ДОДАТОК А

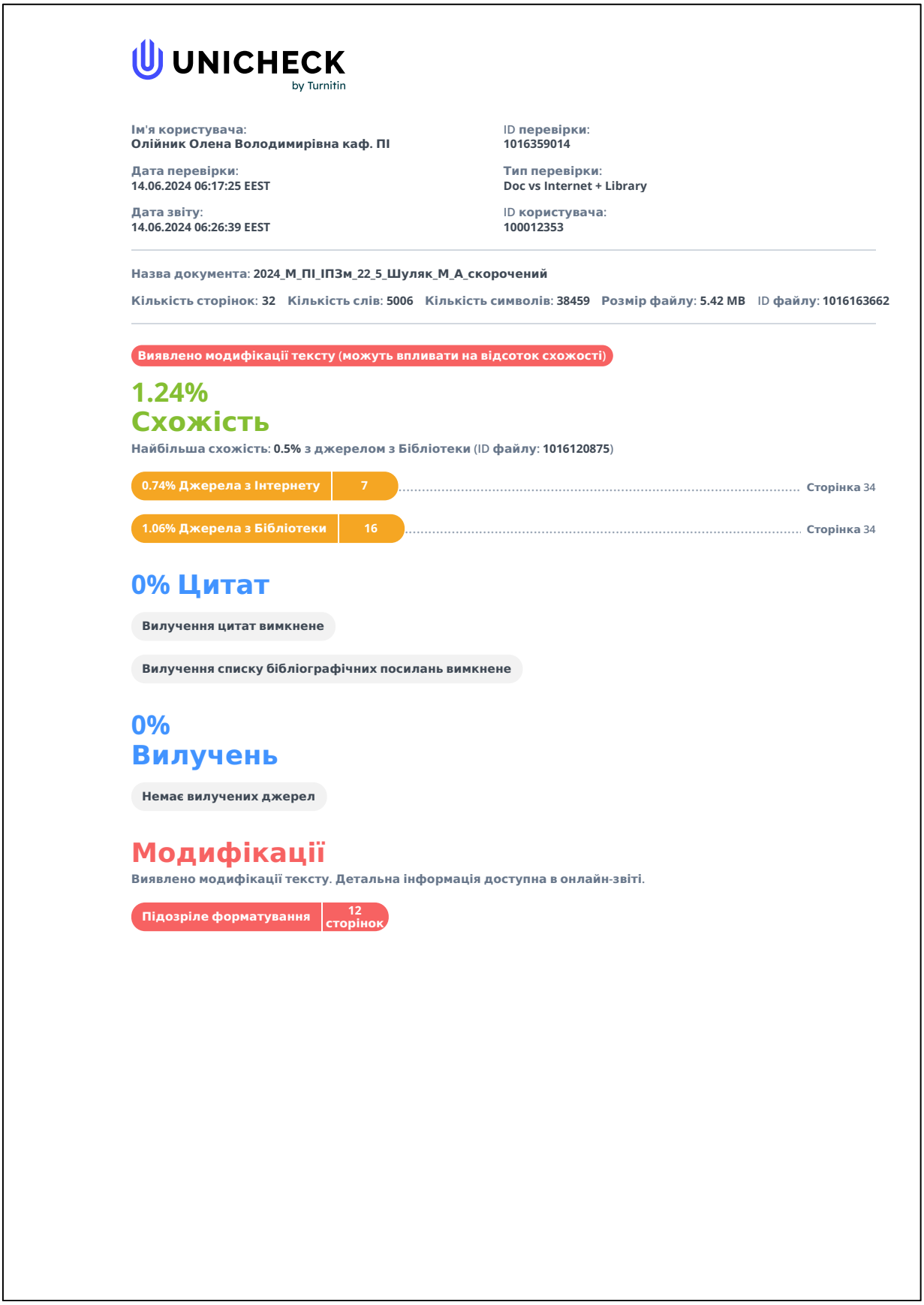
Перелік джерел посилання за науковими напрямами керівника та науковців кафедри програмної інженерії

1. Iryna Gruzdo, Iryna Kyrychenko, Glib Tereshchenko та Oleksandr Shanidze, “Analysis of Models Usability Methods Used on Design Stage to Increase Site Optimization”, CEUR Workshop Proceedings, 2023, 3403, pp. 387–409. URL: https://ceur-ws.org/Vol-3403/paper31.pdf

2. Iryna Gruzdo, Iryna Kyrychenko, Glib Tereshchenko та Nadiya Shanidze, “Metrics applicable for evaluating software at the design stage” // 5th International Conference on Computational Linguistics and In-telligent Systems (COLINS-2021), Kharkiv, Ukraine, April 22-23, 2021. – CEUR Workshop Proceedings, 2021, 2870, Volume I, pp. 916-936. URL: https://ceur-ws.org/Vol-2870/paper69.pdf

ДОДАТОК Б

Звіт результатів перевірки на унікальність тексту в базі



UNICHECK
by Turnitin

Ім'я користувача:
Олійник Олена Володимирівна каф. ПІ

ID перевірки:
1016359014

Дата перевірки:
14.06.2024 06:17:25 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
14.06.2024 06:26:39 EEST

