

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**Лабораторна робота №3-4**

з дисципліни *“Компоненти програмної інженерії 4. Системна архітектура, проектування та розроблення ПЗ”*

**«Проектування архітектури програмного забезпечення Єдиного реєстру боржників України»**

Виконали

студентки III курсу

групи КП-72

Городченко А.В.

Завгородня А.В.

Викладач: Гречко А.В.

Київ 2020

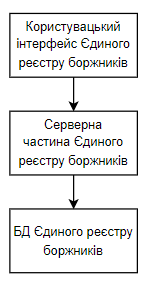
**Мета роботи:** уточнення даних, інтерфейсів, алгоритму функціонування, архітектури програми. Розроблення технічного проекту, яке охоплює проектування системи, що включає проектування архітектури системи і детальне проектування.

**Завдання на лабораторну роботу**

Проаналізувати та спроектувати архітектуру програмного забезпечення для реалізації Національного реєстру відповідно до індивідуального варіанта завдання. Задокументувати результати проектування.

## Структура системи

Структурно система складається із ізольованих один від одного рівнів абстракцій, вона побудована за шаблоном багаторівневневої архітектури. Її можна представити, як ієрархічну структуру (див рис. 1), кожна частина якої взаємодії лише із рівнем, що знаходиться нижче за ієрархією. Засобом спілкування рівнів слугує прикладний програмний інтерфейс (API) яке надає відповідний рівень.



*рис 1. Багаторівнева архітектура системи Єдиного реєстру боржників*

## Опис компонентів системи Єдиного реєстру боржників України

1. **Користувацький інтерфейс Єдиного реєстру боржників України**

Представляє собою веб-додаток, який може бути відкритий у веб-браузері, містить сторінки, необхідні для реалізації функціональних вимог, перелічених у документації до лабораторної роботи 2, надає можливість користувачу взаємодіяти із системою Реєстру боржників та напряму взаємодіє із рівнем Бізнес логіки.

1. **Бізнес логіка Єдиного реєстру боржників України**

Представляє собою веб-сервер, який доступний за адресою, відомою Користувацькому інтерфейсу. Структурно складається із наступних програмних модулів, які в свою чергу складаються із власних підмодулей:

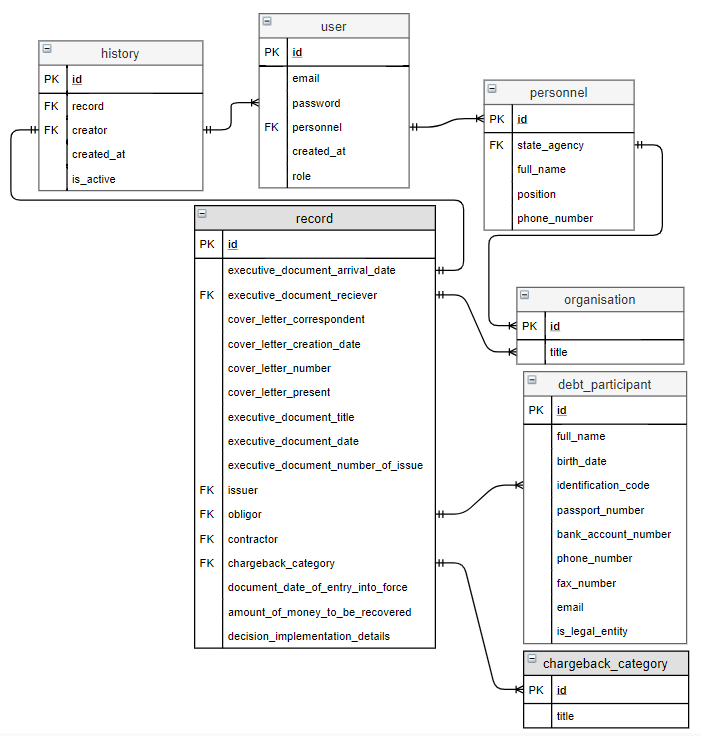
* Модуль абстракції над БД:
  + Моделі сутностей користувачів системи (Користувач, Адміністратор та Реєстратор)
  + Модуль запитів до БД
* Модуль авторизації:
  + Функціональний модуль авторизації Реєстраторів та Адміністраторів системи
* Модуль для роботи із записами про боржників Єдиного реєстру боржників України
  + Функціональний модуль створення нового запису про боржника
  + Функціональний модуль редагування існуючого запису про боржника
  + Функціональний модуль деактивації існуючого запису про боржника
  + Функціональний модуль пошуку записів про боржника у Реєстрі за запитом користувача
* Модуль роботи із профілями Реєстраторів та Адміністраторів
  + Функціональний модуль створення записів про нових Реєстраторів та Адміністраторів системи
  + Функціональний модуль для отримання відомостей про обраного Реєстратора чи Адміністратора
  + Функціональний модуль для деактивації профілю реєстратора чи Адміністратора
  + Функціональний модуль пошуку відомостей про реєстратора чи адміністратора за запитом

1. **БД Єдиного реєстру боржників України**

Являє собою сервер СКБД. Зберігає дані про користувачів (Адміністраторів, та Реєстраторів), записи про боржників та зміни, внесені до записів про боржників Реєстраторами.

