Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Кафедра програмної інженерії

КУРСОВА РОБОТА

ПОЯСНЮВАЛЬНА ЗАПИСКА

з дисципліни “Об’єктно - орієнтоване програмування”

«Картотека Інтерполу»

Керівник , асистент каф. ПІ Ляпота В.М.

Студент гр. ПІ-15-3 Жарков О.Г.

Комісія: проф.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дудар З.В.

проф.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бондарєв В.М.

Харків 2016

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ**

Кафедра: ***Програмної інженерії***

Дисципліна: ***Об’єктно-орієнтоване програмування***

Спеціальність: ***Програмна інженерія***

Курс 1 .Група *ПІ-15-3*. Семестр *2* .

***ЗАВДАННЯ***

***на курсовий проект студента***

***Жаркова Олександра Геннадійовича***

1. Тема проекту: Картотека Інтерполу.
2. Термін здачі студентом закінченого проекту: *“31” - травня - 2016 р*.
3. Вихідні дані до проекту: *специфікація програми, методичні вказівки до виконання курсової роботи*.
4. Зміст розрахунково-пояснювальної записки: *вступ, програмна специфікація, проектна специфікація,* інструкція користувача, стислі висновки.
5. Перелік графічного матеріалу: екранні форми, взаємодія класів.

**КАЛЕНДАРНИЙ ПЛАН**

|  |  |  |
| --- | --- | --- |
| *№* | *Назва етапу* | *Термін виконання* |
| 1 | Видача теми, узгодження і затвердження теми | 1-03-2016 р. |
| 2 | Формулювання вимог до програми | 4-03-2016 – 13-03-2016 р. |
| 3 | Розробка системи класів для «підозрюваного». Створення візуального оформлення головної сторінки. | 16-03-2016 – 22-03-2016 р. |
| 4 | Розробка системи зчитування с файлів та створення методів для роботи з класами для «підозрюваного» | 25-03-2016 – 3-04-2016 р. |
| 5 | Створення екранних форм для пошуку та відображення детальної інформації. Розроблені допоміжні класи та методи для пошуку та відображення. | 5-04-2016 – 14-04-2016 р. |
| 6 | Створення функціоналу відображення переміщення «підозрюваного» у країні, якою розшукується. Розробка статистичних даних. Розроблено редагування. | 15-04-2016 – 26-04-2016 р. |
| 7 | Тестування і доопрацювання певних частин програмної системи. | 26-04-2016 – 29-04-2016 р. |
| 8 | Оформлення пояснювальної записки, додатків, графічного матеріалу | 30-04-2016 – 31-05-2016 р. |
| 9 | Захист | 31-05-2016 р. |

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище, ім’я, по-батькові)

« 1 »\_березня\_ 2016 р.

РЕФЕРАТ

Пояснювальна записка до курсової роботи: 37 с., 36 рис., 1 додаток, 4 джерела.

Мета роботи: розробка програмної системи під назвою «Картотека Інтерполу», призначення якої : зберігати дані про «підозрюваних», яких розшукує Інтерпол, на мові об’єктно-орієнтованого програмування С#.

Метод розробки: Microsoft Visual Studio 2013, WINDOWS FORMS, .NET Framework 4.5.

В результаті розробки отримана програма під назвою “База Інтерполу”, яка дозволяє переглядати справи «підозрюваних» у тяжких злочинах, їх переміщення по країні, редагувати дані, здійснювати пошук по окремих параметрах.

КУРСОВА РОБОТА, ОБ’ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ, МОВА ПРОГРАМУВАННЯ С#, ІНТЕРПОЛ, ПОШУК ЗЛОЧИНЦІВ, ЗЛОЧИНИ.

ЗМІСТ

Вступ 6

1 Специфікація програми 7

* 1. Встановлення мети завдання 7
  2. Функції програми 7
  3. Опис інтерфейсу програмної системи………………………...…….8

2 Проектна специфікація 15

2.1 Об’єктна модель програми .15

2.1.1 Клас HumanPhysData ..17

2.1.2 Клас Suspect ...18

2.1.3 Клас SuspectCollection………………………………………….…..19

2.1.4 Клас Crime………….………………………………………….……19

2.1.5 Клас Story……….………………………………………….....…......21

2.1.6 Клас StoryCollection…….……………………………………..……22

2.1.7 Клас Country………………………………………………………...22

2.1.8 Клас Methods…………………………..……………………………23

3. Інструкція з експлуатації 24

3.1 Встановлення програми 24

3.2 Функціонування програми 24

3.2.1 Карта Світу 24

3.2.2 Перелік «підозрюваних» 25

3.2.3 Додавання нового злочинця .26

3.2.4 Пошук злочинців ..28

3.2.5 Детальна інформація………..……………………………….…...29

3.2.6 Редагування …………..30

3.2.7 Слідкування за переміщенням злочинця …..31

3.2.8 Графік-статистика злочинів за певний проміжок часу………….32

Висновки 33

Перелік посилань 34

Додаток А – Фрагменти коду програми 35

ВСТУП

У сучасному світі питання національної та світової безпеки є вкрай важливим. Сьогодні існує велика кількість спецпідрозділів які займаються такими питаннями: ФБР, АНБ (США), ФСБ (Росія) тощо. І важливою частиною роботи вказаних установ є управління базами даних про злочинців по всій земній кулі.

У шпигунських фільмах ми неодноразово бачили як спеціальні агенти використовували прилади для розпізнавання обличчя або ДНК, але не замислювалися над тим як ці пристрої працюють. Провідну роль у функціонуванні таких гаджетів, після інженерного проектування, є пошук в базі підозрюваних або злочинців. Але кіно – це тільки кіно.

А що ж може допомогти звичайному правозахиснику або громадянину розпізнати злочинця поряд з собою? Перше, що спаде на думку, це звичайно фотороботи, що можуть бути розклеєні по всьому вашому місту. Так зазвичай ми дізнаємося про розшук підозрюваного або повідомлення про зникнувших людей. Пошуком у більш масштабному об’ємі займається міжнародна організація Інтерпол.

Міжнародна організація карної поліції – це установа, яка займається об’єднанням сил правоохоронних органів по всьому світу задля боротьби зі злочинністю.

