

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри
_____ Скарга-Бандурова І.С.
«___» _____ 2018 р.

ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА

ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

ІКС реєстрації та ведення територіальної громади

Освітньо-кваліфікаційний рівень “бакалавр”
Напрямок 6.050102 – “Комп’ютерна інженерія”

Керівник проекту:

(підпис)

доц. Сафонова С.О.

(ініціали, прізвище)

Консультант з охорони праці:

(підпис)

ст.викл. Критська Я.О.

(ініціали, прізвище)

Здобувач вищої освіти:

(підпис)

Газіна Д.І

(ініціали, прізвище)

Група:

КІ-14бд

Севєродонецьк 2018

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітньо-кваліфікаційний рівень бакалавр
Напрямок підготовки 6.050102 Комп'ютерна інженерія
(шифр і назва)
Спеціальність _____
(шифр і назва)

ЗАТВЕРДЖУЮ:

Завідувач кафедри _____
І.С. Скарга-Бандурова
« _____ » _____ 2018 р.

**З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА**

Газиній Дар'ї Ігорівні
(прізвище, ім'я, по батькові)

1. Тема роботи ІКС реєстрації та ведення територіальної громади

керівник проекту (роботи) Сафонова С.О., к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від " 14 " 05 2018 р. № _____

2. Термін подання студентом роботи _____

3. Вихідні дані до роботи матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз і розробка технічного завдання. Вибір і обґрунтування програмних засобів реалізації проекту. Розробка інформаційної системи. Посібник користувача системи. Охорона праці.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Електронні плакати

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст.викл. кафедри КНІ Критська Я.О.		

7. Дата видачі завдання _____

Керівник

_____ (підпис)

Завдання прийняв до виконання

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Аналіз і розробка технічного завдання	14.05.18-16.05.18	
2	Вибір і обґрунтування програмних засобів реалізації проекту	17.05.18-20.05.18	
3	Розробка структури бази даних	21.05.18-24.06.18	
4	Розробка основних компонентів системи	25.06.18-30.06.18	
5	Посібник користувача системи	01.06.18-06.06.18	
6	Розробка розділу охорона праці	07.06.18-09.06.18	
7	Оформлення електронних плакатів	10.06.18-12.06.18	
8	Оформлення пояснювальної записки	13.06.18-14.06.18	

Здобувач вищої освіти

_____ (підпис)

Газіна Д.І.

_____ (прізвище та ініціали)

Керівник

_____ (підпис)

Сафонова С.О.

_____ (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 123 с., 52 рис., 12 табл., 44 бібліографічних джерел посилань, 9 додатків.

Об'єкт розробки: процеси надання послуг реєстрації/зняття з реєстрації фізичних осіб за місцем проживання.

Мета роботи: розробити систему, яка автоматизує процеси надання послуг реєстрації/зняття з реєстрації фізичних осіб за місцем проживання, для підвищення ефективності та якості обслуговування клієнтів.

В проекті виконано:

1. Огляд методів, підходів, технологій до проектування розробки інформаційних систем, сформульована постановка задачі.
2. Вибір засобів для розробки програмного забезпечення.
3. Проектування та розробку бази даних.
4. Проектування та розробку програмної системи.
5. Тестування роботи системи.
6. Здійснений аналіз потенційних небезпечних і шкідливих виробничих чинників проєктованого об'єкта, що впливають на персонал.

Отримано наступні результати: розроблена інформаційна комплексна система реєстрації та ведення територіальної громади.

Практичне значення, галузь застосування роботи: розроблена система буде використовуватися в відділі надання адміністративних послуг, що надасть можливість ефективніше і швидше надавати послуги та знизить показники витрат людських ресурсів.

Ключові слова: ІНФОРМАЦІЙНА СИСТЕМА, БІЗНЕС-ПРОЦЕС, РЕЄСТРАЦІЯ, ФІЗИЧНА ОСОБА, КАРТОГРАФІЧНА СХЕМА, PHP, SQL SERVER, APACHE.

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А, м. Сєверодонецьк, 93400.

ЗМІСТ

1 ПОСТАНОВКА ЗАВДАННЯ	11
1.1 Опис предметної області	11
1.2 Формулювання вимог до розроблюваної системи	12
2 АНАЛІЗ БІЗНЕС-ПРОЦЕСІВ	15
2.1 Статус «як має бути» щодо послуги з реєстрації/ зняття з реєстрації фізичних осіб за місцем проживання.....	15
2.2 Порівняльний аналіз бізнес-процесів «як є» та «як має бути»	19
3 ВИБІР ЗАСОБІВ РОЗРОБКИ.....	29
3.1 Аналіз та вибір мови програмування.....	29
3.2 Огляд та вибір веб-серверу	32
3.3 Огляд та вибір СУБД.....	33
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ.....	37
4.1 Розробка та реалізація бази даних.....	37
4.2 Структура системи.....	45
5 КЕРІВНИЦТВО КОРИСТУВАЧА	51
5.1 Опис інтерфейсу програми	51
5.2 Основні операції користувача.....	53
5.3 Керівництво адміністратора.....	55
6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	61
6.1 Загальні питання з охорони праці	61
6.2 Аналіз стану умов праці	62
6.2.1 Вимоги до організації робочого місця.....	63
6.2.2 Навантаження та напруженість процесу праці.....	64
6.3 Виробнича санітарія	65
6.3.1 Аналіз небезпечних та шкідливих факторів при розробці виробу... ..	65
6.3.2 Пожежна безпека	66
6.3.3 Електробезпека.....	67
6.4 Гігієнічні вимоги до параметрів виробничого середовища.....	68
6.4.1 Мікроклімат.....	68

6.4.2 Освітлення	69
6.4.3 Вентилювання	70
6.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	71
6.6 Висновки до шостого розділу	72
ВИСНОВКИ.....	73
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	74
Додаток А - Комп'ютерна презентація	79
Додаток Б - Бізнес-процес: реєстрація особи за місцем проживання. Статус «як є».....	87
Додаток В - Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як є».....	91
Додаток Г - Бізнес-процес: реєстрація за особи за місцем проживання. Статус «як має бути»	95
Додаток Д - Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як має бути»	98
Додаток Е - Лістинг програми	100
Додаток Ж - Демонстрація вікна пункту меню «Пошук адреси»	124
Додаток И - Демонстрація вікна «Особова картка»	125
Додаток К - Демонстрація вікна «Реєстраційна картка»	126

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ЦНАП	Центр надання адміністративних послуг
БП	бізнес-процес
БД	база даних
ДМС	Державна міграційна служба
ДРВ	Державний Реєстр виборців
ЄДДР	єдиний державний демографічний реєстр
УК	управління кадрами
МОВ	місцеві органи влади
СУБД	система управління базами даних
АТУУ	Адміністративно-територіальний устрій України
ЕЦП	електронно-цифровий підпис
ПЗ	програмне забезпечення
ПК	персональний комп'ютер

ВСТУП

Розвиток інформаційних технологій в даний час відбувається дуже швидкими темпами. Майже для будь-якої області, починаючи від промислової закінчуючи медичною, є спеціалізоване ПЗ, яке автоматизує і спрощує роботу підприємств.

В сучасних умовах на великих підприємствах співробітникам доводиться мати справу з великою кількістю інформації, яка часто оновлюється та яку дуже складно обробити «вручну». На підприємствах, що мають значний оборот продукції, існує необхідність обліку і контролю великого обсягу кадрової, фінансової, маркетингової, виробничої, закупівельно-збутової та іншої інформації. Схожа ситуація спостерігається і в Лисичанському ЦНАПі.

Постійно зростаюча потреба в послугі, яка включає в себе реєстрацію/зняття з реєстрації фізичних осіб за місцем проживання/перебування, супроводжується подальшим ускладненням ключових етапів, з яких вона складається. Все це в свою чергу зумовлює підвищення вимог до надійності наданих послуг та якості отриманих та оброблених даних. Тому швидке, систематизоване виконання цих послуг є невід'ємною складовою роботи відділу надання адміністративних послуг.

1 ПОСТАНОВКА ЗАВДАННЯ

1.1 Опис предметної області

Поточна практика бізнес-процесів надання послуг з реєстрації та ведення територіальної громади показала, що існує низка вимог та процедур, які роблять дані процеси дуже витратними за часом як для ЦНАПу, так і для клієнтів.

На сьогоднішній день існує складність, пов'язана з процесом подання заяви. Для отримання послуги з реєстрації/зняття з реєстрації фізичних осіб за місцем проживання заявник повинен надати велику кількість супровідних документів. Супровідні документи видаються іншими органами, що вимагає чисельних візитів до інших організацій. Окрім супровідних документів в окремих випадках реєстрації/зняття з реєстрації вимагається одночасна фізична присутність всіх співвласників житлового приміщення. Це зумовлює значні складнощі у випадках, коли власників приміщення декілька, і через особисті причини вони не можуть бути присутні одночасно. Процедури, що перелічені вище, роблять процес надання послуги витратним з огляду на час та вимагають від фізичної особи значних зусиль.

Іншим аспектом складності процесу подання заяви є технічна недосконалість бланку заяви, який заявник повинен заповнити вручну. Заява містить велику кількість інформації і роздруковується у форматі А5. Відповідно, заповнення бланку заяви стає складним процесом, що призводить до помилок. Зазвичай бланк заяви заповнюється в офісі ЦНАПу відповідно до зразків розміщених в ньому, але найчастіше заявники звертаються за допомогою до персоналу ЦНАПу, який в свою чергу не завжди може розібрати дані преведені в заяві. Результатом цього є необхідність перезаповнювати бланк заяви, що додатково витрачає час протягом реєстраційних процедур. Окрім того, внесення даних до бази даних із заяви, що заповнена вручну, становить високий ступінь ризику.

Через брак технічної спроможності щодо управління потоком клієнтів, управління чергою здійснюється в ручному режимі через надання заявникам номеру в черзі. З цієї причини заявники не мають чіткої інформації про те, як довго їм потрібно чекати на отримання послуги.

Окрім того, було визначено, що існує низка дублюючих дій, які є частиною внутрішнього бізнес-процесу.

Актуальність теми даного дипломного проекту пов'язана з необхідністю вирішення комплексу задач таких, як організація процесу консультування, спрощення процедури отримання свідоцтва про реєстрацію/зняття з реєстрації, зменшення кількості відвідувань та інших суміжних з нею видів послуг.

Сферою застосування автоматизованого робочого місця є підрозділ місцевого самоврядування, що здійснює надання адміністративних послуг.

Результатом дипломного проекту повинна стати автоматизація робочого місця, яка призвана надати відповідні адміністративні послуги та забезпечити автоматизовану обробку даних.

Користувачами автоматизованих робочих місць є працівники Лисичанського ЦНАПу.

1.2 Формулювання вимог до розроблюваної системи

Автоматизована комплексна система реєстрації/зняття з реєстрації фізичних осіб за місцем проживання повинна вирішувати наступні поставлені завдання:

- а) реєстрація/зняття з реєстрації місця тимчасового проживання з відстеженням виданих довідок;
- б) можливість додавання документів з даними щодо реєстрації/зняття з реєстрації з описовими атрибутами;
- в) можливість генерування звіту «Склад сім'є» для ЦНАПів;
- г) видача звітів до ДМС;

- д) видача звітів до ДРВ;
- е) видача звітів до військомату;
- ж) імпорт даних з існуючої бази даних до нової.

Передбачається, що робота з базою даних буде проводитися щодня.

Основні вимоги до оформлення веб-додатку:

а) оформлення дизайну веб-інтерфейсу має бути максимально простим і ненав'язливим для ока. Загальна колірна гамма повинна бути виконана в світлих тонах з мінімальним набором фонових кольорів. Стартова і внутрішні сторінки повинні мати схожу структуру кольорів: білий фон, графіка фону сіра. Колір основної кнопки – синій, але для деяких інших функціональних кнопок – блакитний/зелений. Шрифт повинен бути чітким і легко читабельним;

б) стартова сторінка повинна містити логотип (до 25% висоти сайту), мати структуру, зібрану з горизонтальних інформаційних блоків (пошук адреси, пошук особи, пошук реєстрації, звіти, довідники, користувачі, адміністратор системи) спільну з основними сторінками;

в) мінімальний дозвіл екрану, на якому веб-інтерфейс буде переглядатися без обмеження від 800x600 пікселів, 15 "монітор. При даному дозволі допускається можливість перегляду сторінок з горизонтальною прокруткою у вікні браузера. Основна роздільна здатність екрану, на якому перегляд буде оптимально зручним 1024x768, 17 "монітор. Веб-інтерфейс повинен займати весь простір екрану по ширині. Висота ж залежить від обсягу інформації, що відображається. Перегляд повинен здійснюватися за допомогою всіх популярних Інтернет-браузерів: Internet Explorer, Mozilla FireFox, Opera, Google Chrome;

г) функціональна частина повинна складатися з реєстрації нового користувача та поділу прав доступу для кожного з них. На початку роботи в програмі буде зареєстровано тільки одного користувача – Адміністратор Системи. Саме він буде мати можливість в будь-який момент додавати,

видаляти, змінювати всю інформацію і дані користувачів в базі. Користувачі додатку не мають доступу в розділ адміністрування.

Вимоги до апаратного забезпечення

- сервер (мінімум один) 2 CPU 16 GB RAM та 2 HDD RAID 0;
- процесор – 64-х розрядний з тактовою частотою не менш 1,4 ГГц;
- тип процесора – процесор x64: AMD Opteron, AMD Athlon 64, Intel Xeon з підтримкою Intel EM64T, Intel Pentium IV з підтримкою EM64T;
- вільне місце на жорсткому диску – не менш 6 Гб;
- монітор – Super VGA з роздільною здатністю не менш 800x600;

Вимоги до програмного забезпечення

Для коректної роботи системи повинно бути встановлено наступне ПЗ:

- операційна система починаючи з Windows 7;
- SQL Server 2012 (можливі як більш пізні, так і більш раніші версії);
- веб-сервер Apache;
- текстові редактори: Microsoft Word, починаючи з версії 2007, та ін.;
- обов'язкова наявність інтернет-браузера Google Chrome.

2 АНАЛІЗ БІЗНЕС-ПРОЦЕСІВ

2.1 Статус «як має бути» щодо послуги з реєстрації/ зняття з реєстрації фізичних осіб за місцем проживання

На основі проведеного аналізу бізнес-процесу «як є» (Додаток Б), (Додаток В) було винайдено рішення для підвищення якості обслуговування, зменшення паперової роботи, вдосконалення управління даними та інформацією послуг з реєстрації/зняття з реєстрації за місцем проживання. Рекомендовані рішення (бізнес-процес «як має бути») ґрунтуються на стандартизації/оптимізації ефективних компонентів існуючих бізнес-процесів, що перешкоджають ефективному наданню послуг, а також автоматизації самого процесу.

Бізнес-процес «як має бути» було розроблено із урахуванням існуючих законодавчих рамок, що регулюють надання послуг з реєстрації/зняття з реєстрації фізичних осіб за місцем проживання. Для наочного бачення усіх складових даного бізнес-процесу розписана схема варіантів дій в залежності від ситуації.

Реєстрація. Консультація:

а) громадянину потрібна реєстрація за конкретною адресою для цього він звертається до адміністратора з рецепції;

б) адміністратор з'ясовує чи відомий громадянину порядок реєстрації:

1) якщо так, то громадянин самостійно збирає документи, проходить необхідні процедури в інших організаціях, сплачує адміністративний збір і отримує квитанцію про сплату після чого приходить до ЦНАПу;

2) якщо ж громадянину не відомий порядок реєстрації, то йому необхідно особисто відвідати ЦНАП і отримати консультацію адміністратора з поясненням підстави та права громадянина на реєстрацію за конкретною адресою;

в) у тому разі коли громадянин має підставу або право, адміністратор рецепції визначає адміністративний збір для сплати, надає роздрукований перелік необхідних документів та банківські реквізити. Адміністратор пропонує громадянину призначити прийом на зручний час та дату. У разі вибору громадянином дати та часу прийому, адміністратор перевіряє у програмному забезпеченні відділу прийому громадян, чи вільний цей час;

г) громадянин отримує талон зі спеціальним кодом, що присвоюється системою електронної черги;

д) якщо у громадянина немає прав або підстав на реєстрацію за конкретною адресою, то дана процедура не може бути реалізована.

Реєстрація. Збір документів та опрацювання їх у відділі прийому громадян:

а) після отримання інформації, громадянин займається збиранням документів та проходить необхідні процедури в інших організаціях;

б) сплачує адміністративний збір та отримує квитанцію про сплату;

в) громадянин приходить до ЦНАПу з необхідними документами та квитанцією;

г) адміністратор звіряє час прийому:

1) якщо прийом призначений, то адміністратор рецепції спрямовує громадянина до адміністратора з реєстрації для подання документів;

2) якщо ж прийом не призначений, то громадянин звертається до адміністратора рецепції для отримання номеру в черзі;

3) громадянин очікує в черзі допоки адміністратор рецепції не відправляє його до адміністратора з реєстрації.

Реєстрація. Подання документів та обробка рішення про реєстрацію:

а) громадянин звертається до адміністратора з рецепції;

б) адміністратор здійснює перевірку посвідчення особи громадянина відповідно до стандартного переліку дій;

в) адміністратор перевіряє правильність поданих документів на основі переліку необхідних документів:

1) якщо усі надані документи подано правильно, то адміністратор вносить дані з них до реєстраційного ПЗ;

2) якщо ж є проблема з документами, то громадянин повторно відвідує ЦНАП з виправленою версією документів;

г) адміністратор роздруковує заяву про реєстрацію, що генерується реєстраційним ПЗ;

д) заявник та адміністратор підписують заяву про реєстрацію;

е) адміністратор сканує подані документи, в тому числі підписану заяву про реєстрацію, та прикладає в реєстр через реєстраційне ПЗ;

ж) громадянин повинен надати паспорт чи інший документ, в якому можна поставити офіційний штамп про реєстрацію:

1) якщо громадянин подав документ, який входить до переліку дозволених, адміністратор проставляє офіційний штамп про реєстрацію;

2) якщо громадянин не надав один із вище зазначених документів, то адміністратор роздруковує довідку про реєстрацію, що генерується реєстраційним ПЗ;

3) довідка підписується та завіряється печаткою адміністратора.

Картографування бізнес-процесу реєстрації особи за місцем проживання статус «як має бути» наведений у Додатку Г.

Зняття з реєстрації. Консультація:

а) громадянин звертається за зняттям з реєстрації місця проживання;

б) адміністратор з'ясовує чи відомий громадянину порядок зняття з реєстрації:

1) якщо так, то громадянин самостійно збирає необхідні документи, проходить необхідні процедури в інших організаціях, сплачує адміністративний збір і отримує квитанцію про сплату після чого приходить до ЦНАПу;

2) якщо ж громадянину не відомий порядок зняття з реєстрації, то адміністратор його інформує;

в) громадянин особисто відвідує офіс ЦНАПу, де адміністратор рецепції пояснює підстави та право громадянина на зняття з реєстрації за конкретною адресою;

г) у разі коли громадянин має підставу або право, адміністратор рецепції визначає адміністративний збір для сплати, надає роздрукований перелік необхідних документів та банківські реквізити. Адміністратор пропонує громадянину призначити прийом на зручний час та дату. У разі вибору громадянином дати та часу прийому, адміністратор перевіряє у програмному забезпеченні відділу прийому громадян, чи вільний цей час;

д) громадянин отримує талон, зі спеціальним кодом, що присвоюється системою електронної черги і залишає ЦНАП;

е) у разі , якщо громадянин не назначив дату та час прийому він також залишає ЦНАП і самостійно збирає документи, проходить необхідні процедури в інших організаціях, сплачує адміністративний збір, отримує квитанцію про сплату після чого приходить до офісу ЦНАПу;

Зняття з реєстрації. Збір документів та опрацювання їх у відділі прийому громадян:

а) після отримання громадянином необхідної інформації він займається збиранням документів та проходить необхідні процедури в інших організаціях;

б) сплачує адміністративний збір та отримує квитанцію про сплату;

в) громадянин приходить до ЦНАПу з необхідними документами та квитанцією;

г) адміністратор перевіряє призначення особою прийому:

1) якщо дата та час прийому підтверджується, то адміністратор рецепції відправляє громадянина до адміністратора з реєстрації для подання документів;

2) якщо ж прийом не призначений, то громадянин звертається до адміністратора рецепції для отримання номеру в черзі;

3) громадянин очікує в черзі доки адміністратор рецепції не відправляє його до адміністратора з реєстрації.

Зняття з реєстрації. Подання документів та обробка рішення про реєстрацію:

а) громадянин звертається до адміністратора з рецепції;

б) адміністратор перевіряє правильність поданих документів на основі стандартного переліку дій:

1) якщо усі документи подано правильно, то адміністратор вносить дані з поданих документів до реєстраційного ПЗ;

2) якщо ж є проблема з документами, то громадянин повторно відвідує ЦНАП з виправленою версією документів;

в) адміністратор роздруковує заяву про зняття з реєстрації, що генерується реєстраційним ПЗ;

г) заявник та адміністратор підписують заяву про зняття з реєстрації;

д) адміністратор сканує подані документи, в тому числі підписану заяву про зняття з реєстрації, та прикладає в реєстр через реєстраційне ПЗ.

Картографування бізнес-процесу зняття з реєстрації за місцем проживання статус «як має бути» наведений у Додатку Д.

2.2 Порівняльний аналіз бізнес-процесів «як є» та «як має бути»

У даному розділі проводиться порівняльний аналіз, що окреслює існуючі сфери для вдосконалення, огляд рекомендованих рішень, а також очікувані результати від запровадження цих рішень щодо підвищення якості надання послуг ЦНАПом Лисичанської міської адміністрації. Порівняльний аналіз охоплює такі послуги:

а) реєстрація місця проживання;

- б) зняття з реєстрації за місцем проживання;
- в) видання довідки про реєстрацію за місцем проживання.

Основною метою рекомендованих бізнес-процесів «як має бути» є підвищення якості надання послуг, що має на меті забезпечення підвищення задоволеності клієнтів якістю надання послуг. Підвищення рівня задоволеності клієнтів буде досягнуто головним чином через:

- а) підвищення ефективності надання послуг;
- б) мінімізацію зусиль громадянина для отримання цих послуг.

Основними очікуваними результатами, які буде досягнуто через інституціоналізацію рекомендованих рішень, будуть:

- а) загальне вдосконалення ефективності роботи через стандартизацію та спрощення процесів надання послуг;
- б) значне зменшення паперової роботи;
- в) вдосконалення управління реєстраційними даними.

Інституціоналізація бізнес-процесів «як має бути» зробить можливим ефективно запровадження ЦНАПом механізмів моніторинга і оцінки для систематичного аналізу ефективності надання послуг, а також оцінки рівня задоволеності клієнтів якістю цих послуг.

Детальний порівняльний аналіз складових бізнес-процесів наведено у таблиці 2.1.

Таблиця 2.1- Реєстрація місця проживання. Під-процес 1: Консультація

Процес R-01: Реєстрація місця проживання		
Під-процес 1: Консультація		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
Протягом під-процесу	Прояснення правової	Переваги для громадянина

Продовження таблиці 2.1 - Реєстрація місця проживання. Під-процес 1:

Консультація

Процес R-01: Реєстрація місця проживання		
Під-процес 1: Консультація		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>консультації адміністратор рецепції прояснює правову підставу та право громадянина на реєстрацію за конкретною адресою, а також надає інформацію щодо необхідних документів для отримання послуги з реєстрації місця проживання. Процедури щодо визначення правових підстав та права, а також надає інформації шо до необхідних документів, не є достатньо стандартизованими. Інформація надається громадянину усно адміністратором рецепції і, зважаючи на кількість документів, що вимагаються у відповідності до конкретного випадку реєстрації, процедура є досить витратною за часом та менш ефективною, а також несе ризик необхідності надання повторної консультації чи неналежного подання документів. На додаток до інформації в усній формі, адміністратор рецепції надає письмову інформацію про банківські реквізити та бланки заяви для реєстрації місця проживання, а також записує дату/час прийому в реєстратора.</p>	<p>підстави та права громадянина на реєстрацію за конкретною адресою (як під час особистої консультації, так і консультації за телефоном) відбуватиметься згідно типового переліку питань. Окрім того, ЦНАП матиме 4 окремі переліки документів та необхідних процедур, що визначені Постановою Кабінету Міністрів України №207. Перелік необхідних документів буде роздруковано та надано громадянину. Перелік необхідних документів буде додатково містити інформацію про банківські реквізити, а також міститиме спеціальне місце для зазначення суми адміністративного збору та відмітки про бажану дату/час прийому. Бланк заяви для реєстрації за місцем проживання буде виключено з процесу.</p>	<p>Переваги для громадянина</p> <ol style="list-style-type: none"> 1) громадянин точно поінформований та має точне посилання на необхідні документи, які потрібно зібрати, а також про проходження необхідних процедур; 2) підвищується рівень задоволеності громадянина через мінімізацію часу, що необхідний на отримання консультації.
		<p>Переваги для організації</p> <ol style="list-style-type: none"> 1) мінімізує час на надання консультації; 2) зменшує обсяг паперового документообігу; 3) спрощує організаційний процес надання послуги з реєстрації місця проживання.

Таблиця 2.2 - Реєстрація місця проживання. Під-процес 2: Прийом

Процес R-01: Реєстрація місця проживання		
Під-процес 2: Прийом		
Опис «як є»	Опис рекомендованого рішення «як маж бути»	Результат зміни
<p>Під час візиту до ЦНАПу для подання заяви та необхідних документів громадянин зазвичай заповнює заяву про реєстрацію місця проживання в офісі ЦНАПу згідно зразка, що розміщується в зоні рецепції.</p>	<p>Громадянам не потрібно буде заповнювати заяву про реєстрацію місця проживання, скільки вона автоматично генеруватиметься реєстраційним програмним забезпеченням.</p>	Переваги для громадянина
		<p>1) підвищує рівень задоволеності громадянина, оскільки від громадянина не вимагатиметься заповнення заяви; 2) зменшує час присутності громадянина в ЦНАПі з метою подання заяви про реєстрацію/ зняття з реєстрації за місцем проживання.</p>
		Переваги для організації
		<p>1) зменшує обсяг паперового документообігу в організації; 2) підвищує ефективність роботи організації.</p>

Таблиця 2.3 - Реєстрація місця проживання. Суб-процес 1: Оформлення рішення про реєстрацію

Суб-процес 1: Оформлення рішення про реєстрацію		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Суб-процес ґрунтується на перегляді документів у паперовій формі. Реєстратор не має доступу онлайн до реєстраційної бази даних. Якщо у громадянина немає паспорта, в якому</p>	<p>Після перегляду поданих документів реєстратор вносимите дані із поданих документів до нового реєстраційного ПЗ. Заява про реєстрацію місця проживання буде автоматично генеруватися</p>	Переваги для громадянина
		<p>1) підвищує рівень задоволеності клієнта, значно спростивши процес подання та обробки заяви;</p>
		Переваги для організації

Продовження таблиці 2.3 - Реєстрація місця проживання. Суб-процес 1:
Оформлення рішення про реєстрацію

Суб-процес 1: Оформлення рішення про реєстрацію		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>можна проставити штамп, видається довідка про реєстрацію місця проживання, яка розробляється на основі заздалегіть підготовленого бланку в форматі MS Word form. Загалом ЦНАП використовує програмне забезпечення управління документообігом «Універсам послуг» для зберігання електронних копій документів, які подаються громадянином. Після сканування та завантаження документів на вказане ПЗ, адміністратор розробляє бланк «Супровідний лист послуги», який зберігається і не передається громадянину. Форма вимагає підписання громадянином. На додаток до сканування реєстратор копіює всі документи, які подаються громадянином, для формування справи в паперовому архіві.</p>	<p>та роздруковуватися ПЗ. Скасовані копії поданих документів будуть додаватися до відповідного реєстру в новому реєстраційному ПЗ. Під-процес закінчуватиметься наданням громадянину проштампованого паспорту чи довідки про реєстрацію місця проживання. Реєстратор не копіюватиме подані документи на цьому етапі, а копії будуть роздруковуватися на запит уповноважених органів спеціалістами бек-офісу.</p>	
		Переваги для громадянина
		<p>2) зменшує загальний час надання послуги громадянину.</p>
		Переваги для організації

Таблиця 2.4 - Реєстрація місця проживання. Під-процес 3: Управління даними

Під-процес 3: Управління даними		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Наприкінці кожного дня реєстратор вручну</p>	<p>Спеціаліст бек-офісу ЦНАПу на регулярній</p>	
		Переваги для громадянина

Продовження таблиці 2.4 - Реєстрація місця проживання. Під-процес 3:

Управління даними

Під-процес 3: Управління даними		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>вручну заповнює Додатки 4 та 5 (бланки, що передбачені законодавством) на основі реєстраційних даних. Адміністратор заповнює Додатки 4 та 5 у відповідності до типового протоколу для паперового архіву «Картотека».</p> <p>Після архівування спеціаліст бек-офісу вручну вносить дані з Додатків 4 та 5 до трьох різних форм:</p> <ul style="list-style-type: none"> – Програмне забезпечення «Паспортист» ЦНАПу в форматі DOS. – Файл в форматі Excel (бланк) Державної міграційної служби. – Файл Реєстру виборців в форматі Excel <p>Наприкінці кожного дня, дата про реєстрацію місця проживання вручну переносяться з існуючої реєстраційної бази даних до програмного забезпечення «Населення» на робочому місці реєстраторів.</p>	<p>основі роздрукуватиме Додатки 4 та 5, які автоматично генеруватиметься новим реєстраційним програмним забезпеченням і здійснюватиме типову процедуру архівування. Спеціаліст бек-офісу на регулярній основі та на запит генеруватиме відповідні звіти для ДМС, Реєстру виборців та військового комісаріату. Бек-офіс ЦНАПу на запит роздрукуватиме відскановані файли з нового реєстраційного програмного забезпечення і здійснюватиме типову процедуру архівування.</p>	Переваги для громадянина
		<p>Підвищує рівень задоволеності клієнтів через підвищення надійності реєстраційних даних для отримання інших адміністративних послуг.</p>
		Переваги для організації
		<ol style="list-style-type: none"> 1) підвищує ефективність роботи організації та спрощує процес надання послуги; 2) значно зменшує обсяг паперового документообігу; 3) формує електронний архів реєстраційних документів, що нап'язаний з реєстраційною базою даних; 4) збільшує кількість клієнтів, яких можна обслужити за день.