## 

## Структура БД



*Рис. ER-модель БД*

Таблиці:

* *user* - зберігає інформацію про акаунти Адміністраторів та Реєстраторів (поле *role* визначає роль користувача) Єдиного реєстру боржників України;
* *record* - зберігає записи про заборгованості юридичних та фізичних осіб*.* Зберігає поля, які перелічені у додатку А, лабораторної роботи 1;
* *chargeback\_category* - зберігає усі можливі категорії стягнення (характер зобов’язання) визначені у пункті 8 розділу I  «Положення про автоматизовану систему виконавчого провадження»;
* *debts\_participant* – зберігає інформацію про стягувача та боржника;
* *organisation* – зберігає найменування органів державної виконавчої служби;
* *personnel* – зберігає інформацію про посадових осіб;
* *history* – зберігає ким і коли був створений запис про заборгованість.

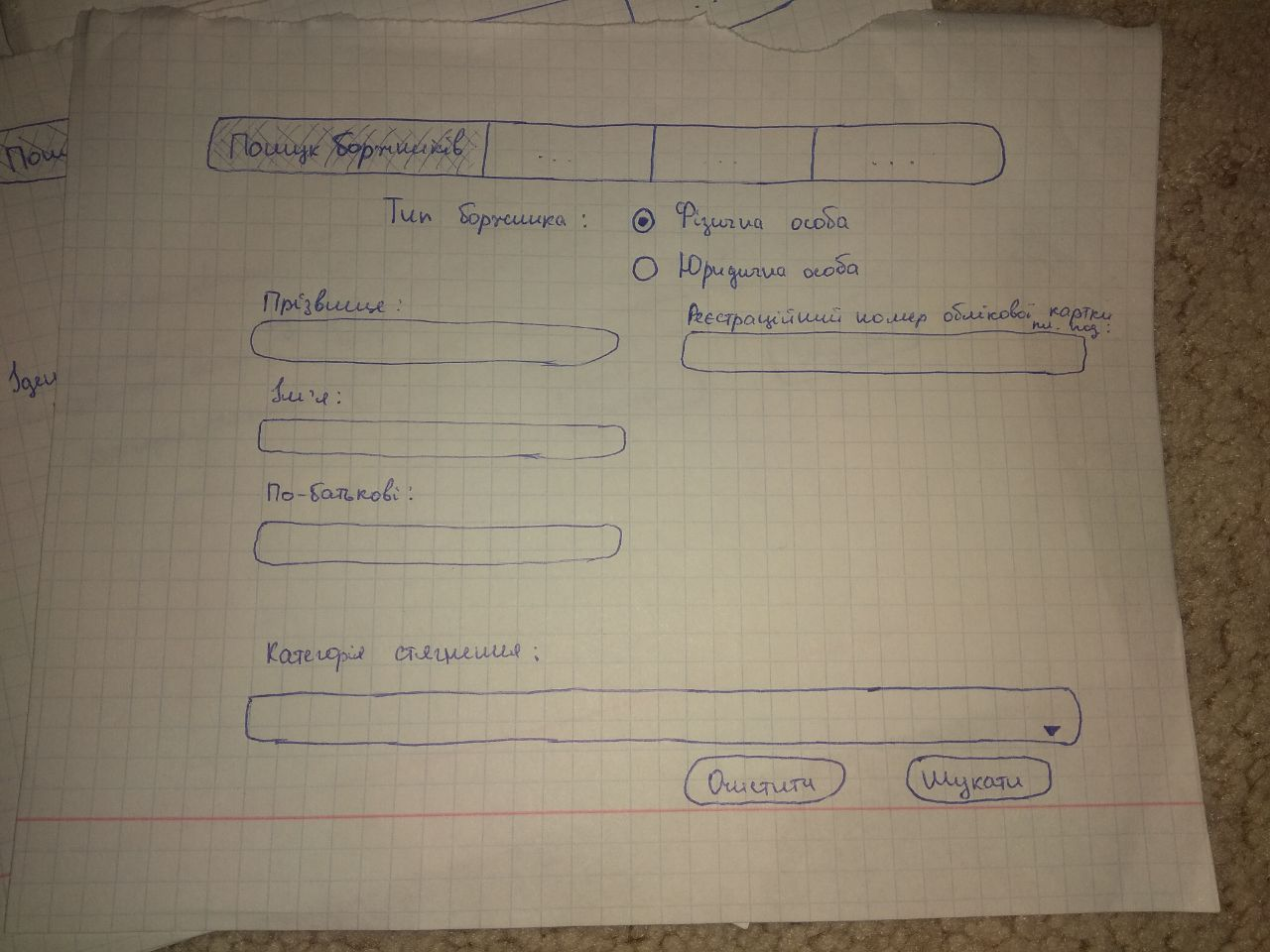
Зв’язки між таблицями: позначені стрілками, які визначають відношення «Обов’язкове one-to-many», «Обов’язкове one-to-one»