ID користувача:
100012353

Назва документа: 2024_М_ПІ_ІПЗм_22_5_Шуляк_М_А_скорочений

Кількість сторінок: 32 **Кількість слів:** 5006 **Кількість символів:** 38459 **Розмір файлу:** 5.42 MB **ID файлу:** 1016163662

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

1.24%
Схожість

Найбільша схожість: 0.5% з джерелом з Бібліотеки (ID файлу: 1016120875)

0.74% Джерела з Інтернету | 7 ..... Сторінка 34

1.06% Джерела з Бібліотеки | 16 ..... Сторінка 34

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

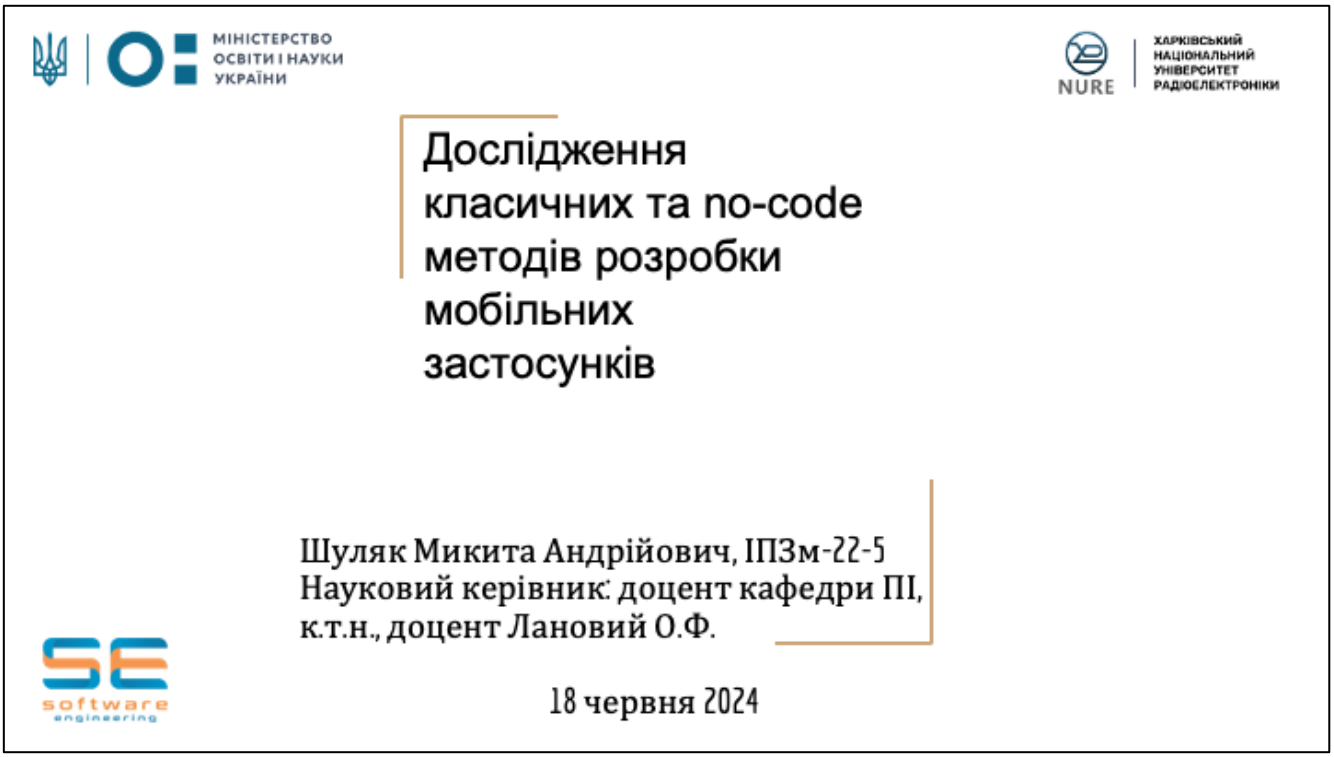
Модифікації

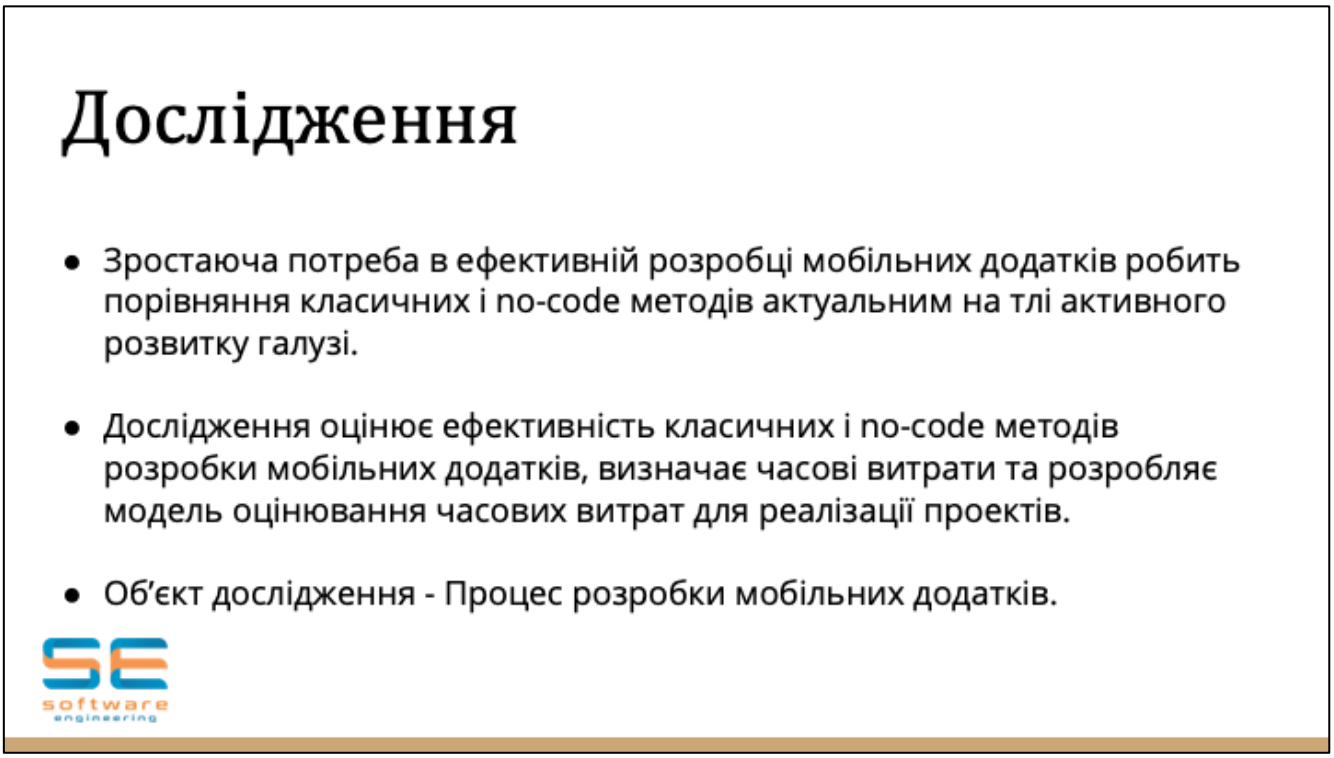
Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Підозріле форматування | 12 сторінок

ДОДАТОК В

Слайди презентації





Огляд літератури (аналогів)

Основні джерела:

- Varajão, João, António Trigo, and Miguel Almeida. 2023. "Low-Code Development Productivity: 'Is Winter Coming' for Code-Based Technologies?»