Таблиця 2.5 - Зняття з реєстрації за місцем проживання. Суб-процес 1:
Консультація

Процес R-02: Зняття з реєстрації за місцем проживання		
Суб-процес 1: Консультація		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Протягом суб-процесу консультації адміністратор рецепції прояснює правову підставу та право громадянина на зняття з реєстрації за конкретною адресою, а також надає інформацію щодо необхідних документів для отримання послуги зі зняття з реєстрації за місцем проживання. Процедури щодо визначення правових підстав та права, а також надання інформації щодо необхідних документів, не є достатньо стандартизованим. Інформація надається громадянину усно адміністратором рецепції, зважаючи на кількість документів, що вимагаються у відповідності до конкретного випадку зняття з реєстрації, процедура є досить витратною за часом та менш ефективною, а також несе ризик необхідності надання повторної консультації чи неналежного подання документів. На додаток до інформації в усній формі, адміністратор рецепції надає письмову інформацію про банківські реквізити та бланки заяви для зняття з реєстрації за місцем проживання, а також записує дату/ час прийому в реєстратора.</p>	<p>Прояснення підстави та права громадянина на зняття з реєстрації за конкретною адресою (як під час особистої консультації, так і консультації за телефоном) відбуватиметься згідно типового переліку питань. Окрім того, ЦНАП матиме 4 окремі переліки документів та необхідних процедур, що визначені Постановою Кабінету Міністрів України №207. Перелік необхідних документів буде роздруковано та надано громадянину. Перелік необхідних документів буде додатково містити інформацію про банківські реквізити, а також міститиме спеціальне місце для зазначення суми адміністративного збору та відмітки про бажану дату/час прийому. Бланк заяви для зняття з реєстрації за місцем проживання буде виключено з процесу.</p>	<p>Переваги для громадянина</p> <ol style="list-style-type: none"> 1) громадянин точно поінформований та має точне посилання на необхідні документи, які потрібно зібрати, а також про проходження необхідних процедур; 2) підвищується рівень задоволеності громадянина через мінімізацію часу, що необхідний на отримання консультації.
		<p>Переваги для організації</p> <ol style="list-style-type: none"> 1) мінімізує час для надання консультації; 2) зменшує обсяг паперового документообігу; 3) спрощує організаційний процес надання послуги з реєстрації місця проживання.

Таблиця 2.6 - Зняття з реєстрації за місцем проживання. Під-процес 1:
Прийом

Під-процес 1: Прийом		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Під час візиту до ЦНАПу для подання заяви та необхідних документів громадянин зазвичай заповнює заяву про зняття з реєстрації за місцем проживання в офісі ЦНАПу згідно зразка, що розміщується в зоні рецепції.</p>	<p>Громадянам не потрібно буде заповнювати заяву про зняття з реєстрації за місцем проживання, оскільки вона автоматично генеруватиметься реєстраційним ПЗ.</p>	<p>Переваги для громадянина</p>
		<p>1) підвищує рівень задоволеності громадянина, оскільки від громадянина не вимагатиметься заповнення заяви; 2) зменшує час присутності громадянина в ЦНАПі з метою подання заяви про реєстрацію/ зняття з реєстрації за місцем проживання.</p>
		<p>Переваги для організації</p>
		<p>1) зменшує обсяг паперового документообігу в організації; 2) підвищує ефективність роботи організації.</p>

Таблиця 2.7 - Зняття з реєстрації за місцем проживання. Під-процес 2:
Оформлення рішення про зняття з реєстрації

Під-процес 2: Оформлення рішення про зняття з реєстрації		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Суб-процес ґрунтується на перегляді документів у паперовій формі. Реєстратор не має доступу онлайн до реєстраційної бази даних. Якщо у громадянина</p>	<p>Після перегляду поданих документів реєстратор вноситиме дані із поданих документів до нового реєстраційного ПЗ. Заява про зняття з реєстрації за місцем проживання буде автоматично генеруватися</p>	<p>Переваги для громадянина</p>

Продовження таблиці 2.7 - Зняття з реєстрації за місцем проживання.

Під-процес 2: Оформлення рішення про зняття з реєстрації

Під-процес 2: Оформлення рішення про зняття з реєстрації		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>немає паспорта, в якому можна проставити штамп, видається довідка про реєстрацію місця проживання, яка розробляється на основі заздалегіть підготовленими бланку в форматі MS Word form. Загалом ЦНАП використовує ПЗ управління документообігом «Універсам послуг» для зберігання електронних копій документів, як подаються громадянином. Після скасування та завантаження документів на вказане ПЗ, адміністратор розробляє бланк «Супровідний лист послуги», який зберігається і не передається громадянину. Форма вимагає підписання громадянином. На додаток до скасування реєстратор копіює всі документи, які подаються громадянином, для формування справи в паперовому архіві.</p>	<p>та роздруковуватися ПЗ. Скановані копії поданих документів будуть додаватися до відповідного реєстру в новому реєстраційному ПЗ. Під-процес закінчуватиметься наданням громадянину проштампованого паспорта чи довідки про зняття з реєстрації за місцем проживання. Реєстратор не копіюватиме подані документи на цьому етапі, а копії будуть роздруковуватися на запит уповноважених органів спеціалістами бек-офісу.</p>	Переваги для громадянина
		<p>1) зменшує загальний час надання послуги громадянину.</p>
		Переваги для організації
		<p>1) підвищує ефективність роботи організації та спрощує процес надання послуги; 2) значно зменшує обсяг паперового документообігу; 3) формує електронний архів реєстраційних документів, що нап'язаний з реєстраційною базою даних; 4) збільшує кількість клієнтів, яких можна обслужити за день.</p>

Таблиця 2.8 - Зняття з реєстрації за місцем проживання. Процес R-02.

Під-процес 3: Управління даними

Під-процес 3: Управління даними		
Опис «як є»	Опис рекомендованого рішення «як має бути»	Результат зміни
<p>Наприкінці кожного дня реєстратор вручну заповнює Додатки 4 та 5 (бланки, що передбачені законодавством) на основі реєстраційних даних. Адміністратор заповнює Додатки 4 та 5 у відповідності до типового протоколу для паперового архіву «Картотека».</p> <p>Після архівування, спеціаліст бек-офісу вручну вносить дані з Додатків 4 та 5 до трьох різних форм:</p> <ul style="list-style-type: none"> – ПЗ «Паспортист» ЦНАПу в форматі DOS. – Файл в форматі Excel (бланк) Державної міграційної служби. – Файл Реєстру виборців в форматі Excel <p>Наприкінці кожного дня, дані про зняття з реєстрації за місцем проживання вручну переносяться з існуючої реєстраційної бази даних до ПЗ «Населення» на робочому місці реєстраторів.</p>	<p>Спеціаліст бек-офісу ЦНАПу на регулярній основі роздруковуватиме Додатки 4 та 5, які автоматично генеруватимуться новим реєстраційним ПЗ і здійснюватиме типову процедуру архівування. Спеціаліст бек-офісу на регулярній основі та на запит генеруватиме відповідні звіти для ДМС, Реєстру виборців та військового комісаріату. Бек-офіс ЦНАПу на запит роздруковуватиме відскановані файли з нового реєстраційного програмного забезпечення і здійснюватиме типову процедуру архівування.</p>	<p>Переваги для громадянина</p> <p>Підвищує рівень задоволеності клієнтів через підвищення надійності реєстраційних даних для отримання інших адміністративних послуг.</p>
		<p>Переваги для організації</p> <p>1) підвищує ефективність роботи організації, а також зменшує зайве навантаження на персонал ЦНАПу через автоматизацію процесів; 2) організація більш ефективно відповідає на запити уповноважених органів.</p>

3 ВИБІР ЗАСОБІВ РОЗРОБКИ

3.1 Аналіз та вибір мови програмування

В даний час немає недоліку у виборі засобів програмування: існує безліч різних мов, як універсальних, використовуваних в багатьох сферах програмування і здатних працювати на декількох апаратних і програмних платформах, так і спеціалізованих, орієнтованих на певну область розробки. Існує безліч класифікацій мов програмування: за рівнем семантики, за різноманітністю підходів, методів, стратегій, ідей і понять, що визначають стиль написання програм, з безпеки і по ряду інших характеристик.

Python – високорівнева мова програмування, має підтримку різних парадигм програмування (структурного, об'єктно-орієнтованого та інших). В ній використовується динамічна типізація змінних, забезпечується періодичне звільнення пам'яті від невикористовуваних об'єктів.

З причини широкого ряду функціональних можливостей Python може використовуватися в самих різних сферах. З його допомогою можна здійснювати розробку web-додатків, автоматизованих інформаційних систем, графічних пакетів.

Основні переваги мови Python:

- а) надає засоби швидкого прототипування і динамічної семантики (конструкція мов програмування);
- б) велике співтовариство;
- в) безліч корисних бібліотек і розширень мови, які можна легко використовувати в проектах завдяки гранично уніфікованому механізму імпорту та програмним інтерфейсам;
- г) добре продумані і легкі у використанні механізми модульності;
- д) абсолютно все в Python є об'єктами в сенсі об'єктно-орієнтованого програмування;

е) при високих навантаженнях на додатки використовують Python, який забезпечує більшу продуктивність, ніж в додатках на Ruby, Java або PHP.

Недоліки мови Python:

- а) недосконала підтримка багатопоточності;
- б) початкова обмеженість засобів для роботи з базами даних.

Ruby – інтерпретована мова високого рівня для швидкого і зручного об'єктно-орієнтованого програмування. мова володіє незалежною від операційної системи реалізацією багатопоточності, динамічною типізацією, «збирачем сміття» (механізм, забезпечує механічне управління пам'яттю без втручання програміста) і багатьма іншими можливостями.

мова Ruby використовується в веб-розробці в складі відкритого веб-фреймворку Rails/Ruby on Rails (RoR).

Основні переваги мови Ruby:

- а) реалізує концептуально чисту об'єктно-орієнтовану парадигму;
- б) просунуті методи маніпуляції рядками і текстом;
- в) легко інтегрує в свої програми високопродуктивні сервери баз даних (DB2, MySQL, Oracle і Sybase);
- г) простий синтаксис;
- д) простий програмний інтерфейс для створення багатопоточних додатків;
- е) сучасні засоби для роботи з масивами;
- ж) можливості мови можна розширити за допомогою бібліотек, написаних на C або Ruby;
- и) зарезервовані слова можуть бути ідентифікаторами, якщо це не створює неоднозначності для парсера;
- к) вбудований відладчик.

Недоліки мови Ruby:

- а) висока складність вивчення мови;
- б) недостатньо інформаційних ресурсів, присвячених Ruby;

в) Ruby менш продуктивна в порівнянні з багатьма іншими мовами, застосовуваними в веб-розробці;

г) Ruby відносно повільно розробляється і розвивається.

PHP – мова програмування загального призначення, створена переважно для роботи над web-додатками, для генерації HTML-сторінок і роботи з базами даних. PHP володіє C – подібним синтаксисом і ядром, яке реалізує засоби для автоматичного керування пам'яттю. Вся виділена пам'ять повертається в систему після завершення роботи скрипта.

Динамічні бібліотеки PHP надають широкі можливості для роботи з базами даних, підтримується DBX для роботи на абстрактному рівні, стандарт ODBC; здійснюється комунікація з використанням різних протоколів (IMMAP, SNMP, POP3, HTTP та інші). Також PHP забезпечує роботу з сокетом, динамічною графікою, криптографічними бібліотеками.

Найбільш широке застосування PHP знаходить при розробці web-додатків: 80% сайтів розроблено саме на ній. Безліч різних web-фреймворків і CMS-систем мають в своїй основі мову PHP.

Основні переваги мови PHP:

а) можна створювати сайти на будь-якій платформі і працювати в зв'язці з різними веб-серверами такими, як: Apache, Nginx, IIS;

б) простота і легкість освоєння. Як правило, маючи навіть невеликий досвід в програмуванні на PHP, можна створювати не складні веб-сайти;

в) різниця між PHP і C++ незначна, тому буде простіше зрозуміти одну, якщо є досвід роботи з іншою;

г) PHP підтримує роботу з великою кількістю систем баз даних (MySQL, MSSQL, Oracle, Postgre, MongoDB та інші);

д) розширеність хостингових послуг і їх доступність. Так як хостингові компанії розміщують веб-сайти на PHP на веб-серверах Apache або Nginx, які є безкоштовними, це значно знижує витрати при розробці продукту;

е) постійний розвиток. PHP продовжує розвиватися, виходять всі нові версії, які несуть нові функції. І, як правило, перейти на нову версію не складає труднощів.

Недоліки мови PHP:

а) не підходить для створення системних додатків або системних компонентів;

б) глобальні параметри конфігурації впливають на базовий синтаксис мови, що ускладнює налаштування сервера і ставить додатки в залежність від конкретного оточення;

в) велика кількість застарілих і дублюючих функцій, залишених для сумісності зі старими версіями, суміш процедурних і об'єктних підходів, відсутність єдиного стандарту найменування функцій і параметрів;

г) неможливість багатопотокової і асинхронної роботи (частково вирішується сторонніми модулями);

д) веб-додатки, написані на PHP, часто мають проблеми з безпекою.

Після остаточного визначення вимог до розроблюваної системи був проведений аналіз найактуальніших засобів реалізації цих вимог. В результаті цього у якості мови розробки web-додатку вибрана PHP.

3.2 Огляд та вибір веб-серверу

Веб-сервер - це сервер, що приймає HTTP-запити від клієнтів, зазвичай веб-браузерів, і видає їм HTTP-відповіді, зазвичай разом з HTML-сторінкою, зображенням, файлом, медіа-потокком або іншими даними. Веб-сервери - основа Всесвітньої павутини.

Основною вимогою для веб-сервера є забезпечення спільної роботи всіх попередніх обраних технологічних рішень.

На даний момент існує всього два веб-сервера, які дозволяють реалізувати усю необхідну роботу:

а) Nginx [engine x] – високопродуктивний HTTP-сервер, призначений в основному для роздачі клієнтам статичного контенту (зображень, java script-файлів, css-стилів і т.д.). В основі лежить технологія неблокуючих з'єднань, що при великій кількості одночасних підключень істотно заощаджує ресурси сервера;

б) Apache HTTP-сервер - вільний веб-сервер. За статистикою на березень 2017 року використовується в 45,82% всіх веб-серверів, що робить його найпопулярнішим веб-сервером. Версія 2.2.x володіє перевіреною роками і мільйонами користувачів стабільністю і надійністю. Сервер має велику кількість модулів для роботи з багатьма серверними технологіями.

Можлива спільна робота двох веб-серверів: nginx обробляє запити на роздачу статичного контенту, в той час як Apache займається виконанням серверних скриптів. Однак, вигреш в економії ресурсів сервера буде помітний тільки при високій відвідуваності сервера (починаючи від 10 одночасних підключень). Тому для роботи в якості веб-сервера вибирається Apache HTTP-сервер.

3.3 Огляд та вибір СУБД

У завданнях побудови складних інформаційних систем однією з головних проблем є обмін даними між різними підсистемами. Завдання значно полегшується, якщо дані певного класу будуть переміщатися між підсистемами, за умови, що в цих підсистемах буде закладена технологічно реалізована можливість сприймати вхідні і віддавати вихідні дані в стандартному форматі імпорту/експорту. Даний підхід є основою для розробки метаданих і інтерфейсів для обміну регулярними даними для різних успадкованих різноформатних систем. Тому для коректної роботи метаданих і інтерфейсів, для обміну регулярними даними необхідно правильно оцінити можливості кожної СУБД, що використовується. В даному розділі проводиться порівняльний аналіз функцій найпоширеніших СУБД Oracle,

SQL Server, DB2(IBM) з метою найбільш раціонального використання для конкретного типу проектів, а також обміну даними між ними.

Oracle Database – об'єктно-реляційна система управління базами даних. В даний час є шість версій СУБД:

- Enterprise Edition. Найбільш повна версія, не має значних обмежень. Дана корпоративна редакція продукту призначається для великих підприємств.

- Standard Edition. Обмеження за кількістю процесорних роз'ємів (небільше 4-х). Редакція призначена для використання на підприємствах середнього розміру або в підрозділі великої компанії.

- Standard Edition One. Обмеження за кількістю процесорних роз'ємів (не більше 2-х). Крім того, не підтримується кластеризація.

- Personal Edition. Версія СУБД розрахована на використання одним користувачем.

- Lite. Дана редакція призначена для використання в мобільних та вбудованих пристроях. Також можливе використання на підприємствах малого розміру.

- Express Edition (XE). Безкоштовна версія СУБД. Використовується тільки один процесор, обмеження за кількістю оперативної пам'яті (1 Гб) і обсягом бази даних (11 Гб). Всі шість редакцій СУБД використовують один вихідний код і мають аналогічний функціонал, виключаючи деякі спеціалізовані інструменти конкретних версій.

MS SQL Server – новітня і найпотужніша система управління базами даних, має клієнт-серверну архітектуру. Крім стандартних для СУБД функцій SQL Server 2012 містить великий набір інтегрованих служб з аналізу даних та забезпечує найвищу в своєму класі масштабованість, продуктивність і безпеку.

Безпека SQL Server 2012:

- а) інтерактивне шифрування баз даних. SQL Server 2012 підтримує прозоре шифрування баз даних і журналів транзакцій;
- б) підтримка PKI. SQL Server 2012 підтримує інфраструктуру і сертифікатів. Доступ до даних можна обмежувати і шифрувати за допомогою сертифікатів та апаратних модулів захисту (TPM);
- в) інтеграція з доменом Active Directory. Доступ до даних можна дозволяти за доменом користувачам і групам;
- г) аудит. SQL Server 2012 підтримує аудит доступу до даних і контроль за здійснюваними з базою даних діями.

Відмовостійкість SQL Server 2012:

- а) SQL Server 2012 має у своєму розпорядженні технологію віддзеркалення бази даних (Database Mirroring), яка передбачає наявність одного активного сервера, на який відбувається передача всіх журналів і їх застосування до віддзеркалення. При відмові активного сервера відбувається перемикання на дзеркальний сервер;
- б) кластери високої доступності (High Availability Cluster). SQL Server 2012 підтримує 16-ти вузлову кластеризацію. Кластер SQL Server 2012 дозволяє забезпечити режим функціонування 24/7 для критичних бізнес-додатків.

Масштабованість SQL Server 2012:

- а) швидке додавання пам'яті і процесорів;
- б) реплікація БД. SQL Server 2012 дозволяє створювати копії бази даних у віддалених філіях. Підтримуються різні механізми реплікації з використанням транзакцій, злиття і моментальних знімків;
- в) розподіл ресурсів (Resource Governor). SQL Server 2012 дозволяє розподілять ресурси і призначати пріоритет для різних завдань. Ця функція дозволяє уникнути випадків різкого падіння продуктивності всього сервера і переривання роботи всіх користувачів при виконання будь-якої складної задачі.

IBM Database 2 – кросплатформена система керування базами даних. Основними мовами розробки є C/ C++, підтримує діалект мови SQL.

DB2 не має власної системи аутентифікації користувачів. Замість цього використовуються спеціалізовані сервери безпеки або засоби операційної системи. Після аутентифікації в DB2 проводиться тільки авторизація користувачів. При роботі з SQL-сервером DB2 має можливість діагностики та обробки помилки, тобто, реалізований принцип компіляції запитів, а не інтерпретації. Інтерфейс кінцевого користувача забезпечується окремим периферійним продуктом – QMF (Query Management Facility). Основними елементами DB2 є: прекомпілятор (Precompiler); генератор планів прикладних задач (Bind); супервізор стадії виконання (Runtime supervisor); програма управління збереженими даними (Stored data manager).

Було порівняно декілька різноманітних популярних СУБД з врахуванням таких критерій, як зручність роботи, масштабованість, спроможність інтегруватися з іншими продуктами, які будуть використовуватись для розробки продукту. Крім того, була звернена увага на вартість системи і підтримки, що надається розробником. В результаті цього була вибрана система управління базою даних – Microsoft SQL Server, яка більш всього відповідає поставленим критеріям.

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ

4.1 Розробка та реалізація бази даних

У базі даних є два типи таблиць: допоміжні таблиці для забезпечення функціонування та розмежування прав користувачів і безпосередньо основні таблиці, що є частиною інформаційної системи (рис. 4.1).

Основні таблиці з даними системи: Regs, Reg_files, UnReg_files, Peoples, Adrs, Ref_mil_doc_type, Mils, Names, Ref_Adm, Ref_Street_Type, Ref_Street, Documents, Users, User_operations, Role_user, Ref_operation.

Regs – таблиця зі списком усіх реєстрацій:

- id – внутрішній ідентифікаційний номер реєстрації/зняття з реєстрації (реєстраційна картка);
- addr_id – ідентифікатор адреси;
- people_id – внутрішній ідентифікатор людини;
- reg_name_id – ідентифікатор реєстраційної дії (картка реєстрації);
- reg_document_id – ідентифікатор реєстраційного документу;
- reg_agent_id – ідентифікаційний номер оператора системи, який проводить операцію;
- reg_date – дата реєстрації;
- reg_place – місце реєстрації;
- reg_place_koatuu – ідентифікатор місця народження згідно АТТУ;
- reg_place_detail – місце народження (детально), якщо назва відсутня в АТТУ або місце народження не є в Україні;
- reg_legal_docs – документи, що підтверджують особу;
- reg_mil_check – наявність військового обов’язку;
- updated_at – дата та час зміни запису;
- updated_by – дата та час створення запису;
- reg_phone – реєстраційний телефон;

- reg_refuse_reason – причина відмови;
- reg_mil_id – ідентифікатор військового обліку (дані щодо деталей військового обліку);
- reg_refuse_type – тип відмови (згідно довідника);
- reg_cancel_date – дата відмови у реєстрації;
- reg_notes – реєстраційні відмітки або нотатки;
- reg_relation_id – ідентифікатор родинного зв'язку;
- reg_type – тип реєстрації;
- reg_date_end – дата закінчення реєстрації;
- reg_archive_info – номер архівної справи;
- reg_case_number – номер шафи в архіві;
- reg_number – номер полиці в архіві;
- reg_create_date – дата створення запису реєстрації.

Reg_files – таблиця з посиланнями на відскановані документи:

- id – ідентифікатор реєстрації (картка реєстрації);
- reg_id – реєстраційний номер;
- file_id – ідентифікатор файлу;
- type_id – тип файлу (pdf, doc та інші);
- page – відскановані документи;
- comment – коментарі щодо сканованих документів;
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису.

UnReg_files – таблиця зняття з реєстрації.

- id – ідентифікатор зняття з реєстрації (картка реєстрації);
- unreg_id – реєстраційний номер;
- file_id – ідентифікатор файлу;
- type_id – тип файлу (pdf, doc та інші);
- page – відскановані документи;
- comment – коментарі щодо сканованих документів;

- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису.

Peoples – таблиця в яку заносяться дані особи під час реєстрації:

- id – ідентифікаційний номер;
- birthdate – дата народження;
- sex – стать;
- nationality_id – громадянство;
- birth_place – місце народження;
- birth_place_id – код місця народження;
- ipn – індивідуальний податковий номер ;
- eddr – № запису у єдиному державному демографічному реєстрі ;
- death_date – дата смерті, якщо помер;
- is_dead – відмітка про смерть людини;
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису;
- note – примітки/нотатки.

Adrs – таблиця з адресами осіб:

- id – ідентифікаційний номер;
- street_id – код вулиці;
- building – будівля;
- section – секція;
- template_id – шаблон для цієї адреси (дім, або дім та квартира);
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису;
- complete – дані повні;
- incomplete – дані не повні;
- has_privat – чи приватизовано житло;
- archive_info – номер архівної справи (картка Б).

Ref_mil_doc_type – таблиця типів військових документів:

- id – ідентифікатор військового документу;
- name – назва військового документу.

Mils – таблиця військовозобов'язаності осіб:

- id – ідентифікаційний номер;
- mil_category – категорія військового обліку;
- mil_issuer – ким виданий документ;
- mil_doc_type_id – тип військового обов'язку;
- mil_seria – серія документу;
- mil_number – номер документу;
- mil_salutation – військове звання;
- mil_specialization – військова спеціальність;
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису.

Names – таблиця осіб:

- id – ідентифікаційний номер;
- people_id – людина;
- surname – по батькові;
- name – ім'я;
- patronymic – по батькові;
- date_from – дата від початку зміни даних;
- updated_at – дата закінчення зміни даних;
- updated_by – дата створення запису;
- strong – повна ПІБ особи;
- fonetic_surname – по батькові (фонетика для використання семантичного словника для пошуку);
- fonetic_name – ім'я (фонетика для використання семантичного словника для пошуку);

– fonetic_patronymic – по батькові (фонетика для використання семантичного словника для пошуку).

Ref_Adm – таблиця АТУУ:

- id – ідентифікатор населеного пункту;
- name – назва населеного пункту;
- type – тип населеного пункту (село, місто тощо);
- full_name – повна назва населеного пункту;
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису;
- district – район;
- oblast – область.

Ref_Street_Type – таблиця типів вулиць:

- id – ідентифікатор типу вулиці;
- name – ім'я типу вулиці;
- shortname – скорочена назва вулиці;
- drv_name – назва вулиці згідно даних ДРВ.

Ref_Street – таблиця вулиць:

- id – ідентифікатор вулиці;
- name – назва вулиці;
- type_id – тип вулиці;
- updated_at – дата останньої зміни запису;
- updated_by – дата створення запису;
- city_id – код міста;
- merged_with – нова назва вулиці після перейменування;
- deleted – скасування;
- approved – затвердження.

Documents – таблиця документів:

- id – ідентифікатор військового обліку;

- `people_id` – ідентифікатор людини;
- `type_id` – тип військового документу;
- `seria` – серія документу;
- `number` – номер документу;
- `date_from` – документ дійсний від;
- `date_to` – документ дійсний до;
- `issuer` – ким виданий документ;
- `updated_at` – дата останньої зміни запису;
- `updated_by` – дата створення запису.

`Users` – таблиця даних користувача системи:

- `id` – ідентифікатор користувача;
- `name` – ім'я користувача;
- `username` – логін користувача;
- `password` – пароль користувача;
- `disabled` – активний користувач чи ні;
- `remember_token` – тип ідентифікації (логін та пароль чи токен);
- `updated_at` – дата останньої зміни запису;
- `updated_by` – дата створення запису;
- `key_token` – цифровий ключ ЕЦП;
- `settings_id` – настройки;
- `api_token` – як обробляти перевірку ЕЦП.

`User_operations` – таблиця операцій виконуваних користувачами:

- `id` – ідентифікаційний номер;
- `oper_type` – тип операції;
- `user_id` – номер користувача;
- `oper_date` – дата виконання операції.

`Role_user` – таблиця ролей користувачів:

- `user_id` – ідентифікатор користувача;

- role_id – ідентифікатор ролі.

Ref_operation – таблиця повідомлень про зміни зроблені користувачем:

- id – ідентифікаційний номер;
- name – ім'я.

