(15));
```

2. Очікування, коли елемент стане видимим та клік на ньому:

```
wait.Until(ExpectedConditions.ElementIsVisible(By.CssSelector("СЕЛЕКТОР"))).Click();
```

3. Очікування появи елемента у DOM-структурі сторінки та передача йому певного тексту (використовувалося для полів введення):

```
IWebElement inputFile =  
wait.Until(ExpectedConditions.ElementExists(By.Id("СЕЛЕКТОР")));  
inputFile.SendKeys("ТЕКСТ");
```

На початковому етапі також було встановлено через Selenium WebDriver налаштування для браузера Google Chrome каталог за замовленням для автоматичного зберігання файлів та відключено появу запиту на збереження файлів. Це дозволило автоматизувати збереження підписаних файлів у потрібному каталозі:

```
ChromeOptions options = new ChromeOptions();  
options.AddUserProfilePreference("download.default_directory", TempPath);  
options.AddUserProfilePreference("download.prompt_for_download", false);
```

Описані дії дозволили розробити віконний додаток, який дозволив автоматизувати підписування набору файлів електронними підписами через офіційний вебсервіс <https://ca.djia.gov.ua/sign>.

### **Список використаних джерел:**

1. Краковська А., Бабик М. Цифровізація адміністративних послуг в Україні: проблеми та перспективи розвитку // Науковий вісник Ужгородського Національного Університету, 2022, С. 329–334.
1. Губарев С.В., Лов'як О.О. Електронний підпис як складова електронних правочинів // Вчені записки ТНУ імені В.І. Вернадського. Серія: юридичні науки, Т. 33 (72). – № 4 2022. – С. 18–23.