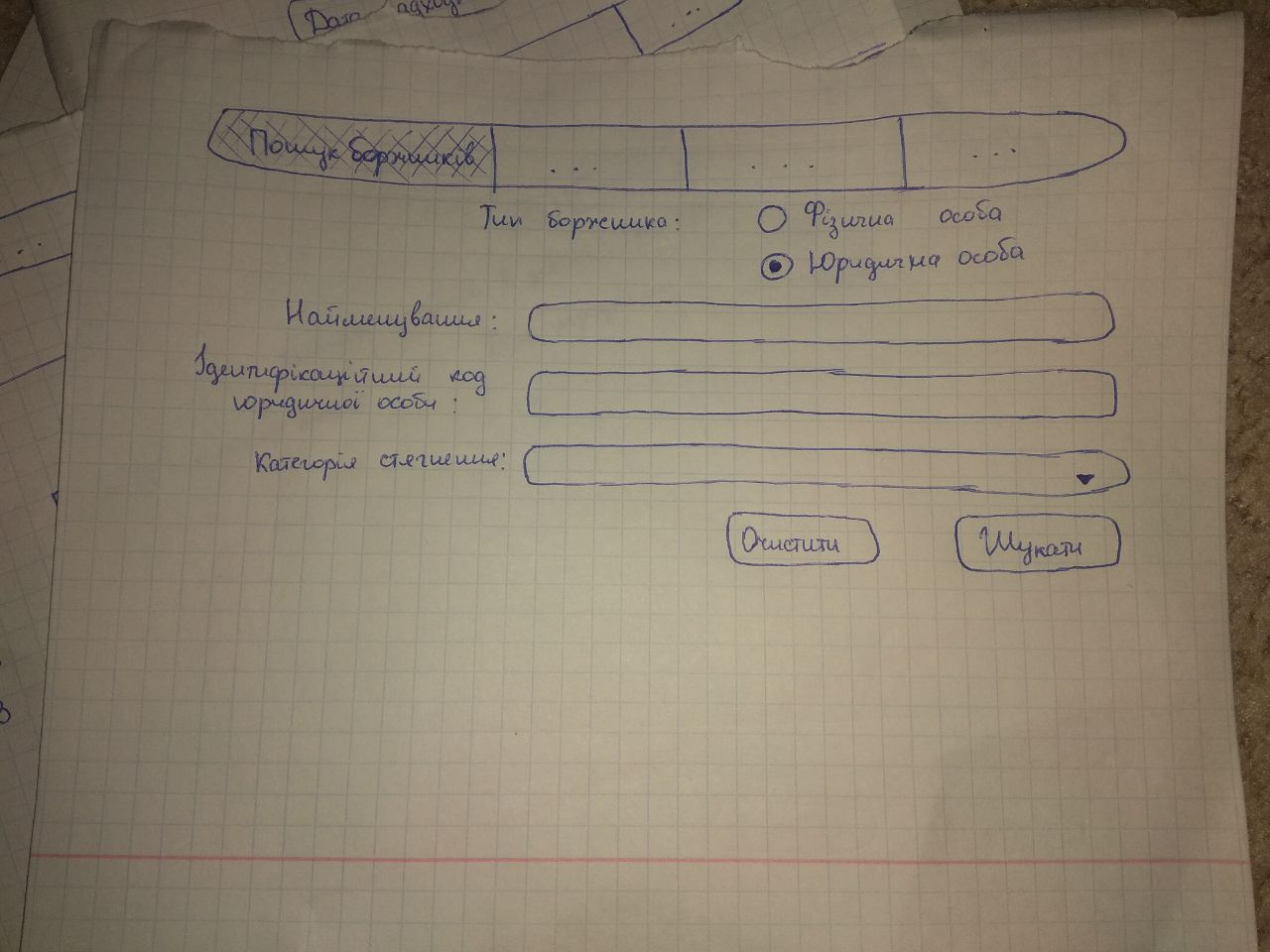
Первинні та зовнішні ключі: на схемі визначені позначками “[PK]” та “[FK]” відповідно

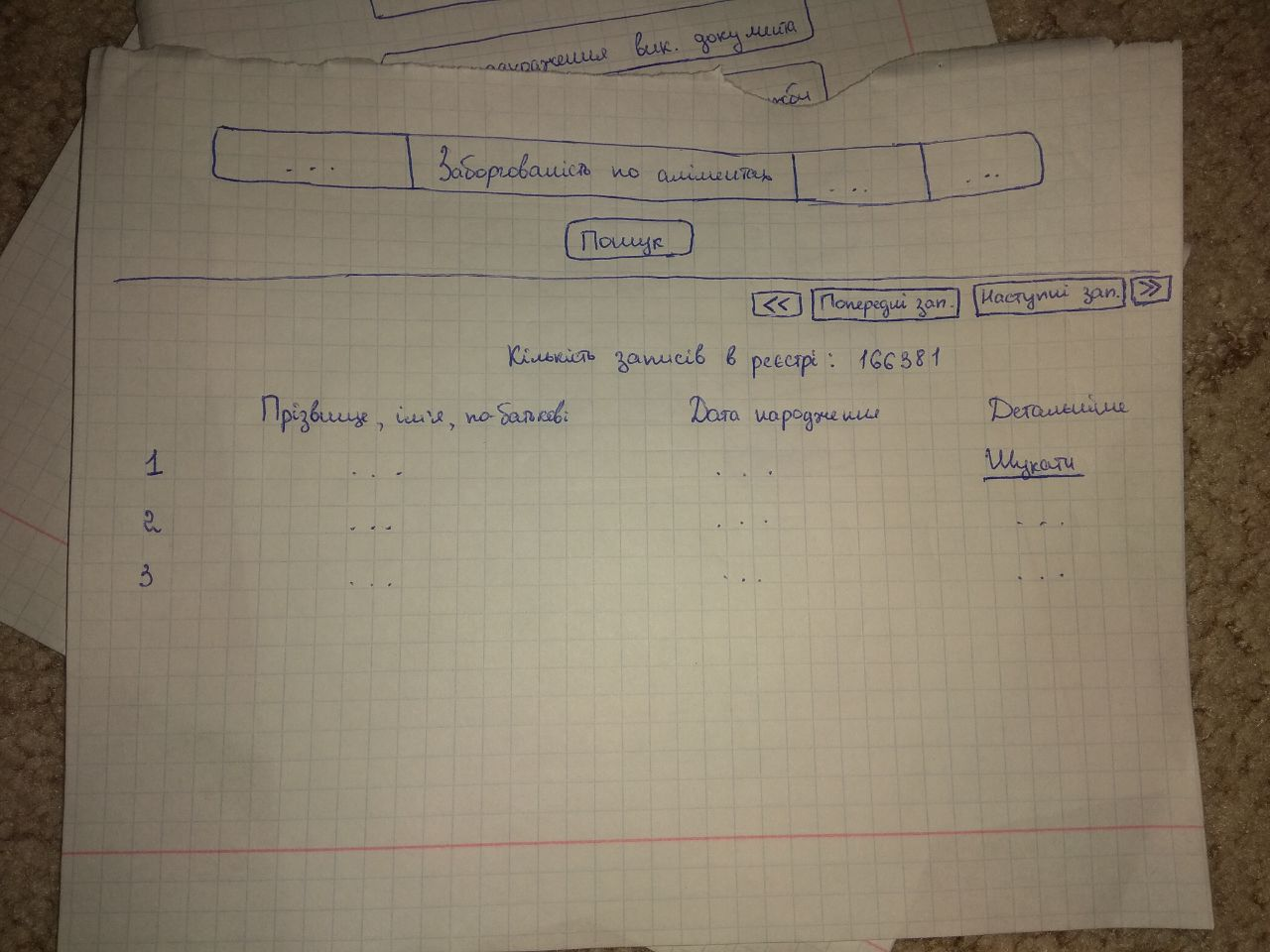
Тип СУБД: реляційна база даних PostgreSQL 11.

