

ГЕНЕРАЦІЯ БАЗ ДАНИХ ДЛЯ РІЗНИХ МОБІЛЬНИХ ПЛАТФОРМ

БОДНЯ О.С. ТВ-51М

ПРОБЛЕМА



ПРОБЛЕМА

СХЕМА БАЗИ
ДАНИХ

СХЕМА БАЗИ
ДАНИХ (2)

СХЕМА БАЗИ
ДАНИХ (3)

РЕАЛІЗАЦІЯ
РОБОТИ З
БАЗОЮ ДАНИХ

РЕАЛІЗАЦІЯ
РОБОТИ З
БАЗОЮ ДАНИХ

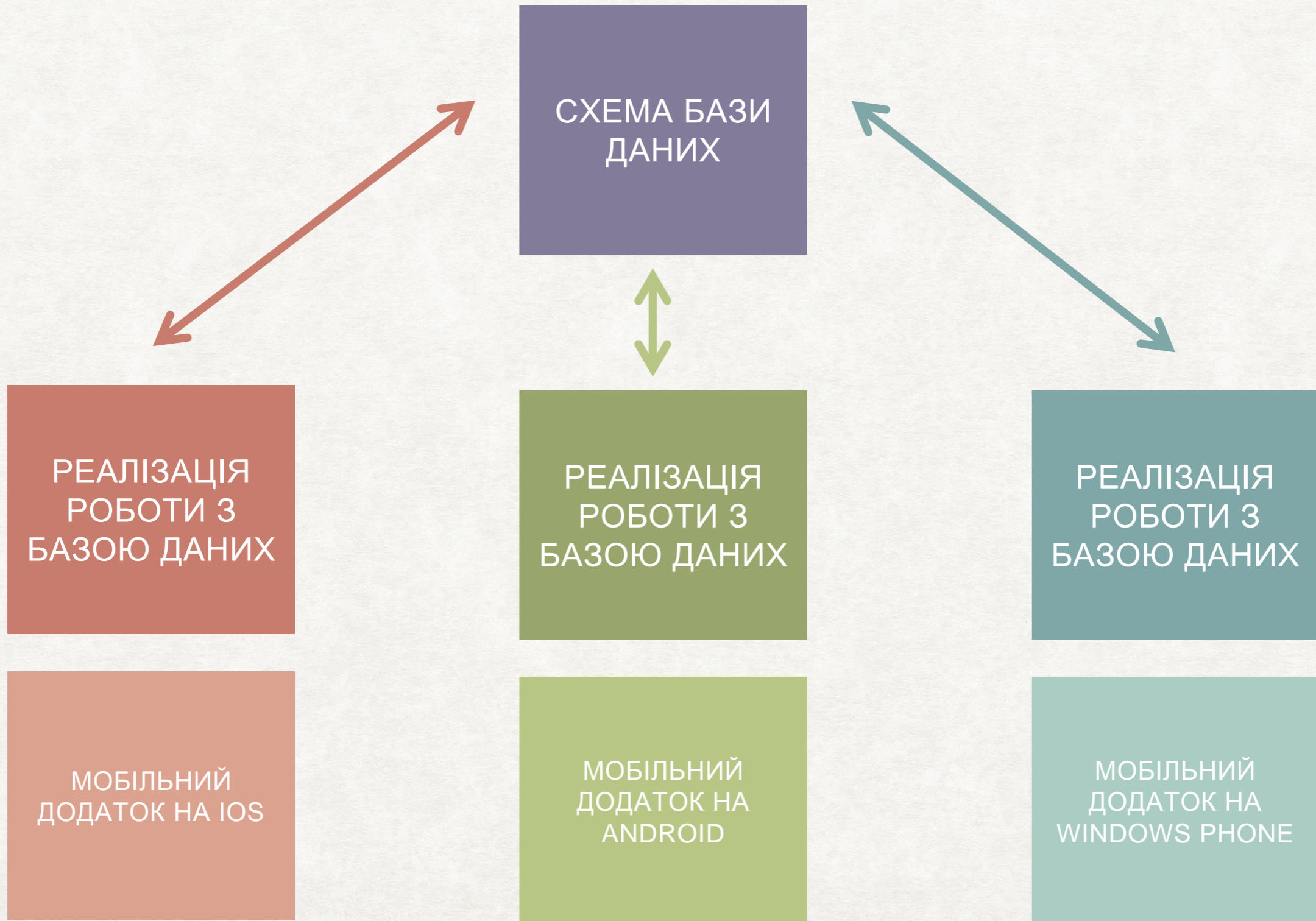
РЕАЛІЗАЦІЯ
РОБОТИ З
БАЗОЮ ДАНИХ

МОБІЛЬНИЙ
ДОДАТОК НА IOS

МОБІЛЬНИЙ
ДОДАТОК НА
ANDROID

МОБІЛЬНИЙ
ДОДАТОК НА
WINDOWS PHONE

ПРОБЛЕМА



ПРОБЛЕМА

- Схема бази даних однакова для різних мобільних клієнтів
- При поточній моделі розробки, код роботи з БД доводиться писати щоразу заново на кожній мобільній платформі
- Це займає час і ресурси, які є в обмеженому вигляді в молодих стартапах і студенських проектах

ПРОБЛЕМА



KPImaps

ПРОБЛЕМА

	2015 2кв	2015 3кв	2015 4кв	2016 1кв	2016 2кв
Готовність модельної частини	50 %	80 %	25 %	45 %	75 %
Готовність клієнт-серверної частини	80 %	95 %	75 %	75 %	75 %
Готовність UI	10 %	12 %	35 %	50 %	75 %

ПРОБЛЕМА

- При розробці версії KPImaps під Android довелось переписувати модельний код через низьку швидкодію
- Переписування бази даних вручну на основі існуючої схеми зайняло близько 7 місяців. Багато однакового по семантиці коду писалось вручну довгий час.
- На це був алокований окремий розробник
- Розробка модельної частини тормозила розробки UI та клієнт-серверної частини
- Проект так і не зарелізився, бо команда стомилась так довго над ним працювати безкоштовно, а питання моделі вже всіх дратувало

ДОСЛІДЖЕННЯ