- Yan Zhaohang. "The Impacts of Low/No-Code Development on Digital Transformation and Software Development"

Порівняння продуктивності

TABLE 2: **PRODUCTIVITY DIFFERENCES (WITHOUT QF)**

TECHNOLOGY	CODE-BASED	LOW-CODE	EXTREME LOW-CODE
Code-Based	-	0.330 x	0.113 x
Low-Code	3.031 x	-	0.342 x
Extreme Low-Code	8.849 x	2.920 x	-

TABLE 3: **PRODUCTIVITY DIFFERENCES (WITH QF)**

TECHNOLOGY	CODE-BASED	LOW-CODE	EXTREME LOW-CODE
Code-Based	-	0.315 x	0.102 x
Low-Code	3.176 x	-	0.323 x
Extreme Low-Code	9.823 x	3.093 x	-

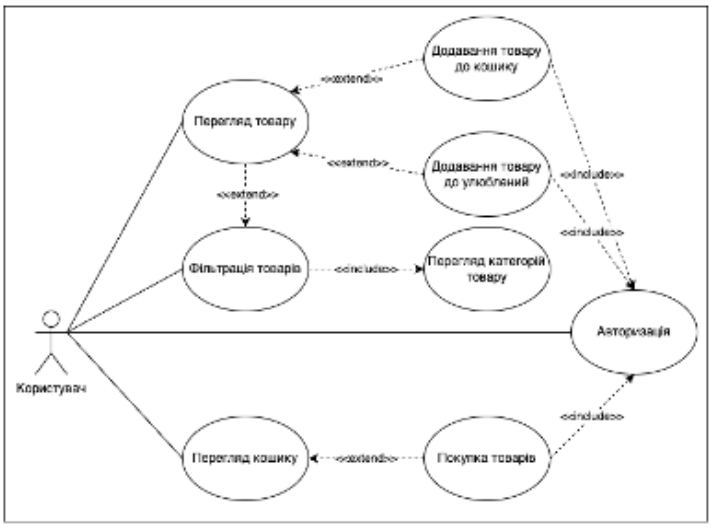
Постановка задачі

- Проблема - невизначеність часових витрат на розробку різного функціоналу мобільних проектів за допомогою no-code та low-code методів у порівнянні з класичними методами
- У ході дослідження будуть виявлені переваги та недоліки кожного підходу для різних типів функціоналу