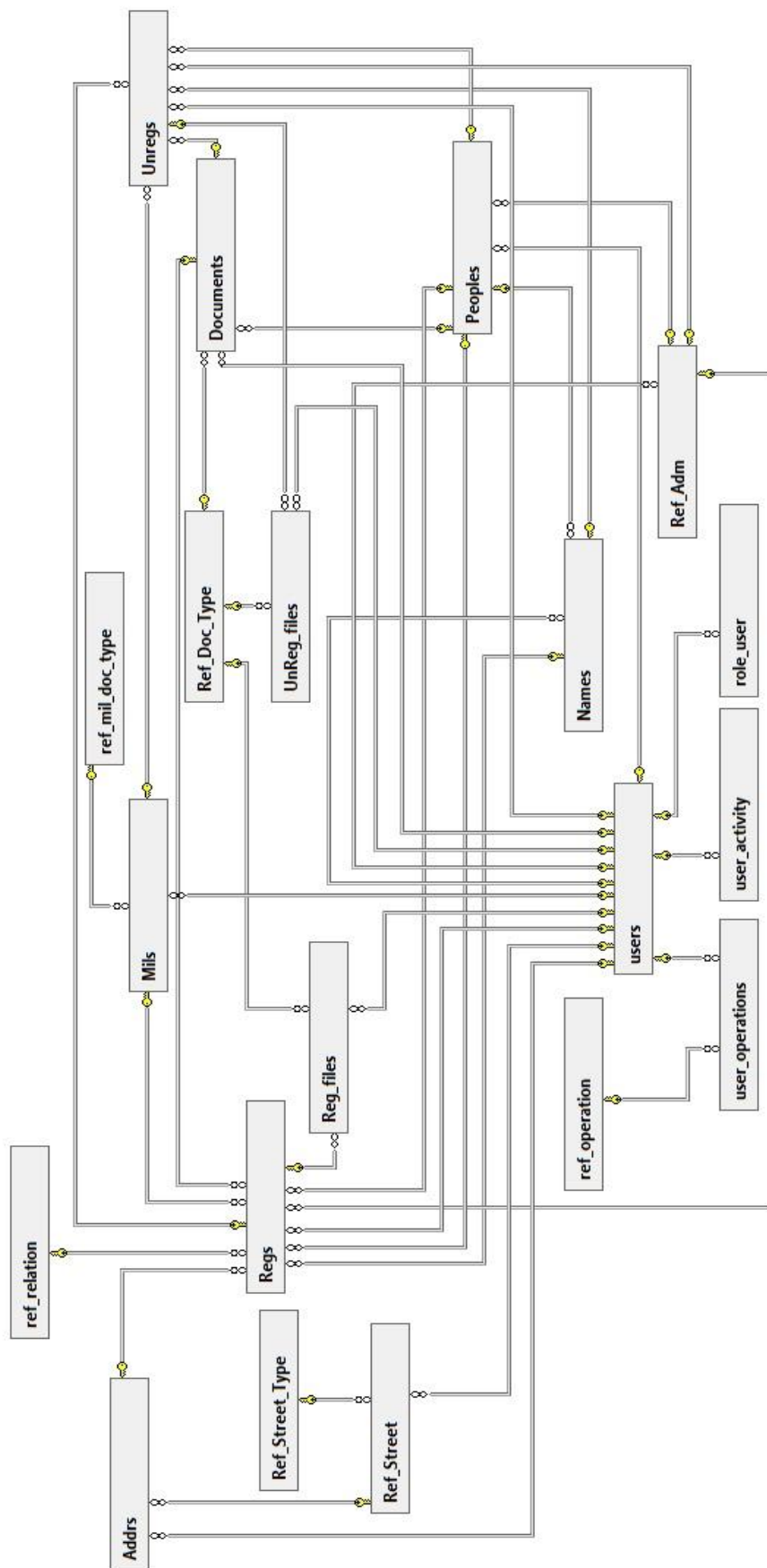


Рисунок 4.1 – Структура бази даних розроблюваної системи

4.2 Структура системи

Основні часто використовувані функції системи призначені безпосередньо для операторів даної системи. Дивлячись на це будуть розглянуті схеми алгоритмів роботи користувача у розробленому web-додатку. Лістинг програми наведений у Додатку Е.



Рисунок 4.2 – Схема алгоритма входу користувача до системи

На рисунку 4.3 наведена блок-схема алгоритму створення нового користувача адміністратором системи. Для цього йому необхідно відкрити розділ меню «Користувачі» і вибрати пункт «Управління ролями».



Рисунок 4.3 – Схема алгоритма створення нового користувача

На рисунку 4.4 зображена блок-схема алгоритму редагування профілю вже зареєстрованого користувача. Для цього адміністратору системи потрібно поетапно виконати ряд операцій.

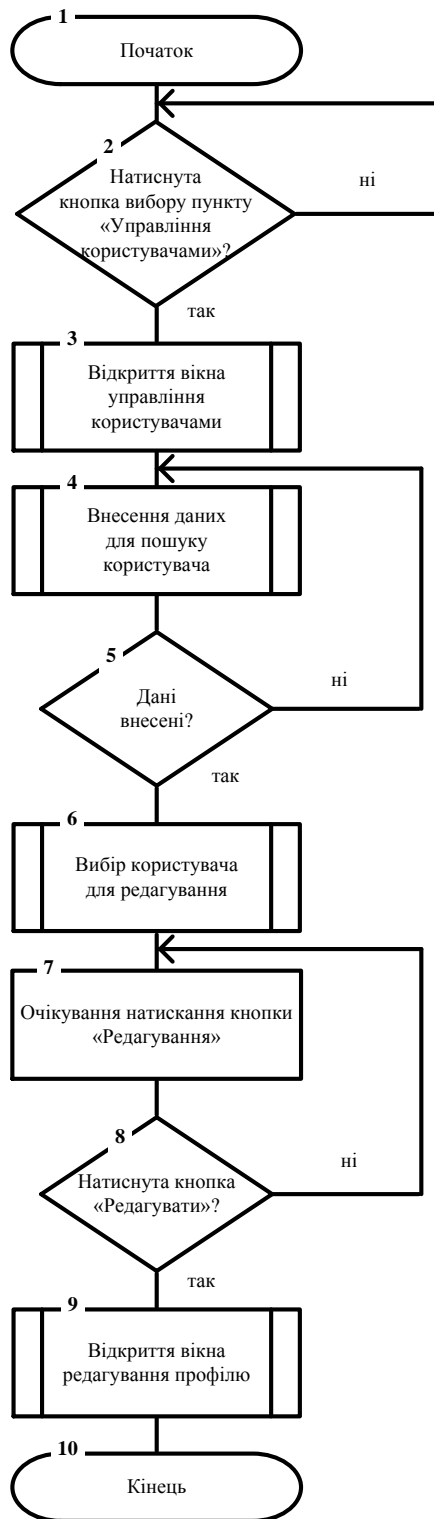


Рисунок 4.4 – Схема алгоритму редагування профілю користувача

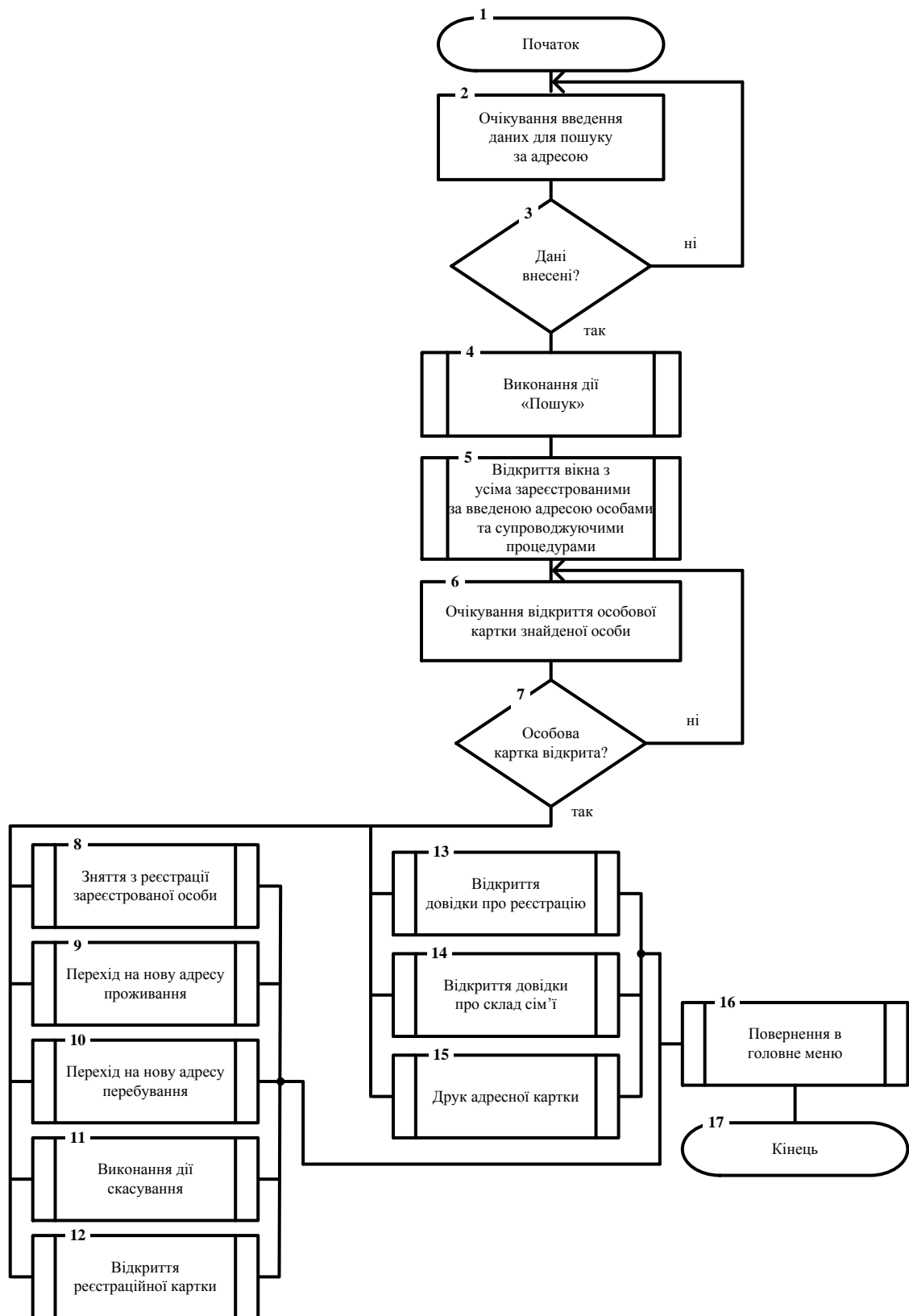


Рисунок 4.5 – Схема алгоритму роботи у розділі головного меню «Пошук за адресою»

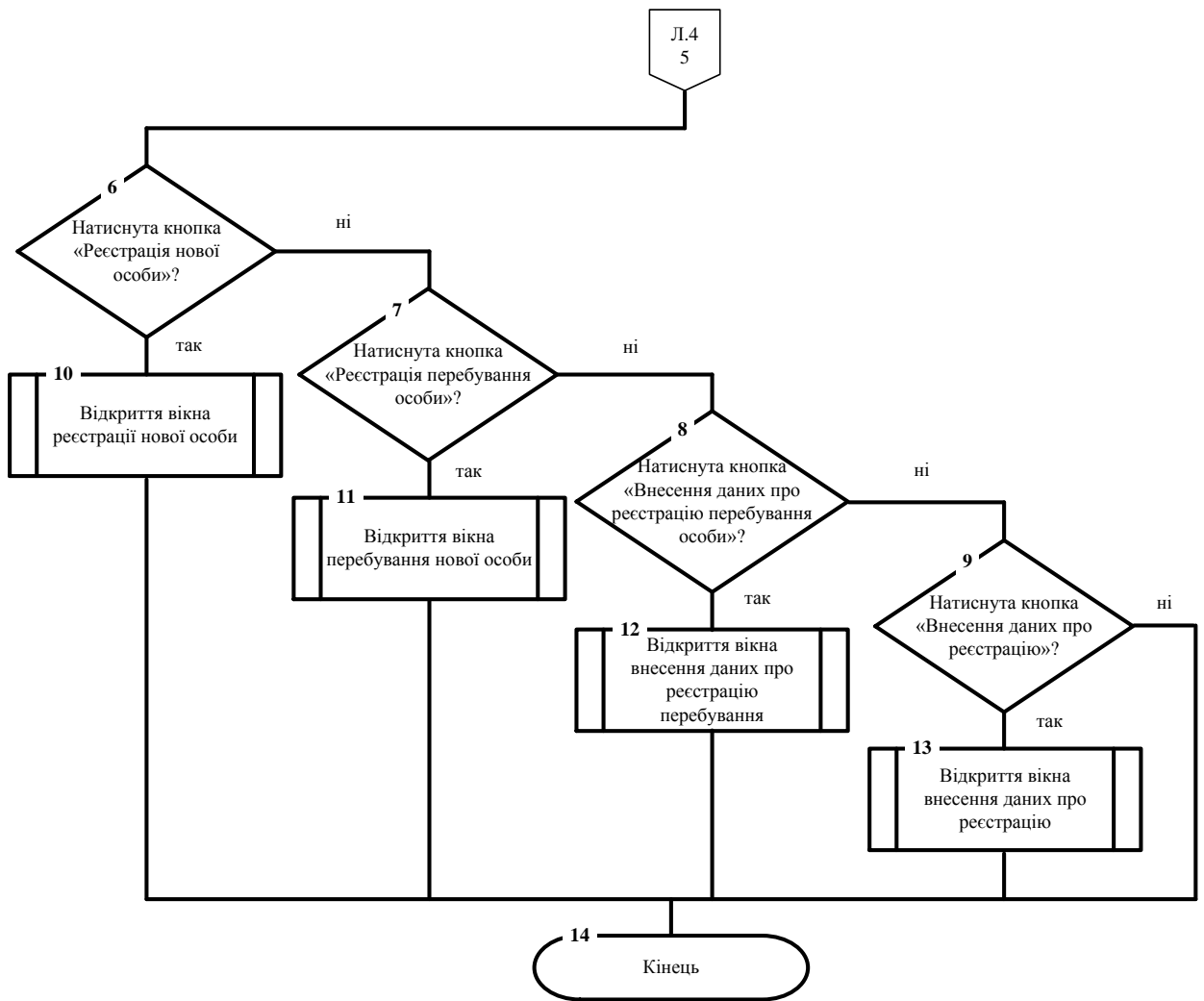


Рисунок 4.6 – Схема роботи функціоналу у розділі головного меню «Пошук за адресою».

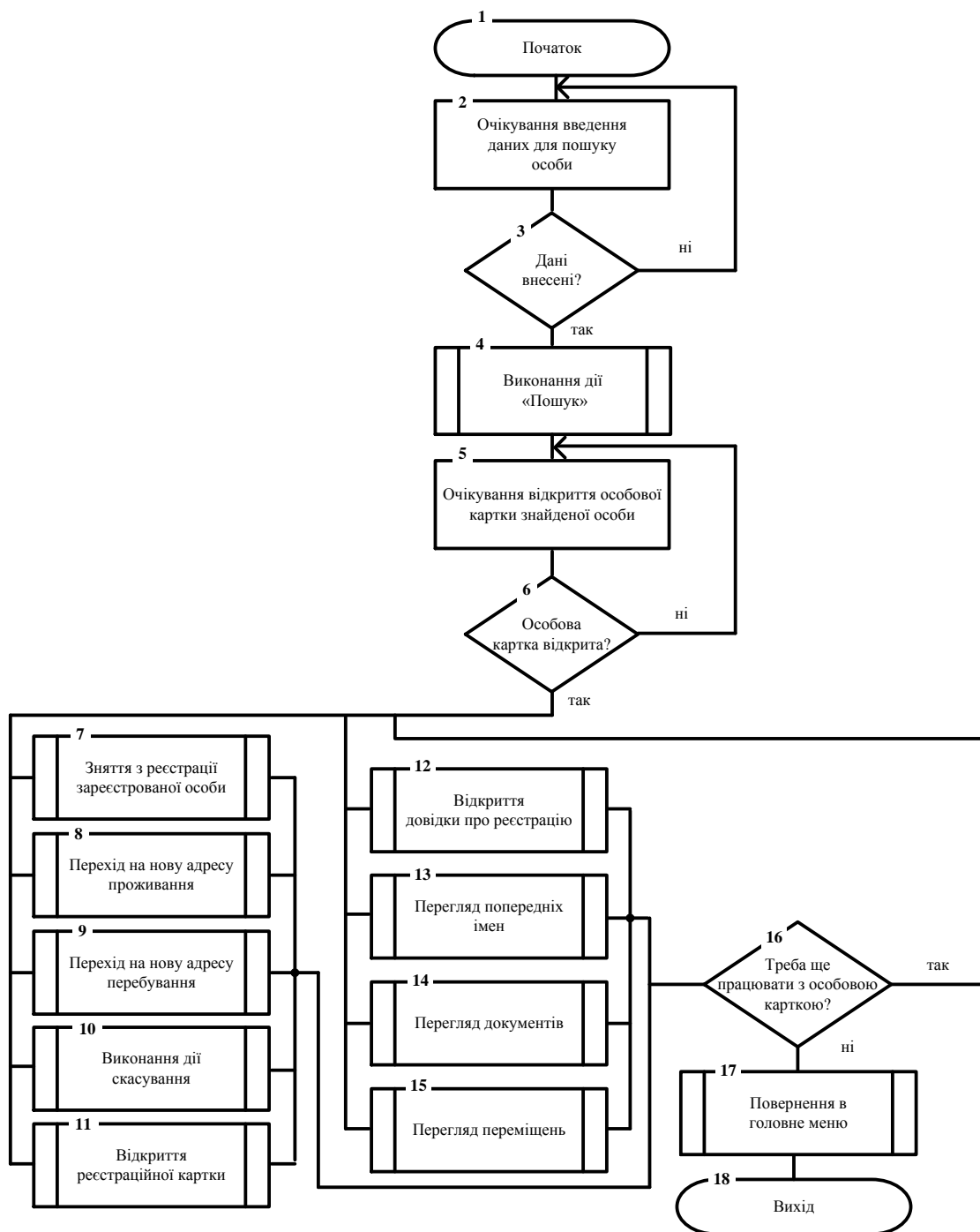


Рисунок 4.6 – Схема алгоритму роботи розділу головного меню «Пошук особи».

5 КЕРІВНИЦТВО КОРИСТУВАЧА

5.1 Опис інтерфейсу програми

Додаток являє собою веб-інтерфейс, а безпосередній зв'язок з самим додатком здійснюється по локальній мережі.

Після введення адреси додатка, з'являється меню авторизації (рис. 5.1), яке вимагає введення імені користувача і пароля для подальшої роботи.

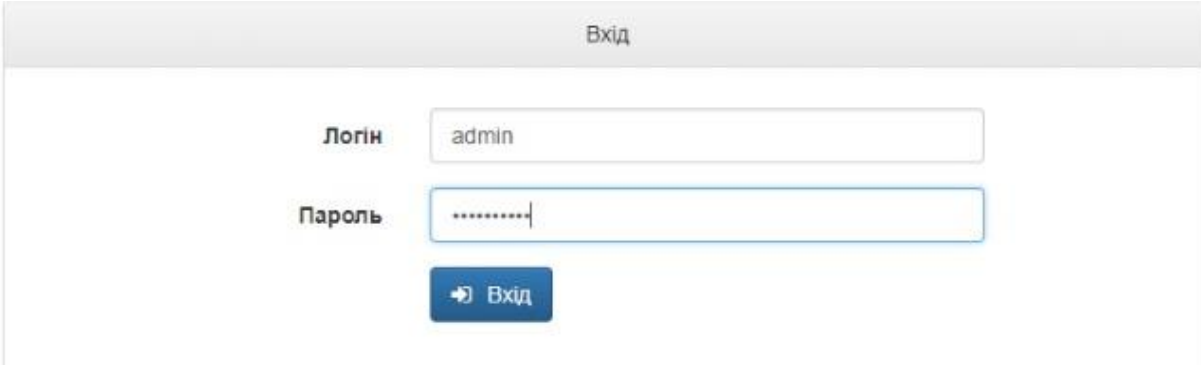


Рисунок 5.1 – Процес авторизації користувача

Після входу в систему користувач потрапляє на сторінку головного меню, в якому знаходиться весь доступний інструментарій (рис. 5.2).

З головного меню можливий перехід в будь-який конкретний інформаційний розділ.

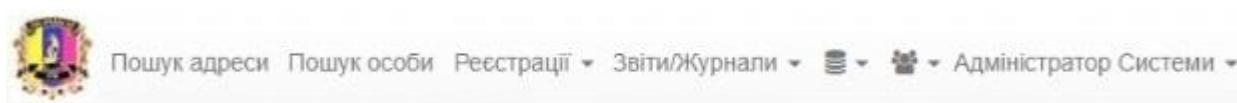


Рисунок 5.2 – Структура головного меню сайту

Структура головного меню сайту:

- пошук адреси;
- пошук особи;
- пошук реєстрації;
- поточна реєстрація;

- звіти/журнали (вбудовані звіти, зовнішні звіти);
- довідники (типи документів, родинні стосунки, імена, по батькові);
- користувачі (список користувачів, управління ролями, дії користувачів);
- адміністратор системи.

На початку роботи в програмі зареєстровано тільки одного користувача, а саме – Адміністратора Системи. Для коректного обліку дій, чіткого розмежування обов'язків відповідно до посади працівника ЦНАПу кожному оператору надається окремий доступ до системи. Для цього потрібно зареєструвати оператора як нового користувача програмного комплексу. Для цього адміністратору треба обрати пункт [Користувачі > Список користувачів] (рис.5.3) і ввести дані оператора, під якими він надалі буде працювати в системі (рис. 5.4).

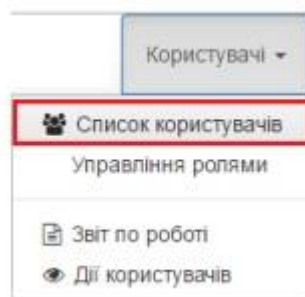


Рисунок 5.3– Вибір пункту списку користувачів

Повне ім'я	Адміністратор Системи		
Логін	admin	Заборонений	<input type="checkbox"/>
Пароль	<input type="password"/>	Повторний пароль	<input type="password"/>
Підпис	Адміністратор управління забезпечення надання адміністративних послуг Рівненської міської ради		

Ролі			
Адміністратор системи	<input checked="" type="checkbox"/>	Керування користувачами	<input checked="" type="checkbox"/>
Робота зі сторожами	<input checked="" type="checkbox"/>	Виправлення помилок	<input checked="" type="checkbox"/>
Виконання звітів	<input checked="" type="checkbox"/>	Робота з довідниками	<input checked="" type="checkbox"/>
Скасування реєстрацій	<input checked="" type="checkbox"/>	Робота з вхідними повідомленнями	<input checked="" type="checkbox"/>
Необмежена к-ть входів	<input checked="" type="checkbox"/>	Зняття по рішенням суду	<input checked="" type="checkbox"/>
Видалення записів	<input checked="" type="checkbox"/>	Комунальна служба	<input type="checkbox"/>
Реєстратор ЦНАП	<input checked="" type="checkbox"/>	Керування ролями	<input checked="" type="checkbox"/>
Адміністратор ЦНАП	<input type="checkbox"/>		

Рисунок 5.4 – Реєстрація нового користувача

Після цього оператор має можливість виконувати роботу в програмному комплексі відповідно до наданого йому набору функцій.

5.2 Основні операції користувача

Кожен з користувачів має обмежене право доступу до функціоналу, відносно своєї ролі. У декількох користувачів може бути однаковий доступ до однієї та тієї ж функції.

У цьому розділі наведено основні дії оператора при відвідуванні розробленого web-додатку. Користувачам, які пройшли процедуру реєстрації, можуть призначити певну роль. По замовчуванню в системі доступний набір ролей такі рекомендовані ролі:

а) робота зі сторожами. Доступ до встановлення юридично впровадженої заборони на здійснення будь-яких дій по певній адресі;

б) виконання звітів. Користувач цієї функції має можливість формування звітів про реєстрацію/зняття з реєстрації зареєстрованих та осіб знятих з реєстрації, тощо;

в) скасування реєстрацій. Доступ до скасування реєстрацій дозволяє скасування некоректно або помилково зареєстрованої особи;

г) видалення записів. Доступ до даної функції дає можливість видалення записів про особу та місце її реєстрації;

д) керування користувачами. Дана роль дає доступ до всіх функціональних можливостей вкладки «Користувачі»;

е) робота з довідниками. Роль роботи з довідниками дозволяє редагування інформації та внесення нових даних до вкладки «Довідники»;

ж) робота з вхідними повідомленнями. Доступ до роботи в «Режимі внесення даних за повідомленням»;

и) керування ролями. Дана роль дає можливість розподілу функціоналу між користувачами системи;

к) реєстратор ЦНАПу. Реєстратор системи має значно обмежений доступ до такого функціоналу, як:

- 1) робота з іменами;
- 2) робота з особовими картками;
- 3) довідниками;
- 4) робота з реєстраціями;
- 5) робота зі зняттям з реєстрації.

з) адміністратор ЦНАПу має більш розширений доступ ніж Реєстратор ЦНАПу, але менший ніж у Адміністратор Системи:

- 1) робота з адресними картками;
- 2) робота з документами;
- 3) робота зі сканованими документами;
- 4) робота з фотоданими;
- 5) робота з іменами;
- 6) робота з особовими картками;

- 7) довідниками;
- 8) робота з реєстраціями;
- 9) робота зі зняттям з реєстрації.

к) адміністратор системи. Адміністратор має найвищий рівень доступу та можливість редагування всіх ролей. Крім цього він має доступ до розділу адміністратора, де може керувати іншими користувачами.

Деякі користувачі можуть одночасно мати доступ до деяких функцій. Будь-який користувач може одночасно виконувати кілька ролей. Доступні ролі відмічаються галочками. Виняток – привілеї адміністратора системи. При коректній роботі системи користувач не матиме можливості працювати із закритими для нього розділами.

5.3 Керівництво адміністратора

Як зазначалося у попередньому розділі, Адміністратор Системи має необмежені можливості роботи в програмі. Сайт має спеціалізований розділ меню для адміністратора системи, з якого він може здійснювати управління сайтом. У розділі «Пошук за адресою» (рис. 5.5) адміністратору доступний пошук особи за веденою адресою, під якою вона занесена до бази, перегляд даних про цю особу, реєстрація нової особи/зняття її з реєстрації, тощо.

вулиця	Будинок	Квартира
Л. України Українки_Л.	28	
вулиця Л. України Українки_Л.	Будинок	Корпус/квартира
вулиця Л. України Українки_Л.	Будинок 28	квартира 1
вулиця Л. України Українки_Л.	Будинок 28	квартира 2

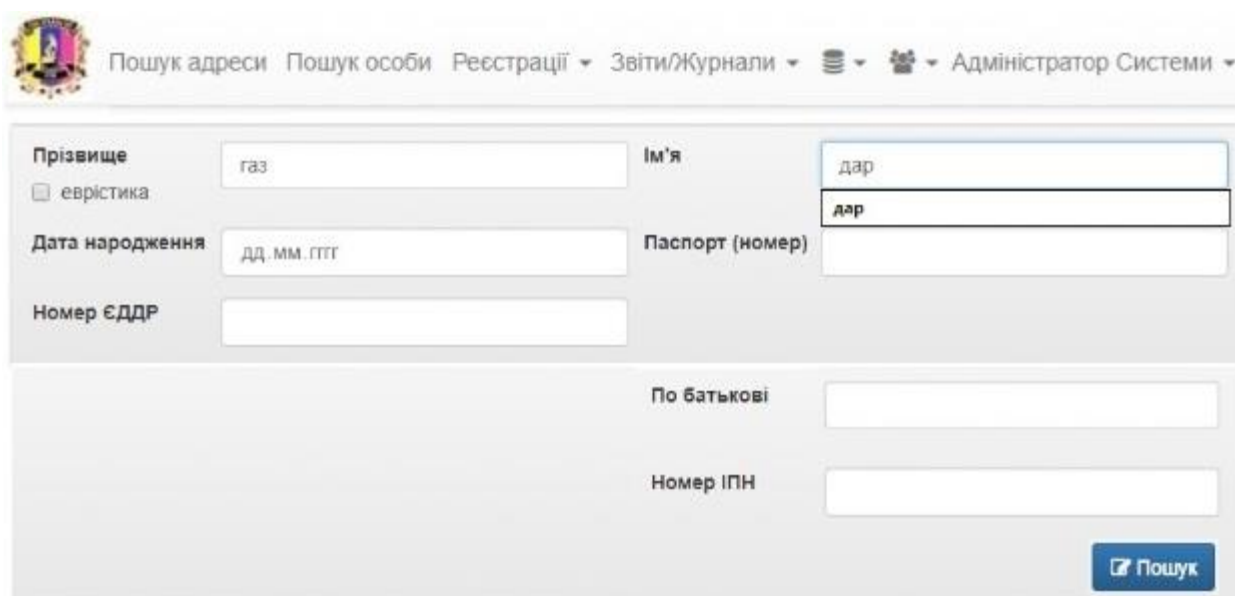
Рисунок 5.5 – Сторінка введення даних для пошуку за адресою

Після введення адреси певної особи і відкриття головного вікна даного інформаційного розділу (Додаток Ж) є можливість виконання таких дій, як: «Реєстрація нової особи», «Реєстрація перебування нової особи», «Внесення даних про реєстрацію перебування», «Внесення даних про реєстрацію»,

«Друк адресної картки», «Склад сім'ї». Для перегляду відомостей про вищезазначену особу та даних про її реєстрацію/зняття та скасування з реєстрації, Адміністратор Системи може відкрити особову (Додаток И) чи реєстраційну картки (Додаток К).

На сторінці перегляду реєстраційної картки (Додаток И) оператор має можливість перегляду реєстраційних даних певної особи та перевірки її активності.

Для роботи з пошуком особи адміністратору необхідно так само, як і під час роботи з розділом «Пошук адреси», ввести орієнтовні дані (рис. 5.6).



The screenshot shows a web interface for searching a person. At the top, there is a navigation bar with a logo on the left and menu items: «Пошук адреси», «Пошук особи», «Реєстрації», «Звіти/Журнали», and «Адміністратор Системи». Below the navigation bar is a search form with the following fields and options:

- Прізвище** (Last Name): Input field containing «газ».
- Ім'я** (First Name): Input field containing «дар». Below it is a dropdown menu with «дар» selected.
- Дата народження** (Date of Birth): Input field with a date format «дд. мм. гггг».
- Паспорт (номер)** (Passport Number): Input field.
- Номер ЄДДР** (ID Number): Input field.
- По батькові** (Patronymic): Input field.
- Номер ІПН** (Tax ID Number): Input field.
- евристика** (heuristics): A checkbox.
- Пошук** (Search): A blue button with a magnifying glass icon.

Рисунок 5.6 – Сторінка пошуку особи

Після цього відкривається головне вікно «Пошук особи» з повним набором функціоналу (рис. 5.7).