ШЛЯХИ ВИРІШЕННЯ

- Використання засобів кросплатформеної розробки
- Використання спільного коду за рахунок зворотньої сумістності мов Java та Objective-C до мови C
- **Повторне використання схеми бази даних для генерації програмного коду на її основі за допомогою заготовлених шаблонів**

ДОСЛІДЖЕННЯ

БАЗА ДАНИХ

- Аналіз основних функцій бази даних
- Аналіз SQL мови
- Аналіз базових можливостей СУБД

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

- Буде використовуватись об'єктно-орієнтований мапінг
- SQLite не може гарантувати цілісності зв'язків, їх гарантуватиме редактор
- UI має надавати можливість CRUD на сутностях, атрибутах і зв'язках

ДОСЛІДЖЕННЯ

БАЗА ДАНИХ В МОБІЛЬНИХ ПРИСТРОЯХ

Аналіз використання бази даних на мобільних платформах:

- Патерн DAO в Android
- Бібліотека Core Data в iOS

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

- Шаблони для Android містимуть класичну реалізацію CRUD за патерном DAO
- Шаблони для Core Data міститимуть класи сутностей, а також контролер для CRUD над CoreData

DONEC QUIS NUNC

ДОСЛІДЖЕННЯ

ГЕНЕРАЦІЯ ПРОГРАМНОГО КОДУ

- Аналіз загальних практик і підходів до генерації коду
- Пошук необхідного інструменту для генерації коду

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

- Обрані шаблони Mustache для генерації коду і бібліотека GRMustache для інтеграції в програмний засіб
- Для сутностей будуть створені окремо класи Human та Machine