Виникла комісія карної поліції у 1923 році у Відні. Це була перша установа, яка займалась підтриманням порядку по всьому світі. Під час Другої світової війни її функціонування будо ненадовго призупинено, хоча формально Інтерпол ще існував. У 1946 році штаб-квартира Інтерполу була перенесена до Парижу, де у 1956 році було прийнято офіційний статут, за яким Міжнародна організація карної поліції отримала нову назву – Інтерпол.

Основною задачею Інтерполу є : проведення єдиної політики у боротьбі зі злочинністю. Серед другорядних завдань виділяють такі: боротьба з торгівлею людьми, організованою злочинністю, контрабандою наркотиків, корупцією тощо. Останнім часом велика увага приділяється боротьбі з тероризмом.

Сьогодні у складі Інтерполу наявні правоохоронні установи 191 країни. І кожна з них всіма силами намагається зберігати спокій своїх громадян.

1. Специфікація завдання

1.1 Встановлення мети завдання

В даній курсовій роботі треба розробити програмну систему під назвою «Картотека Інтерполу».

Програма повинна зберігати базу даних про злочини та злочинців та вільно оперувати даною базою.

У програмі наявні такі функції, як: Редагування даних про злочинця, створення нових даних про злочинця та занесення у базу, вільний пошук по базі (по багатьох параметрах пошуку), відстеження дій злочинця на певній частині карти світу, статистичні дані про скоєні злочини у певній державі або частині світу за певні проміжки часу (за день, за тиждень, за місяць, за рік та протягом всього часу).

Також створений зручний та практичний інтерфейс для ефективного та швидкого маніпулювання базою даних.

1.2 Функції програми

а) вибір країни для пошуку за допомогою карти Світу (див. рис. 1.1, 1.2);

б) відображення статистики злочинності у певній країні за допомогою діаграми (див. рис. 1.3);

в) додавання нового злочинця у загальну базу (див. рис. 1.4);

г) редагування даних певного злочинця : додавання нових злочинів або фактів, додавання нового фото злочинця або зміна значень деяких даних (див. рис. 1.7);

д) пошук серед злочинців (по багатьох параметрах) (див. рис. 1.5);

е) перегляд списку скоєних злочинів певним злочинцем (див. рис. 1.9);

є) видалення злочинів або певних фактів;

ж) відображення переміщення злочинця у певній країні: карта з контрольними точками та шляхами руху (див. рис. 1.8);

1.3 Опис інтерфейсу програмної системи

Під час запуску програми ви побачите екранну форму із зображенням мапи Світу (див. рис. 1.1).

А за кліком на певну частину Світу відкривається екранна форма з головним меню або вибрана частина карти збільшується (якщо це регіон, у якому на невеликій частині мапи присутня велика кількість країн).

Наприклад, за кліком по Європі відкриється більш детальна мапа (див. рис. 1.2).



Рисунок 1.1 – Мапа Світу

Рисунок 1.2 – Мапа Європи

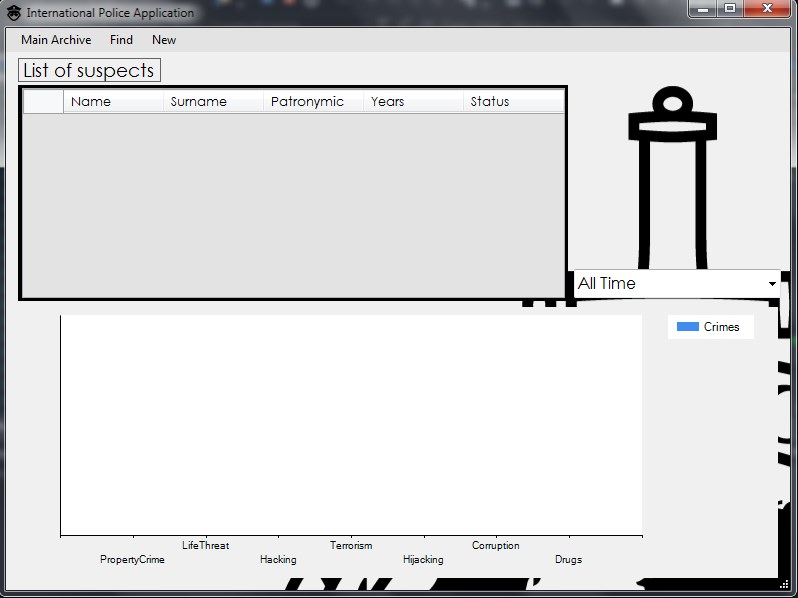
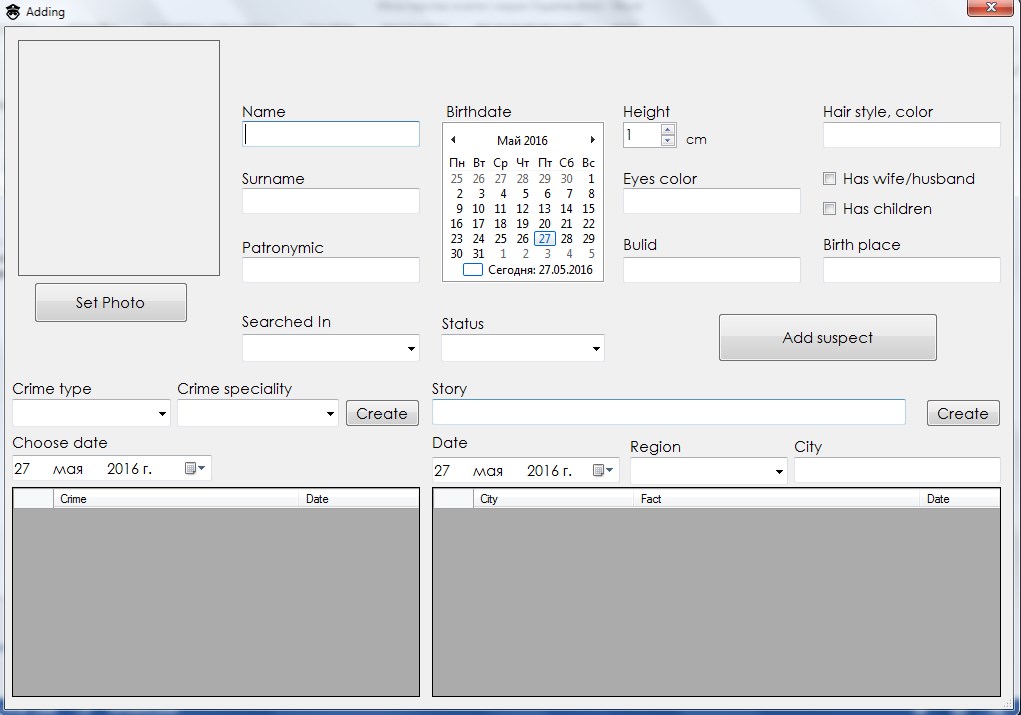
Далі, після вибору певної країни, відкриється нове вікно – Головне меню (див. рис. 1.3).