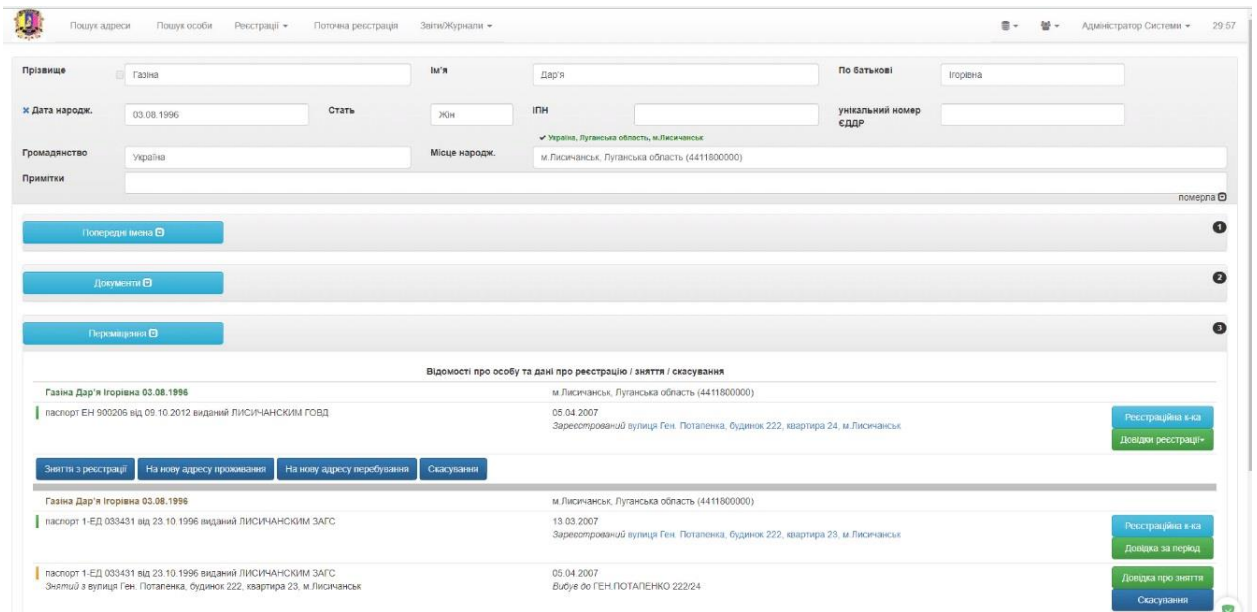


Рисунок 5.7 – Демонстрація вікна «Пошук особи»

Для перегляду особистої інформації необхідної особи потрібно відкрити її реєстраційну картку (рис. 5.8).

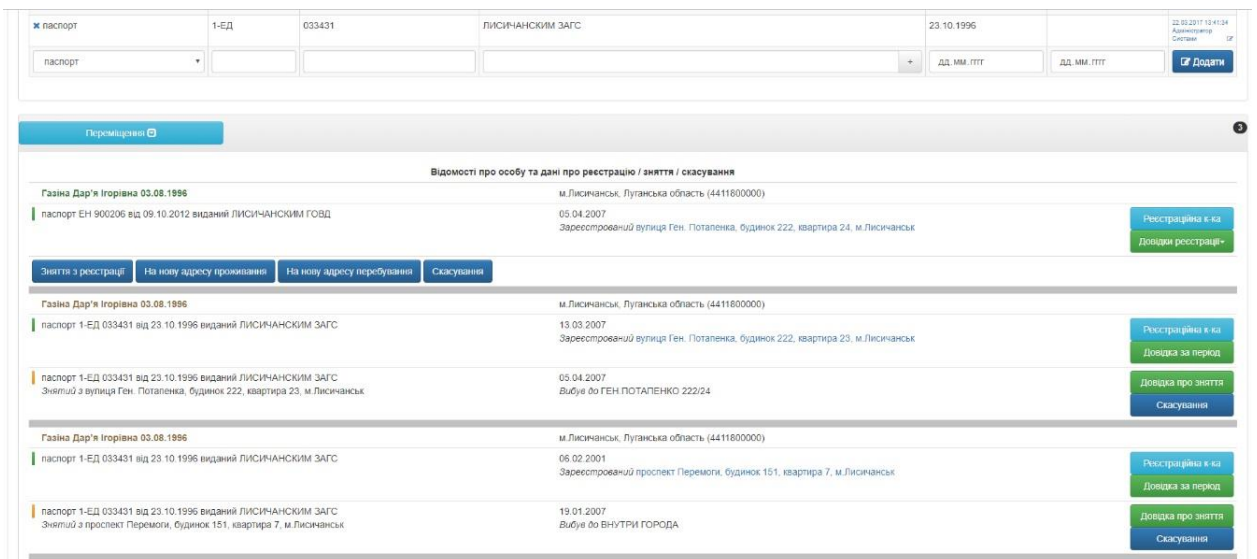


Рисунок 5.8 – Демонстрація вікна «Реєстраційна картка»

На сторінці розділу меню «Реєстрації» (рис. 5.9) є можливість пошуку та перегляду зареєстрованих осіб за певний проміжок часу з використанням додаткового фільтрів.

Дата з Проживання По даті початку реєстрації

Дата до Перебування По даті закінчення реєстрації

Рисунок 5.9 – Сторінка введення даних для пошуку за реєстрацією

Результатом пошуку є поява вікна з особами, які задовільняють параметрам пошуку.

При натисканні на розділ меню «Звіти/Журнали» з'являється випадаючий список (рис. 5.13) в якому можливий перехід у будь-який із запропонованих підрозділів.

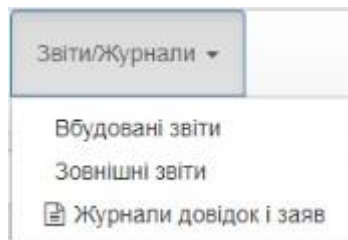


Рисунок 5.10 – Демонстрація випадаючого списку розділу «Звіти/Журнали»

У разі вибору розділу списку «Вбудовані звіти» можливо вибрати тип звіту:

- відомості для статистики;
- періодичне поновлення даних державного реєстру виборців;
- звіт про склад сім'ї.

На сторінці «Журнали довідок і заяв» (рис. 5.11) надані шаблони усіх заяв використовуваних для реалізації реєстрації/зняття з реєстрації та сумісних з нею процесів.

	Назва	Варіанти друку
upload	Заява на реєстрацію проживання	звичайна з представником
upload	Заява на реєстрацію проживання з відмовою	звичайна з представником
upload	Заява на реєстрацію проживання малолітньої	звичайна
upload	Заява на реєстрацію проживання малолітньої з відмовою	звичайна
upload	Заява на зняття з реєстрації проживання	звичайна з представником
upload	Заява на зняття з реєстрації проживання з відмовою	звичайна з представником
upload	Заява на зняття з реєстрації проживання малолітньої	звичайна
upload	Заява на зняття з реєстрації проживання малолітньої з відмовою	звичайна
upload	Заява на реєстрацію перебування	звичайна з представником
upload	Заява на реєстрацію перебування з відмовою	звичайна з представником
upload	Заява на реєстрацію перебування малолітньої	звичайна
upload	Заява на реєстрацію перебування малолітньої з відмовою	звичайна
upload	Заява на зняття з реєстрації перебування	звичайна з представником
upload	Заява на зняття з реєстрації перебування малолітньої	звичайна
upload	Заява на зняття з реєстрації померлої особи	з представником
upload	Заява на зняття з реєстрації по рішення суду	з представником
upload	Заява на довідку про померлу особу	звичайна
upload	Заява на довідку по відсутність зареєстрованих	звичайна
upload	Заява на довідку про склад сім'ї	звичайна

Рисунок 5.11 – Демонстрація вікна «Журнали довідок і заяв»

При натисканні на розділ меню «Довідники» з'являється випадаючий список (рис. 5.12), в якому можливий перехід у будь-який із запропонованих підрозділів.

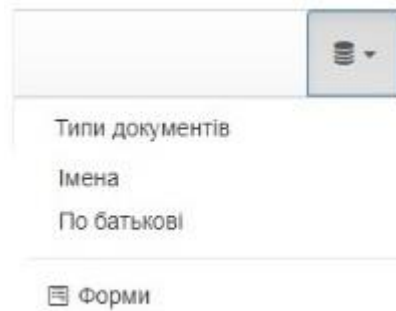


Рисунок 5.12 – Демонстрація випадаючого списку розділу «Довідники»

При відкритті вікна «Типи документів» з’являється перелік усіх типів документів, які можуть приймати участь у процесі реєстрації/зняття з реєстрації (паспорт, ідентифікаційний код, письмова згода, довіреність, іноземний паспорт, службове посвідчення).

У розділі списку «Імена» та «По батькові» можливо побачити абсолютно усі імена осіб занесених до бази.

На сторінці меню «Адміністратор Системи» при натисненні на розділ «Профіль» (рис. 5.13) даного списку адміністратор має можливість повернення до налаштування профілю.

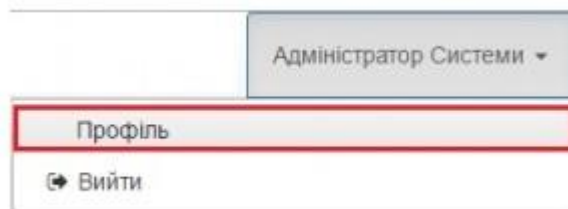


Рисунок 5.13 – Демонстрація випадаючого списку розділу «Адміністратор Системи».

При натисканні на розділ списку «Вийти» відбувається перехід на сторінку з меню авторизації (рис. 5.1).

6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної роботи бакалавра являє собою дослідження методів, підходів, технологій розробки комплексної системи реєстрації та ведення територіальної громади. Метою роботи є автоматизація процесів надання послуг реєстрації/зняття з реєстрації фізичних осіб за місцем проживання, для підвищення ефективності та якості обслуговування клієнтів. Так як в процесі проектування виконувалось у побутових умовах, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера, на якому розробляється система.

6.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [31] визначається, що охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика,

електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» [31] (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України, Закону України «Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності» та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

6.2 Аналіз стану умов праці

Робота над проектуванням та розробка навчального додатку проходить в побутовому приміщенні. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

6.2.1 Вимоги до приміщення

Геометричні розміри приміщення зазначені у таблиці 6.1

Таблиця 6.1 – Розміри робочого місця

Параметр	Значення
Довжина, м	5

Продовження таблиці 6.1 – Розміри робочого місця

Параметр	Значення
Ширина, м	3
Висота, м	2,5
Площа, м ²	15
Об'єм, м ³	37,5

Згідно до санітарних норм мікроклімату виробничих приміщень [32] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 м², а об'єм — не менше 20 м³. Отже, дане приміщення цілком відповідає зазначеним нормам.

6.2.1 Вимоги до організації робочого місця

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця [33] (табл. 6.2) і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 6.2 – Характеристика робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибин сидіння, мм	400	не менше 400

Продовження таблиці 6.2 – Характеристика робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700÷800

Приміщення кабінету знаходиться на другому поверсі чотирьох поверхової будівлі і має об'єм 37,5 м³, площу – 15 м². У цьому кабінеті обладнано одне робоче місце, яке укомплектовано одним персональним комп'ютером.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Система вентилявання приміщення — природна, а опалення — централізоване.

6.2.2 Навантаження та напруженість процесу праці

За фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити

додаткові регламентовані перерви тривалістю 15 хв через кожну годину роботи [34].

6.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

6.3.1 Аналіз небезпечних та шкідливих факторів при розробці виробу

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 6.3). Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00.-1.28-10 [35], які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери та периферійні пристрої. Основними робочими характеристиками персонального комп'ютера є:

- робоча напруга $U=+220\text{В} \pm 5\%$;
- робочий струм $I=2\text{А}$;
- споживана потужність $P=600\text{ Вт}$.

Робочі місця мають відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин [34].

Таблиця 6.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Нормативні документи
Фізичні		
підвищена температура поверхонь обладнання	експлуатація ЕОМ, серверного обладнання для роботи	ДСН 3.3.6.042-99 [32]
підвищена або знижена вологість повітря	-//-	ДСН 3.3.6.042-99 [32]
підвищена або знижена рухливість повітря	-//-	ДСН 3.3.6.042-99 [32]
підвищений рівень напруги електричної мережі	-//-	ГОСТ 12.1.030-81 [36] ГОСТ 13109-97 [37]
підвищений рівень статичної електрики	-//-	ГОСТ 12.1.030-81 [36]
підвищена напруженість електромагнітного поля	-//-	ГОСТ 12.1.006-84 [38]
недостатність природного світла	порушення умов праці (вимог до приміщень)	ДБН В.2.5-28:2015 [39]
недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	ДБН В.2.5-28:2015 [39]
Психофізіологічні		
нервово-психічне перевантаження	розумова робота над проектом	НПАОП 0.00-1.28-10 [35] ДСанПіН 3.3.2.007-98 [34]
фізичні (статичне)	порушення умов праці та організації робочого часу	НПАОП 0.00-1.28-10 [35]

6.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ, вентиляції і кондиціонування. Небезпека загоряння пов'язана з особливістю комп'ютерів - із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Надійна

робота окремих елементів і мікросхем в цілому забезпечується тільки в певних інтервалах температури, вологості і при заданих електричних параметрах. При відхиленні реальних умов експлуатації від розрахункових можуть виникнути пожежонебезпечні ситуації.

Потенційними джерелами запалювання можуть бути:

- іскри і дуги короткого замикання;
- електрична іскра при замиканні і розмиканні ланцюгів;
- перегріву від тривалого перевантаження,
- відкритий вогонь і продукти горіння,
- наявність речовин, нагрітих вище за температуру самозаймання, розрядна статична електрика.

Причинами можливого загоряння і пожежі можуть бути:

- несправність електроустановки;
- конструктивні недоліки устаткування;
- коротке замикання в електричних мережах;
- запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки[40].

6.3.3 Електробезпека

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК, периферійних пристроїв і устаткування для обслуговування, виконана як окрема групова три- провідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення

(занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

6.4 Гігієнічні вимоги до параметрів виробничого середовища

6.4.1 Мікроклімат

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. В даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, то для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають нормам [40] і наведені в табл. 6.4.

Таблиця 6.4 – Норми мікроклімату робочої зони об'єкту

Період року	Категорія робіт	Температура С ⁰	Відносна вологість %	Швидкість руху повітря, м/с
Холодна	легка-1 а	22 - 24	40 – 60	0,1
Тепла	легка-1 а	23 - 25	40 – 60	0,1

6.4.2 Освітлення

Світло є природною умовою існування людини. Воно впливає на стан вищих психічних функцій і фізіологічні процеси в організмі. Хороше освітлення діє тонізуюче, створює гарний настрій, покращує протікання основних процесів вищої нервової діяльності.

У проекті, що розробляється, передбачається використовувати суміщене освітлення. У світлий час доби використовуватиметься природне освітлення приміщення через віконні отвори, в решту часу використовуватиметься штучне освітлення. Штучне освітлення створюється газорозрядними лампами.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2015 Природне і штучне освітлення [39]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє ДБН і для даного приміщення в світлий час доби достатньо природного освітлення.

Розрахунок освітлення.

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше $1/8$, в побутових – $1/10$:

$$S_b = \left(\frac{1}{5} \div \frac{1}{10} \right) \cdot S_n, \quad (6.1)$$

де S_b – площа віконних прорізів, m^2 ;

S_n – площа підлоги, m^2 .

$$S_n = a \cdot b = 5 \cdot 3 = 15 \text{ м}^2,$$

$$S = 1/10 \cdot 15 = 1,5 \text{ м}^2.$$

Приймаємо 1 вікно площею $S=1,5 \text{ м}^2$.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для

створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників n виробляється по формулі (6.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (6.2)$$

де E – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

S – освітлювана площа, m^2 ; $S = 15 m^2$;

Z – поправочний коефіцієнт світильника (1,1 для люмінесцентних ламп);

K – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

U – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. [41] – 0,575

M – число люмінесцентних ламп в світильнику – 3;

F – світловий потік лампи – 5200лм (для ЛБ-80-7).

Підставивши числові значення у формулу (6.2), отримуємо:

$$n = \frac{300 \times 15 \times 1.1 \times 1.5}{5200 \times 0.54 \times 3} = 0,8 \approx 1$$

Приймаємо освітлювальну установку, яка складається з 1-го світильника, які складаються з трьох люмінесцентних ламп загальною потужністю 80 Вт, напругою – 220 В.

6.4.3 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП.

6.5 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

1) Заходи безпеки під час експлуатації персонального комп'ютера та периферійних пристроїв передбачають:

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала 2/3 нормальної освітленості приміщення);
- облаштовуючи приміщення для роботи з ПК, потрібно передбачити припливно-витяжну вентиляцію або кондиціонування повітря:

2) Заходи безпеки під час експлуатації інших електричних приладів передбачають дотримання таких правил:

- постійно стежити за справним станом електромережі, розподільних щитків, вимикачів, штепсельних розеток, лампових патронів, а також мережевих кабелів живлення, за допомогою яких електроприлади під'єднують до електромережі;
- постійно стежити за справністю ізоляції електромережі та мережевих кабелів, не допускаючи їхньої експлуатації з пошкодженою ізоляцією;
- не тягнути за мережевий кабель, щоб витягти вилку з розетки;
- не закривати меблями, різноманітним інвентарем вимикачі, розетки;

- не підключати одночасно декілька потужних електропристроїв до однієї розетки, що може викликати надмірне нагрівання провідників, руйнування їхньої ізоляції, розплавлення і загоряння полімерних матеріалів;

- не залишати включені електроприлади без нагляду;

- не допускати потрапляння всередину електроприладів крізь вентиляційні отвори рідин або металевих предметів, а також не закривати їх та підтримувати в належній чистоті, щоб уникнути перегрівання та займання приладу;

не ставити на електроприлади матеріали, які можуть під дією теплоти, що виділяється, загорітися (канцелярські товари, сувенірну продукцію тощо).

6.6 Висновки до шостого розділу

В результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом написаному в кваліфікаційній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені розміри приміщення, значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.

ВИСНОВКИ

В даному проекті були розглянуті проблеми пов'язані з організацією процесу консультування, процедури отримання свідоцтва про реєстрацію/зняття з реєстрації, кількості відвідувань та ряду інших суміжних видів послуг.

Після впровадження розробленого програмного продукту робота в відділі надання адміністративних послуг стане ефективніше і швидше, тому показники витрат людських ресурсів значно знизяться. Економія була досягнута шляхом зниження трудомісткості процесів реєстрації, що є заслугою автоматизації всіх основних функцій, що здійснювались раніше вручну. Це добре видно по коефіцієнту зниження трудових витрат, що дорівнює приблизно 30%. такий відсоток зниження трудових витрат оцінюється, як достатній, що доводить ефективність впровадження ІКС.

Проводячи аналіз вищевказаного результату можна відзначити, що мета проекту була досягнута з високим ступенем реалізації, що дає підставу говорити про ефективність проекту в цілому. Для підтримки або збільшення показників ефективності проекту необхідно чітко дотримуватися регламент роботи з проектним комплексом.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. ЦНАП м. Лисичанська. URL: <http://adminposluga.lis.lurenet.ua/>
2. Ст. 33 Конституції України «Про свободу пересування та вільний вибір місця проживання в Україні» URL: <http://zakon2.rada.gov.ua/laws/show/254к/96-вр>
3. Постанова КМУ № 207 «Про затвердження Правил реєстрації місця проживання та Порядку передачі органами реєстрації інформації до Єдиного державного демографічного реєстру» (надалі – «Постанова №207»)
4. Ст. 7 Закону України «Про адміністративні послуги» URL: <http://zakon4.rada.gov.ua/laws/show/5203-17>
5. Закон України «Про захист персональних даних» URL: <http://zakon5.rada.gov.ua/laws/show/2297-17>
6. Закон України «Про правовий статус іноземців та осіб без громадянства» <http://zakon3.rada.gov.ua/laws/show/3773-17>
7. Закон України «Про військовий обов'язок та військову службу» URL: <http://zakon3.rada.gov.ua/laws/show/2232-12>
8. Сімейний кодекс України URL: <http://zakon3.rada.gov.ua/laws/show/2947-14>
9. Закон України «Про електронні документи та електронний документообіг» URL: <http://zakon3.rada.gov.ua/laws/show/851-15>
10. Постанова КМУ №1453 «Про затвердження Типового порядку здійснення електронного документообігу в органах виконавчої влади» URL: <http://zakon3.rada.gov.ua/laws/show/1453-2004-п>
11. Наказ МІОУ №1000/5 «Про затвердження Правил організації діловодства та архівного зберігання документів у державних органах, органах місцевого самоврядування, на підприємствах, в установах і організаціях» URL: <http://zakon3.rada.gov.ua/laws/show/z0736-15>

12. Наказ МІОУ №1886/5 «Про затвердження Порядку роботи з електронними документами у діловодстві та їх підготовки до передавання на архівне зберігання» URL: <http://zakon3.rada.gov.ua/laws/show/z1421-14>

13. Рішення Лисичанської міської адміністрації №9/149 «Про внесення змін і доповнень до Положення про відділ надання адміністративних послуг Лисичанської міської адміністрації» URL: http://lis.gov.ua/attachments/article/13746/RIGS_442_01_11_2016.doc

14. Борсук Н.А., Козєєва О.О. Порівняльний аналіз мов програмування Python і PHP. – Москва: МНЖ «Символ науки» ФДБОУ ВПО МДУ ім. Н.Е. Баумана, 2016. URL: <https://cyberleninka.ru/article/v/sravnitelnyy-analiz-yazykov-programirovaniya-python-i-php>

15. Лукін І.К. Порівняльний аналіз мов програмування для розробки веб-додатків. – Самара: МНПЖ «Теорія і практика сучасної науки» ПДУТІ, 2017. – 6 с.

16. Oleg. PHP vs Python vs Ruby для розробки веб-додатків: докладний порівняння. URL: <https://8d9.ru/php-vs-python-vs-ruby-dlya-razrabotki-veb-prilozhenij-podrobnoe-sravnenie>

17. Ковбурн П. Керівництво PHP. Мануал. URL: <http://php.net/manual/ru/index.php#index>

18. Хокінс С. Адміністрування вебсервера Apache і керівництво по електронній комерції. - Київ: Вільямс, 2001.

19. Харлашкін А.І., Ревунков Г.І. Порівняльний аналіз реляційних систем управління базами даних. – Москва: ЕЖ «Молодіжний науково-технічний вісник» ФДБОУ ВПО МДУ ім. Н.Е. Баумана, 2016. URL: <http://docplayer.ru/42646091-Sravnitelnyy-analiz-relyacionnyh-sistem-upravleniya-bazami-dannyh.html>

20. Oracle Database URL: https://ru.wikipedia.org/wiki/Oracle_Database

21. Калягіна Н.В. Інформаційні технології у організації виробництва. Порівняльний аналіз функцій СУБД, з якими працюють корпоративні інформаційні системи. URL: <https://cyberleninka.ru/article/v/sravnitelnyy-analiz->

[funktsiy-subd-s-kotorymi-rabotayut-sovremennye-korporativnye-informatsionnye-sistemy](#)

22. Бен-Ган І., Сарка Д., Талмейдж Р. Microsoft SQL Server 2012. Створення запитів. Навчальний курс Microsoft. - Москва: Російська Редакція, 2014.

23. Вільям Р. Станек. Microsoft SQL Server 2012. Довідник адміністратора. - Санкт-Петербург: БХВ- Петербург, 2014.

24. Стандарт вищої освіти України першого (бакалаврського) рівня ступеня «бакалавр» за галуззю знань 12 «Інформаційні технології» спеціальністю 122 «Комп'ютерні науки та інформаційні технології» // Видання Міністерства освіти і науки України. 2016. 25 с. URL: - <http://mon.gov.ua/activity/education/reforma-osviti/naukovo-metodichna-rada-ministerstva/proekti-standartiv-vishhoyi-osviti.html>

25. Стандарт вищої освіти України перший (бакалаврський) рівень. Галузь знань 12 Інформаційні технології. Спеціальність 123 Комп'ютерна інженерія // Видання Міністерства освіти і науки України. - 2016. – 14 с. URL: - <http://mon.gov.ua/activity/education/reforma-osviti/naukovo-metodichna-rada-ministerstva/proekti-standartiv-vishhoyi-osviti.html>

26. Стандарт вищої освіти України перший (бакалаврський) рівень. Галузь знань 12 Інформаційні технології. Спеціальність 125 // Видання Міністерства освіти і науки України. 2016. 14 с. URL: <http://mon.gov.ua/activity/education/reforma-osviti/naukovo-metodichna-rada-ministerstva/proekti-standartiv-vishhoyi-osviti.html>

27. Державний стандарт України. ДСТУ 3008-2015 «Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання») / [На заміну ДСТУ 3008-95; чинний від 2017-07-01].- Київ: ДП «УкрНДНЦ», 2016. 31 с. URL: http://www.knmu.kharkov.ua/attachments/3659_3008-2015.PDF

28. Державний стандарт України. ДСТУ 8302:2015 «Інформація та документація. Бібліографічне посилання. Загальні вимоги та правила складання» URL: <http://lib.npu.edu.ua/files/dstu-8302-2015.pdf>

29. Стандарти з інформації, бібліотечної і видавничої справи. URL: <http://wwwv.library.univ.kiev.ua/ukr/about/dstu.html>

30. ДСТУ ISO 5807:2016 Оброблення інформації. Символи та угоди щодо документації стосовно даних, програм та системних блок-схем, схем мережевих програм та схем системних ресурсів (ISO 5807:1985, IDT).

31. Закон України «Про охорону праці»

32. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень.

33. НАПБ Б.03.001-2004 Типових норм належності вогнегасників

34. ДСанПіН 3.3.2.007-98 Гігієнічні вимоги до організації роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

35. НПАОП 0.00-1.28-10 Правила охорони праці під час експлуатації електронно- обчислювальних машин

36. ГОСТ 12.1.030-81 ССБТ. Электробезопасность. Защитное заземление. Зануление

37. ГОСТ 13109-97 «Электрическая энергия. Совместимость технических средств электромагнитных. Нормы качества электроэнергоснабжения общего назначения»

38. ГОСТ 12.1.006-84 ССБТ. Электромагнитные поля радиочастот. Общие требования безопасности. Допустимые уровни на рабочих местах и требования к проведению контроля

39. ДБН В.2.5-28:2015 Природне і штучне освітлення

40. ГОСТ 12.1.044-89 ССБТ. Пожаровзрывоопасность веществ и материалов. Номенклатура показателей и методы их определения.

41. НАПБ Б.03.002-2007 Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою

42. НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці.