ДОСЛІДЖЕННЯ

ФОРМАТИ ЗБЕРЕЖЕННЯ ДАНИХ

Аналіз переваг і недоліків існуючих форматів збереження даних:

JSON; XML; CSV

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

- Для збереження схеми бази даних обрано формат JSON
- Для збереження конфігурації шаблонів обрано формат JSON

ДОСЛІДЖЕННЯ

МОВИ ПРОГРАМУВАННЯ В МОБІЛЬНИХ ДОДАТКАХ

Аналіз мов програмування, що використовуються при розробці мобільних додатків для написання шаблонів:

- Java / Kotlin для Android
- Objective-C / Swift для iOS
- C# (Xamarin) як універсальний підхід

РЕЗУЛЬТАТ ДОСЛІДЖЕННЯ

- Для шаблонів Android обрано мову Kotlin
- Для шаблогів iOS обрано мову Swift

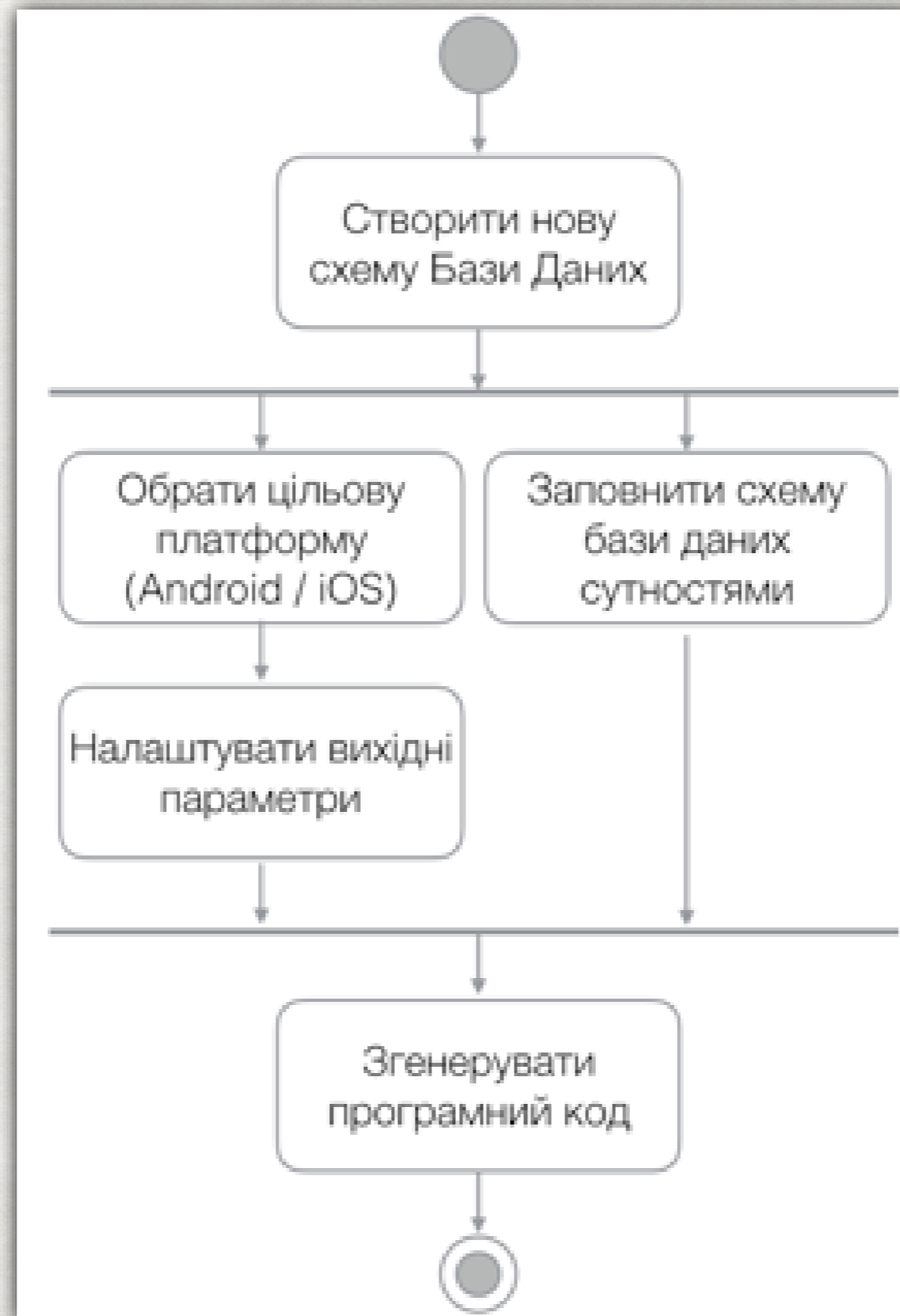
ВИРІШЕННЯ

- Маючи схему бази даних та знаючи який саме код має бути використаний для роботи з моделлю в програмному застосунку, треба вивести шаблони класів
- На основі схеми бази даних та шаблонів класів генерувати модельний код

ВИРІШЕННЯ



РОБОЧИЙ ПРОЦЕС



РЕАЛІЗАЦІЯ

DBCreator - */Users/abodnya/KPI/KMModelClassesGenerator/KMDatabaseDesigner/Resources/TestDataba...

Entities

Attributes Relationships

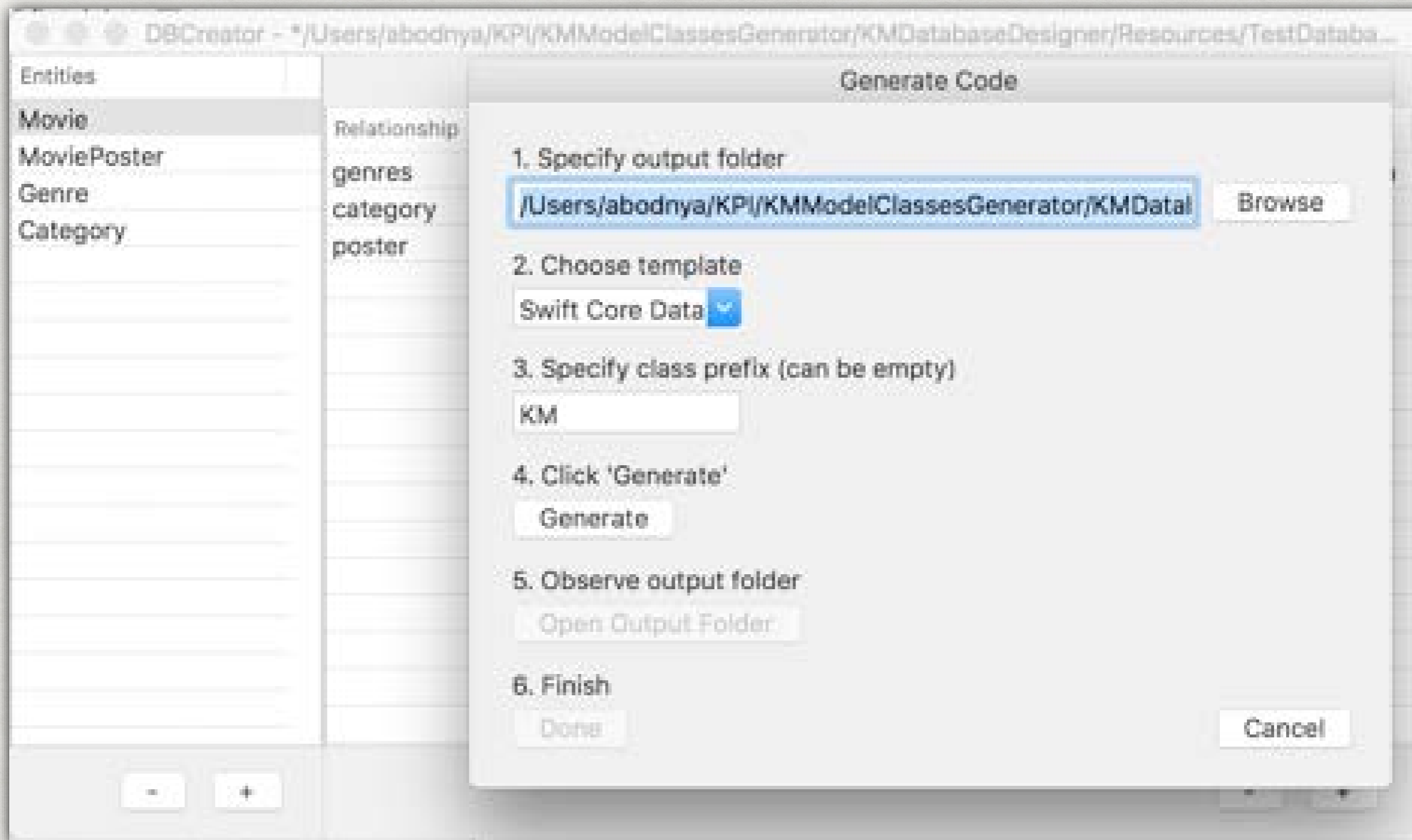
Attribute	Type	Comment
title	String	⌘ Назва фільму
year	Integer	⌘ Рік виходу фільму
description	String	⌘ Опис фільму
identifier	String	⌘ Ідентифікатор

