# CRM banking

## 1. Business (functional) requirements

### 1.1. Idea

CRM is a system for financial companies, which helps to hold clients' databases and fixes the history of all actions with clients. For example, there are clients' databases, which we can export in the system as files. The manager of the company proposes financial products (loans, deposits) to clients. If one is interested, then the manager creates a lead (as potential interest) and specifies a product and amount (sum of future contract). And when the interest grown to agreeing of client to make a deal, the manager creates a Contract from the Lead.

### 1.2. General actions

Actions, which allowed for users with all roles:

* Possibility to login using Username (should be unique) and Passwort (should match the stored username and password).
* Show data in Sections:
  + Leads,
  + Contacts,
  + Contracts.

### 1.3. Manager’s actions

Actions, which allowed for users with only an Manager role:

* Import data in Contacts Section.
* Export data from Leads and Contacts Sections.
* Possibility to create new Contact:
  + Name,
  + Phone number,
  + Email address
  + Gender (male, female)
* Possibility to create new Lead:
  + Name,
  + Phone number,
  + Email address,
  + Product type,
  + Amount,
  + Term
* Possibility to create new Lead from Contact:
  + Contact,
  + Product type,
  + Amount,
  + Term
* Possibility to create Contract from Lead:
  + Customer,
  + Manager,
  + Stage,
  + Product type,
  + Amount,
  + Start date,
  + End date

### 1.4. Admin’s actions

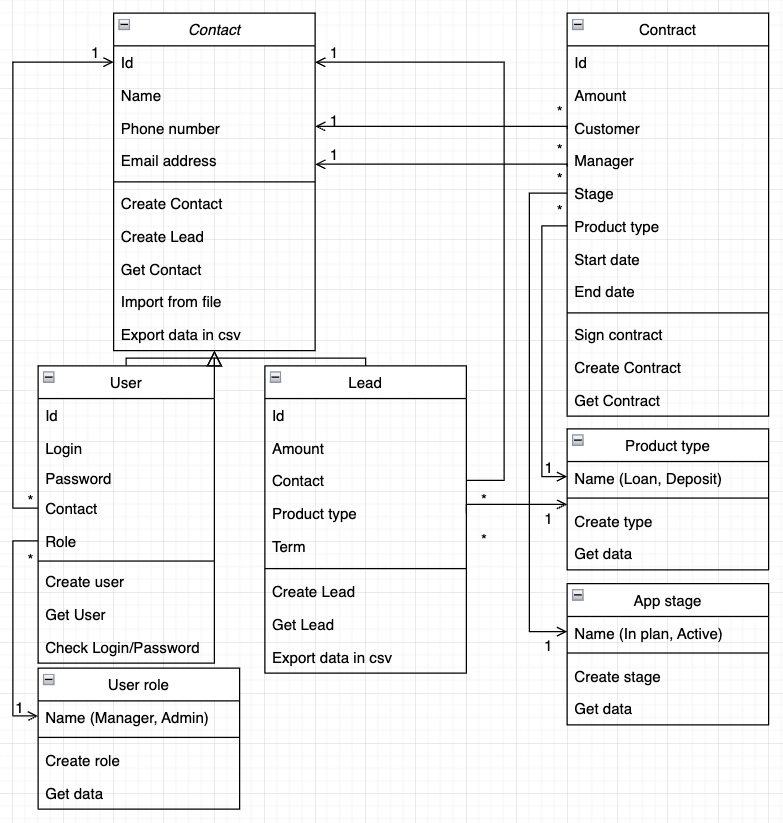
For users with only an Admin role:

* Possibility to see data in all Sections.
* Possibility to create a user with parameters:
  + username,
  + password
  + role.
* Possibility to create new records in Dictionaries:
  + User roles,
  + Product types,
  + App Stages

## 2. Non-functional requirements

* Programming Language: Python. DB: Sqlite3, ORM: Peewee.
* Submission: Github
* The application should start with the main screen. As this will be a console application.
* The application should be fully automated. (OOP, Requests, Modules)
* There will be two main user roles: Manager and Admin.
* Should be able to import Customer data (File Processing)
* The application should be able to handle exceptions (Error Handling).
* Logging should be present (Error Handling).
* Unit Test should be Present (Testing).

## 3. Diagram of classes



[Link in drawio](https://drive.google.com/file/d/1r3gu2W6AvA7fDeCBMkTaTJbUO8YGVgqw/view?usp=sharing)

Presentation

Why did I choose ORM: Peewee?

* Low code
* Possibility to change the DB type
* Support to use a few DBs at the same time (SQLite, MS SQL, PostgreSQL)
* Native Python code (without direct SQL-requests)

## Installation guide

install sqlite3 database: https://www.sqlite.org/download.html

pip install --requirement requirements.txt

Git repo: https://github.com/IvashDima/CRMbanking.git

## User guide

Run 'Main'.

In Python console:

1. Enter username and password. For the first login use test/test. Only logged users can create new users.

2. Follow the recommendations in Console:

1) Choose section!

1 - Contacts, 2 - Contracts, 3 - Leads, 4 - Users, 5 - Products, 0 - Exit

2) Choose operation in section:

1) Contacts.

1 - Create, 2 - Get info, 3 - Import from file, 4 - Export to file, 9 - Go to Section, 0 - Exit

2) Contracts.

1 - Create, 2 - Get info, 9 - Go to Section, 0 - Exit

3) Leads.

9 - Go to Section, 0 - Exit

4) Users.

1 - Create, 2 - Get info, 9 - Go to Section, 0 - Exit

5) Products.

9 - Go to Section, 0 - Exit

3. There is possible to check the age and gender by the name of the contact.

It automatically works when creating a contract. Or manually need to run functions:

checks/public\_data/get\_age\_by\_name

checks/public\_data/get\_gender\_by\_name

4. Show log: Log file is located in the root folder of the application with name: my\_log\_{currentdate}.log.

5. Run Pytest: TBD