43. НПАОП Б.02.005-2003 Про інструктаж, спецнавчання з питань пожежної безпеки

44. НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці

Додаток А

Комп'ютерна презентація

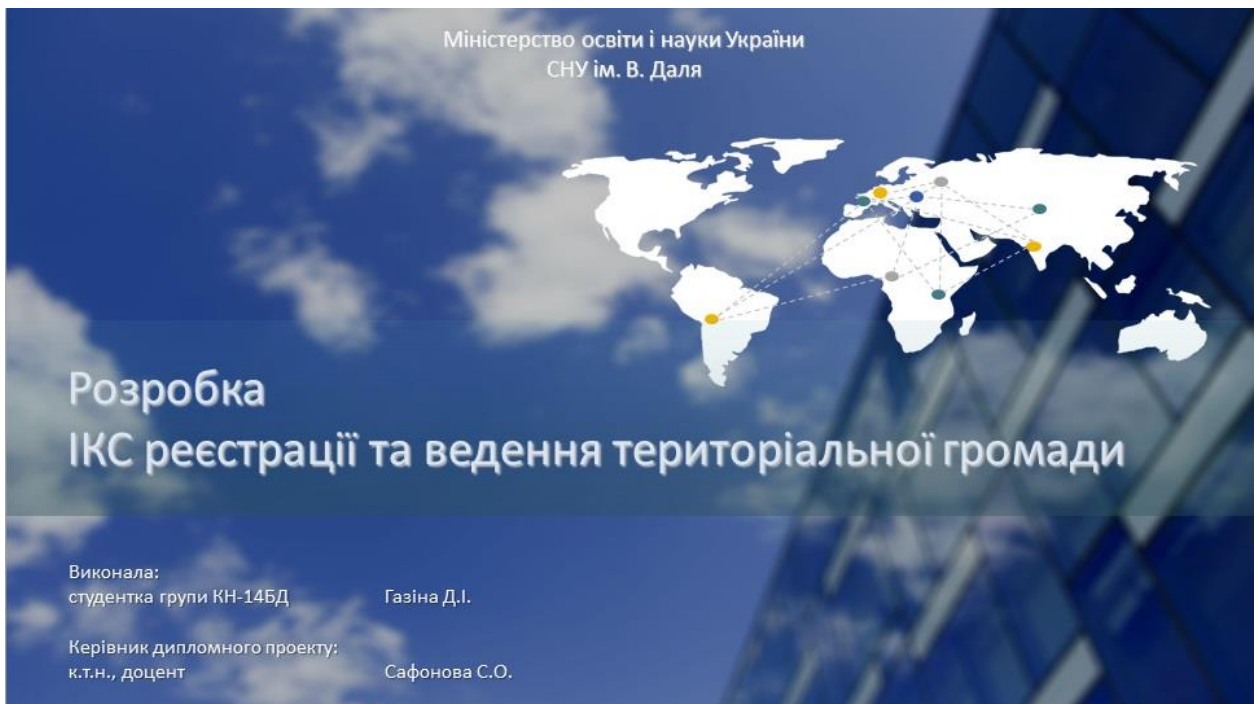


Рисунок А.1 – Слайд №1

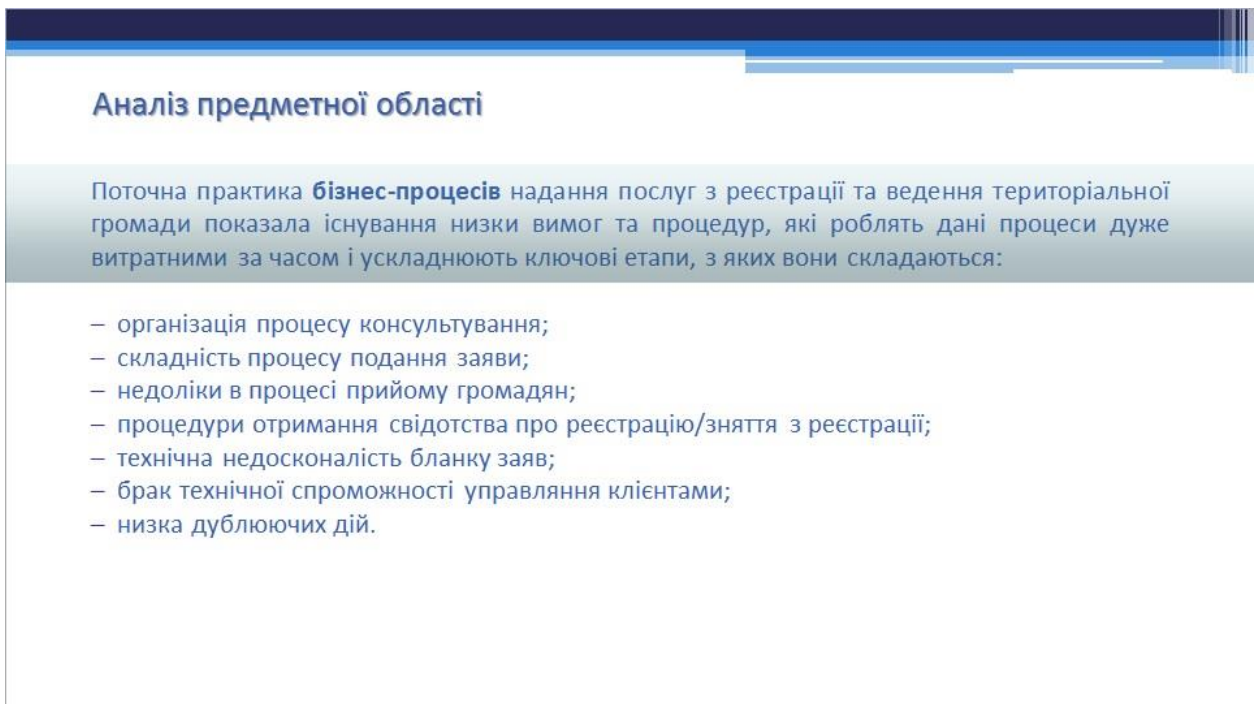


Рисунок А.1 – Слайд №2

Мета проекту

Метою проекту є розробка інформаційної комплексної системи формування і ведення реєстру територіальної громади в електронному вигляді, яка підвищить ефективність роботи та якість надання найбільш запитуваних громадянами послуг – **реєстрації/зняття з реєстрації фізичних осіб за місцем проживання**, через автоматизацію процесів:

- надання самої послуги;
- процедур обробки;
- паперової роботи;
- управління даними та інформацією стосовно вказаної послуги.

З урахуванням існуючих законодавчих рамок, що регулюють надання реєстраційних послуг проаналізовано і розроблено бізнес-процеси («як є», «як має бути»).

Рисунок А.1 – Слайд №3

Вимоги до розроблюваної системи

Автоматизована комплексна система реєстрації/зняття з реєстрації фізичних осіб за місцем проживання повинна вирішувати наступні завдання:

- реєстрація/зняття з реєстрації місця тимчасового проживання з відстеженням виданих довідок;
- можливість додавання документів з даними щодо реєстрації/зняття з реєстрації з описовими атрибутами;
- можливість генерування звіту «Склад сім'ї»;
- видача звітів до ДНС;
- видача звітів до ДРВ;
- видача звітів до військокомату;
- імпорт даних з існуючої бази даних до нової.

Рисунок А.1 – Слайд №4

Картографування бізнес-процесу реєстрації особи за місцем проживання. Статус «як є»

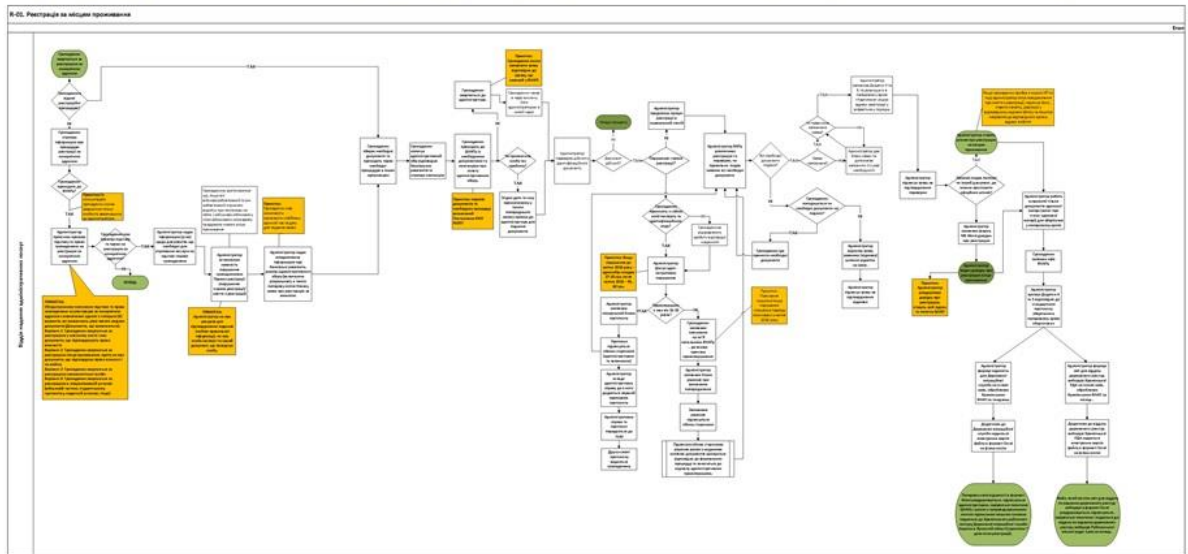


Рисунок А.1 – Слайд №5

Картографування бізнес-процесу зняття з реєстрації особи за місцем проживання. Статус «як є»

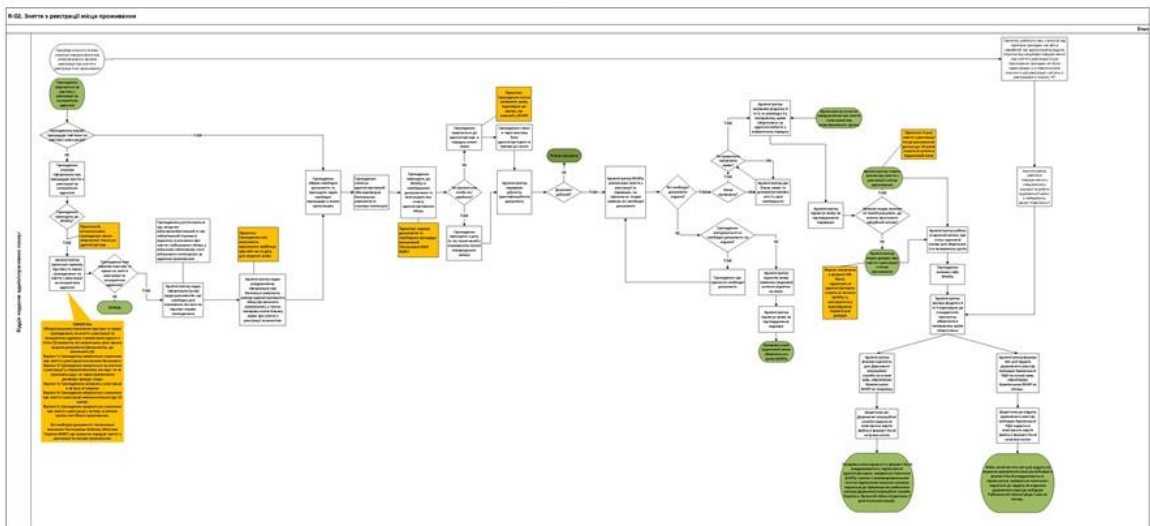


Рисунок А.1 – Слайд №6

Статус «як має бути» щодо процесу реєстрації фізичних осіб за місцем проживання

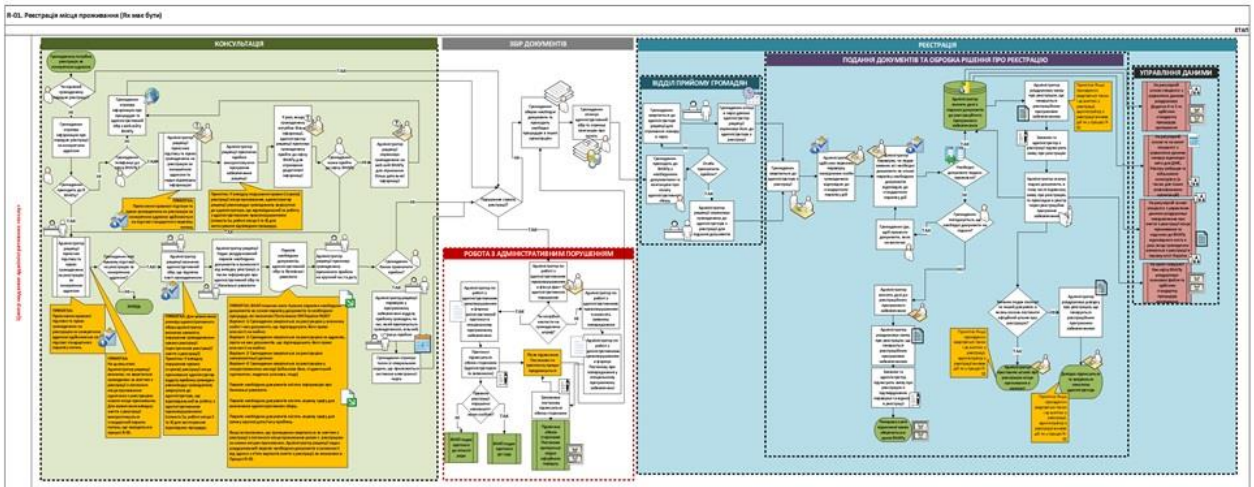


Рисунок А.1 – Слайд №7

Статус «як має бути» щодо процесу зняття з реєстрації фізичних осіб за місцем проживання

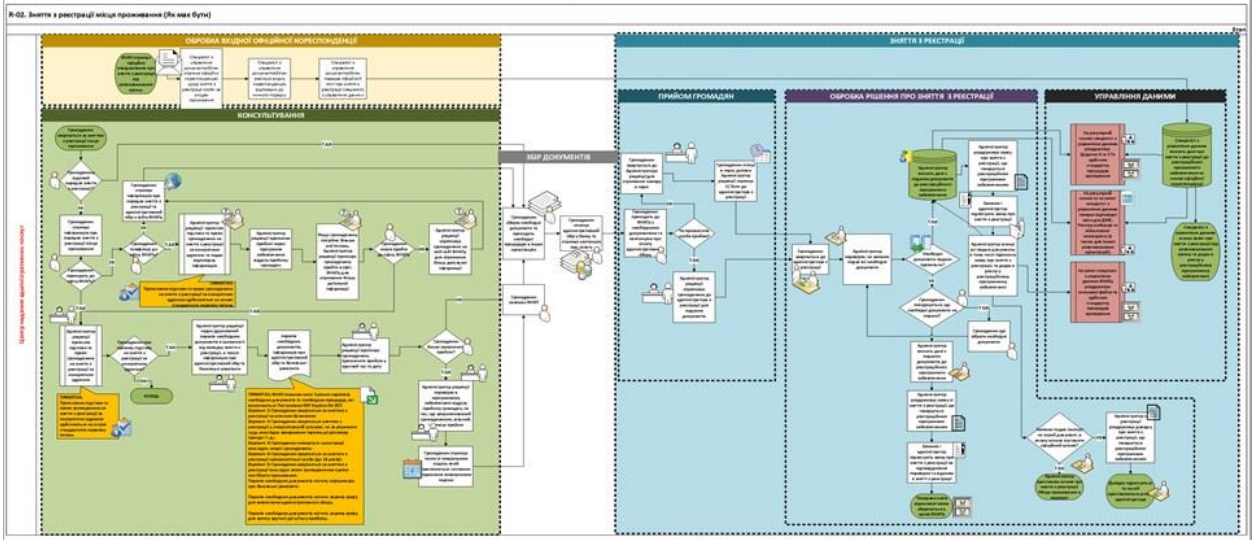


Рисунок А.1 – Слайд №8

Засоби розробки проекту

Вибір засобів розробки відбувався з урахуванням таких критеріїв, як зручність роботи, спроможність інтегрування з іншими продуктами, які використовуються для розробки системи. Була звернена увага на вартість програмного продукту і підтримку розробника. Для досягнення поставлених цілей були вибрані наступні засоби:

- мова програмування PHP;
- веб-сервер Apache;
- СУБД Microsoft SQL Server.

Рисунок А.1 – Слайд №9

Результати розробленого проекту

На основі аналізу бізнес-процесів «як є» та «як має бути» розроблений веб-додаток з повним набором функціоналу.

На початку роботи в додатку зареєстровано тільки одного користувача – Адміністратора Системи. Для коректного обліку дій, чіткого розмежування обов'язків відповідно до посади працівника кожному оператору надається окремий доступ до системи (логін і пароль).

Додаток має головне меню з якого можливий перехід в будь-який конкретний інформаційний розділ.

Рисунок А.1 – Слайд №10

Структура головного меню



Пошук адреси Пошук особи Реєстрації Звіти/Журнали Адміністратор Системи

- пошук адреси;
- пошук особи;
- пошук адреси;
- поточна реєстрація;
- звіти/журнали (вбудовані звіти, зовнішні звіти);
- довідники (типи документів, родинні стосунки, імена, по батькові);
- користувачі (список користувачів, управління ролями, дії користувачів);
- адміністратор системи.

Рисунок А.1 – Слайд №11

Сторінка введення даних для пошуку за адресою

The screenshot shows a web application interface for searching addresses. At the top, there is a navigation menu with the following items: "Пошук адреси", "Пошук особи", "Реєстрації", "Звіти/Журнали", and "Адміністратор Системи". The main search area contains several input fields: "вулиця" (street) with a dropdown menu showing "П. України" and "України_П.", "Будинок" (building) with the value "20", and "Квартира" (apartment) with a dropdown menu. There are also buttons for "Пошук" (Search) and "Нова карта" (New map). Below the search fields, there is a table with the following data:

Вулиця	Будинок/корпус/квартира	Зареєстровано	
вулиця П. України України_П.	Будинок 28, квартира 1	+	Карта
вулиця П. України України_П.	Будинок 28, квартира 2	+	Карта

Рисунок А.1 – Слайд №12

Пункт меню «Пошук адреси»

Пошук адреси Пошук особи Реєстрації Звітність

м.Лисичанськ, Луганська область

вулиця: Л. Українки будинок: 26 квартира: 2

Інформація по адресній картці [натиснути](#)

Реєстрації нової особи Реєстрації перебування нової особи Внесення даних про реєстрацію перебування Внесення даних про реєстрацію

Друк адресної картки Відомість зареєстрованих Склад сім'ї Додати на тримісяччю

Архівний номер адресної картки: АКА4128 Архівна інформація

Контроль

Зареєстровано Відомості про особу та дані про реєстрацію / квитка / скасування

Кудряков Віктор Олександрович 27.04.1974	м.Лисичанськ, Луганська область (4411800000)	Особова картка
паспорт ЕМ 620630 від 13.12.2000 виданий Лисичанським МВ УМВС України в Луганській області	26.11.1990 Зареєстрований вулиця Л. Українки, будинок 26, квартира 2, м.Лисичанськ	Реєстраційна картка Дані про реєстрацію
Кудряков Олександр Олександрович 07.11.1961	м.Лисичанськ, Луганська область (4411800000)	Особова картка
паспорт ЕМ 144691 від 25.06.2002 виданий ЛИСИЧАНСЬКИМ ГОВД	06.12.1968 Зареєстрований вулиця Л. Українки, будинок 26, квартира 2, м.Лисичанськ	Реєстраційна картка
свідчення про смерть І-БД 512635 від 12.12.2017 виданий Лисичанським МВ ДРАЦС ЄТЮУ у Луганській області Звітний з вулиця Л. Українки, будинок 26, квартира 2, м.Лисичанськ	12.12.2017 у зв'язку зі смертю віз 1467 від 12.12.2017р	Додати про смерть
Гурбань Спар Іванович 25.10.1911	с.Тичинька, Красноградський район, Харківська область (6323382002)	Особова картка
паспорт ЕМ 144690 від 25.06.2002 виданий ЛИСИЧАНСЬКИМ ГОВД	22.08.2000 Зареєстрований вулиця Л. Українки, будинок 26, квартира 2, м.Лисичанськ	Реєстраційна картка

Рисунок А.1 – Слайд №13

Вікно «Особова картка»

Пошук адреси Пошук особи Реєстрації Звітність

Прізвище: гав ім'я: дпр По батькові: Пошук

Дата народження: ДД.ММ.РРРР Паспорт (номер): Номер ІПН: Нова особова картка

Номер ЄДРП:

ПІВ	Дата	Остання реєстрація
Гавина Дар'я Ігорівна 03.08.1996 м.Лисичанськ, Луганська область	05.04.2007	вулиця Гев. Полтавська, будинок 222, квартира 24, м.Лисичанськ паспорт ЕМ 900209 від 09.10.2012 виданий ЛИСИЧАНСЬКИМ ГОВД

Вибрати Особова картка

Рисунок А.1 – Слайд №14

Вікно «Реєстраційна картка»

Рисунок А.1 – Слайд №15

Результати розроблюваного проекту

- демо версія через інтернет-браузер Google Chrome;
- керівництво користувача з детальним описом;
- готове рішення для впровадження.

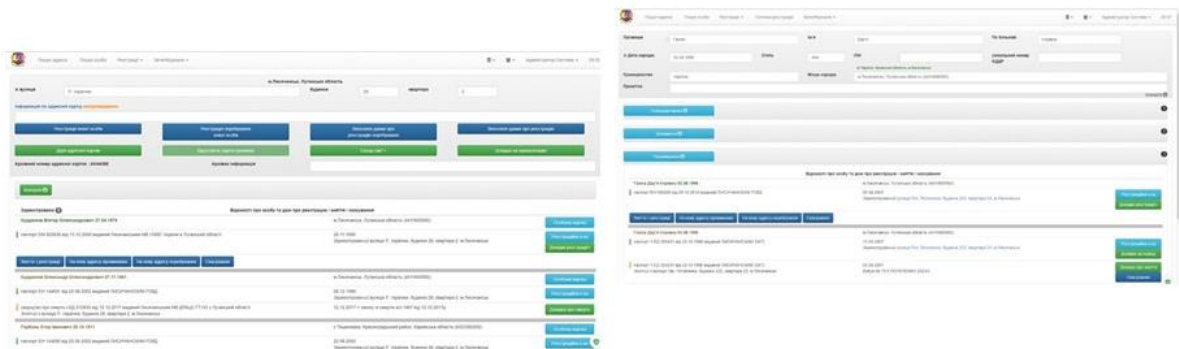
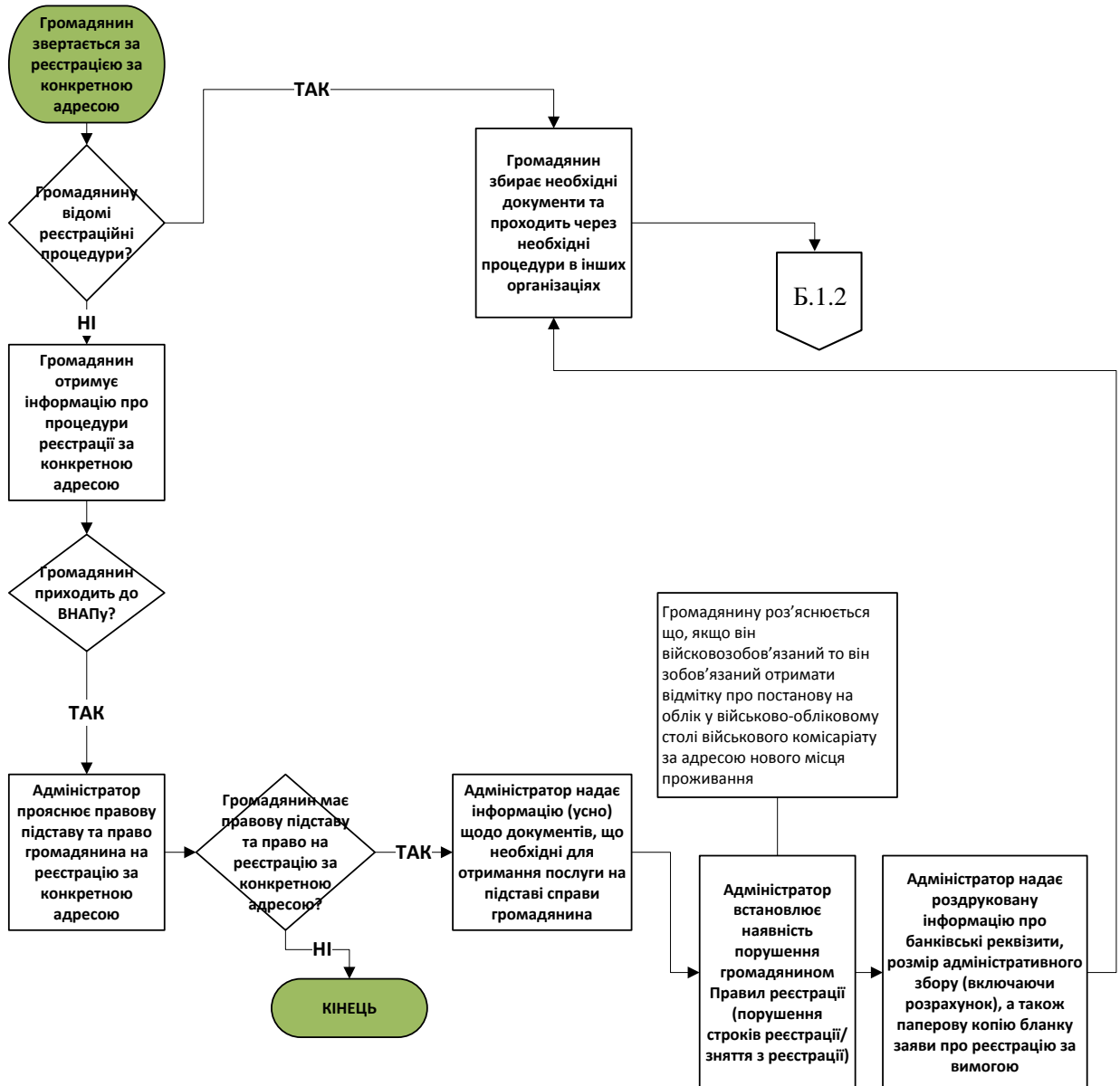


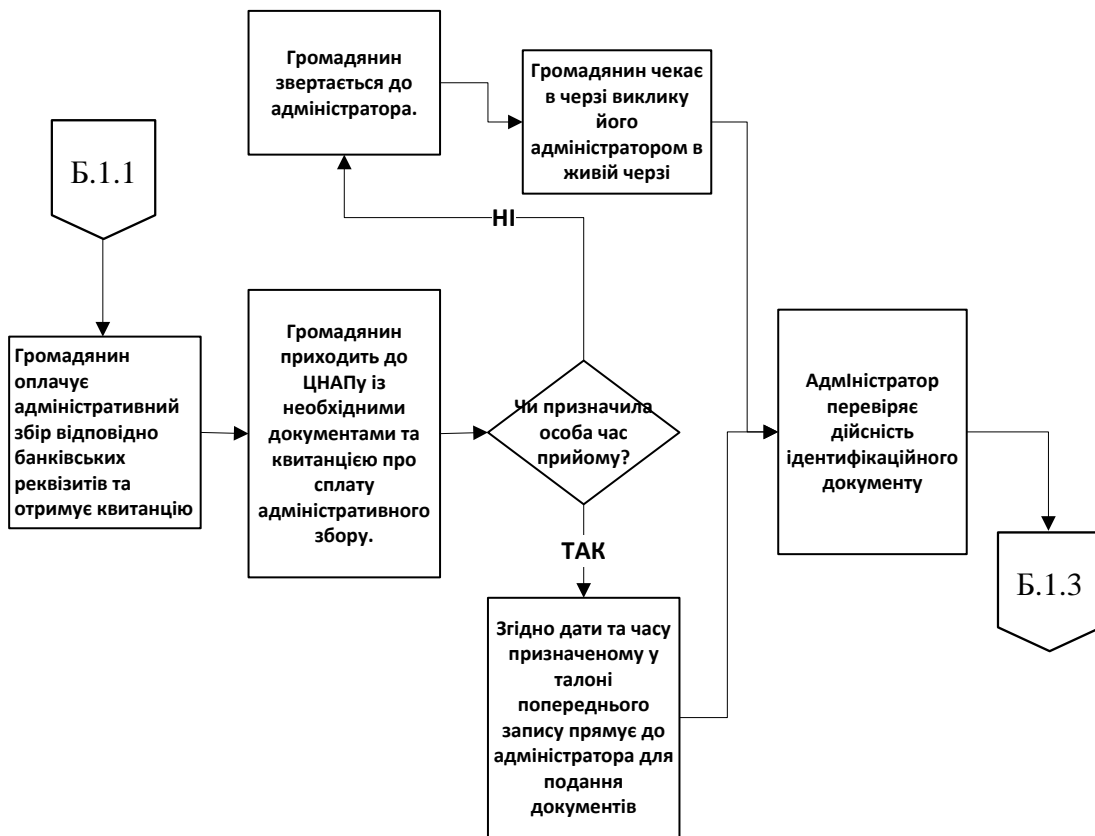
Рисунок А.1 – Слайд №16

Додаток Б

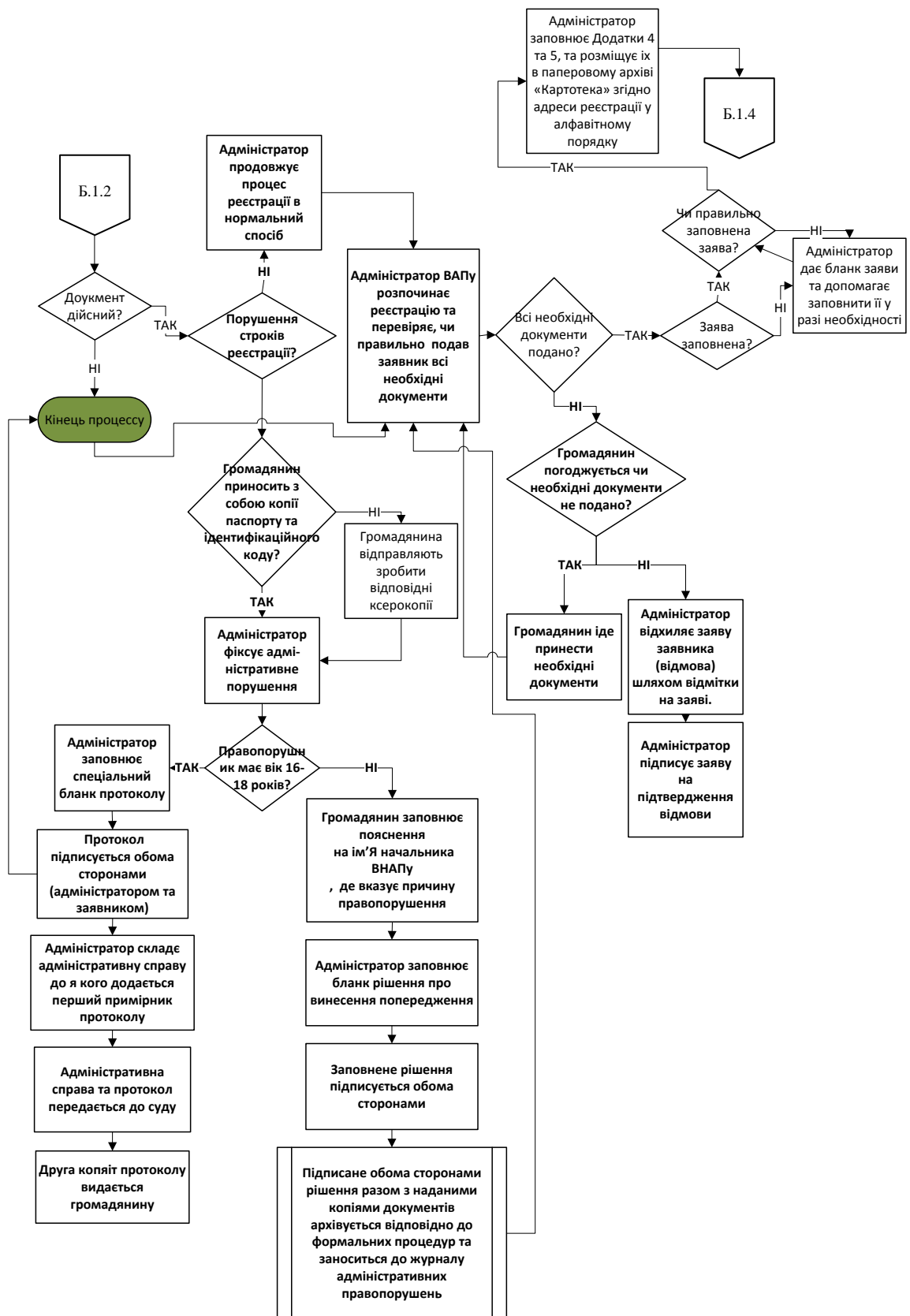
Бізнес-процес: реєстрація особи за місцем проживання. Статус «як є»