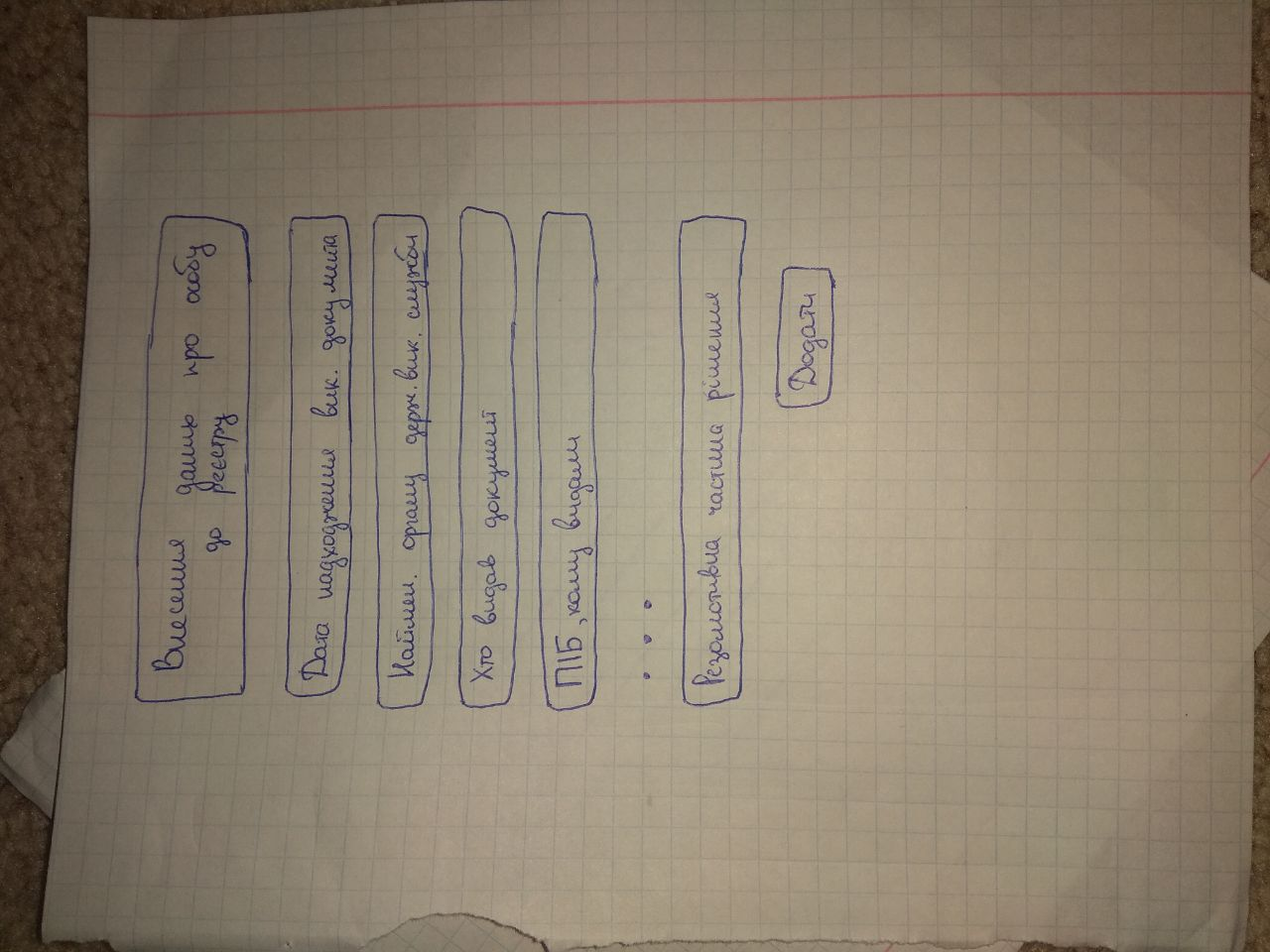
## Архітектура користувацького інтерфейсу програми

Основні елементи мають збігатись із існуючим рішенням Єдиного Реєстру боржників, який доступний за адресою <https://erb.minjust.gov.ua/#/search-debtors>.