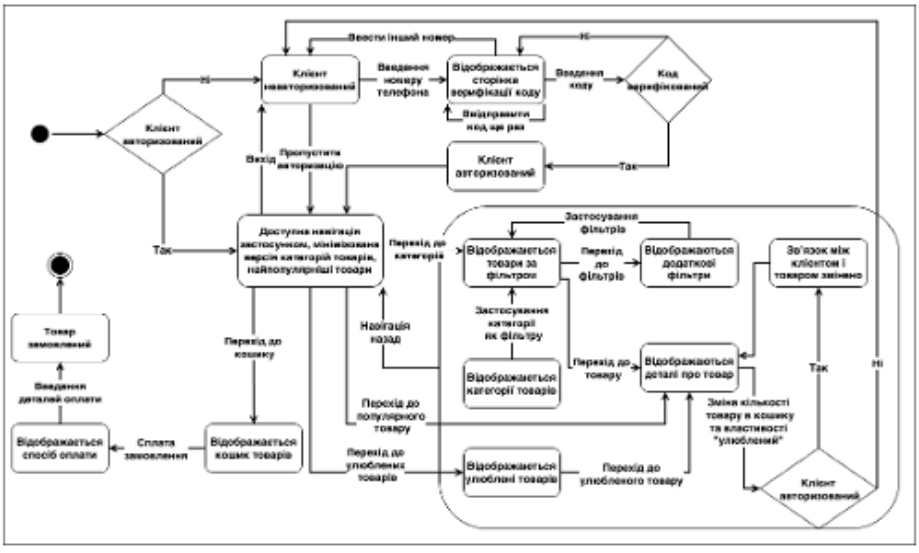
Методологія

- Порівняльний аналіз для оцінки ефективності різних підходів до розробки
- Емпіричні методи для вимірювання часових витрат на розробку
- Математичне моделювання для виведення формул оцінки вартості розробки

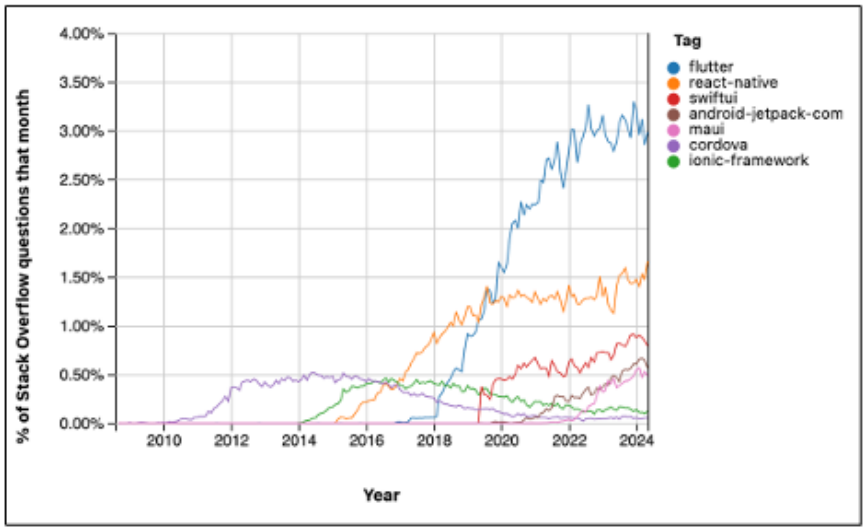
Діаграми для проведення експериментального дослідження



Діаграма станів



Опис програмного забезпечення, що було використано у дослідженні



Опис програмного забезпечення, що було використано у дослідженні

- Початкова сторінка
- Авторизація та реєстрація за номером мобільного телефону
- Навігація додатком
- Сторінка з головними продуктами
- Сторінка категорій товарів
- Фільтрація продуктів
- Сторінка користувача
- Улюблені продукти
- Кошик з обраними продуктами

Flutter

FlutterFlow

Зміст проведеного експерименту

Назва функціоналу	Властивість			Класичний метод, години	No-code метод, години
	Присутнє підключення до API	Присутня бізнес логіка	Присутній складний інтерфейс		
Початкова сторінка	Ні	Ні	Ні	2,82	2,11
Аутентифікація за номером телефона	Так	Ні	Ні	12,32	4,37
Елемент навігації застосунком	Ні	Ні	Так	6,33	7,54
Головна сторінка	Так	Ні	Так	5,2	7,34
Сторінка категорій	Так	Ні	Ні	8,9	3,26
Фільтрація товарів	Так	Так	Ні	26,04	29,48

Назва функціоналу	Властивість			Класичний метод, години	No-code метод, години
	Присутнє підключення до API	Присутня бізнес логіка	Присутній складний інтерфейс		
Деталі продукту	Так	Ні	Ні	16,98	9,68
Профіль користувача	Ні	Ні	Ні	5,97	2,29
Сторінка з улюбленими товарами	Так	Так	Ні	6,39	6,22
Кошик	Так	Ні	Ні	8,12	4,76
Пупка товарів	Так	Ні	Так	5	7,95