Є»

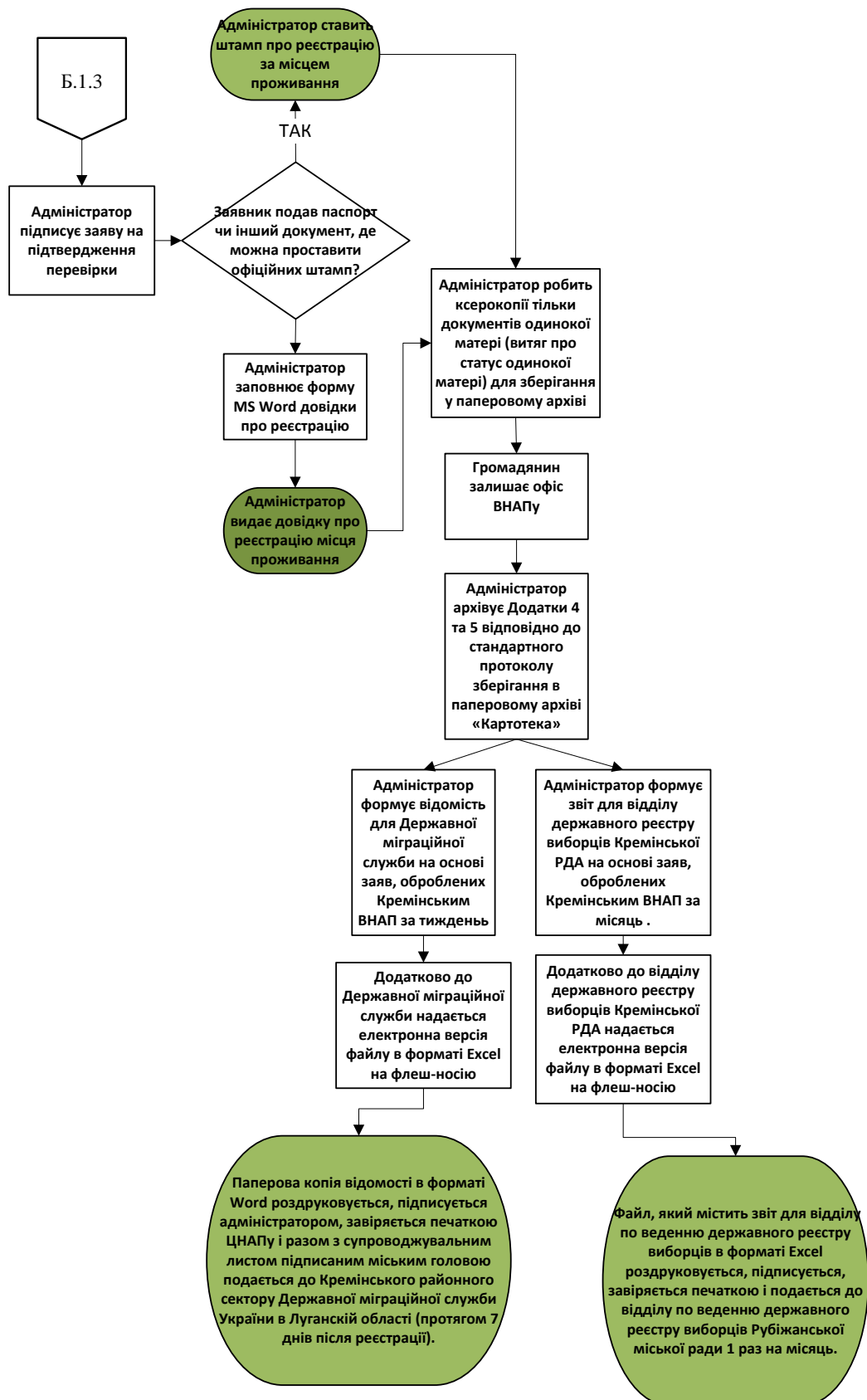




Продовження рисунку Б.1 – Бізнес-процес: реєстрація особи за місцем проживання. Статус «як є»



Продовження рисунку Б.1 – Бізнес-процес: реєстрація особи за місцем проживання. Статус «як є»

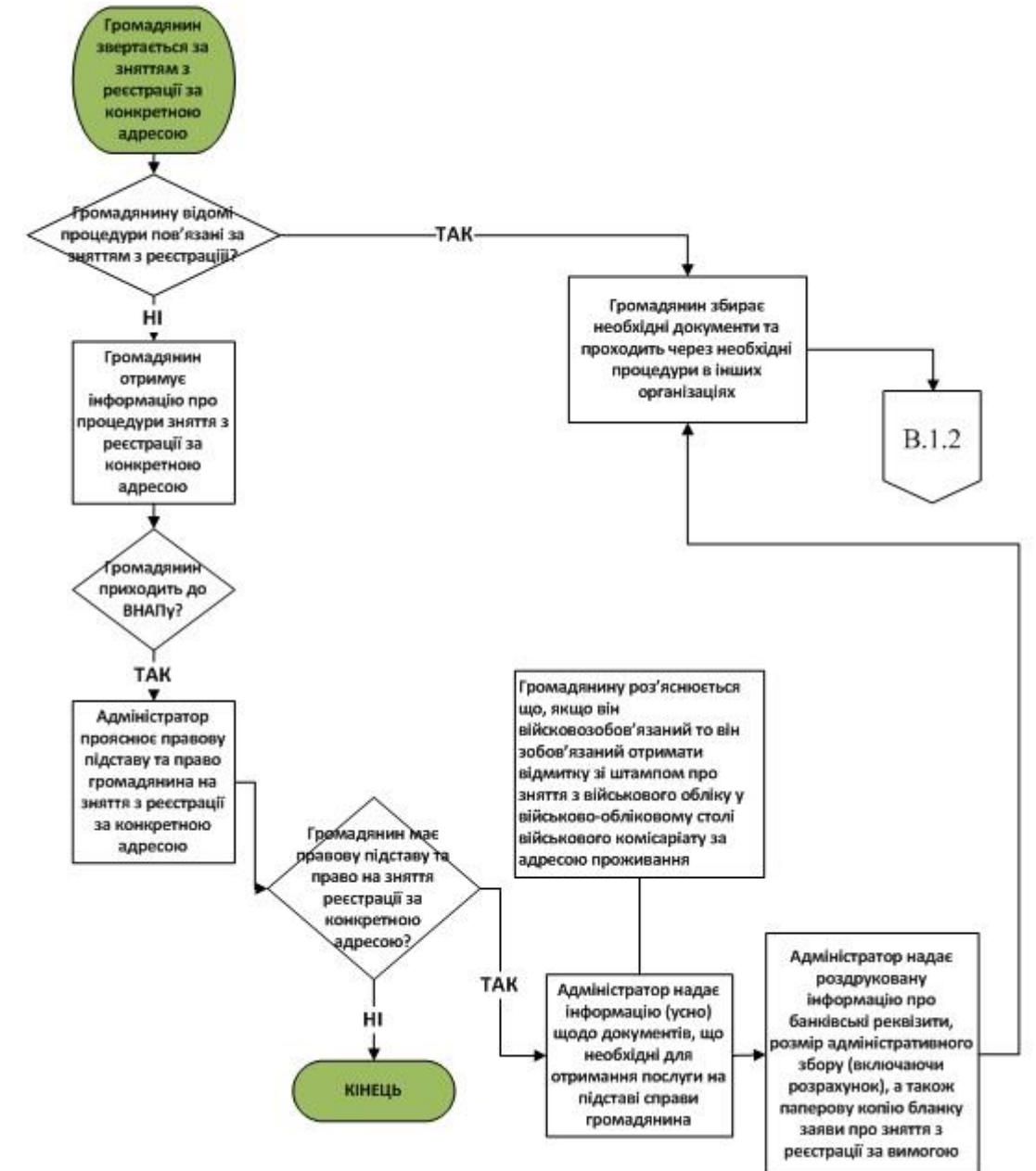


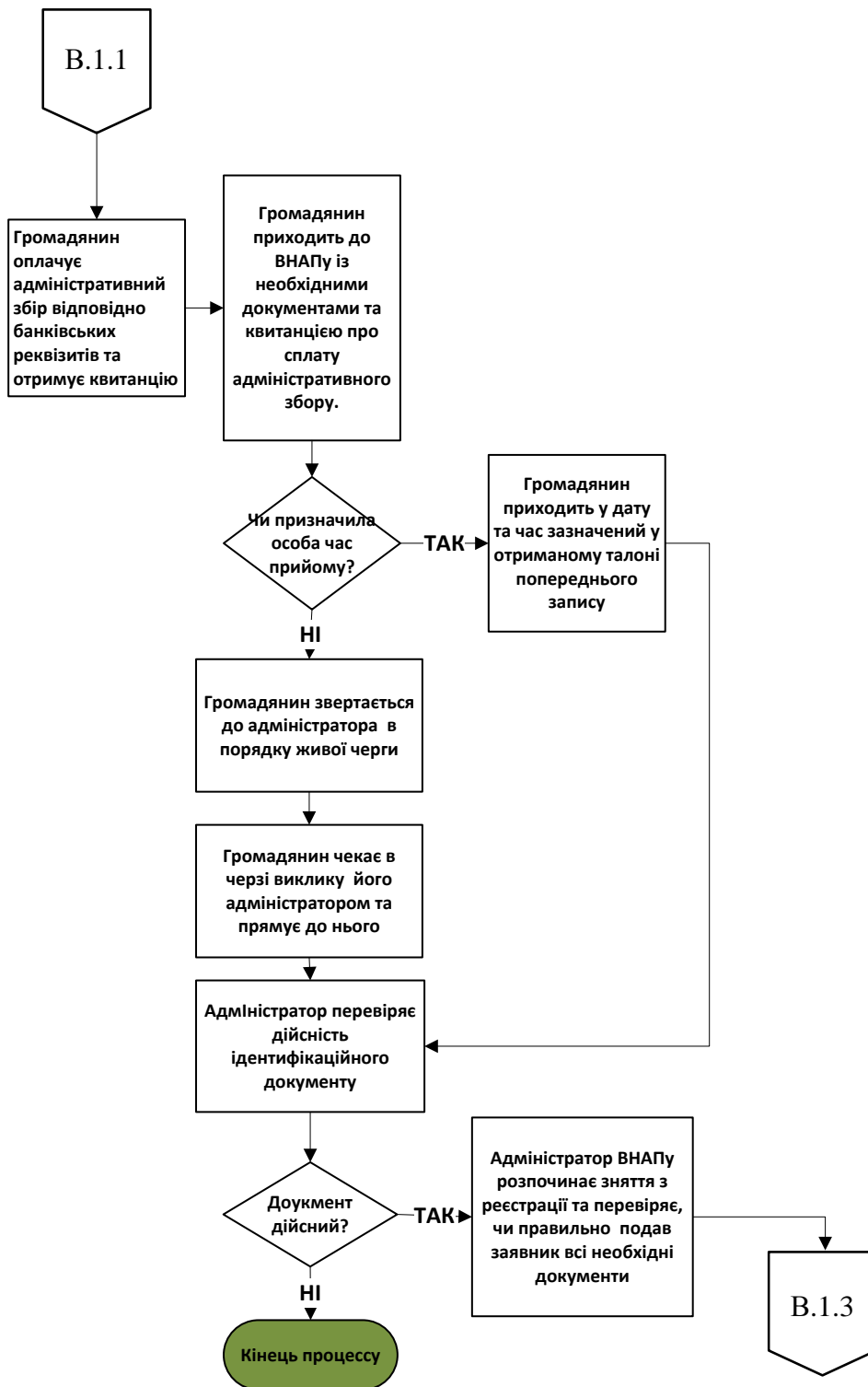
Продовження рисунку Б.1 – Бізнес-процес: реєстрація особи за місцем проживання. Статус «як є»

Додаток В

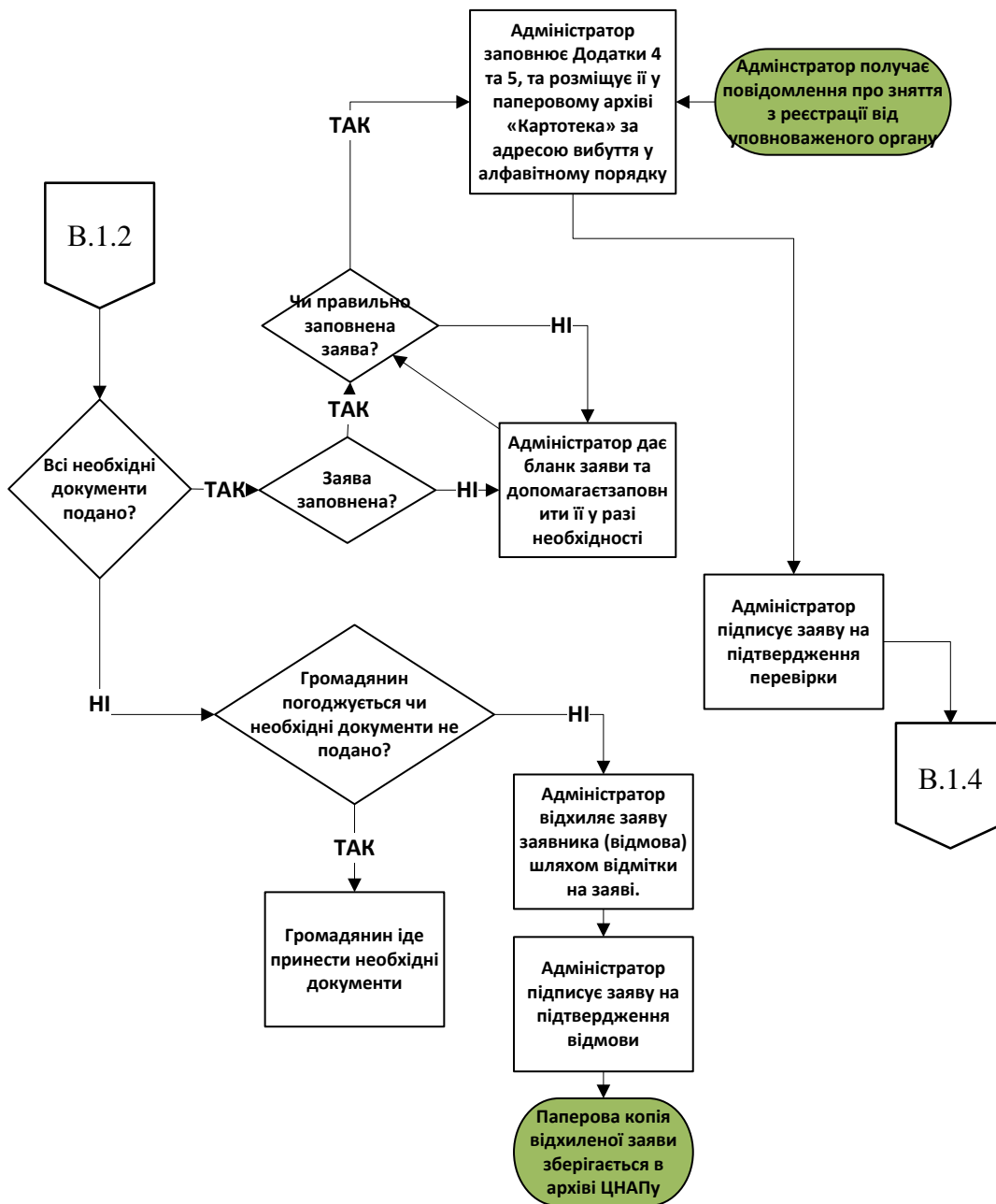
Бізнес-процес: зняття з реєстрації особи за місцем проживання.

Статус «як є»

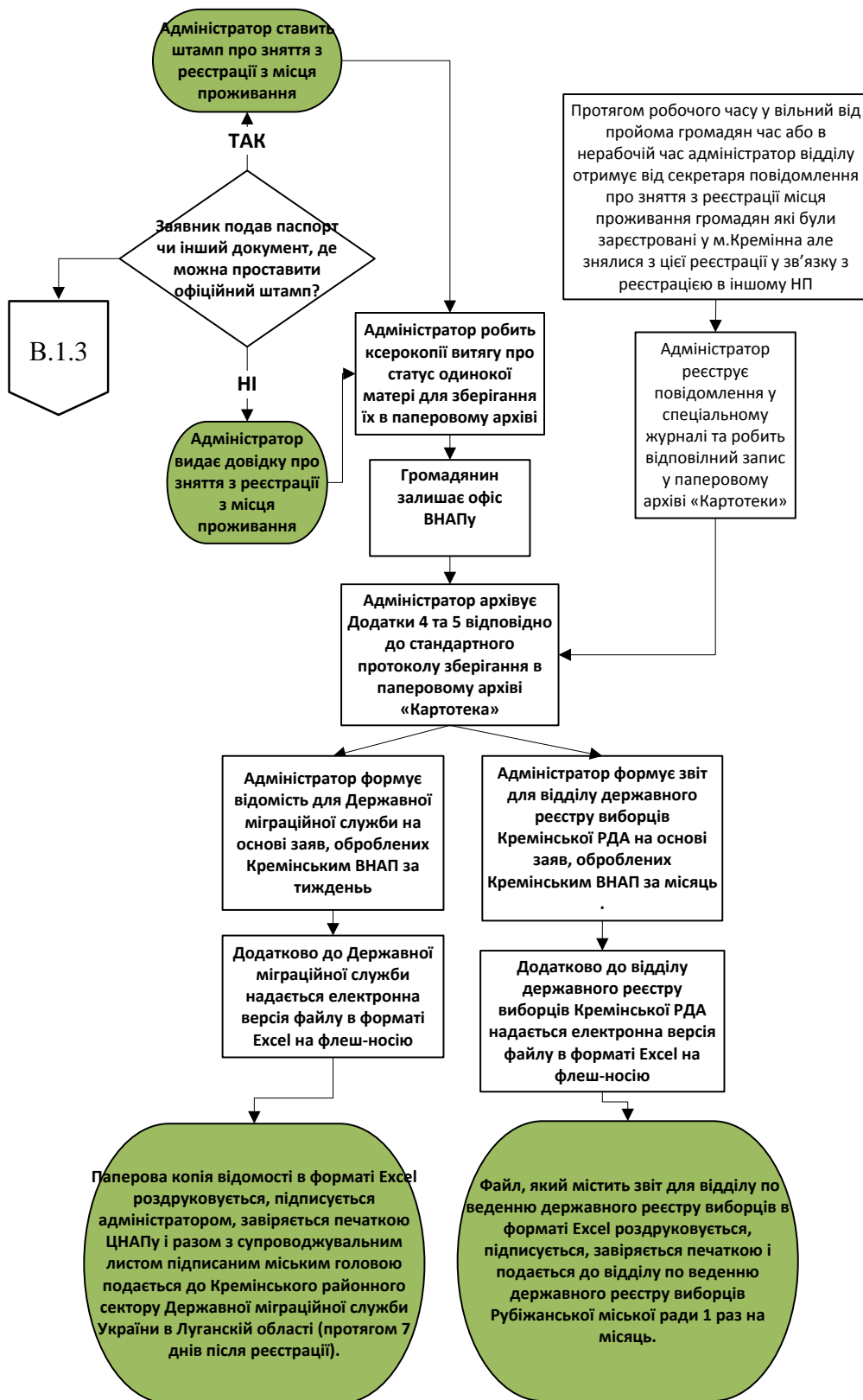




Продовження рисунку В.1 – Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як є»



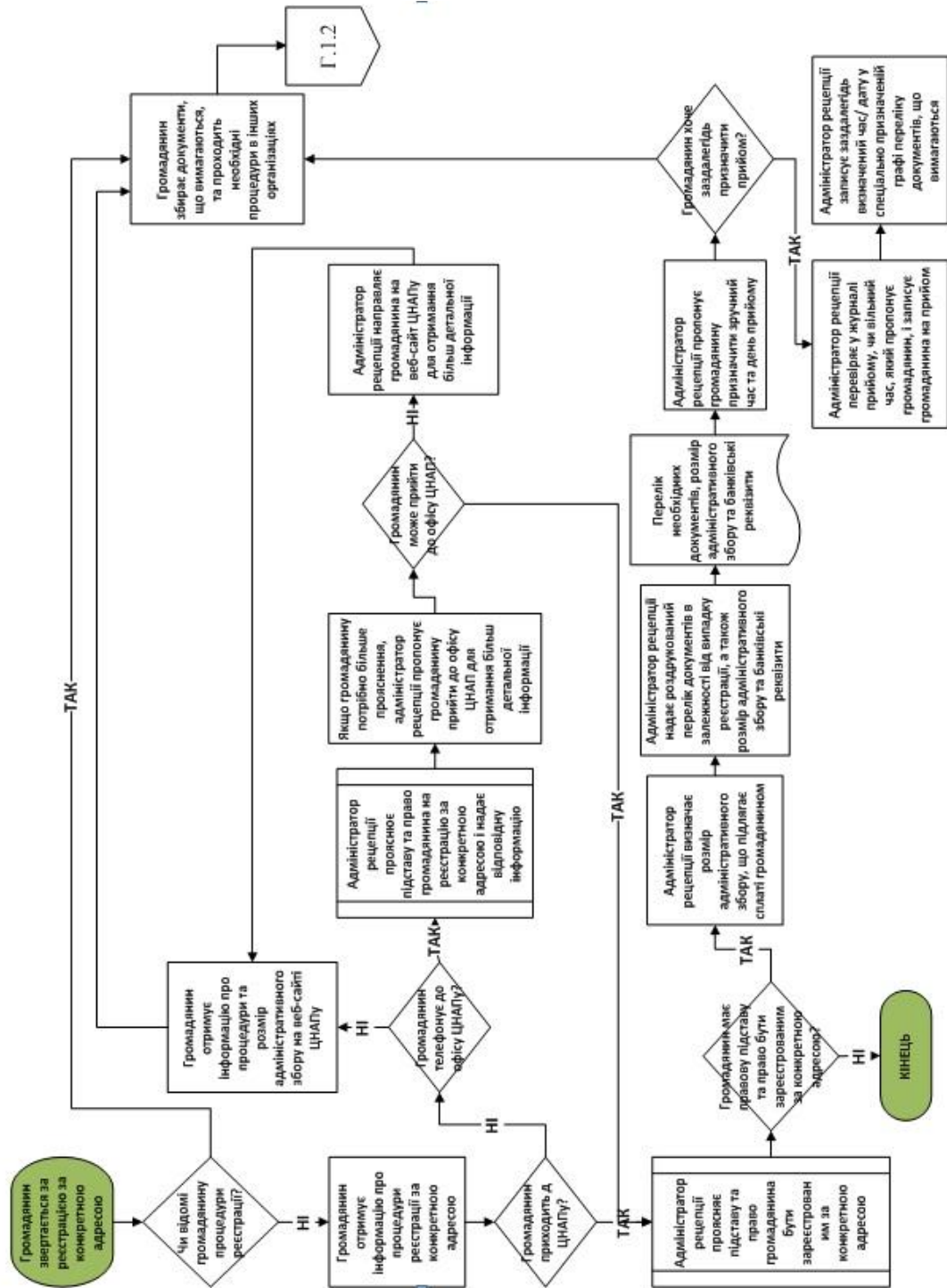
Продовження рисунку В.1 – Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як є»



Продовження рисунку В.1 – Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як є»

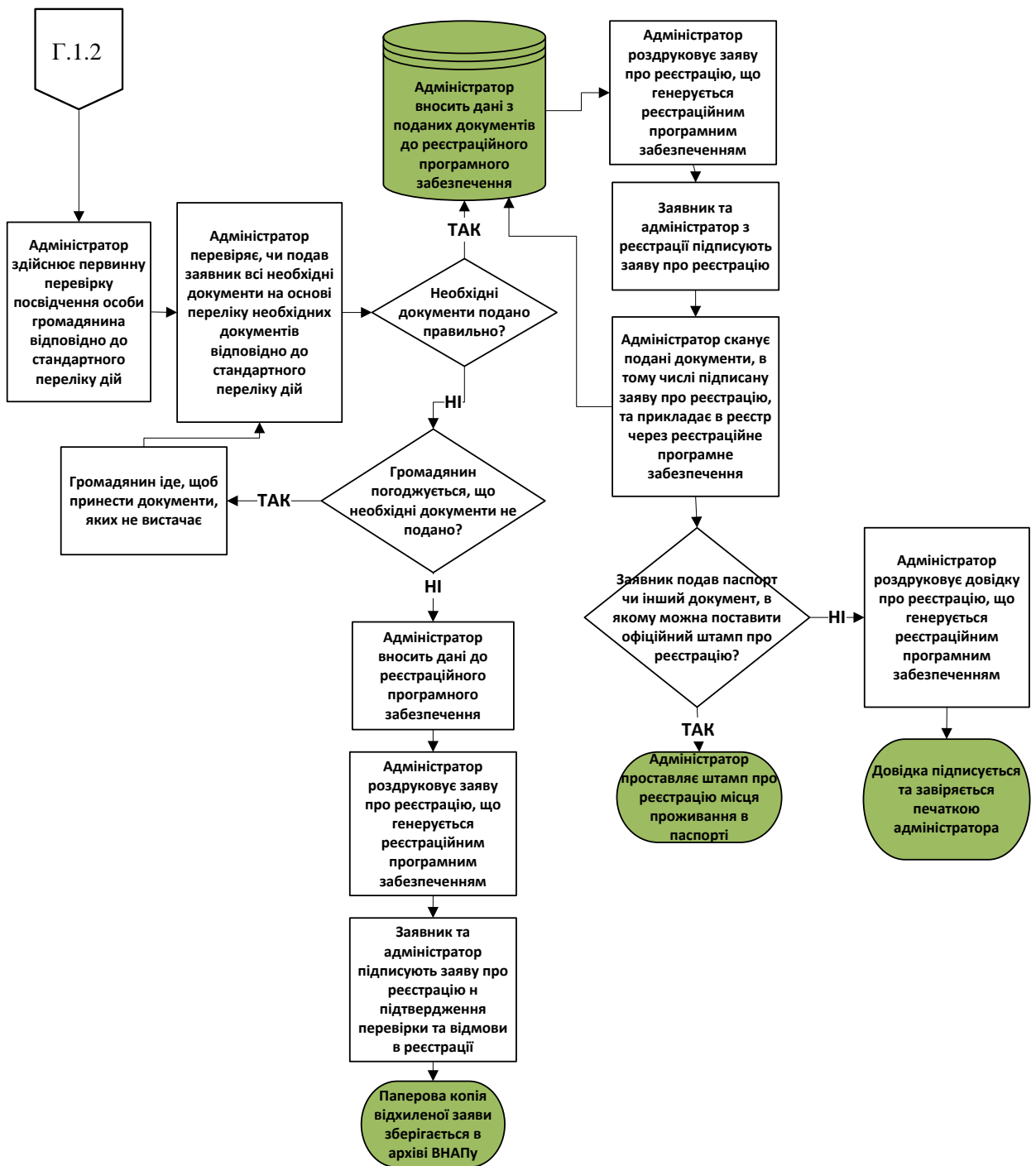
Додаток Г

Бізнес-процес: реєстрація за особи за місцем проживання. Статус «як має бути»