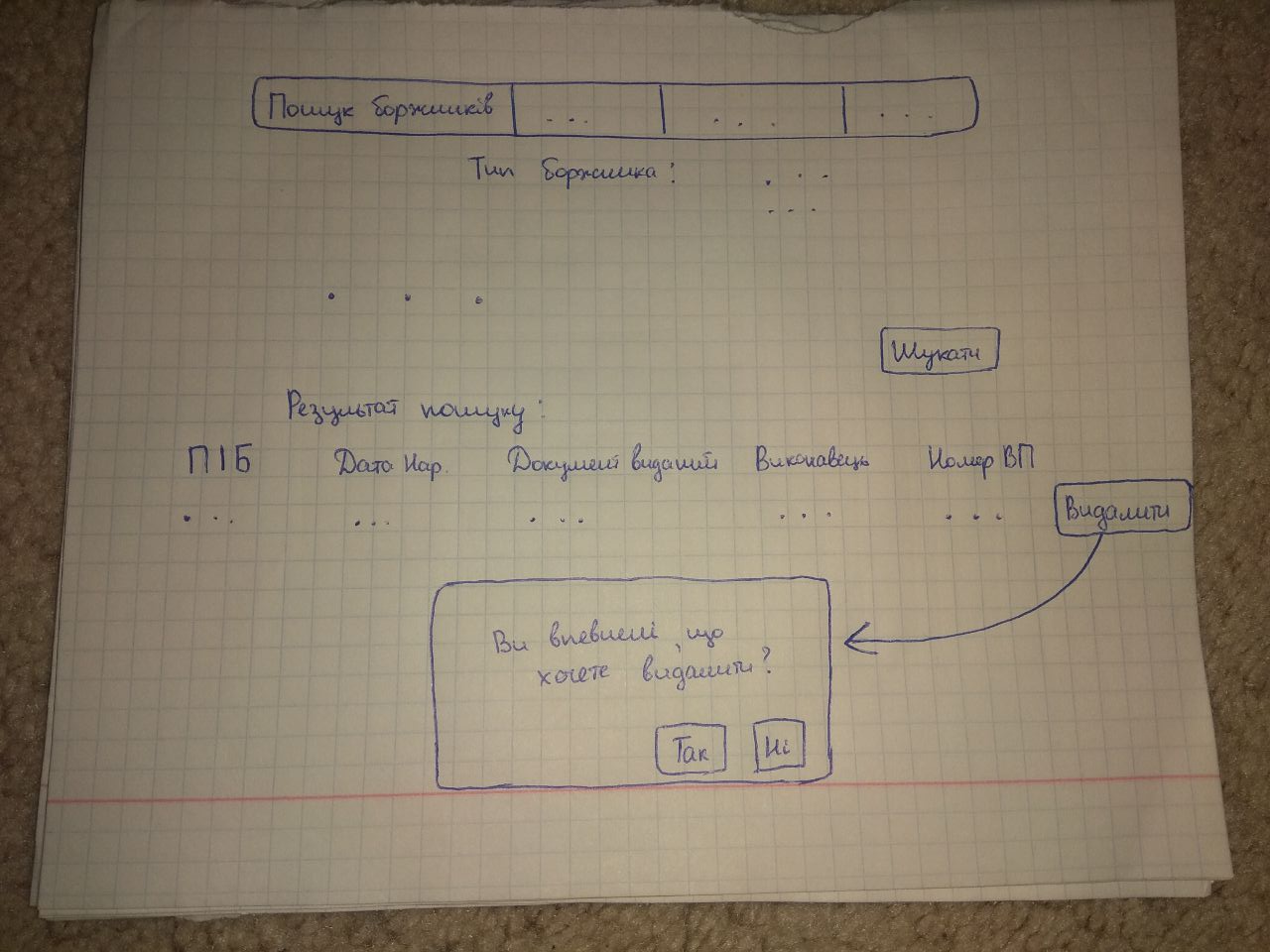
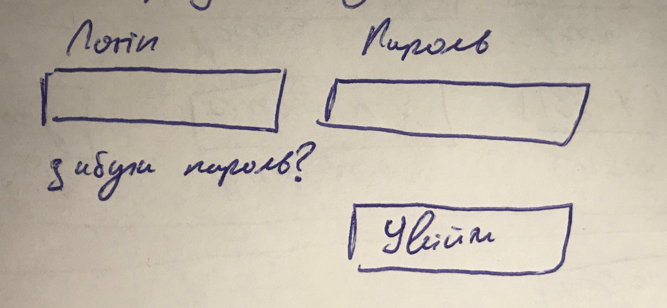
Наведемо макет системи основних елементів користувацького інтерфейсу:

*Рис 1. Сторінка пошуку даних про фізичних боржників, внесених до реєстру*

*Рис 2. Сторінка пошуку даних про юридичних боржників, внесених до реєстру*

*Рис 3. Сторінка зі списком боржників за певним критерієм*

*Рис 4. Сторінка внесення даних про боржника до Реєстру*

*Рис 5. Видалення запису про боржника з реєстру*

*Рис 6. Сторінка авторизації Реєстратора або Адміністратора у системі Реєстру боржників*

## Засоби розробки і склад програмного забезпечення системи

Розроблюване програмне забезпечення буде включати в себе наступні програмні компоненти:

* Серверної частини (server-side) – включає в себе модулі частини бізнес логіки Реєстру боржників та СКБД
* Клієнтської частини (client-side) – включає в себе керування користувацькою частиною реєстру та обмін даними з серверною частиною
* Тести серверної та клієнтської частини – автоматизовані тести основного функціоналу системи

Отже, розроблювана система буде веб-орієнтованою. Як засоби розробки буде використано мову програмування Java для серверної частини та мова програмування JavaScript для клієнтської частини. В якості СУБД виступатиме PostgreSQL 11.