Рисунок 1.3 – Головне меню

На формі головного меню присутні : панель функцій: «Main Archive», функція, що відображає весь список злочинців у даній країні та статистику за весь час, «Find», система пошуку в базі, «New», функція, що створює нового злочинця та вносить його дані до бази, таблиця для відображення злочинців у базі за критерієм вибраної на карті країни та діаграма статистики злочинів.

При виборі функції «New» ввідкриється форма створення нового злочинця (див. рис. 1.4).

 Рисунок 1.4 – Форма створення нового злочинця

При правильному заповненні полів форми, злочинець буде внесений до бази.

Рисунок 1.5 – Форма для пошуку

При виборі функції «Find» відкриється діалогове вікно пошуку злочинця (див. рис. 1.5).

Далі, за подвійним кліком на будь-якому заповненому рядку таблиці злочинців відкриється форма з детальною інформацією про підозрюваного (див. рис. 1.6).



Рисунок 1.6 – Форма з детальною інформацією

Форма можливого редагування відкривається за допомогою кнопки «Edit» (див. рис. 1.7).

За кліком по кнопці «Show Аction Story» відкриється екранна форма з інформацією про переміщення злочинця (див. рис. 1.8).

На даній формі присутня карта країни або частини світу де розшукується злочинець, контрольні точки, де відбулися певні події та переміщення між містами. Таблиця в нижньому кутку відображає інформацію у певному регіоні.

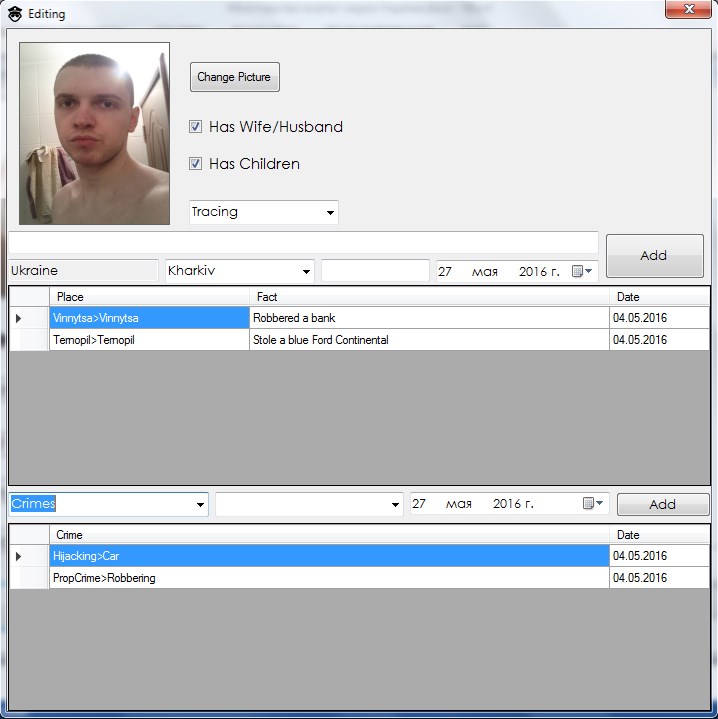
Рисунок 1.7 – Форма для редагування



Рисунок 1.8 – Форма з відстеженням переміщення злочинця

Також на формі детальної інформації присутня кнопка «Show Crimes», що відкриє вікно з переліком скоєних злочинів даного суб’єкта, де будуть наявні злочин, його спеціалізація (детальний опис злочину) та дата скоєння (див. рис. 1.9).

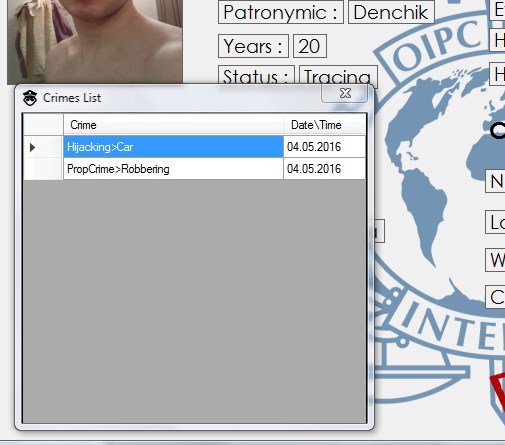


Рисунок 1.9 – Список скоєних злочинів

Після закриття форми з картою світу дані бази, з якою користувач працював під час функціонування програми, записуються до файлів.

2 Проектна специфікація

2.1 Об’єктна модель програми

Для програми була розроблена об’єктна модель для демонстрації взаємодії класів.

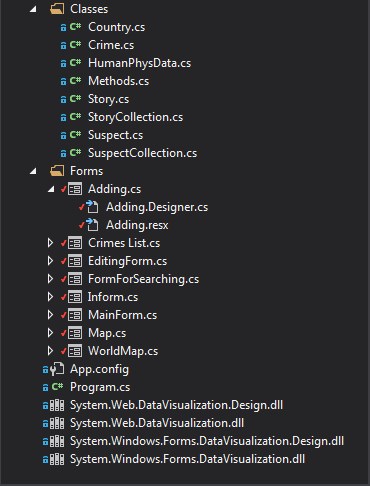


Рисунок 2.1 – Демонстрація об’єктної моделі

Папка «Classes» містить в собі класи : «Country» для відображення історії переміщення злочинця в певній країні, «Crime» для реалізації різноманітних злочинів, «HumanPhysData», що містить в собі загальну інформацію про підозрюваного (ім’я, вік, ріст, місце народження тощо), «Methods», що містить допоміжні статичні методи для обробки деяких даних, «Story» для зберігання історії переміщення та дій злочинця, «StoryCollection» для зберігання екземплярів класу «Story», «Suspect» для зберігання та обробки даних про злочинців та «SuspectCollection» для зберігання колекції злочинців.