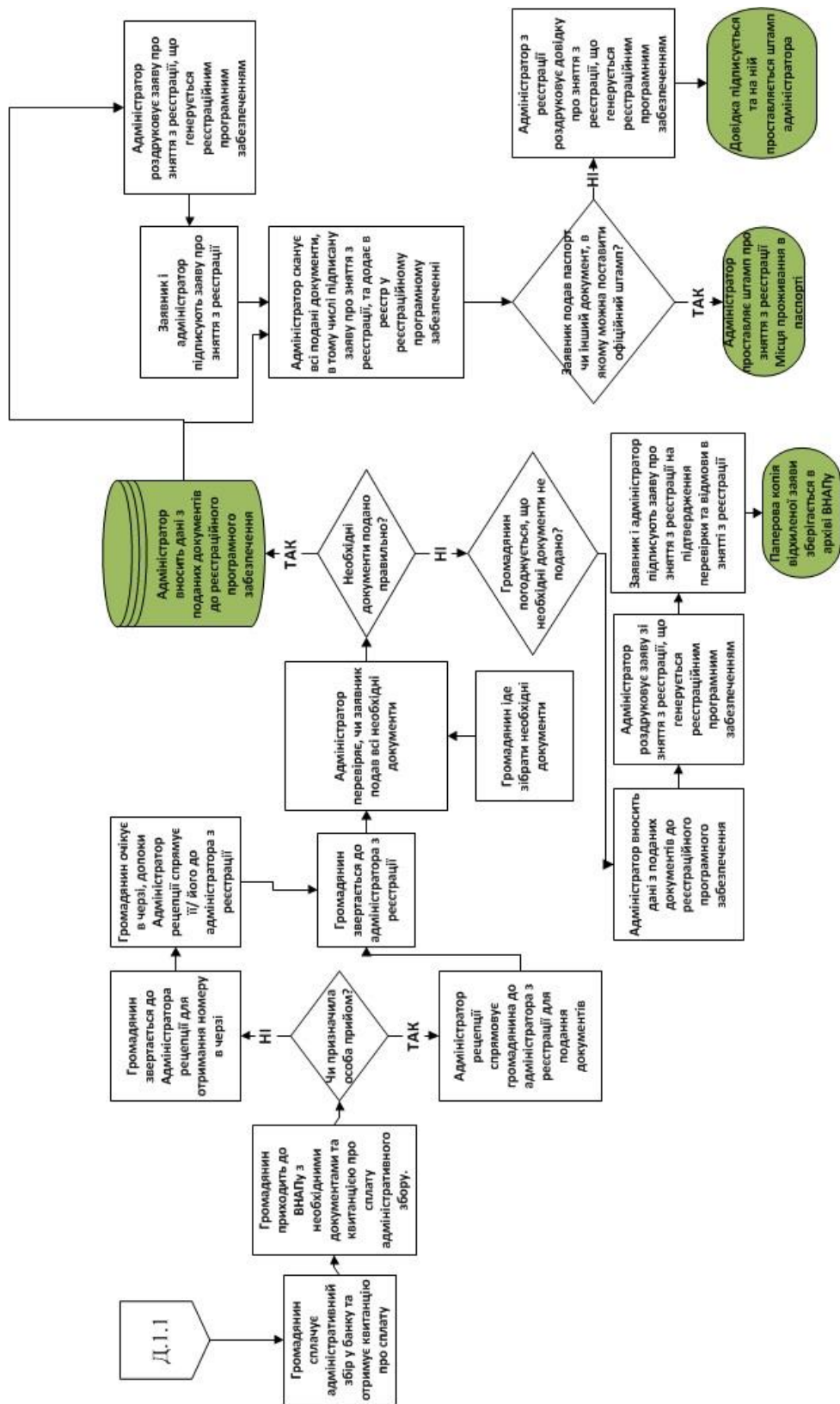




Продовження рисунку Г.1 – Бізнес-процес: реєстрація за особи за місцем проживання. Статус «як має бути»



Продовження рисунку Г.1 – Бізнес-процес: реєстрація за особи за місцем проживання. Статус «як має бути»



Продовження рисунку Д.1 – Бізнес-процес: зняття з реєстрації особи за місцем проживання. Статус «як має бути»

Додаток Е
Лістинг програми
appi.php

```
1 <?php
2 // IIT key support
3 Route::post('/api/proxy_handler',
4 ['as'=>'api.proxy_handler', 'middleware' => [],
5 'uses'=>'IIT@proxy_handler']);
6 Route::get('/api/proxy_handler', ['as'=>'api.proxy_handler',
7 'middleware' => [], 'uses'=>'IIT@proxy_handler']);
8 Route::get('/api/server_certificates',
9 ['as'=>'api.server_certificates', 'middleware' => [],
10 'uses'=>'IIT@server_certificates']);
11 // User support
12 Route::post('/api/user/allowed', ['as'=>'api.user.allowed',
13 'middleware' => [], 'uses'=>'User@allowed']);
14 Route::post('/api/user/unknown', ['as'=>'api.user.unknown',
15 'middleware' => [], 'uses'=>'User@unknown']);
16 // Updater
17 Route::post('/api/update', ['as'=>'api.updater',
18 'middleware' => [], 'uses'=>'Updater@server_side']);
```

CnapLoader.php

```
1 <?php
2 /**
3  * Created by PhpStorm.
4  * User: zserh
5  * Date: 17.10.2017
6  * Time: 13:30
7  */
8 namespace App\Http;
9 use Illuminate\Routing\Router;
10 use Illuminate\Support\Facades\View;
11 class CnapLoader
12 {
13     protected $paths = [];
14     protected $region = '';
15     public function add_region($region)
16     {
17         if (!$region) return;
18         $this->region = getRegionRoot($region);
19         $this->add_path($this->region);
```

```

20     }
21     public function add_profile($profile)
22     {
23         $path = app()-
24 >basePath().DIRECTORY_SEPARATOR.'profiles'.DIRECTORY_SEPARATOR.$
25 profile.DIRECTORY_SEPARATOR;
26         $this->add_path($path);
27         $path = $this-
28 >region.'profiles'.DIRECTORY_SEPARATOR.$profile.DIRECTORY_SEPARA
29 TOR;
30         $this->add_path($path);
31     }
32     public function add_profiles($profiles)
33     {
34         if (!$profiles) return;
35         $profiles = explode(',', $profiles);
36         foreach ($profiles as $profile) {
37             $profile = trim($profile);
38             if ($profile) $this->add_profile($profile);
39         }
40     }
41     protected function add_path($base_path)
42     {
43         if (!file_exists($base_path)) return;
44         if (key_exists($base_path, $this->paths)) return;
45
46         $this->paths = array_merge([$base_path], $this-
47 >paths);
48         $def =
49 array_merge([$base_path.'resources'.DIRECTORY_SEPARATOR.'views'
50 , config('view.paths')]);
51         config(['view.paths' => $def]);
52         View::setFinder(app('view.finder'));
53         $region_routes =
54 $base_path.'App'.DIRECTORY_SEPARATOR.'Http'.DIRECTORY_SEPARATOR.
55 'init.php';
56         if (file_exists($region_routes)) {
57             $router = app(Router::class);
58             $router->group(['namespace' =>
59 'App\\Http\\Controllers',],
60             function ($router) use ($region_routes) {
61                 require $region_routes;
62             });
63         }
64     }
65     public function getTemplateFile($file_name, $always =
66 false)
67     {
68         $file_part =
69 config('cnap.paper_forms_templates_root').DIRECTORY_SEPARATOR.$f
70 ile_name;
71         foreach ($this->paths as $path) {
72             $file = realpath($path.$file_part);

```

```

73         if ($file) return $file;
74     }
75     return $always ? $this->region.$file_part : null;
76 }
77 public function register()
78 {
79     spl_autoload_register(array($this, 'loadClass'),
80 true, true);
81 }
82 public function loadClass($class)
83 {
84     if (substr($class, 0, 4) != 'App\\') return false;
85     foreach ($this->paths as $path) {
86         $file = $path.$class.'.php';
87         if (file_exists($file)) {
88             includeFile($file);
89             return true;
90         }
91     }
92     return false;
93 }
94 protected $callbacks = [];
95 public function add_callback($type, \Closure $closure) {
96     if (!key_exists($type, $this->callbacks)) $this-
97 >callbacks[$type] = [];
98     $this->callbacks[$type][] = $closure;
99 }
100 public function process($type)
101 {
102     if (!key_exists($type, $this->callbacks)) return
103 null;
104     $callbacks = $this->callbacks[$type];
105     foreach ($callbacks as $closure) {
106         $result = $closure();
107         if ($result) return $result;
108     }
109     return null;
110 }
111 }
112 function includeFile($file)
113 {
114     include $file;
115 }

```

Kernel.php

```

1 <?php
2 namespace App\Http;
3 use Illuminate\Foundation\Http\Kernel as HttpKernel;

```

```

4 class Kernel extends HttpKernel
5 {
6     /**
7      * The application's global HTTP middleware stack.
8      *
9      * These middleware are run during every request to your
10 application.
11     *
12     * @var array
13     */
14     protected $middleware = [
15
16 \Illuminate\Foundation\Http\Middleware\CheckForMaintenanceMode::
17 class,
18     \App\Http\Middleware\AutoloaderByConfig::class,
19     \App\Http\Middleware\InputCleanUp::class,
20
21     \App\Http\Middleware\EncryptCookies::class,
22
23 \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
24     \Illuminate\Session\Middleware\StartSession::class,
25
26     \App\Http\Middleware\LoadUserProfiles::class,
27 //     \App\Http\Middleware\LogSQL::class,
28 ];
29
30     /**
31     * The application's route middleware groups.
32     *
33     * @var array
34     */
35     protected $middlewareGroups = [
36         'web_login' => [
37
38 \Illuminate\View\Middleware\ShareErrorsFromSession::class,
39     \App\Http\Middleware\VerifyCsrfToken::class,
40     ],
41
42     'web' => [
43         'web_login',
44         'auth',
45         'user_profile:init',
46     ],
47
48     'api' => [
49         'throttle:60,1',
50     ],
51
52     'api_token' => [
53         'auth:api',
54         'check_logged_user',
55         'user_profile',

```

```

56         ],
57     ];
58
59     /**
60      * The application's route middleware.
61      *
62      * These middleware may be assigned to groups or used
63 individually.
64      *
65      * @var array
66      */
67     protected $routeMiddleware = [
68         'auth' => \App\Http\Middleware\Authenticate::class,
69         'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
70         'can' => \Illuminate\Foundation\Http\Middleware\Authorize::class,
71         'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
72         'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
73         'role' => \Zizaco\Entrust\Middleware\EntrustRole::class,
74         'permission' => \Zizaco\Entrust\Middleware\EntrustPermission::class,
75         'ability' => \Zizaco\Entrust\Middleware\EntrustAbility::class,
76         'reflash' => \App\Http\Middleware\SessionReflash::class,
77         'user_profile' => \App\Http\Middleware\LoadUserProfiles::class,
78
79         'check_logged_user' => \App\Http\Middleware\CheckLoggedUser::class
80     ];
81
82     },];}

```

routes_ref.php

```

1     <?php
2     Route::get('ref/street_type',
3     ['as'=>'ref.street_type.index', 'middleware' => [],
4     'uses'=>'Refs\RefStreetTypeController@index']);
5     Route::get('ref/street_type/edit',
6     ['as'=>'ref.street_type.edit', 'middleware' => [],
7     'uses'=>'Refs\RefStreetTypeController@editor']);
8     Route::post('ref/street_type/edit',
9     ['as'=>'ref.street_type.edit', 'middleware' => [],
10    'uses'=>'Refs\RefStreetTypeController@update']);
11    Route::get('ref/street_type/delete',
12    ['as'=>'ref.street_type.delete', 'middleware' => [],
13    'uses'=>'Refs\RefStreetTypeController@deleter']);
14

```

```

15     Route::get('ref/street/{city}', ['as'=>'ref.street.index',
16 'middleware' => [],
17 'uses'=>'Refs\\RefStreetController@street_index']);
18     Route::get('ref/street/edit', ['as'=>'ref.street.edit',
19 'middleware' => [],
20 'uses'=>'Refs\\RefStreetController@editor']);
21     Route::post('ref/street/edit', ['as'=>'ref.street.edit',
22 'middleware' => [],
23 'uses'=>'Refs\\RefStreetController@update']);
24     Route::get('ref/street/delete', ['as'=>'ref.street.delete',
25 'middleware' => [],
26 'uses'=>'Refs\\RefStreetController@deleter']);
27
28     Route::get('ref/street/olds/{street_id}',
29 ['as'=>'ref.street.olds', 'middleware' => [],
30 'uses'=>'Refs\\RefStreetController@olds_editor']);
31     Route::post('ref/street/olds/{street_id}',
32 ['as'=>'ref.street.olds', 'middleware' => [],
33 'uses'=>'Refs\\RefStreetController@olds_update']);
34     Route::get('ref/street/olds/{street_id}/delete/{id}',
35 ['as'=>'ref.street.olds.delete', 'middleware' => [],
36 'uses'=>'Refs\\RefStreetController@olds_delete']);
37
38     Route::get('ref/doc_type', ['as'=>'ref.doc_type.index',
39 'middleware' => [],
40 'uses'=>'Refs\\RefDocTypeController@index']);
41     Route::get('ref/doc_type/edit', ['as'=>'ref.doc_type.edit',
42 'middleware' => [],
43 'uses'=>'Refs\\RefDocTypeController@editor']);
44     Route::post('ref/doc_type/edit', ['as'=>'ref.doc_type.edit',
45 'middleware' => [],
46 'uses'=>'Refs\\RefDocTypeController@update']);
47     Route::get('ref/doc_type/delete',
48 ['as'=>'ref.doc_type.delete', 'middleware' => [],
49 'uses'=>'Refs\\RefDocTypeController@deleter']);
50
51     Route::get('ref/doc_issuer', ['as'=>'ref.issuers.index',
52 'middleware' => [],
53 'uses'=>'Refs\\RefDocIssuerController@index']);
54     Route::get('ref/doc_issuer/edit', ['as'=>'ref.issuers.edit',
55 'middleware' => [],
56 'uses'=>'Refs\\RefDocIssuerController@editor']);
57     Route::post('ref/doc_issuer/edit',
58 ['as'=>'ref.issuers.edit', 'middleware' => [],
59 'uses'=>'Refs\\RefDocIssuerController@update']);
60     Route::get('ref/doc_issuer/delete',
61 ['as'=>'ref.issuers.delete', 'middleware' => [],
62 'uses'=>'Refs\\RefDocIssuerController@deleter']);
63
64     Route::get('ref/country', ['as'=>'ref.country.index',
65 'middleware' => [],
66 'uses'=>'Refs\\RefCountryController@index']);

```

```

67     Route::get('ref/country/edit',    ['as'=>'ref.country.edit',
68 'middleware'                        =>                                [],
69 'uses'=>'Refs\\RefCountryController@editor']);
70     Route::post('ref/country/edit',   ['as'=>'ref.country.edit',
71 'middleware'                        =>                                [],
72 'uses'=>'Refs\\RefCountryController@update']);
73     Route::get('ref/country/delete',
74 ['as'=>'ref.country.delete',        'middleware'      =>            [],
75 'uses'=>'Refs\\RefCountryController@deleter']);
76
77     Route::get('ref/building_template',
78 ['as'=>'ref.building_template.index', 'middleware'      =>            [],
79 'uses'=>'Refs\\RefBuildingController@index']);
80     Route::get('ref/building_template/edit',
81 ['as'=>'ref.building_template.edit',  'middleware'      =>            [],
82 'uses'=>'Refs\\RefBuildingController@editor']);
83     Route::post('ref/building_template/edit',
84 ['as'=>'ref.building_template.edit',  'middleware'      =>            [],
85 'uses'=>'Refs\\RefBuildingController@update']);
86     Route::get('ref/building_template/delete',
87 ['as'=>'ref.building_template.delete', 'middleware'      =>            [],
88 'uses'=>'Refs\\RefBuildingController@deleter']);
89
90     Route::get('ref/signs',           ['as'=>'ref.signs.index',
91 'middleware' => [], 'uses'=>'Refs\\RefSignController@index']);
92     Route::get('ref/signs/edit',      ['as'=>'ref.signs.edit',
93 'middleware' => [], 'uses'=>'Refs\\RefSignController@editor']);
94     Route::post('ref/signs/edit',     ['as'=>'ref.signs.edit',
95 'middleware' => [], 'uses'=>'Refs\\RefSignController@update']);
96     Route::get('ref/signs/delete',    ['as'=>'ref.signs.delete',
97 'middleware' => [], 'uses'=>'Refs\\RefSignController@deleter']);
98
99     Route::get('ref/relation',        ['as'=>'ref.relation.index',
100 'middleware'   =>                                [],
101 'uses'=>'Refs\\RefRelationController@index']);
102     Route::get('ref/relation/edit',   ['as'=>'ref.relation.edit',
103 'middleware'   =>                                [],
104 'uses'=>'Refs\\RefRelationController@editor']);
105     Route::post('ref/relation/edit',  ['as'=>'ref.relation.edit',
106 'middleware'   =>                                [],
107 'uses'=>'Refs\\RefRelationController@update']);
108     Route::get('ref/relation/delete',
109 ['as'=>'ref.relation.delete',        'middleware'      =>            [],
110 'uses'=>'Refs\\RefRelationController@deleter']);
111
112     Route::get('ref/mil_doc_type',
113 ['as'=>'ref.mil_doc_type.index',      'middleware'      =>            [],
114 'uses'=>'Refs\\RefMilDocTypeController@index']);
115     Route::get('ref/mil_doc_type/edit',
116 ['as'=>'ref.mil_doc_type.edit',      'middleware'      =>            [],
117 'uses'=>'Refs\\RefMilDocTypeController@editor']);

```



```

118     Route::post('ref/mil_doc_type/edit',
119 ['as'=>'ref.mil_doc_type.edit',      'middleware'    =>    [],
120 'uses'=>'Refs\\RefMilDocTypeController@update']);
121     Route::get('ref/mil_doc_type/delete',
122 ['as'=>'ref.mil_doc_type.delete',    'middleware'    =>    [],
123 'uses'=>'Refs\\RefMilDocTypeController@deleter']);
124
125     Route::get('ref/mil_issuer',  ['as'=>'ref.mil_issuer.index',
126 'middleware'                    =>                                [],
127 'uses'=>'Refs\\RefMilIssuerController@index']);
128     Route::get('ref/mil_issuer/edit',
129 ['as'=>'ref.mil_issuer.edit',      'middleware'    =>    [],
130 'uses'=>'Refs\\RefMilIssuerController@editor']);
131     Route::post('ref/mil_issuer/edit',
132 ['as'=>'ref.mil_issuer.edit',      'middleware'    =>    [],
133 'uses'=>'Refs\\RefMilIssuerController@update']);
134     Route::get('ref/mil_issuer/delete',
135 ['as'=>'ref.mil_issuer.delete',    'middleware'    =>    [],
136 'uses'=>'Refs\\RefMilIssuerController@deleter']);
137
138     Route::get('ref/mil_salutation',
139 ['as'=>'ref.mil_salutation.index',  'middleware'    =>    [],
140 'uses'=>'Refs\\RefMilSalutationController@index']);
141     Route::get('ref/mil_salutation/edit',
142 ['as'=>'ref.mil_salutation.edit',   'middleware'    =>    [],
143 'uses'=>'Refs\\RefMilSalutationController@editor']);
144     Route::post('ref/mil_salutation/edit',
145 ['as'=>'ref.mil_salutation.edit',   'middleware'    =>    [],
146 'uses'=>'Refs\\RefMilSalutationController@update']);
147     Route::get('ref/mil_salutation/delete',
148 ['as'=>'ref.mil_salutation.delete', 'middleware'    =>    [],
149 'uses'=>'Refs\\RefMilSalutationController@deleter']);
150
151     Route::get('ref/mil_specialization',
152 ['as'=>'ref.mil_specialization.index', 'middleware'    =>    [],
153 'uses'=>'Refs\\RefMilSpecializationController@index']);
154     Route::get('ref/mil_specialization/edit',
155 ['as'=>'ref.mil_specialization.edit', 'middleware'    =>    [],
156 'uses'=>'Refs\\RefMilSpecializationController@editor']);
157     Route::post('ref/mil_specialization/edit',
158 ['as'=>'ref.mil_specialization.edit', 'middleware'    =>    [],
159 'uses'=>'Refs\\RefMilSpecializationController@update']);
160     Route::get('ref/mil_specialization/delete',
161 ['as'=>'ref.mil_specialization.delete', 'middleware'    =>    [],
162 'uses'=>'Refs\\RefMilSpecializationController@deleter']);
163
164     Route::get('ref/adm',  ['as'=>'ref.adm.index',  'middleware'
165 => [], 'uses'=>'Refs\\RefAdmController@index']);
166     Route::get('ref/adm/edit',  ['as'=>'ref.adm.edit',
167 'middleware' => [], 'uses'=>'Refs\\RefAdmController@editor']);
168     Route::get('ref/adm/rename',  ['as'=>'ref.adm.rename',
169 'middleware' => [], 'uses'=>'Refs\\RefAdmController@renamer']);

```

```

170     Route::post('ref/adm/edit',          ['as'=>'ref.adm.edit',
171 'middleware' => [], 'uses'=>'Refs\\RefAdmController@update']);
172     Route::get('ref/adm/refresh',        ['as'=>'ref.adm.refresh',
173 'middleware'          => [],
174 'uses'=>'Refs\\RefAdmController@refresh_admin']);
175
176     Route::get('roles',                  ['middleware'          => [],
177 'as'=>'roles.index', 'uses' => 'Refs\\RolesController@index']);
178     Route::get('roles/edit',            ['middleware'          => [],
179 'as'=>'roles.edit', 'uses' => 'Refs\\RolesController@editor']);
180     Route::post('roles/edit',           ['middleware'          => [],
181 'as'=>'roles.edit', 'uses' => 'Refs\\RolesController@update']);
182     Route::get('roles/delete',          ['middleware'          => [],
183 'as'=>'roles.delete',          'uses'          =>
184 'Refs\\RolesController@deleter']);
185
186     Route::get('ref/name',               ['as'=>'ref.name.index', 'middleware'
187 => [], 'uses'=>'Refs\\RefNameController@index']);
188     Route::get('ref/name/edit',          ['as'=>'ref.name.edit',
189 'middleware' => [], 'uses'=>'Refs\\RefNameController@editor']);
190     Route::post('ref/name/edit',         ['as'=>'ref.name.edit',
191 'middleware' => [], 'uses'=>'Refs\\RefNameController@update']);
192     Route::get('ref/name/delete',        ['as'=>'ref.name.delete',
193 'middleware' => [], 'uses'=>'Refs\\RefNameController@deleter']);
194
195     Route::get('ref/patronymic',         ['as'=>'ref.patronymic.index',
196 'middleware'          => [],
197 'uses'=>'Refs\\RefPatronymicController@index']);
198     Route::get('ref/patronymic/edit',    ['as'=>'ref.patronymic.edit', 'middleware'
199 => [],
200 'uses'=>'Refs\\RefPatronymicController@editor']);
201     Route::post('ref/patronymic/edit',   ['as'=>'ref.patronymic.edit', 'middleware'
202 => [],
203 'uses'=>'Refs\\RefPatronymicController@update']);
204     Route::get('ref/patronymic/delete',  ['as'=>'ref.patronymic.delete', 'middleware'
205 => [],
206 'uses'=>'Refs\\RefPatronymicController@deleter']);

```

Web.php

```

1     <?php
2     /*
3     Application Routes
4     | Here is where you can register all of the routes for an
5     application.
6     | It's a breeze. Simply tell Laravel the URIs it should
7     respond to
8     | and give it the controller to call when that URI is
9     requested.
10    |

```

```

11     */
12     Route::get('/', ['as'=>'home', 'middleware' => [],
13 'uses'=>'HomeController@index']);
14     Route::post('/', ['as'=>'home', 'middleware' => [],
15 'uses'=>'HomeController@index']);
16
17     Route::get('/asearch', ['as'=>'addr.search', 'middleware' =>
18 [], 'uses'=>'AddrController@search_form']);
19     Route::post('/asearch', ['as'=>'addr.search', 'middleware'
20 => [], 'uses'=>'AddrController@do_search']);
21     Route::get('/acard/{id?}', ['as'=>'addr.card', 'middleware'
22 => [], 'uses'=>'AddrController@show']);
23     Route::post('/acard/{id?}', ['as'=>'addr.card.post',
24 'middleware' => [], 'uses'=>'AddrController@update']);
25     Route::post('/acard/{id}/control/add',
26 ['as'=>'addr.card.control.add', 'middleware' => [],
27 'uses'=>'AddrController@control_add']);
28     Route::post('/acard/{id}/control/close',
29 ['as'=>'addr.card.control.close', 'middleware' => [],
30 'uses'=>'AddrController@control_close']);
31     Route::get('/acard/{id}/control/{controlid}/delete',
32 ['as'=>'addr.card.control.delete', 'middleware' => [],
33 'uses'=>'AddrController@control_delete']);
34     Route::get('/acard/control/{controlid}/history',
35 ['as'=>'addr.card.control.history', 'middleware' => [],
36 'uses'=>'AddrController@control_history']);
37     Route::get('/acard/{id}/edit', ['as'=>'addr.card.edit',
38 'middleware' => [], 'uses'=>'AddrController@editor']);
39     Route::get('/acard/{id}/delete', ['as'=>'addr.card.delete',
40 'middleware' => [], 'uses'=>'AddrController@deleter']);
41     Route::get('/acard/{id}/printing', ['as'=>'addr.print',
42 'middleware' => [], 'uses'=>'AddrController@printing']);
43     Route::get('/acard/{id}/history', ['as'=>'addr.history',
44 'middleware' => [], 'uses'=>'AddrController@history']);
45     Route::get('/acard/{id}/toogle_incomplete',
46 ['as'=>'addr.card.toogle_incomplete', 'middleware' => [],
47 'uses'=>'AddrController@toogle_incomplete']);
48     Route::post('/acard/{id}/incomplete',
49 ['as'=>'addr.card.incomplete', 'middleware' => [],
50 'uses'=>'AddrController@incomplete_text']);
51     Route::get('/acard/{id}/toogle_privat',
52 ['as'=>'addr.card.toogle_privat', 'middleware' => [],
53 'uses'=>'AddrController@toogle_privat']);
54     Route::get('/acard/{id}/paper/active',
55 ['as'=>'addr.paper.active', 'middleware' => [],
56 'uses'=>'AddrController@paper_active']);
57     Route::post('/acard/{id}/paper/active',
58 ['as'=>'addr.paper.active', 'middleware' => [],
59 'uses'=>'AddrController@paper_active_reload']);
60     Route::get('/acard/{id}/paper/history',
61 ['as'=>'addr.paper.history', 'middleware' => [],
62 'uses'=>'AddrController@paper_history']);

```

```

63     Route::post('/acard/{id}/paper/history',
64 ['as'=>'addr.paper.history',      'middleware'      =>      [],
65 'uses'=>'AddrController@paper_history_reload']);
66     Route::post('/acard/{id}/archive',
67 ['as'=>'addr.card.archive',        'middleware'      =>      [],
68 'uses'=>'AddrController@update_archive']);
69     Route::get('/acard/{id}/paper/card',
70 ['as'=>'addr.paper.card',          'middleware'      =>      [],
71 'uses'=>'AddrController@paper_card_show']);
72     Route::post('/acard/{id}/paper/card',
73 ['as'=>'addr.paper.card',          'middleware'      =>      [],
74 'uses'=>'AddrController@paper_card_process']);
75
76     Route::get('/psearch', ['as'=>'people.search', 'middleware'
77 => [], 'uses'=>'PeopleController@search_form']);
78     Route::post('/psearch', ['as'=>'people.search', 'middleware'
79 => [], 'uses'=>'PeopleController@do_search']);
80     Route::get('/pscanner', ['as'=>'people.search.scanner',
81 'middleware'                =>                [],
82 'uses'=>'PeopleController@search_form_facade']);
83     Route::post('/pscanner', ['as'=>'people.search.scanner',
84 'middleware' => [], 'uses'=>'PeopleController@scanner_search']);
85     Route::get('/pcard/{id?}', ['as'=>'people.card',
86 'middleware' => [], 'uses'=>'PeopleController@show']);
87     Route::post('/pcard/{id?}', ['as'=>'people.card.post',
88 'middleware' => [], 'uses'=>'PeopleController@update']);
89     Route::get('/pcard/{id}/edit', ['as'=>'people.card.edit',
90 'middleware' => [], 'uses'=>'PeopleController@editor']);
91     Route::get('/pcard/{id}/delete',
92 ['as'=>'people.card.delete',      'middleware'      =>      [],
93 'uses'=>'PeopleController@deleter']);
94     Route::get('/pcard/{id}/name/{nameid}/edit',
95 ['as'=>'people.card.name.edit',   'middleware'      =>      [],
96 'uses'=>'PeopleController@edit_name']);
97     Route::post('/pcard/{id}/name/{nameid}/edit',
98 ['as'=>'people.card.name.edit',   'middleware'      =>      [],
99 'uses'=>'PeopleController@update_name']);
100    Route::get('/pcard/{id}/name/{nameid}/delete',
101 ['as'=>'people.card.name.delete', 'middleware'      =>      [],
102 'uses'=>'PeopleController@delete_name']);
103    Route::post('/pcard/{id}/death', ['as'=>'people.card.death',
104 'middleware' => [], 'uses'=>'PeopleController@add_death']);
105    Route::get('/pcard/{id}/doc/{docid}/edit',
106 ['as'=>'people.card.doc.edit',    'middleware'      =>      [],
107 'uses'=>'PeopleController@edit_doc']);
108    Route::post('/pcard/{id}/doc/{docid}/edit',
109 ['as'=>'people.card.doc.edit',    'middleware'      =>      [],
110 'uses'=>'PeopleController@update_doc']);
111    Route::get('/pcard/{id}/doc/{docid}/delete',
112 ['as'=>'people.card.doc.delete',  'middleware'      =>      [],
113 'uses'=>'PeopleController@delete_doc']);
114    Route::get('/pcard/{id}/history', ['as'=>'people.history',
115 'middleware' => [], 'uses'=>'PeopleController@history']);