Результати експерименту

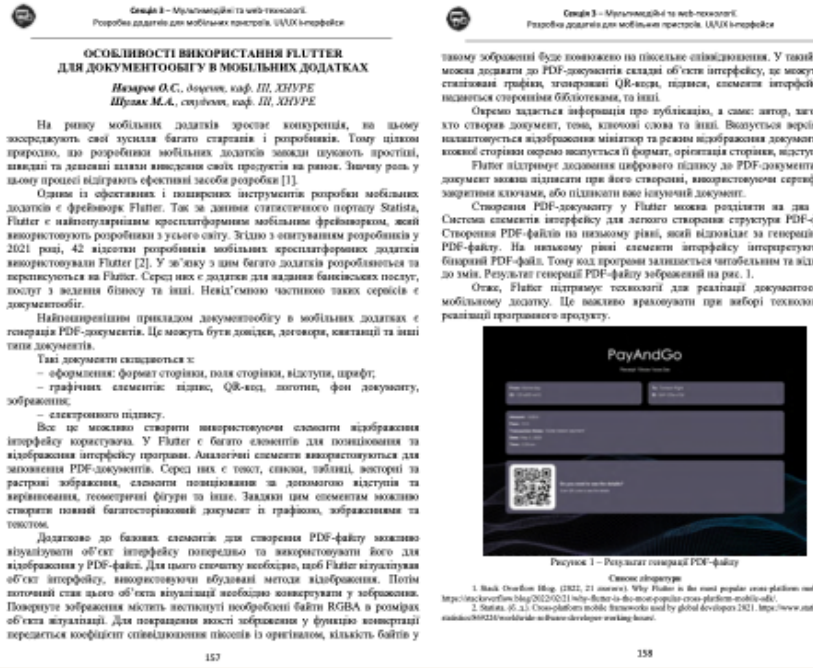
Порівняння no-code з класичним методом з приводу часу розробки:

- Функціонал без залежностей – швидше на 53%
- Функціонал з підключенням API – швидше на 50%
- Функціонал з бізнес логікою – довше на 121%
- Функціонал з складним інтерфейсом – довше на 117%

Аналіз отриманих результатів

Назва функціоналу	Класичний метод, години	No-code метод, години	No-code метод, очікуваний результат, години
Початкова сторінка	2,82	2,11	1,41
Аутентифікація за номером телефона	12,32	4,37	5,86
Елемент навігації застосунком	6,33	7,54	13,75
Головна сторінка	5,2	7,34	5,38
Сторінка категорій	8,9	3,26	4,24
Фільтрація товарів	26,04	29,48	27,39
Деталі продукту	16,98	9,68	8,08
Профіль користувача	5,97	2,29	2,99
Сторінка з улюбленими товарами	6,39	6,22	6,72
Кошик	8,12	4,76	3,87
Пупка товарів	5	7,95	5,17

Апробація



Секція 3 – Мультимедійні та web-технології.
Розробка додатків для мобільних пристроїв. UI/UX інтерфейси

ОСОБЛИВОСТІ ВИКОРИСТАННЯ FLUTTER ДЛЯ ДОКУМЕНТООБІГУ В МОБІЛЬНИХ ДОДАТКАХ

***Назаров О.С.**, доцент, каф. ІП, ХНУРЕ*
***Шуляк М.А.**, студент, каф. ІП, ХНУРЕ*

На ринку мобільних додатків зростає конкуренція, на цьому зосереджують свої зусилля багато стартапів і розробників. Тому цілком природно, що розробники мобільних додатків зокрема шукають простішні, швидші та дешевші шляхи виведення своїх продуктів на ринок. Значну роль у цьому процесі відіграють ефективні засоби розробки [1].

Одним із ефективних і поширених інструментів розробки мобільних додатків є фреймворк Flutter. Так за даними статистичного порталу Statista, Flutter є найпопулярнішим кроссплатформним мобільним фреймворком, який використовують розробники з усього світу. Згідно з опитуванням розробників у 2021 році, 42 відсотки розробників мобільних кроссплатформних додатків використовували Flutter [2]. У зв'язку з цим багато додатків розробляються та переписуються на Flutter. Серед них є додатки для надання банківських послуг, послуг з ведення бізнесу та інші. Невід'ємною частиною таких сервісів є документообіг.

Найпоширенішим прикладом документообігу в мобільних додатках є генерація PDF-документів. Це можуть бути довідки, договори, квитанції та інші типи документів.

Такі документи складаються з:
– оформлення: формат сторінки, поля сторінки, відступи, шрифт;
– графічних елементів: підпис, QR-код, логотип, фон документу, зображення;
– електронного підпису.

Все це можливо створити використовуючи елементи відображення інтерфейсу користувача. У Flutter є багато елементів для позиціонування та відображення інтерфейсу програми. Аналогічні елементи використовуються для заповнення PDF-документів. Серед них є текст, списки, таблиці, векторні та растрові зображення, елементи позиціювання за допомогою відступів та вирівнювання, геометричні фігури та інше. Завдяки цим елементам можливо створити повний багатосторінковий документ із графікою, зображеннями та текстом.