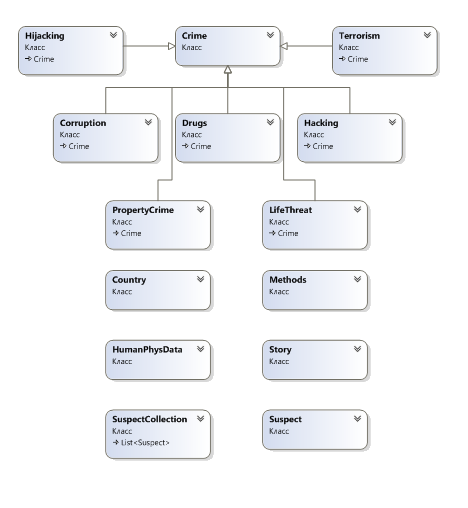


Рисунок 2.2 – Об’єктна модель класів

2.1.1 Клас «HumanPhysData»

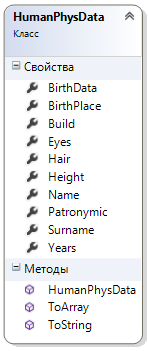


Рисунок 2.3 – Клас HumanPhysData

Клас «HumanPhysData» створений для зберігання загальної інформації про злочинця (див. рис. 2.3).

Він має поля: “Name” для зберігання імені, “Surname” для зберігання прізвища (другого імені), “Patronymic” для зберігання прізвиська, “Years” поле для зберігання віку злочинця, “BirthDate” для зберігання дати народження, “BirthPlace” для зберігання місця народження, “Build”, “Eyes”, “Hair” для зберігання особливостей зовнішнього вигляду.

Клас також має методи: HumanPhysData - конструктор класу, ToArray метод, що повертає масив з даними цього класу та ToString що повертає інформацію про клас у вигляді рядка.

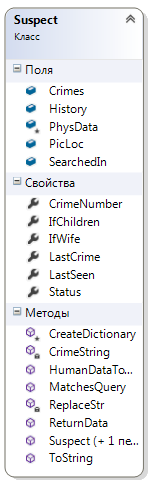
2.1.2 Клас «Suspect»

Рисунок 2.4 – Клас Suspect

Клас «Suspect» створений для зберігання інформації про злочинця

(див. рис. 2.4).

Він має такі поля: “Crimes”, це колекція скоєних злочинів, “History” - колекція з даними про переміщення злочинця, “PhysData” загальна інформація про злочинця, “PicLoc” для збереження місця розташування фотографії, “SearchedIn” для збереження назви країни, якою розшукується.

Клас має властивості: “CrimeNumber” для зберігання та відображення кількості скоєних злочинів, “IfChildren” вказує на те, чи має злочинець дітей, “IfWife”, вказує на те, чи має злочинець дружину або чоловіка, ”LastCrime” вказує на останній скоєний злочин, ”LastSeen” вказує на останнє місце перебування, “Status” вказує на теперішний статус.

Він також має методи: “CreateDictionary” повертає інформацію про злочини у вигляді списку із елементами ключ, значення, де ключ – злочин, значення – дата скоєння, “CrimeString” що формує із всіх елементів колекції злочинів строку певного вигляду для запису в базу, “HumanDataToArrayForDataBase” повертає декілька даних про злочинця, “MatchesQuery” для перевірки чи задовольняють критерії пошуку інформацію, що наявна у даному екземплярі класу, ”ReplaceStr” створений для співвідношення певних строкових значень, значенням ключів у колекції , що повертається методом “ReturnData”, “Suspect” - конструктор та “ToString” що представляє інформацію про клас у вигляді строки та “ReturnData”, що повертає колекцію з даними про підозрюваного.

2.1.3 Клас «SuspectCollection»

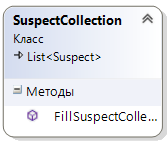


Рисунок 2.5 – Клас SuspectCollection

Клас створений для зберігання колекції злочинців (див. рис. 2.5).

Клас «SuspectCollection» має лише один метод для заповнення колекції даними про злочинців, зчитуючи дані з файлів.

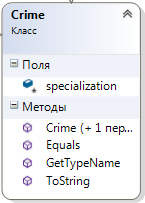
2.1.4 Клас «Crime»

Клас створений для зберігання інформації про певний злочин.

Клас «Crime» має методи: “Crime” - конструктор, “Equals” для порівняння з іншими злочинами, “GetTypeName”, що повертає ім’я класу у вигляді строки та “ToString”, що повертає дані про клас у вигляді строки.

Він має одне поле “specialization” , що вказує на певну спеціалізацію злочину.

Від класу «Crime» наслідуються такі класи: «Drugs», «Terrorism», «Hacking», «LifeThreat», «PropertyCrime», «Hijacking», «Corruption».



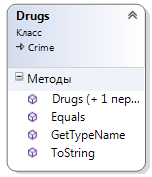
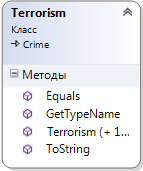
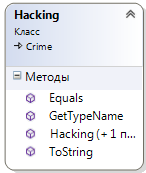
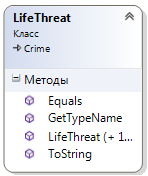


Рисунок 2.6 – Клас Crime

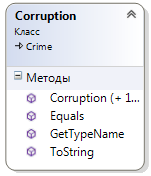
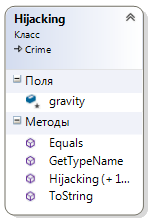
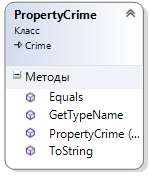


Рисунок 2.7 – Класи Drugs, Terrorism, Hacking, LifeThreat, PropertyCrime, Hijacking, Corrruption

Всі представлені класи мають: свій конструктор, та перевантажені методи: “GetTypeName”, “ToString” та “Equals”.