Movie
MoviePoster
Genre
Category

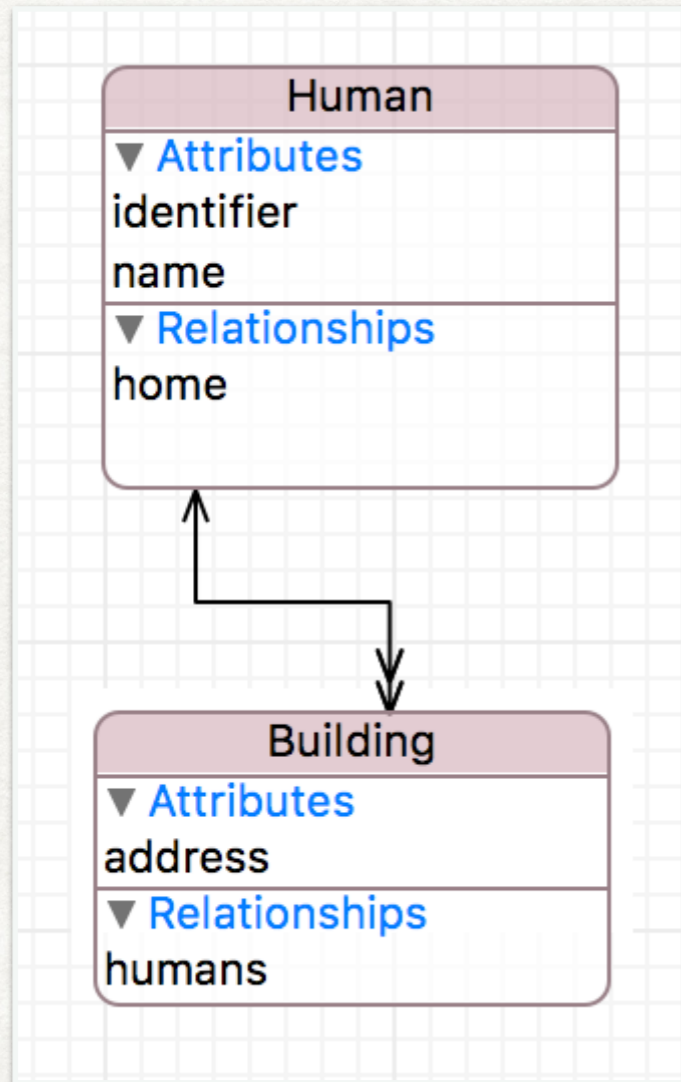
- +

- +

РЕАЛІЗАЦІЯ



ПРИКЛАД РЕАЛІЗАЦІЇ



DBCreator - /Users/abodnya/KPI/KMModelClassesGenerator/KMDatabaseDe...