```

```

116     Route::post('/pcard/{id}/notes', ['as'=>'people.card.notes',
117 'middleware' => [], 'uses'=>'PeopleController@update_notes']);
118
119     Route::get('/name/{id}/history',      ['as'=>'name.history',
120 'middleware' => [], 'uses'=>'PeopleController@name_history']);
121     Route::get('/doc/{id}/history',      ['as'=>'doc.history',
122 'middleware' => [], 'uses'=>'PeopleController@doc_history']);
123
124     Route::get('/rsearch', ['as'=>'reg.search', 'middleware' =>
125 [], 'uses'=>'RegController@search_form']);
126     Route::post('/rsearch', ['as'=>'reg.search', 'middleware' =>
127 [], 'uses'=>'RegController@do_search']);
128     Route::get('/reg/{id}', ['as'=>'reg.card', 'middleware' =>
129 [], 'uses'=>'RegController@show']);
130     Route::get('/reg/{id}/edit',      ['as'=>'reg.card.reg.edit',
131 'middleware' => [], 'uses'=>'RegController@reg_edit']);
132     Route::get('/reg/{id}/delete', ['as'=>'reg.card.reg.delete',
133 'middleware' => [], 'uses'=>'RegController@reg_delete']);
134     Route::get('/reg/{id}/history',
135 ['as'=>'reg.card.reg.history',      'middleware' => [],
136 'uses'=>'RegController@reg_history']);
137     Route::post('/reg/{id}/notes',      ['as'=>'reg.card.notes',
138 'middleware' => [], 'uses'=>'RegController@reg_update_notes']);
139     Route::post('/reg/{id}/regdate',      ['as'=>'reg.card.regdate',
140 'middleware' => [], 'uses'=>'RegController@reg_update_reg_date']);
141     Route::get('/reg/addr/{id}',      ['as'=>'reg.by_addr',
142 'middleware' => [], 'uses'=>'RegController@reg_by_addr']);
143     Route::get('/reg/people/{id}',      ['as'=>'reg.by_people',
144 'middleware' => [], 'uses'=>'RegController@reg_by_people']);
145     Route::get('/reg/add/people/{id}',      ['as'=>'reg.add.people',
146 'middleware' => [], 'uses'=>'RegController@reg_add_people']);
147     Route::get('/reg/add/addr/{id}',      ['as'=>'reg.add.addr',
148 'middleware' => [], 'uses'=>'RegController@reg_add_addr']);
149     Route::post('/reg', ['as'=>'reg.card.post', 'middleware' =>
150 [], 'uses'=>'RegController@update']);
151     Route::get('/reg', ['as'=>'reg.current', 'middleware' => [],
152 'uses'=>'RegController@show_current']);
153     Route::get('/reg/{id}/printing',      ['as'=>'reg.print',
154 'middleware' => [], 'uses'=>'RegController@printing']);
155     Route::get('/reg/{id}/printing/history',
156 ['as'=>'reg.print.history',      'middleware' => [],
157 'uses'=>'RegController@printing_history']);
158     Route::get('/reg/{id}/close',      ['as'=>'reg.close',
159 'middleware' => [], 'uses'=>'RegController@reg_close']);
160     Route::get('/reg/{id}/cancel',      ['as'=>'reg.cancel',
161 'middleware' => [], 'uses'=>'RegController@reg_cancel']);
162     Route::get('/reg/{regid}/move',
163 ['as'=>'reg.move.addr.start',      'middleware' => [],
164 'uses'=>'RegController@move_addr_start']);
165     Route::get('/reg/{regid}/move/addr/{addrid}',
166 ['as'=>'reg.move.addr.finish',      'middleware' => [],
167 'uses'=>'RegController@move_addr_finish']);

```

```

169     Route::get('/reg/{regid}/form/death',
170 ['as'=>'reg.form.death',      'middleware' => [],
171 'uses'=>'RegController@form_death']);
172     Route::get('/reg/regfiles/{file_id}/show',
173 ['as'=>'reg.files.reg.show',   'middleware' => [],
174 'uses'=>'RegController@reg_files_show']);
175     Route::get('/reg/unregfiles/{file_id}/show',
176 ['as'=>'reg.files.unreg.show', 'middleware' => [],
177 'uses'=>'RegController@unreg_files_show']);
178     Route::get('/reg/regfiles/{file_id}/preview',
179 ['as'=>'reg.files.reg.preview', 'middleware' => [],
180 'uses'=>'RegController@reg_files_preview']);
181     Route::get('/reg/unregfiles/{file_id}/preview',
182 ['as'=>'reg.files.unreg.preview', 'middleware' => [],
183 'uses'=>'RegController@unreg_files_preview']);
184     Route::post('/reg/{id}/archive', ['as'=>'reg.card.archive',
185 'middleware' => [],
186 'uses'=>'RegController@reg_update_archive']);
187
188     if (config('cnap.reg_virtual_enable')) {
189         Route::get('/reg/pending', ['as' => 'reg.pending',
190 'middleware' => [], 'uses' => 'RegController@pending_index']);
191         Route::post('/reg/pending/search', ['as' =>
192 'reg.pending.search', 'middleware' => [], 'uses' =>
193 'RegController@pending_search']);
194         Route::get('/reg/pending/reg/{id}', ['as' =>
195 'reg.pending.reg', 'middleware' => [], 'uses' =>
196 'RegController@pending_show_reg']);
197         Route::get('/reg/pending/unreg/{id}', ['as' =>
198 'reg.pending.unreg', 'middleware' => [], 'uses' =>
199 'RegController@pending_show_unreg']);
200         Route::get('/reg/pending/reg/{id}/delete', ['as' =>
201 'reg.pending.reg.delete', 'middleware' => [], 'uses' =>
202 'RegController@pending_delete_reg']);
203         Route::get('/reg/pending/unreg/{id}/delete', ['as' =>
204 'reg.pending.unreg.delete', 'middleware' => [], 'uses' =>
205 'RegController@pending_delete_unreg']);
206     }
207
208     Route::get('/unreg/{id}/history',
209 ['as'=>'reg.card.unreg.history', 'middleware' => [],
210 'uses'=>'RegController@unreg_history']);
211     Route::get('/unreg/{regid}/delete/{id}',
212 ['as'=>'reg.card.unreg.delete', 'middleware' => [],
213 'uses'=>'RegController@unreg_delete']);
214     Route::get('/unreg/{regid}/edit/{id}',
215 ['as'=>'reg.card.unreg.edit', 'middleware' => [],
216 'uses'=>'RegController@unreg_edit']);
217     Route::get('/unreg/{id}/cancel', ['as'=>'unreg.cancel',
218 'middleware' => [], 'uses'=>'RegController@unreg_cancel']);
219     Route::post('/unreg/{id}/archive',
220 ['as'=>'unreg.card.archive', 'middleware' => [],
221 'uses'=>'RegController@unreg_update_archive']);

```

```

222
223     Route::get('/mil/{id}/history',          ['as'=>'mil.history',
224 'middleware' => [], 'uses'=>'RegController@mil_history']);
225
226     Route::get('users',                    ['middleware' => [],
227 'as'=>'users.index', 'uses' => 'UsersController@index']);
228     Route::get('users/edit/{id?}',         ['middleware' => [],
229 'as'=>'users.edit', 'uses'=>'UsersController@show']);
230     Route::post('users/edit/{id?}',        ['middleware' => [],
231 'as'=>'users.update', 'uses'=>'UsersController@update']);
232     Route::get('users/justice/toogle',     ['middleware' => [],
233 'as'=>'users.justice', 'uses' => 'UsersController@justice']);
234     Route::get('users/notice/toogle',     ['middleware' => [],
235 'as'=>'users.notice', 'uses' => 'UsersController@notice']);
236     Route::get('users/report',             ['middleware' => [],
237 'as'=>'users.report', 'uses' => 'UserReportController@index']);
238     Route::post('users/report',           ['middleware' => [],
239 'as'=>'users.report.make',
240 'UserReportController@report']);
241     Route::get('users/activity',           ['middleware' => [],
242 'as'=>'users.activity',
243 'UserActivityController@index']);
244     Route::post('users/activity',         ['middleware' => [],
245 'as'=>'users.activity.make',
246 'UserActivityController@make']);
247     Route::get('users/{item_id}/bindings/edit', ['middleware' =>
248 [],
249 'as'=>'users.bindings.edit',
250 'uses'=>'BindedAddrController@user_bindings_shows']);
251     Route::post('users/{item_id}/bindings/edit', ['middleware'
252 => [],
253 'as'=>'users.bindings.update',
254 'uses'=>'BindedAddrController@user_bindings_update']);
255     Route::get('users/{item_id}/bindings/{id}/delete',
256 ['middleware' => [], 'as'=>'users.bindings.delete',
257 'uses'=>'BindedAddrController@user_bindings_delete']);
258     Route::get('users/impersonate/{id?}', ['middleware' => [],
259 'as'=>'users.impersonate',
260 'uses'=>'UsersController@impersonate']);
261
262     Route::get('groups',                   ['middleware' => [],
263 'as'=>'groups.index', 'uses' => 'GroupsController@index']);
264     Route::get('groups/edit/{id?}',        ['middleware' => [],
265 'as'=>'groups.edit', 'uses'=>'GroupsController@show']);
266     Route::post('groups/edit/{id?}',       ['middleware' => [],
267 'as'=>'groups.update', 'uses'=>'GroupsController@update']);
268     Route::get('groups/{item_id}/bindings/edit', ['middleware'
269 => [],
270 'as'=>'groups.bindings.edit',
271 'uses'=>'BindedAddrController@group_bindings_shows']);
272     Route::post('groups/{item_id}/bindings/edit', ['middleware'
273 => [],
274 'as'=>'groups.bindings.update',
275 'uses'=>'BindedAddrController@group_bindings_update']);
276     Route::get('groups/{item_id}/bindings/{id}/delete',
277 ['middleware' => [], 'as'=>'groups.bindings.delete',
278 'uses'=>'BindedAddrController@group_bindings_delete']);

```

```

275
276     Route::get('login/failed',      ['middleware' => [],
277 'as'=>'auth.failed.index',        'uses' =>
278 'FailedLoginsController@index']);
279     Route::post('login/failed',     ['middleware' => [],
280 'as'=>'auth.failed.index',        'uses' =>
281 'FailedLoginsController@at_date']);
282     Route::get('login/failed/pairing/{id}/{uid}', ['middleware'
283 => [], 'as'=>'auth.failed.pairing', 'uses' =>
284 'FailedLoginsController@pairing']);
285
286     Route::get('report', ['as'=>'report.index', 'middleware' =>
287 [], 'uses' => 'ReportController@index']);
288     Route::post('report/dms', ['as'=>'report.dms', 'middleware'
289 => [], 'uses' => 'ReportController@dms_maker']);
290     Route::post('report/drv', ['as'=>'report.drv', 'middleware'
291 => [], 'uses' => 'ReportController@drv_maker']);
292     Route::post('report/stat',      ['as'=>'report.stat',
293 'middleware' => [], 'uses' => 'ReportController@stat_maker']);
294     Route::post('report/csv', ['as'=>'report.csv', 'middleware'
295 => [], 'uses'=>'ReportController@csv_maker']);
296     Route::post('report/mil', ['as'=>'report.mil', 'middleware'
297 => [], 'uses'=>'ReportController@mil_maker']);
298     Route::post('/report/family-members',
299 ['as'=>'report.family', 'middleware' => [],
300 'uses'=>'FamilyMembersReportController@formSubmit']);
301     Route::post('/report/reg_place', ['as'=>'report.reg_place',
302 'middleware' => [],
303 'uses'=>'ReportController@reg_place_maker']);
304
305     Route::get('jurnals', ['as'=>'jurnals.report', 'middleware'
306 => [], 'uses'=>'JurnalsController@report_index']);
307     Route::post('jurnals', ['as'=>'jurnals.report', 'middleware'
308 => [], 'uses'=>'JurnalsController@report_make']);
309     Route::get('jurnals/addr/req',   ['as'=>'jurnals.addr.req',
310 'middleware' => [],
311 'uses'=>'JurnalsController@addr_req_index']);
312     Route::post('jurnals/addr/req', ['as'=>'jurnals.addr.req',
313 'middleware' => [], 'uses'=>'JurnalsController@addr_req_list']);
314     Route::post('jurnals/addr/paper',
315 ['as'=>'jurnals.addr.paper', 'middleware' => [],
316 'uses'=>'JurnalsController@addr_paper_make']);
317
318     Route::get('ext_report',         ['as'=>'report.external',
319 'middleware' => [], 'uses' => 'ExtReportController@index']);
320     Route::get('ext_report/{id}/delete',
321 ['as'=>'report.external.remove', 'middleware' => [], 'uses' =>
322 'ExtReportController@remove']);
323     Route::post('ext_report/update',
324 ['as'=>'report.external.change', 'middleware' => [], 'uses' =>
325 'ExtReportController@change']);
326

```



```

327     Route::get('/foto/{id}', ['as'=>'foto.get', 'middleware' =>
328 [], 'uses'=>'ImagesController@get_foto']);
329     Route::post('/foto/add/{id}', ['as'=>'foto.add',
330 'middleware' => [], 'uses'=>'ImagesController@add_foto']);
331     Route::get('/foto/{id}/delete', ['as'=>'foto.delete',
332 'middleware' => [], 'uses'=>'ImagesController@delete_foto']);
333     Route::get('/foto/last/{id}', ['as'=>'foto.last',
334 'middleware' => [], 'uses'=>'ImagesController@last_foto']);
335
336     Route::get('/regfiles/{file_id}/history',
337 ['as'=>'files.reg.history', 'middleware' => [],
338 'uses'=>'FilesController@reg_history']);
339     Route::get('/regfiles/{file_id}', ['as'=>'files.reg.get',
340 'middleware' => [], 'uses'=>'FilesController@reg_download']);
341     Route::get('/regfiles/{file_id}/show',
342 ['as'=>'files.reg.show', 'middleware' => [],
343 'uses'=>'FilesController@reg_show']);
344     Route::get('/regfiles/{file_id}/preview',
345 ['as'=>'files.reg.preview', 'middleware' => [],
346 'uses'=>'FilesController@reg_preview']);
347     Route::post('/regfiles/update', ['as'=>'files.reg.post',
348 'middleware' => [], 'uses'=>'FilesController@reg_update']);
349     Route::get('/regfiles/{file_id}/delete',
350 ['as'=>'files.reg.delete', 'middleware' => [],
351 'uses'=>'FilesController@reg_delete']);
352
353     Route::get('/unregfiles/{file_id}/history',
354 ['as'=>'files.unreg.history', 'middleware' => [],
355 'uses'=>'FilesController@unreg_history']);
356     Route::get('/unregfiles/{file_id}',
357 ['as'=>'files.unreg.get', 'middleware' => [],
358 'uses'=>'FilesController@unreg_download']);
359     Route::get('/unregfiles/{file_id}/show',
360 ['as'=>'files.unreg.show', 'middleware' => [],
361 'uses'=>'FilesController@unreg_show']);
362     Route::get('/unregfiles/{file_id}/preview',
363 ['as'=>'files.unreg.preview', 'middleware' => [],
364 'uses'=>'FilesController@unreg_preview']);
365     Route::post('/unregfiles/update', ['as'=>'files.unreg.post',
366 'middleware' => [], 'uses'=>'FilesController@unreg_update']);
367     Route::get('/unregfiles/{file_id}/delete',
368 ['as'=>'files.unreg.delete', 'middleware' => [],
369 'uses'=>'FilesController@unreg_delete']);
370
371     Route::get('/forms', ['as'=>'forms.index', 'middleware' =>
372 [], 'uses'=>'FormsController@index']);
373     Route::get('/forms/download', ['as'=>'forms.download',
374 'middleware' => [], 'uses'=>'FormsController@download']);
375     Route::post('/forms/upload', ['as'=>'forms.upload',
376 'middleware' => [], 'uses'=>'FormsController@upload']);
377     Route::get('/forms/print_regs', ['as'=>'forms.print_regs',
378 'middleware' => [], 'uses'=>'FormsController@regBussiness']);

```

```

379     Route::get('/forms/print_acard', ['as'=>'forms.print_acard',
380 'middleware' => [], 'uses'=>'FormsController@addrBussiness']);
381     Route::get('/forms/print_acard2',
382 ['as'=>'forms.print_acard2', 'middleware' => [],
383 'uses'=>'FormsController@addrCardBussiness']);
384     Route::get('/forms/print_pacard',
385 ['as'=>'forms.print_pacard', 'middleware' => [],
386 'uses'=>'FormsController@paperAddrBussiness']);
387
388     Route::get('logo', ['uses'=>'HomeController@logo']);
389
390     // web API
391     Route::get('/api/adm', ['as'=>'api.adm', 'middleware' =>
392 ['reflash'], 'uses'=>'APIController@adm']);
393     Route::get('/api/adm/helper', ['as'=>'api.adm.helper',
394 'middleware' => ['reflash'],
395 'uses'=>'APIController@adm_helper']);
396     Route::get('/api/street', ['as'=>'api.street', 'middleware'
397 => ['reflash'], 'uses'=>'APIController@street']);
398     Route::get('/api/doc_issuer', ['as'=>'api.doc_issuer',
399 'middleware' => ['reflash'],
400 'uses'=>'APIController@doc_issuer']);
401     Route::get('/api/relations', ['as'=>'api.relations',
402 'middleware' => ['reflash'],
403 'uses'=>'APIController@relations']);
404     Route::get('/api/mil_issuers', ['as'=>'api.mil_issuers',
405 'middleware' => ['reflash'],
406 'uses'=>'APIController@mil_issuers']);
407     Route::get('/api/mil_salutations',
408 ['as'=>'api.mil_salutations', 'middleware' => ['reflash'],
409 'uses'=>'APIController@mil_salutations']);
410     Route::get('/api/mil_specializations',
411 ['as'=>'api.mil_specializations', 'middleware' => ['reflash'],
412 'uses'=>'APIController@mil_specializations']);
413     Route::get('/api/pasp', ['as'=>'api.pasp', 'middleware' =>
414 ['reflash'], 'uses'=>'APIController@pasp']);
415     Route::get('/api/name', ['as'=>'api.ref_name', 'middleware'
416 => ['reflash'], 'uses'=>'APIController@ref_name']);
417     Route::get('/api/patronymic', ['as'=>'api.ref_patronymic',
418 'middleware' => ['reflash'],
419 'uses'=>'APIController@ref_patronymic']);
420     // Updater
421     Route::get('/updater', ['as'=>'updater.index', 'middleware'
422 => [], 'uses'=>'UpdaterController@client_side']);
423     Route::post('/updater', ['as'=>'updater.run', 'middleware'
424 => [], 'uses'=>'UpdaterController@run_update']);
425
426     include 'routes_ref.php';

```

Authenticate.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use Closure;
4     use Illuminate\Support\Facades\Auth;
5
6     use Illuminate\Support\Facades\Log;
7
8     class Authenticate
9     {
10        /**
11         * Handle an incoming request.
12         *
13         * @param \Illuminate\Http\Request $request
14         * @param \Closure $next
15         * @param string|null $guard
16         * @return mixed
17         */
18        public function handle($request, Closure $next, $guard =
19 null)
20        {
21            if ($guard) Auth::shouldUse($guard);
22            $user = Auth::user();
23            if (!$user) {
24                if ($request->ajax() || $request->wantsJson()) {
25                    return response('Unauthorized.', 401);
26                } else {
27                    return redirect()->guest('login');
28                }
29            } else {
30                if (config('cnap.log_user_input', false)) {
31                    Log::info("log      ".$request->method().      '
32 user:'. $user->id.      ' ('. $user->name.      ')      '. $request-
33 >fullUrl());
34                    if ($request->isMethod('post')) {
35                        self::log_input($request);
36                    }
37                }
38            }
39            return $next($request);
40        }
41        public static function log_input($request)
42        {
43            $input = $request->all();
44            $message = 'log input';
45            $user = Auth::user();
46            if ($user) $message .= ' user:' . $user->id . ' ('.
47 $user->name . ')';
48            Log::info($message, $input);
49        }

```

AutoloaderByConfig.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use App\Http\CnapLoader;
4     use Closure;
5     class AutoloaderByConfig
6     {
7         /**
8          * Handle an incoming request.
9          *
10         * @param  \Illuminate\Http\Request  $request
11         * @param  \Closure  $next
12         * @return mixed
13         */
14         public function handle($request, Closure $next)
15         {
16             app()->singleton('cnap.loader', function ($app) {
17                 $loader = new CnapLoader();
18                 $loader->register();
19                 return $loader;
20             });
21             app('cnap.loader')-
22 >add_region(config('cnap.region'));
23             $GLOBALS['cnap_xls_extension'] =
24 config('cnap.XLSX_downgrade_to_XLS', false) ? 'xls' : 'xlsx';
25             return $next($request);
26         }
27     }

```

CheckLoggedUser.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use App\User;
4     use Closure;
5     use Illuminate\Support\Facades\Auth;
6     class CheckLoggedUser
7     {
8         /**
9          * Handle an incoming request.
10         *
11         * @param  \Illuminate\Http\Request  $request
12         * @param  \Closure  $next

```

```

13     * @return mixed
14     */
15     public function handle($request, Closure $next)
16     {
17         $user = Auth::user();
18         if ($user && ($msg = $user->
19 >check_logged_on(User::LoginType_api))) {
20             if ($request->ajax() || $request->wantsJson()) {
21                 return response("Unauthorized.\r\n".$msg, 401);
22             } else {
23                 return redirect()->guest('login')-
24 >withErrors(['username' => $msg,]);
25             } }
26     return $next($request);}}

```

LoadUserProfiles.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use Closure;
4     use Illuminate\Support\Facades\Auth;
5     class LoadUserProfiles
6     {
7         /**
8          * Handle an incoming request.
9          *
10         * @param \Illuminate\Http\Request $request
11         * @param \Closure $next
12         * @return mixed
13         */
14         public function handle($request, Closure $next, $options
15 = null)
16         {
17             $user = Auth::user();
18             if ($user) {
19                 $loader = app('cnap.loader');
20                 if (is_null($options)) {
21                     $user->load_impersonate();
22                     $loader->add_profiles($user->getProfile());
23                 } else {
24                     $result = $loader->process($options);
25                     if ($result) return $result;
26                 }

```

```

27         }
28
29         $result = $next($request);
30
31         if ($options == 'init') {
32             $loader = app('cnap.loader');
33             $loader->process('deinit');
34         }
35
36         return $result;
37     }
38 }

```

LogSQL.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use Closure;
4     use Illuminate\Support\Facades\DB;
5     use Illuminate\Support\Facades\Log;
6
7     class LogSQL
8     {
9         /**
10          * Handle an incoming request.
11          *
12          * @param  \Illuminate\Http\Request  $request
13          * @param  \Closure  $next
14          * @return mixed
15          */
16
17         public function handle($request, Closure $next)
18         {
19             DB::listen(
20                 function ($query, $bindings = null, $time =
21 null, $connectionName = null) {
22                 // Laravel 5.2 changed the way some core
23 events worked. We must account for
24                 // the first argument being an "event
25 object", where arguments are passed
26                 // via object properties, instead of
27 individual arguments.
28                 if ( $query instanceof
29 \Illuminate\Database\Events\QueryExecuted ) {
30                     $bindings = $query->bindings;
31                     $time = $query->time;
32                     $connection = $query->connection;
33
34                     $query = $query->sql;
35                 } else {

```

```

36         $connection =
37 DB::connection($connectionName);
38     }
39     $this->addQuery((string) $query, $bindings,
40 $time, $connection);
41     }
42     );
43     $res = $next($request);
44     return $res;
45 }
46
47     protected function checkBindings($bindings)
48     {
49         foreach ($bindings as &$binding) {
50             if (is_string($binding) &&
51 !mb_check_encoding($binding, 'UTF-8')) {
52                 $binding = '[BINARY DATA]';
53             }
54         }
55         return $bindings;
56     }
57
58     public function addQuery($query, $bindings, $time,
59 $connection) {
60         $pdo = $connection->getPdo();
61         $bindings = $connection->prepareBindings($bindings);
62         $bindings = $this->checkBindings($bindings);
63         if (!empty($bindings)) {
64             foreach ($bindings as $key => $binding) {
65                 // This regex matches placeholders only, not
66 the question marks,
67                 // nested in quotes, while we iterate
68 through the bindings
69                 // and substitute placeholders by suitable
70 values.
71                 $regex = is_numeric($key)
72                     ?
73 "/\?(?=(?:[^\'\\\']*'[^\'\\\']*')*['\\\']*$)/"
74                     :
75 "/{:$key}(?=(?:[^\'\\\']*'[^\'\\\']*')*['\\\']*$)/";
76                 $query = preg_replace($regex, $pdo-
77 >quote($binding), $query, 1);
78             }
79         }
80         Log::info("{:$time} {$this->formatSql($query)}");
81     }
82     protected function formatSql($sql)
83     {
84         return trim(preg_replace("/\s*\n\s*/", "\n", $sql));
85     }

```

```

1     <?php
2     namespace App\Http\Middleware;
3     use Closure;
4     use Illuminate\Support\Facades\Auth;
5
6     class RedirectIfAuthenticated
7     {
8         /**
9          * Handle an incoming request.
10         *
11         * @param  \Illuminate\Http\Request  $request
12         * @param  \Closure  $next
13         * @param  string|null  $guard
14         * @return mixed
15         */
16         public function handle($request, Closure $next, $guard =
17 null)
18         {
19             if (Auth::guard($guard)->check()) {
20                 return redirect('/');
21             }
22             return $next($request);
23         }

```

SessionReflash.php

```

1     <?php
2     namespace App\Http\Middleware;
3     use Closure;
4     use Illuminate\Support\Facades\Session;
5     class SessionReflash
6     {
7         /**
8          * Handle an incoming request.
9          *
10         * @param  \Illuminate\Http\Request  $request
11         * @param  \Closure  $next
12         * @return mixed
13         */
14         public function handle($request, Closure $next)
15         {
16             $result = $next($request);
17             Session::reflash();
18             return $result;
19         }
20     }

```

Request.php


```
1     <?php
2     namespace App\Http\Requests;
3     use Illuminate\Foundation\Http\FormRequest;
4     abstract class Request extends FormRequest
5     { // }
```

Додаток Ж

Демонстрація вікна пункту меню «Пошук адреси»

Пошук адреси | Пошук особи | Реєстрації | Залиї журналі | 29/02

П. Україна | м.Лисичанськ, Луганська область | будинок 28 | квартира 2

Відомості про адресу: **неідентифікована**

Реєстрація нової особи	Реєстрація перебування люди особи	Висловити дані про реєстрацію	Висловити дані про реєстрацію
Друк адресної картки	Відсутність зареєстрованих	Сторінка 0/1	Доклад на реєстрацію

Архівна інформація | Архівний номер адресної картки : AK44385

Копіювати


Зареєстровано 4 | Відомості про особу та дані про реєстрацію / зняття / списування

Курдюков Віктор Олександрович 27.04.1974	м.Лисичанськ, Луганська область (4411600000)	Особова картка
паспорт ЕМ 620630 від 13.12.2000 виданий Лисичанським ІМБ УМВС України в Луганській області	28.11.1990 Зареєстрованої вулиці П. України, будинок 28, квартира 2, м.Лисичанськ	Реєстрація в сім'ю Діагностика реєстрації
Зняття з реєстрації	На нову адресу проживання	Особова картка
Курдюков Олександр Олександрович 07.11.1961	м.Лисичанськ, Луганська область (4411600000)	Реєстрація в сім'ю
паспорт ЕМ 144691 від 25.06.2002 виданий ЛИСИЧАНСЬКИМ ГОВД	08.12.1968 Зареєстрованої вулиці П. України, будинок 28, квартира 2, м.Лисичанськ	Доклад про смерть
свідчення про смерть №Д 912839 від 12.12.2017 виданий Лисичанським ІМБ ДРАЦС ПТЮО у Луганській області	12.12.2017 у зв'язку з смертю а/з 1467 від 12.12.2017р.	Особова картка
Знятий з вулиці П. України, будинок 28, квартира 2, м.Лисичанськ	с.Тисівська, Красноармійський район, Харківська область (633360002)	Реєстрація в сім'ю
Горбань Стеніслав Іванович 25.10.1911	22.08.2000 Зареєстрованої вулиці П. України, будинок 28, квартира 2, м.Лисичанськ	
паспорт ЕМ 144690 від 25.06.2002 виданий ЛИСИЧАНСЬКИМ ГОВД		

Додаток И

Демонстрація вікна «Особова картка»

Пошук адреси Пошук особи Реєстрації **Звіт Кур'янина** Адміністратор Системи 29.63



ервистика

Прізвище **Ім'я** **По батькові**

Дата народження **Паспорт (номер)** **Номер ІПН**

Номер ЄДРР

Пошук **Нова особова картка**

ПІБ **Дата**

Газна Дар'я Ігорівна 03.08.1996 05.04.2007

вулиця Ген.Потапенка, будинок 222, квартира 24, м.Лисичанськ
паспорт ЕН 900206 від 09.10.2012 виданий ЛІСУ/НАНС/ММ ГОВД

Остання реєстрація

Вибрати
Особова картка

Додаток К

Демонстрація вікна «Ресстраційна картка»

Пошук адреси | Пошук особи | Ресстрації | Поточна ресстрація | Звіт/Журнали | 29:55 | Адміністратор Системи

м.Лисичанськ, Луганська область, вулиця Л. Українки, будинок 28, квартира 2

10.05.2018

Особа має активну ресстрацію за адресою м.Лисичанськ, Луганська область, вулиця Ген.Поталенка, будинок 222, квартира 24. Програма автоматично зробить зняття з ресстрації!

Прізвище	Ім'я	По батькові	Дата зміни
Газна	Дар'я	Ігорівна	21.03.2017

Тип	Серія	Номер	Виданий	Дата видачі	Дійсний по
паспорт	ЕН	900206	ЛІСІЧАНСЬКИМ ГОВІД	09.10.2012	до м.г.г.г.

Увага: Луганська область, м.Лисичанськ
м.Лисичанськ, Луганська область (4411600000)

Адреса ресстрації: мiсце проживання на дату звернення
вул. Бульвар, буд. 111, корп. 123, кв. 124
вулиця Ген. Поталенка, будинок 222, квартира 24

Адреса:

Документ на право власності:

Представник:

Телефон:

Відмова:

Адмін порушення:

Родинні стосунки:

[Заява на зняття з ресстрації](#) | [Заява на ресстрацію](#) | [Документи](#)