## Технічне забезпечення системи

Технічне забезпечення системи Реєстру боржників складається із cерверного обладнання, який зберігає код бізнес логіки, СКБД та клієнтської частини, а також код автоматичних тестів до серверної частини.

Даний сервер має забезпечувати швидку та безперебійну роботу системи, тому він має відповідати наступним вимогам:

* ОС: Linux Ubuntu > 16.10
* CPU: Quad core 3.8 GHz+
* RAM: мінімум 6 GB
* Database: space: 128 GB

## Персонал розробки системи

Персонал: Городченко Анна, Завгородня Анна

Чисельність персоналу: 2 людини.

Кваліфікації персоналу: 2 студенти кафедри програмного забезпечення комп’ютерних систем ФПМ НТУУ “КПІ ім Ігоря Сікорського”

Функції персоналу: вчасне і якісне завершення процесу створення та тестування ПЗ відповідно до вимог, які містяться у ТЗ системи, розгортання системи на виділеному сервері.

Розподіл робіт між персоналом:

* Завгородня Анна– розробник клієнтської та серверної частин системи
* Городченко Анна – розробник тестів та технічної документації системи

## Структура бази даних

Далі наведено код SQL-скриптів, які використовуються частиною бізнес логіки системи Реєстру боржників для керування даними, збереженими у БД.