Entities: Human, Building

Attributes

Attribute	Type	Comment
name	String	Full name
identifier	String	some id

DBCreator - /Users/abodnya/KPI/KMModelClassesGenerator/KMDatabaseDe...

Entities: Human, Building

Relationships

Relationship	Destination	Inverse	Type	Con
home	Building	humans	To One	

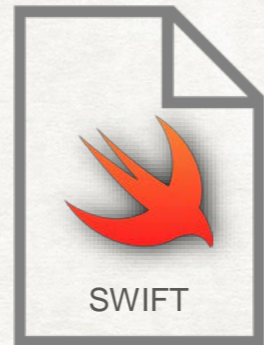
ПРИКЛАД РЕАЛІЗАЦІЇ

IOS SWIFT

DataProviders



DTLocalStorage



DTLocalStorage+
Entities



DTLocalStorage+
Context



DTLocalStorage+
Fetch

Entities



DTBuilding



DTHuman



DTBuildingMachine



DTHumanMachine

ПРИКЛАД РЕАЛІЗАЦІЇ

ANDROID KOTLIN

dao



DTAbstractDAO



DTDAOFactory



DTBuildingDAO



DTHumanDAO

entity



DTBuilding



DTHuman



DTBuildingMachine



DTHumanMachine

ПРИКЛАД РЕАЛІЗАЦІЇ

DTHumanMachine.kt

```
package model.entity
import java.util.UUID

/**
 * Created by Alexey Bodnya on 28.04.17.
 */
open class DTHumanMachine :
    DTDatabaseEntity {

    var name: String? = null
    var identifier: String? = null
    var homeId: String? = null

    constructor() {

    }

    constructor(id: String?, name:
String?, identifier: String?, home:
DTBuilding?) {
        this.id = id ?:
UUID.randomUUID().toString()
        this.name = name
        this.identifier = identifier
        this.homeId = home?.id
    }
}
```