Додатково до базових елементів для створення PDF-файлу можливо візуалізувати об'єкт інтерфейсу попередньо та використовувати його для відображення у PDF-файлі. Для цього спочатку необхідно, щоб Flutter візуалізував об'єкт інтерфейсу, використовуючи вбудовані методи відображення. Потім поточний стан цього об'єкта візуалізації необхідно конвертувати у зображення. Повернуте зображення містить нестиснуті необроблені байти RGBA в розмірах об'єкта візуалізації. Для покращення якості зображення у функцію конвертації передається коефіцієнт співвідношення пікселів із оригіналом, кількість байтів у

Секція 3 – Мультимедійні та web-технології.
Розробка додатків для мобільних пристроїв. UI/UX інтерфейси

такому зображенні буде пов'язано із пікселями співвідношення. У такий спосіб можна дізнатися до PDF-документів складові об'єкти інтерфейсу, це можуть бути стилізовані графіки, згенеровані QR-коди, підписи, елементи інтерфейсу, що надаються сторонніми бібліотеками, та інші.

Окремо задається інформація про публікацію, а саме: автор, заголовок, хто створив документ, теми, ключові слова та інші. Вказується версія PDF, налаштовується відображення мініатюр та режим відображення документу. Для кожної сторінки окремо вказується її формат, орієнтація сторінки, відступи.

Flutter підтримує додавання цифрового підпису до PDF-документа. PDF-документ можна підписати при його створенні, використовуючи сертифікат із закритими ключами, або підписати вже існуючий документ.

Створення PDF-документу у Flutter можна розділити на два етапи. Система елементів інтерфейсу для легкого створення структури PDF-файлів. Створення PDF-файлів по низькому рівні, який відповідає за генерацію бітів PDF-файлу. На низькому рівні елементи інтерфейсу інтерпретуються у бінарний PDF-файл. Тому код програми залишається читабельним та відкритим до змін. Результат генерації PDF-файлу зображений на рис. 1.

Отже, Flutter підтримує технології для реалізації документообігу в мобільному додатку. Це важливо враховувати при виборі технології для реалізації програмного продукту.

Рисунок 1 – Результат генерації PDF-файлу

Список літератури

1. Stack Overflow Blog. (2022, 21 листопад). Why Flutter is the most popular cross-platform mobile SDK. https://stackoverflow.blog/2022/02/21/why-flutter-is-the-most-popular-cross-platform-mobile-sdk/.
2. Statista. (б. д.). Cross-platform mobile frameworks used by global developers 2021. https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/.

Висновки

- Результати співвідносяться з іншими дослідженнями, але додають ясність в областях, які раніше не були досліджені
- Можливість більш точно прогнозувати часові витрати
- У подальшому можна дослідити інші технології no-code та класичних методів розробки, описати інші функціональні особливості додатки та вплив технологій на швидкість їх реалізації

ДОДАТОК Г

Апробація результатів роботи

Секція 3 – Мультимедійні та web-технології.
Розробка додатків для мобільних пристроїв. UI/UX інтерфейси

ОСОБЛИВОСТІ ВИКОРИСТАННЯ FLUTTER ДЛЯ ДОКУМЕНТООБІГУ В МОБІЛЬНИХ ДОДАТКАХ

***Назаров О.С.**, доцент, каф. ПІ, ХНУРЕ*
***Шуляк М.А.**, студент, каф. ПІ, ХНУРЕ*

На ринку мобільних додатків зростає конкуренція, на цьому зосереджують свої зусилля багато стартапів і розробників. Тому цілком природно, що розробники мобільних додатків завжди шукають простіші, швидші та дешевші шляхи виведення своїх продуктів на ринок. Значну роль у цьому процесі відіграють ефективні засоби розробки [1].

Одним із ефективних і поширених інструментів розробки мобільних додатків є фреймворк Flutter. Так за даними статистичного порталу Statista, Flutter є найпопулярнішим кросплатформним мобільним фреймворком, який використовують розробники з усього світу. Згідно з опитуванням розробників у 2021 році, 42 відсотки розробників мобільних кросплатформних додатків використовували Flutter [2]. У зв'язку з цим багато додатків розробляються та переписуються на Flutter. Серед них є додатки для надання банківських послуг, послуг з ведення бізнесу та інші. Невід'ємною частиною таких сервісів є документообіг.

Найпоширенішим прикладом документообігу в мобільних додатках є генерація PDF-документів. Це можуть бути довідки, договори, квитанції та інші типи документів.

Такі документи складаються з:

– оформлення: формат сторінки, поля сторінки, відступи, шрифт;

– графічних елементів: підпис, QR-код, логотип, фон документу, зображення;

– електронного підпису.

Все це можливо створити використовуючи елементи відображення інтерфейсу користувача. У Flutter є багато елементів для позиціювання та відображення інтерфейсу програми. Аналогічні елементи використовуються для заповнення PDF-документів. Серед них є текст, списки, таблиці, векторні та растрові зображення, елементи позиціювання за допомогою відступів та вирівнювання, геометричні фігури та інше. Завдяки цим елементам можливо створити повний багатосторінковий документ із графікою, зображеннями та текстом.