*Таблиця 1. Створення структури БД Єдиного*

*реєстру боржників України*

|  |
| --- |
| init.sql |
| CREATE TABLE reestr\_borzhnikiv.chargeback\_categories  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.chargeback\_categories\_id\_seq'::regclass),     title *CHARACTER* varying(100) COLLATE pg\_catalog."default" NOT NULL,      CONSTRAINT chargeback\_categories\_pkey PRIMARY KEY (id)  )   ALTER TABLE reestr\_borzhnikiv.chargeback\_categories      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.debt\_participants  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.debt\_participants\_id\_seq'::regclass),     full\_name *CHARACTER* varying(200) COLLATE pg\_catalog."default" NOT NULL,      birth\_date date,      identification\_code CHARACTER VARYING(10) COLLATE pg\_catalog."default" NOT NULL,      passport\_number CHARACTER VARYING(9) COLLATE pg\_catalog."default",      bank\_account\_number CHARACTER VARYING(29) COLLATE pg\_catalog."default",      phone\_number CHARACTER VARYING(12) COLLATE pg\_catalog."default",      fax\_number CHARACTER VARYING(10) COLLATE pg\_catalog."default",      email CHARACTER VARYING(30) COLLATE pg\_catalog."default",      CONSTRAINT debt\_participants\_pkey PRIMARY KEY (id)  )   ALTER TABLE reestr\_borzhnikiv.debt\_participants      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.history  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.history\_id\_seq'::regclass),      debt *BIGINT* NOT NULL,      creator *BIGINT* NOT NULL,      created\_at timestamp without time zone NOT NULL,      CONSTRAINT history\_pkey PRIMARY KEY (id),      CONSTRAINT creator FOREIGN KEY (creator)          REFERENCES reestr\_borzhnikiv.users (id) MATCH simple          ON UPDATE no action          ON DELETE no action,      CONSTRAINT debt FOREIGN KEY (debt)          REFERENCES reestr\_borzhnikiv.records (id) MATCH simple          ON UPDATE no action          ON DELETE no action  )   ALTER TABLE reestr\_borzhnikiv.history      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.organisations  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.organisations\_id\_seq'::regclass),      title *CHARACTER* varying(100) COLLATE pg\_catalog."default" NOT NULL,      CONSTRAINT organisations\_pkey PRIMARY KEY (id)  )   ALTER TABLE reestr\_borzhnikiv.organisations      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.personnel  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.personnel\_id\_seq'::regclass),      state\_agency *BIGINT* NOT NULL,      full\_name *CHARACTER* varying(100) COLLATE pg\_catalog."default" NOT NULL,      "position" CHARACTER VARYING(30) COLLATE pg\_catalog."default" NOT NULL,      phone\_number CHARACTER VARYING(12) COLLATE pg\_catalog."default",      CONSTRAINT personnel\_pkey PRIMARY KEY (id),      CONSTRAINT state\_agency FOREIGN KEY (state\_agency)          REFERENCES reestr\_borzhnikiv.organisations (id) MATCH simple          ON UPDATE no action          ON DELETE no action  )   ALTER TABLE reestr\_borzhnikiv.personnel      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.records  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.debts\_id\_seq'::regclass),      executive\_document\_recieved\_date timestamp without time zone NOT NULL,      executive\_document\_reciever BIGINT NOT NULL,      cover\_letter\_correspondent CHARACTER VARYING(100) COLLATE pg\_catalog."default",      cover\_letter\_creation\_date timestamp without time zone,      cover\_letter\_number BIGINT,      cover\_letter\_present boolean NOT NULL,      executive\_document\_title CHARACTER VARYING(100) COLLATE pg\_catalog."default" NOT NULL,      executive\_document\_date date NOT NULL,      executive\_document\_number\_of\_issue BIGINT NOT NULL,      issuer BIGINT NOT NULL,      obligor BIGINT NOT NULL,      contractor BIGINT NOT NULL,      chargeback\_category BIGINT NOT NULL,      document\_date\_of\_entry\_into\_force timestamp without time zone NOT NULL,      amount\_of\_money\_to\_be\_recovered BIGINT NOT NULL,      decision\_implementation\_details CHARACTER VARYING(400) COLLATE pg\_catalog."default",      CONSTRAINT debts\_pkey PRIMARY KEY (id),      CONSTRAINT chargeback\_category FOREIGN KEY (chargeback\_category)          REFERENCES reestr\_borzhnikiv.chargeback\_categories (id) MATCH simple          ON UPDATE no action          ON DELETE no action,      CONSTRAINT contractor FOREIGN KEY (contractor)          REFERENCES reestr\_borzhnikiv.debt\_participants (id) MATCH simple          ON UPDATE no action          ON DELETE no action,      CONSTRAINT executive\_document\_reciever FOREIGN KEY (executive\_document\_reciever)          REFERENCES reestr\_borzhnikiv.organisations (id) MATCH simple          ON UPDATE no action          ON DELETE no action,      CONSTRAINT issuer FOREIGN KEY (issuer)          REFERENCES reestr\_borzhnikiv.personnel (id) MATCH simple          ON UPDATE no action          ON DELETE no action,      CONSTRAINT obligor FOREIGN KEY (obligor)          REFERENCES reestr\_borzhnikiv.debt\_participants (id) MATCH simple          ON UPDATE no action          ON DELETE no action  )   ALTER TABLE reestr\_borzhnikiv.records      owner TO postgres;CREATE TABLE reestr\_borzhnikiv.users  (      id *BIGINT* NOT NULL DEFAULT **Nextval**('reestr\_borzhnikiv.users\_id\_seq'::regclass),      email *CHARACTER* varying(50) COLLATE pg\_catalog."default" NOT NULL,      password CHARACTER VARYING(20) COLLATE pg\_catalog."default" NOT NULL,      personnel BIGINT NOT NULL,      created\_at timestamp without time zone NOT NULL,      role CHARACTER VARYING(15) COLLATE pg\_catalog."default" NOT NULL,      CONSTRAINT id PRIMARY KEY (id),      CONSTRAINT personnel FOREIGN KEY (personnel)          REFERENCES reestr\_borzhnikiv.personnel (id) MATCH simple          ON UPDATE no action          ON DELETE no action  )   ALTER TABLE reestr\_borzhnikiv.users      owner TO postgres;  COMMENT ON COLUMN reestr\_borzhnikiv.records.executive\_document\_recieved\_date IS  'дата надходження виконавчого документа';   COMMENT ON COLUMN reestr\_borzhnikiv.records.executive\_document\_reciever IS  'найменування органу державної виконавчої служби або прізвище, ім’я, по батькові приватного виконавця, до якого надійшов виконавчий документ' ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.cover\_letter\_correspondent IS  'кореспондент супровідного листа про відкриття виконавчого провадження'  ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.cover\_letter\_creation\_date IS  'дата супровідного листа про відкриття виконавчого провадження'  ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.cover\_letter\_number IS  'номер супровідного листа про відкриття виконавчого провадження'  ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.cover\_letter\_present IS  'помітка про наявність супровідного листа';   COMMENT ON COLUMN reestr\_borzhnikiv.records.executive\_document\_title IS  'назва виконавчого документа';   COMMENT ON COLUMN reestr\_borzhnikiv.records.executive\_document\_date IS  'дата видачі виконавчого документа';   COMMENT ON COLUMN reestr\_borzhnikiv.records.executive\_document\_number\_of\_issue  IS 'номер виконавчого документа';   COMMENT ON COLUMN reestr\_borzhnikiv.records.issuer IS  'найменування органу, прізвище, ім’я, по батькові та посада  посадової особи, що видав(ла) документ' ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.obligor IS 'боржник';   COMMENT ON COLUMN reestr\_borzhnikiv.records.contractor IS 'стягувач';   COMMENT ON COLUMN reestr\_borzhnikiv.records.chargeback\_category IS  'категорія стягнення за виконавчим документом згідно з пунктом 8 розділу І «Положення про автоматизовану систему виконавчого провадження»' ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.document\_of\_entry\_into\_force IS  'дата набрання рішенням законної (юридичної) сили'  ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.amount\_of\_money\_to\_be\_recovered IS  'сума грошових коштів, яка підлягає стягненню за виконавчим документом'  ;   COMMENT ON COLUMN reestr\_borzhnikiv.records.decision\_implementation\_details IS  'відомості про негайне виконання рішення (за наявності)' |