Клас “Hijacking” має поле “gravity”, що вказує на тяжкість скоєння даного злочину.

2.1.5 Клас «Story»

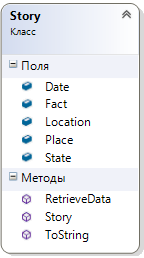


Рисунок 2.8 – Клас Story

Клас «Story» створений для зберігання певного моменту в історії дій або переміщення злочинця (див. рис. 2.8).

Цей клас має поля: “Date” для зберігання дати, коли був помічений, -а, “Fact”, що вказує на факт з історії злочинця,”Location” координати певного місця на карті країни, “Place”, що вказує на місто, де перебував злочинець, ”State” - регіон де перебував злочинець.

Він також має методи: “RetrieveData”, що повертає дані, які наявні в екземплярі класу, “Story” - конструктор, “ToString”, що повертає інформацію про клас у вигляді рядка.

2.1.6 Клас «StoryCollection»

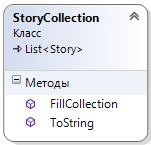


Рисунок 2.9 – Клас StoryCollection

Клас «StoryCollection» створений для групування у колекцію окремих фактів з історії злочинця (див. рис. 2.9).

Клас StoryCollection має методи :”FillCollection”, що заповнює колекцію, зчитуючи дані з файлів (у файлах дані про положення регіонів на картах), “ToString” повертає всі елементи колекції у вигляді рядку, який буде записано до файлу.

2.1.7 Клас «Country»

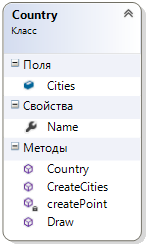


Рисунок 2.10 – Клас Country

Клас «Country» призначений для відображення шляхів переміщення злочинця на карті країни (див. рис. 2.10).

Він має поле “Cities” , що являє собою колекцію з даними у вигляді ключ, значення, де ключ це назва регіону, значення це точка на карті та також має властивість “Name” , що вказує на ім’я країни.

Та методи: “Country” - конструктор, “CreateCities” - заповнює колекцію назвами регіонів та їх положенням, ”createPoint” що повертає точку із строки певного вигляду та , “Draw”, що відображає точки та регіони, які зберігаються в колекції, на карті.

2.1.8 Клас «Methods»

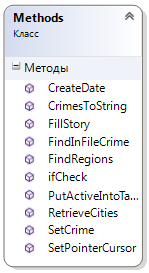


Рисунок 2.11 – Клас Methods

Клас призначений для зберігання статичних методів, що неодноразово використовуються у багатьох класах (див. рис. 2.11).

Цей клас має методи: ”CreateDate” , що створює дату із строки певного вигляду, повертає всі елементі колекції, “CrimesToString” що містить злочини у вигляді рядків в один цілісний рядок, що буде використовуватись при запису для файлу, або створення нового екземпляру класу «Suspect», “FillStory” метод, що вставляє дані колекції з історією у певну таблицю, “FindInFileCrime” для пошуку у файлі спеціалізації злочину для певного типу злочину, “FindRegions” знаходить координати регіонів у файлі з відповідною назвою, “ifCheck” перевіряє поля для набору тексту на правильність заповнення, “PutActiveIntoTable” заповнює таблицю активною колекцією злочинців, “RetrieveCities” знаходить назви регіонів для певних країн (обласних центрів), “SetCrime” повертає екземпляр класу «Crime» та наслідуваних класів відповідно до складу рядка (строки) та , “SetPointerCursor” що перетворює курсор із Arrow на Pointer, якщо курсор користувача потрапив у певну зону на мапі.

1. Інструкція з експлуатації

3.1. Встановлення програми.

1. запуск установочного файлу;
2. розпакування файлів у певну директорію;
3. встановлення додаткового програмного забезпечення;

3.2 Функціонування програмної системи

3.2.1 Карта Світу

Коли програма буде запущена, ви побачите екранну форму з картою світу (див. рис. 3.1). Регіони цієї карти можна вибирати за допомогою миші.

За кліком на певний регіон відкриється більш детальна карта області Світу. З’явиться можливість вибору країни.

.

Рисунок 3.1 – Мапа Світу

3.2.2 Перелік підозрюваних

Після вибору певної країни, вам відкриється вікно з таблицею, що заповнена колекцією підозрюваних, діаграмою статистики та меню з інструментами (див. рис. 3.2).

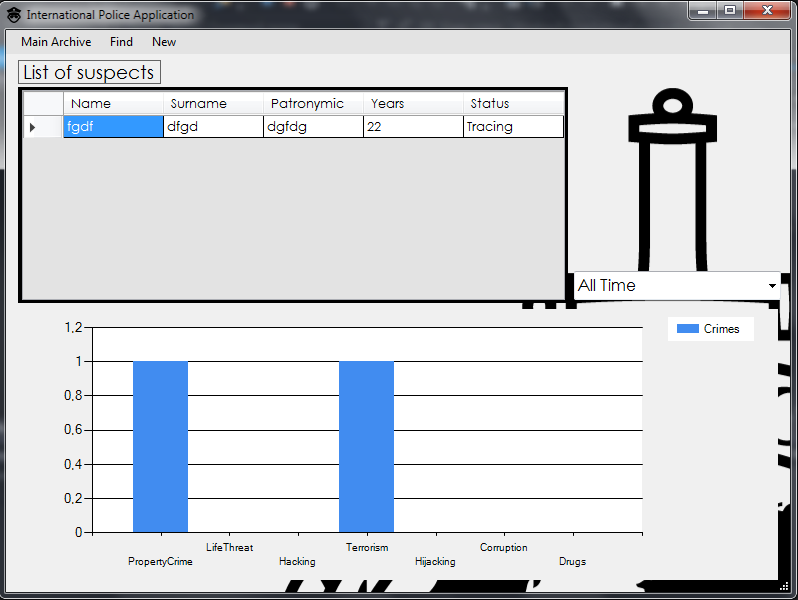


Рисунок 3.2 – Головне меню

3.2.3 Додавання нового злочинця

При виборі з меню інструментів функції «New» , ви матимете можливість занести в базу дані про нового злочинця. В текстові поля не дозволяється вводити знаки “~”, “=”, ”#”. Кожний злочинець повинен мати хоч один злочин та хоч один факт з історії дій. Додавати фото не обов’язково. В полі «Статус» вибрати дійсний статус злочинця: В розшуку, Заарештований.