Додатково до базових елементів для створення PDF-файлу можливо візуалізувати об'єкт інтерфейсу попередньо та використовувати його для відображення у PDF-файлі. Для цього спочатку необхідно, щоб Flutter візуалізував об'єкт інтерфейсу, використовуючи вбудовані методи відображення. Потім поточний стан цього об'єкта візуалізації необхідно конвертувати у зображення. Повернуте зображення містить нестиснуті необроблені байти RGBA в розмірах об'єкта візуалізації. Для покращення якості зображення у функцію конвертації передається коефіцієнт співвідношення пікселів із оригіналом, кількість байтів у

такому зображенні буде помножено на піксельне співвідношення. У такий спосіб можна додавати до PDF-документів складні об'єкти інтерфейсу, це можуть бути стилізовані графіки, згенеровані QR-коди, підписи, елементи інтерфейсу, що надаються сторонніми бібліотеками, та інші.

Окремо задається інформація про публікацію, а саме: автор, заголовок, хто створив документ, тема, ключові слова та інші. Вказується версія PDF, налаштовується відображення мініатюр та режим відображення документу. Для кожної сторінки окремо вказується її формат, орієнтація сторінки, відступи.

Flutter підтримує додавання цифрового підпису до PDF-документа. PDF-документ можна підписати при його створенні, використовуючи сертифікат із закритими ключами, або підписати вже існуючий документ.

Створення PDF-документу у Flutter можна розділити на два етапи. Система елементів інтерфейсу для легкого створення структури PDF-файлів. Створення PDF-файлів на низькому рівні, який відповідає за генерацію бітів PDF-файлу. На низькому рівні елементи інтерфейсу інтерпретуються у бінарний PDF-файл. Тому код програми залишається читабельним та відкритим до змін. Результат генерації PDF-файлу зображений на рис. 1.

Отже, Flutter підтримує технології для реалізації документообігу в мобільному додатку. Це важливо враховувати при виборі технології для реалізації програмного продукту.

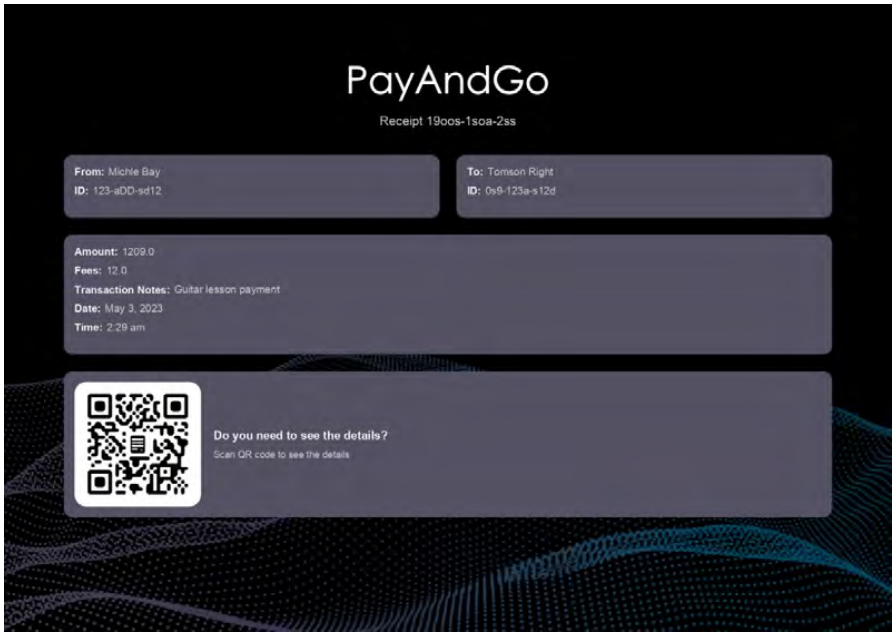


Рисунок 1 – Результат генерації PDF-файлу

Список літератури

1. Stack Overflow Blog. (2022, 21 лютого). Why Flutter is the most popular cross-platform mobile SDK. https://stackoverflow.blog/2022/02/21/why-flutter-is-the-most-popular-cross-platform-mobile-sdk/.

2. Statista. (б. д.). Cross-platform mobile frameworks used by global developers 2021. https://www.statista.com/statistics/869224/worldwide-software-developer-working-hours/.