DTHumanMachine.swift

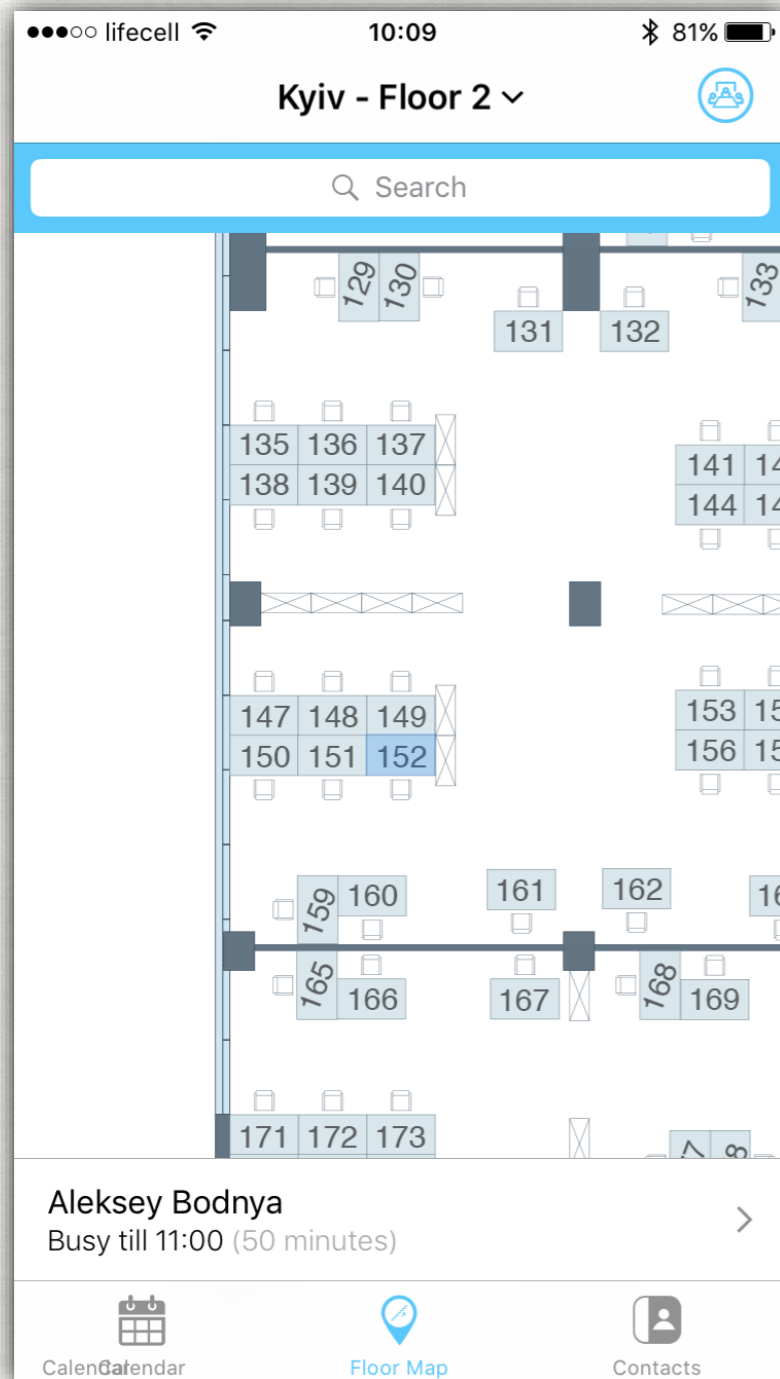
```
//
// DTHumanMachine.swift
//
// Created by Alexey Bodnya on 28.04.17.
// Copyright (c) 2017 Alexey Bodnya. All rights reserved.
//

import Foundation
import CoreData

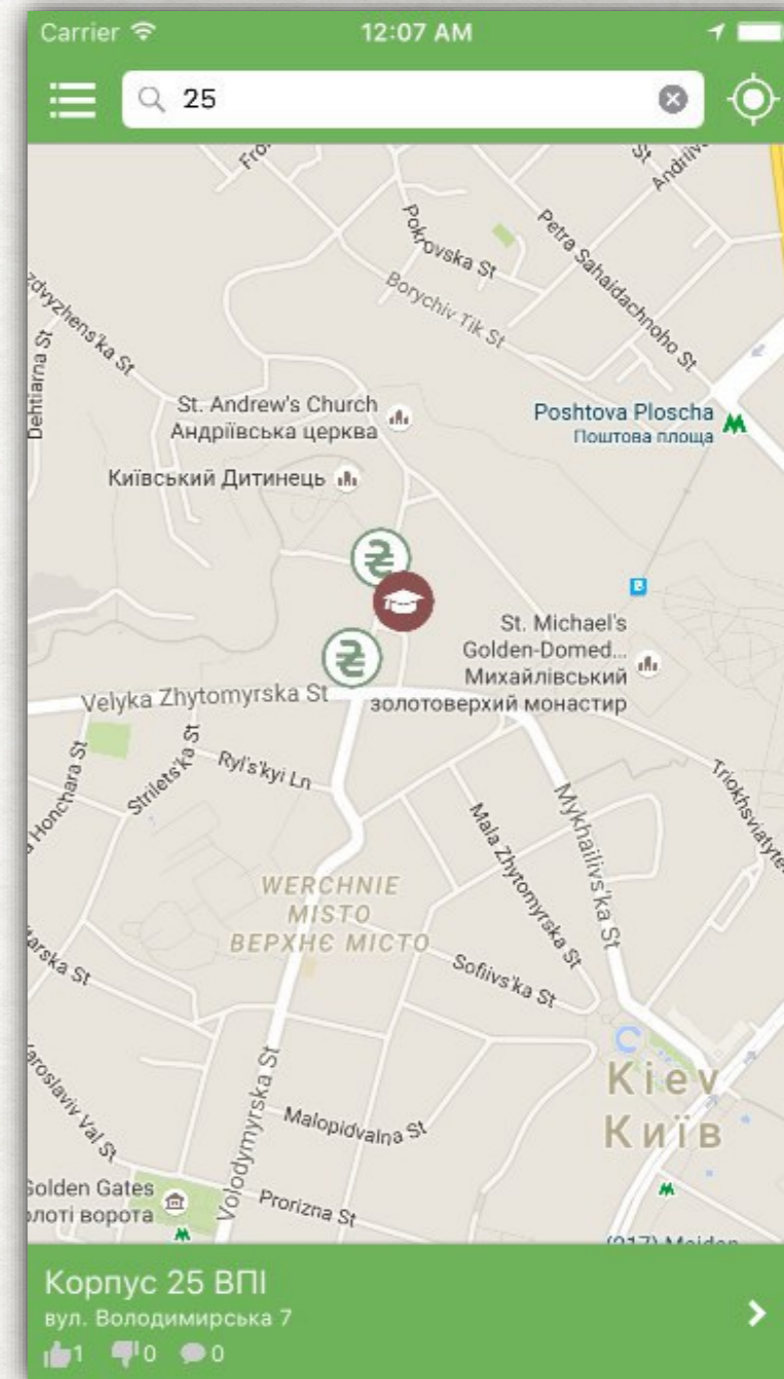
@objc(DTHumanMachine)
public class DTHumanMachine: NSObject {

    @NSManaged public var name: String?
    @NSManaged public var identifier: String?
    @NSManaged public var home: DTBuilding?
}
```