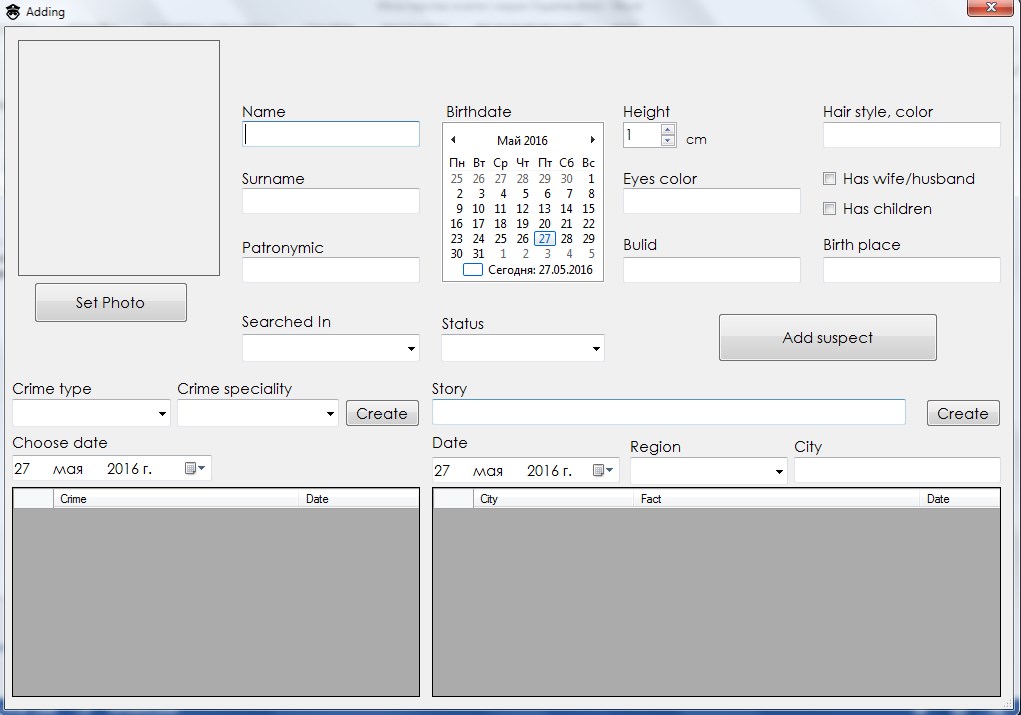
Можливість запису до таблиці з історією переміщення можливий тільки після вибору країни, де злочинець розшукується.

Рисунок 3.3 – Форма створення нового злочинця

Також не рекомендовано вибирати дати, пізніші сьогоднішньої, але при таких діях програма сама повідомить про помилку.

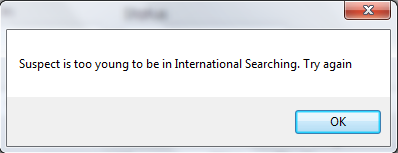


Рисунок 3.4 – Вікно з попереджувальним повідомленням про невірно введену дату

Вік злочинця складав менш ніж 18 років.

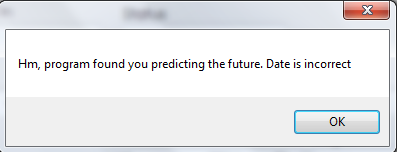


Рисунок 3.5 – Вікно з попереджувальним повідомленням про введену дату, пізнішу ніж сьогоднішня дата

Дата, що була використана при додаванні фактів до загальної історії або що використовувалась при додаванні нового злочину була пізніше ніж сьогоднішня дата.

За натисканням кнопки Add Suspect, якщо дані у всіх текстових полях будуть вірними, злочинець буде записаний до загальної бази.

3.2.4 Пошук злочинців

При виборі з меню інструментів функції «Find» , ви матимете можливість пошуку в колекції злочинців за певними параметрами.

Рисунок 3.6 – Форма для пошуку

Для користувача створений шаблон для пошуку по певних параметрах. Реєстр не важливий. Кожен параметр треба розділяти символом “;”. Після натискання кнопки «Find» таблиця на головному меню програми буде заповнятись злочинцями, що відповідають заданим критеріям.

3.2.5 Детальна інформація

Якщо на рядку в таблиці буде здійснено подвійний клік, то відкриється екранна форма з детальною інформацією про злочинця.

Рисунок 3.7 – Форма з детальною інформацією

На даній формі присутні функції перегляду списку злочинів, редагування інформації про даного злочинця та слідкування за переміщенням злочинця.

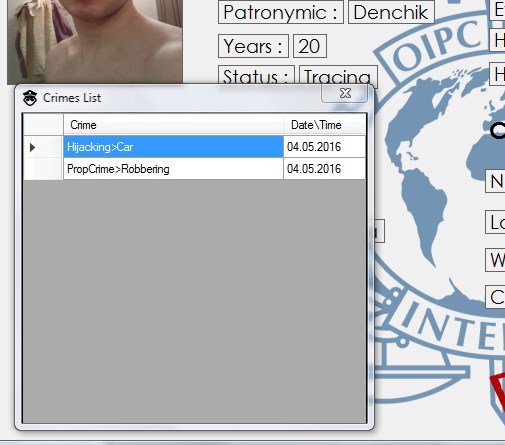


Рисунок 3.8 – Список скоєних злочинів

3.2.6 Редагування

При натисканні кнопки «Edit», відкриється вікно для редагування даних про злочинця.