ДОДАТОК Д

Код фільтрації товарів

```
@freezed
class CategoryModel with _$CategoryModel {
 const factory CategoryModel({
  required String id,
  required String catalogueId,
  required String name,
  @JsonKey(fromJson: DateUtils.convertToDateTime)
  required DateTime? createdAt,
 }) = _CategoryModel;

 factory CategoryModel.fromJson(Map<String, dynamic> json) =>
   _$CategoryModelFromJson(json);

 factory CategoryModel.fromJsonWithId(Map<String, dynamic> json, String id)
=>
   _$CategoryModelFromJson(<String, dynamic>{'id': id, ...json});

 const CategoryModel._();

 static const empty = CategoryModel(
  id: '',
  catalogueId: '',
  name: '',
  createdAt: null,
 );
}

@Freezed(makeCollectionsUnmodifiable: false)
class ClothingFilterPreferences with _$ClothingFilterPreferences {
 const factory ClothingFilterPreferences({
  double? minPrice,
  double? maxPrice,
  List<String>? brands,
  List<String>? colorNames,
  List<String>? sizes,
  SortType? sortType,
  SubCategoryModel? subCategoryModel,
 }) = _ClothingFilterPreferences;

 const ClothingFilterPreferences._();

 static const empty = ClothingFilterPreferences();

 bool get isNotEmpty => this != ClothingFilterPreferences.empty;
}

@freezed
class ProductModel with _$ProductModel {
 const factory ProductModel({
  required String id,
  required String categoryId,
  required String subCategoryId,
  required String brand,
```

```
  required String title,
  required double price,
  required double currentPrice,
  required bool inStock,
  required bool isFeatured,
  required List<String> mainUrls,
  required List<String> inFavorite,
  @JsonKey(fromJson: DateUtils.convertToDateTime) required DateTime
createdAt,
  String? description,
  Map<String, bool>? colorNames,
  List<String>? colorUrls,
  Map<String, bool>? sizes,
  @Default(0.0) double rating,
  double? discountPrice,
 }) = _ProductModel;

 factory ProductModel.fromJson(Map<String, dynamic> json) =>
   _$ProductModelFromJson(json);

 factory ProductModel.fromJsonWithId(Map<String, dynamic> json, String id)
=>
   _$ProductModelFromJson(<String, dynamic>{'id': id, ...json});
}

@override
Future<List<ProductModel>> fetchProductsCatalogue({
 required int limit,
 required CategoryModel categoryModel,
 required ClothingFilterPreferences filterPreferences,
 ProductModel? lastVisibleProduct,
}) async {
 return _fetchProducts(
  limit: limit,
  query: _getCatalogueClothingQuery(
   categoryModel: categoryModel,
   filterPreferences: filterPreferences,
  ),
  sortField: filterPreferences.sortType,
  lastVisibleProduct: lastVisibleProduct,
 );
}

Future<List<ProductModel>> _fetchProducts({
 required int limit,
 required Query query,
 SortType? sortField,
 ProductModel? lastVisibleProduct,
}) async {
 final lastVisibleProductLocal = lastVisibleProduct;

 final lastDocument = lastVisibleProductLocal != null
   ? await _productCollectionRef.doc(lastVisibleProductLocal.id).get()
   : null;

 final currentSortField = sortField ?? _orderField;
 final productsSnapshot = lastDocument == null
   ? await query
```

```
      .orderBy(
       currentSortField.dbField,
       descending: currentSortField.isDescending,
      )
      .limit(limit)
      .get()
    : await query
      .orderBy(
       currentSortField.dbField,
       descending: currentSortField.isDescending,
      )
      .startAfterDocument(lastDocument)
      .limit(limit)
      .get();

 return _productsSnapshotToModelList(productsSnapshot);
}

Query<Object?> _getCatalogueClothingQuery({
 required CategoryModel categoryModel,
 required ClothingFilterPreferences filterPreferences,
}) {
 final subCategoryModel = filterPreferences.subCategoryModel;

 var query = _productCollectionRef.where(
  subCategoryModel == null
    ? Filter('categoryId', isEqualTo: categoryModel.id)
    : Filter('subCategoryId', isEqualTo: subCategoryModel.id),
 );

 final brands = filterPreferences.brands;
 if (brands != null && brands.isNotEmpty) {
  query = query.where('brand', whereIn: brands);
 }

 final colorNames = filterPreferences.colorNames;
 if (colorNames != null && colorNames.isNotEmpty) {
  if (colorNames.length == 1) {
   query = query.where('colorNames.${colorNames.first}', isEqualTo: true);
  } else {
   for (final colorName in colorNames) {
    query = query.where('colorNames.$colorName', isEqualTo: true);
   }
  }
 }

 final sizes = filterPreferences.sizes;
 if (sizes != null && sizes.isNotEmpty) {
  if (sizes.length == 1) {
   query = query.where('sizes.${sizes.first}', isEqualTo: true);
  } else {
   for (final size in sizes) {
    query = query.where('sizes.$size', isEqualTo: true);
   }
  }
 }

 final minPrice = filterPreferences.minPrice;
```

```
 final maxPrice = filterPreferences.maxPrice;
 final sortType = filterPreferences.sortType;

 if (minPrice != null || maxPrice != null) {
  query = query
    .where('currentPrice', isGreaterThanOrEqualTo: minPrice)
    .where('currentPrice', isLessThanOrEqualTo: maxPrice);
  if (!(sortType == SortType.priceAsc || sortType == SortType.priceDesc)) {
   query = query.orderBy('currentPrice');
  }
 }

 return query;
}
```

ДОДАТОК Е

Експертний висновок результатів перевірки кваліфікаційної роботи на відповідність оформлення вимогам ДСТУ 3008: 2015

Експертний висновок результатів перевірки кваліфікаційної роботи

студент (посада) | програмної інженерії (кафедра) | ІПЗм-22-5 (група)

Шуляк Микита Андрійович

(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні познаки	
	7.15 Додатки	

зауважень немає

Експерт _____ (підпис) Олена ОЛІЙНИК (прізвище, ініціали)

14.06.2024