*Таблиця 2. Додавання в БД Єдиного*

*реєстру боржників України*

|  |
| --- |
| insert\_debt.sql |
| INSERT INTO reestr\_borzhnikiv.organisations              (title)       VALUES(  'Глухівський міськрайонний відділ державної виконавчої служби Північно-Східного міжрегіонального управління Міністерства юстиції (м. Суми)'  );   SELECT \*    FROM reestr\_borzhnikiv.organisations;   INSERT INTO reestr\_borzhnikiv.personnel              (state\_agency,               full\_name,               position,               phone\_number)       VALUES(1,              'Обуховська Ірина Анатоліївна',              'головний керівник',              '(05444)23354');   INSERT INTO reestr\_borzhnikiv.users              (email,               password,               personnel,               created\_at,               role)       VALUES('info@gl.sm.dvs.gov.ua',              'siw$#%#$erg#$',              1,              CURRENT\_TIMESTAMP,              'Реєстратор');   INSERT INTO reestr\_borzhnikiv.debt\_participants              (full\_name,               birth\_date,               identification\_code,               passport\_number)       VALUES('Воловик Леонід Васильович',              '1985-08-12',              '1234567891',              '000111001');   INSERT INTO reestr\_borzhnikiv.debt\_participants              (full\_name,               identification\_code)       VALUES(  'ДП "ДНІПРОВСЬКИЙ ЕЛЕКТРОВОЗОБУДІВНИЙ ЗАВОД"'  ,  '1224367896');   INSERT INTO reestr\_borzhnikiv.records              (executive\_document\_recieved\_date,               executive\_document\_reciever,               cover\_letter\_present,               executive\_document\_title,               executive\_document\_date,               executive\_document\_number\_of\_issue,               issuer,               obligor,               contractor,               chargeback\_category,               document\_date\_of\_entry\_into\_force,               amount\_of\_money\_to\_be\_recovered,               is\_legal\_entity)       VALUES(CURRENT\_TIMESTAMP,              1,              false,              'Назва документу',              '2019-04-02',              12345,              1,              1,              2,              1,              CURRENT\_TIMESTAMP,              12345000,              false);   INSERT INTO reestr\_borzhnikiv.history              (record,               creator,               created\_at,               is\_active)       VALUES(2,              2,              CURRENT\_TIMESTAMP,              true); |

*Таблиця 3. Деактивація запису про заборгованість*

|  |
| --- |
| update.sql |
| UPDATE reestr\_borzhnikiv.history h     SET is\_active = false    FROM reestr\_borzhnikiv.records r   WHERE h.record = r.id         AND r.id = 2; |

## Класи об’єктно-орієнтованого програмування

Нижче наведено класи об’єктно-орієнтованого програмування, які відповідають реляційним сутностям Єдиного реєстру боржників України.

Сутність User, що представляє собою відображення таблиці user БД Єдиного реєстру боржників України:

*Таблиця 4. Сутність User*

|  |
| --- |
| User.java |
| @Data @Entity **public class** User{  **private** Long **id**;  **private** String **email**;  **private** String **password**;  **private** Personnel **personnel**;  **private** LocalDateTime **createdAt**;  **private** String **role**; } |

Сутність Debt, що представляє собою відображення таблиці record БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність Debt*

|  |
| --- |
| Record.java |
| @Data @Entity **public class** Record {  **private** Long **id**;  **private** LocalDateTime **executiveDocumentArrivalDate**;  **private** Organisation **executiveDocumentReceiver**;  **private** String **coverLetterCorrespondent**;  **private** String **coverLetterNumber**;  **private** Boolean **coverLetterPresence**;  **private** String **executiveDocumentTitle**;  **private** LocalDateTime **executiveDocumentDate**;  **private** Long **executiveDocumentNumberOfIssue**;  **private** String **executiveDocumentIssuer**;  **private** DebtParticipant **obligor**;  **private** DebtParticipant **contractor**;   **private** Long **chargebackCategory**;  **private** LocalDateTime **documentDateOfEntryIntoForce**;  **private** Long **moneyAmountToBeRecovered**;  **private** String **informationAboutImplementationOfDecision**;  } |

Сутність ChargebackCategory, що представляє собою відображення таблиці chatgeback\_category БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність ChargebackCategory*

|  |
| --- |
| ChargebackCategory.java |
| @Data @Entity **public class** ChargebackCategory {  **private** Long **id**;  **private** String **title**; } |

Сутність Personnel, що представляє собою відображення таблиці personnel БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність Personnel*

|  |
| --- |
| Personnel.java |
| @Data @Entity **public class** Personnel{  **private** Long **id**;  **private** Organisation **state\_agency**;  **private** String **full\_name**;  **private** String **position**;  **private** String **phone\_number**; } |

Сутність DebtParticipant, що представляє собою відображення таблиці debt\_participant БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність DebtParticipant*

|  |
| --- |
| DebtParticipant.java |
| @Data @Entity **public class** DebtParticipant{  **private** Long **id**;  **private** LocalDate **birth\_date**;  **private** String **identification\_code**;  **private** String **passport\_number**;  **private** String **bank\_account\_number**;  **private** String **phone\_number**;  **private** String **fax\_number**;  **private** String **email**; } |

Сутність History, що представляє собою відображення таблиці history БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність History*

|  |
| --- |
| History.java |
| @Data @Entity **public class** History{  **private** Long **id**;  **private** Record **record**;  **private** User **user**;  **private** LocalDateTime **created\_at**;  **private** Boolean **is\_active**; } |

Сутність Organisation, що представляє собою відображення таблиці organisation БД Єдиного реєстру боржників України:

*Таблиця 5. Сутність Organisation*

|  |
| --- |
| Organisation.java |
| @Data @Entity **public class** Organisation{  **private** Long **id**;  **private** String **title**; } |