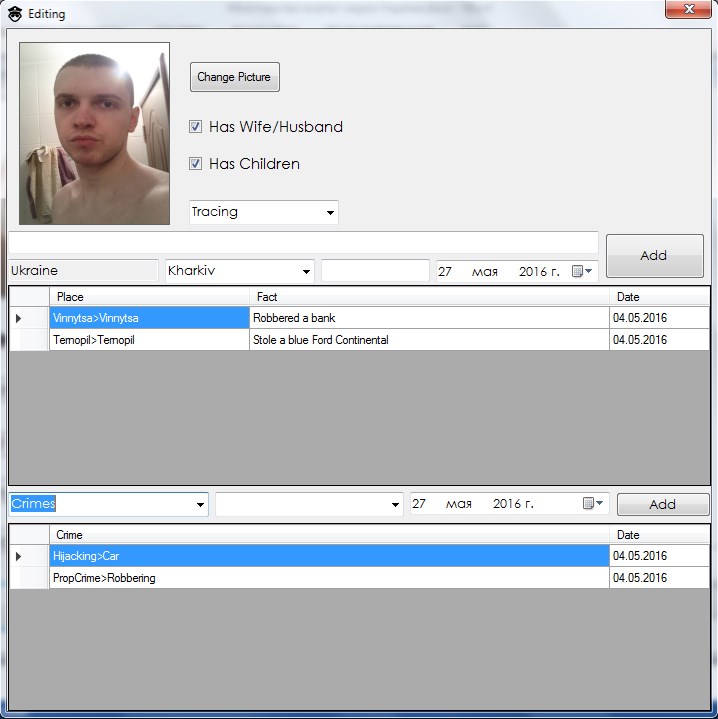


Рисунок 3.9 – Вікно редагування інформації

Можна змінити фото злочинця, дійсний статус, вказати чи є дружина або чоловік, вказати чи є діти. Також можна додавати нові факти історії та злочини і також, видаляти дійсні, але неможливо видалити усі злочини або історію (повинен існувати хоч один елемент у кожній колекції). Дані редагуються по закриттю даного вікна.

3.2.7 Слідкування за переміщенням злочинця

При натисканні кнопки «Show Action Story» на формі з детальною інформацією, відкривається вікно, що дозволяє побачити дії злочинця у певних регіонах країни у певний час.



Рисунок 3.10 – Форма для слідкування за історією переміщення

При натисканні на сині точки, що вказують на перебування злочинця у даній місцевості, ви отримаєте детальну інформацію про його, її дії у цьому регіоні.

3.2.8 Графік-статистика злочинів за певний проміжок часу

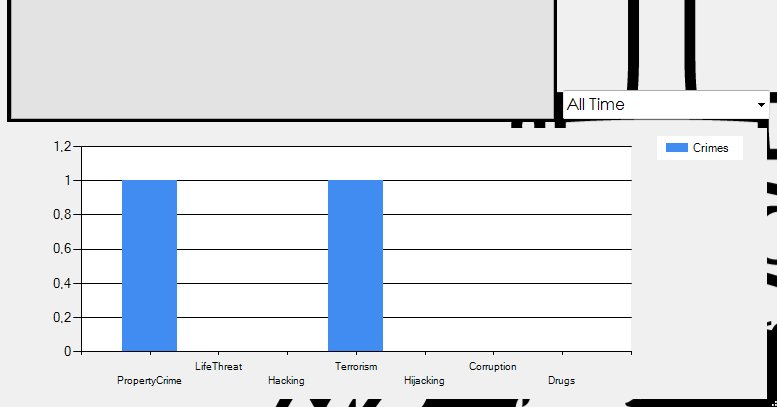
 На формі з переліком підозрюваних у певній країні, також наявна діаграма-статистика злочинів за певний час. За допомогою комбінованого поля, можна переглядати статистику скоєних злочинів за сьогоднішній день, за останній тиждень, за останній місяць, за останній рік та за увесь час.

Рисунок 3.11 – Діаграма зі статистикою злочинів

ВИСНОВКИ

В результаті розробки курсового проекту була отримана програма під назвою «Картотека Інтерполу», яка дозволяє маніпулювати базою даних, яка містить інформацію про злочинність по всьому світу : редагувати дані бази, додавати нові, передивлятись статистику та слідкувати за переміщеннями та діями злочинців.

Курсовий проект був розроблений за допомогою мови програмування C# на платформі .NET в системі для розробки під назвою Microsoft Visual Studio 2013.

Під час виконання курсового проекту, я детальніше розібрав принципи об’єктно-орієнтовного програмування, навчився працювати з графікою та текстовими файлами. Також були вивчені етапи розробки програмного забезпечення загалом, що допомогло мені у проектуванні курсового проекту.

Надалі планую продовжити розробку та вдосконалення даного програмного продукту. В етапи подальшої розробки будуть входити: проектування бази даних для зберігання відбитків пальців та пошук по відбитках, створення системи розпізнавання обличь по картинках, запити до баз даних поліції у різноманітних країнах світу, створення особистого кабінету офіцера поліції.

ПЕРЕЛІК ПОСИЛАНЬ

1. Бондарев В. М. «Объектно-ориентированное программирование на С#» [Текст]: учеб. Пособ / В. М. Бондарев. – Х.: Смит, 2009. – 224 с.;
2. Microsoft Developer Network. Библиотека MSDN. Разработка на .NET [Электронный ресурc] / Interpol –connecting police for a safer world. – Режим доступа : www/URL:https://msdn.microsoft.com/ru-ru/ – 25.05.2016 г. – Загл. с экрана.
3. Internet / Home – INTERPOL [Электронный ресурc] / INTERPOL – connecting police for a safer world. – Режим доступа : [www.interpol.int](http://www.interpol.int).
4. Интерпол – Википедия [Электронный ресурc] / Интерпол – Википедия . – Режим доступа : https://ru.wikipedia.org/wiki/Интерпол

Додаток А – Фрагмент коду форми головного меню

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using System.Windows.Forms.DataVisualization.Charting;

namespace WindowPolice

{

public partial class Interpole : Form

{

private SuspectCollection suspects;

private SuspectCollection wholeCollection;

private string country;

public Interpole(SuspectCollection Suspects, SuspectCollection WholeCollection, string Country)

{

InitializeComponent();

suspects = Suspects;

wholeCollection = WholeCollection;

country = Country;

}

private void Interpole\_Load(object sender, EventArgs e)

{

SuspectTable.Rows.Clear();

Methods.PutActiveIntoTable(SuspectTable, suspects);

FindStatistic("All Time");