ТЕСТУВАННЯ



Cogniance maps



KPImaps

МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ

ФАКТОРИ ЗАГРОЗ

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Підходить для нових проектів	Для існуючих проектів виникне потреба переносити базу даних. Продукт більше підходить для нових проектів	Додавання можливості автоматизованого імпорту з різних типів сховищ
2	Власний формат зберігання	При необхідності змінити інструмент, компанії доведеться це робити вручну, оскільки використовується власна структура збереження даних	Додавання можливості автоматизованого експорту в різні типи сховищ
3	Обмеженість функцій	Інструмент обмежений наявними функціями і не має деяких функцій, які мають конкуренти (наприклад: індекси)	Додавання нових функцій за потреби

МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ

ФАКТОРИ МОЖЛИВОСТЕЙ

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Популярність мобільних платформ	Мобільна індустрія наразі друга за розвитком після веб-індустрії в постійно набирає обертів	Вихід на мобільний ринок
2	Потреба у використанні баз даних	SQL бази даних використовуються у більшості мобільних додатків	Надання інструменту для генерування нативного коду для роботи з БД
3	Відсутність повноцінних альтернатив	Існуючі альтернативи не надають можливості генерувати код для декількох платформ	Розширена підтримка існуючих платформ

МАРКЕТИНГОВЕ ДОСЛІДЖЕННЯ

SWOT-АНАЛІЗ СТАРТАП-ПРОЕКТУ

Сильні сторони:

Генерація коду для декількох платформ
Можливість модифікації шаблонів

Слабкі сторони:

Відсутність графічного редактора
Відсутність можливості задати індекси

Можливості:

Популярність мобільних платформ
Потреба у використанні баз даних
Відсутність повноцінних альтернатив

Загрози:

Підходить для нових проектів
Власний формат зберігання
Обмеженість функцій

ВИСНОВКИ

- Генерація даних дасть можливість суттєво зекономити час мобільних розробників
- Позбавить необхідності робити зайву роботу
- В перспективі надасть готовий код для серверної частини
- Дасть поштовх розвитку нових проектів, в т.ч. стартапів і студенських проектів

ДЯКУЮ ЗА УВАГУ