}

private void archiveToolStripMenuItem\_Click(object sender, EventArgs e)

{

FormForSearching search = new FormForSearching(this.suspects, SuspectTable);

search.Location = this.Location;

if(search.ShowDialog() == DialogResult.OK)

{

search.Close();

}

}

private void Interpole\_SizeChanged(object sender, EventArgs e)

{

for (int i = 0; i < SuspectTable.Columns.Count; i++)

{

SuspectTable.Columns[i].MinimumWidth = Convert.ToInt32(Math.Floor(this.Width / 7.93));

}

SuspectTable.Width = Convert.ToInt32(Math.Floor(this.Width / 1.4732965009208103130755064456722));

SuspectTable.Height = Convert.ToInt32(Math.Floor(this.Height / 2.8571428571428571428571428571429));

panel1.Width = SuspectTable.Width + 7;

panel1.Height = SuspectTable.Height + 7;

}

private void SuspectTable\_CellDoubleClick(object sender, DataGridViewCellEventArgs e)

{

try

{

DataGridViewRow cell = this.SuspectTable.Rows[e.RowIndex];

}

catch(ArgumentOutOfRangeException ex)

{

return;

}

DataGridViewRow cells = this.SuspectTable.Rows[e.RowIndex];

Inform form = new Inform(suspects.ElementAt(Convert.ToInt32(cells.Cells[cells.Cells.Count - 1].Value)));

Dictionary<string, string> Dict = suspects.ElementAt(Convert.ToInt32(cells.Cells[cells.Cells.Count - 1].Value)).ReturnData();

for (int i = 0; i < form.Controls.Count; i++)

{

foreach(KeyValuePair<string, string> comparer in Dict)

{

if (form.Controls[i].Text.Equals(comparer.Key))

form.Controls[i].Text = comparer.Value;

if (form.Controls[i].Name.Equals(comparer.Key) && form.Controls[i].Name.Equals("Photo"))

{

PictureBox box = new PictureBox();

box = (PictureBox)form.Controls[i];

Bitmap Img = new Bitmap(comparer.Value);

box.Image = Img;

box.SizeMode = PictureBoxSizeMode.StretchImage;

}

if (comparer.Key.Equals("Status") && form.Controls[i].Name.Equals("InterpolAction"))

{

PictureBox box = new PictureBox();

box = (PictureBox)form.Controls[i];

Bitmap Img;

if (comparer.Value == "In Jail")

{

Img = new Bitmap(@"Images\busted.png");

box.SizeMode = PictureBoxSizeMode.Zoom;

}

else

{

Img = new Bitmap(@"Images\LOGO\_WANTED\_V3\_TORCIDO\_ROJO.png");

box.SizeMode = PictureBoxSizeMode.StretchImage;

}

box.Image = Img;

}

}

}

if(form.ShowDialog() != DialogResult.None)

{

this.chart1.Series[0].Points.Clear();

FindStatistic(comboBox1.Text);

}

}

private void newToolStripMenuItem\_Click(object sender, EventArgs e)

{

Adding AddForm = new Adding(SuspectTable, wholeCollection);

if(AddForm.ShowDialog() != DialogResult.None)

{

loadList();

Methods.PutActiveIntoTable(SuspectTable, suspects);

comboBox1.Text = "All Time";

this.chart1.Series[0].Points.Clear();

FindStatistic("All Time");

}

}

private void FindStatistic(string Time)

{

string[] crimes = { "PropertyCrime", "LifeThreat", "Hacking", "Terrorism", "Hijacking", "Corruption", "Drugs" };

int[] count = new int[crimes.Length];

for (int i = 0; i < crimes.Length; i++)

{

foreach(Suspect compr in suspects)

{

foreach(KeyValuePair<Crime,DateTime> comparer in compr.Crimes)

{

if (Time.Equals("Today"))

{

if (comparer.Key.GetTypeName().Equals(crimes[i]) && DateTime.Today.ToString("dd/MM/yyyy").Equals(comparer.Value.ToString("dd/MM/yyyy")))

count[i] = count[i] + 1;

}

if(Time.Equals("Week"))

{

if (comparer.Key.GetTypeName().Equals(crimes[i]) && comparer.Value.Day >= DateTime.Today.Day - 7 && comparer.Value.Year == DateTime.Today.Year && comparer.Value.Month == DateTime.Today.Month)

count[i] = count[i] + 1;

}

if (Time.Equals("Month"))

{

if (comparer.Key.GetTypeName().Equals(crimes[i]) && comparer.Value.Year == DateTime.Today.Year && comparer.Value.Month == DateTime.Today.Month)

count[i] = count[i] + 1;

}

if (Time.Equals("Year"))

{

if (comparer.Key.GetTypeName().Equals(crimes[i]) && comparer.Value.Year == DateTime.Today.Year)

count[i] = count[i] + 1;

}

if(Time.Equals("All Time"))

{

if(comparer.Key.GetTypeName().Equals(crimes[i]))

count[i] = count[i] + 1;

}

}

}

}

for (int i = 0; i < count.Length; i++)

this.chart1.Series["Crimes"].Points.AddXY(crimes[i], count[i]);

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

this.chart1.Series[0].Points.Clear();

FindStatistic(comboBox1.Text);

}

private void mainToolStripMenuItem\_Click(object sender, EventArgs e)

{

SuspectTable.Rows.Clear();

Methods.PutActiveIntoTable(SuspectTable, suspects);

}

private void Interpole\_FormClosing(object sender, FormClosingEventArgs e)

{

bool exit = false;

foreach(Suspect suspected in suspects)

{

foreach(Suspect allsuspcets in wholeCollection)

{

if (allsuspcets.PicLoc == suspected.PicLoc && allsuspcets.LastSeen == suspected.LastSeen && allsuspcets.LastCrime.ToString() == allsuspcets.LastCrime.ToString())

{

exit = true;

break;

}

}

if(!exit)

wholeCollection.Add(suspected);

exit = false;

}

}

private void loadList()

{

SuspectCollection ToLoad = new SuspectCollection();

foreach (Suspect suspected in wholeCollection)

{

if (suspected.SearchedIn.Equals(country))

ToLoad.Add(suspected);

}

suspects = ToLoad;

}

}

}