

DIY ATM

Драговоз D, Надєєв І, Нагорний Y



ПЛАН

Постановка задачі

Основний функціонал АТМ

Другорядний функціонал АТМ

Архітектура

Безпека системи

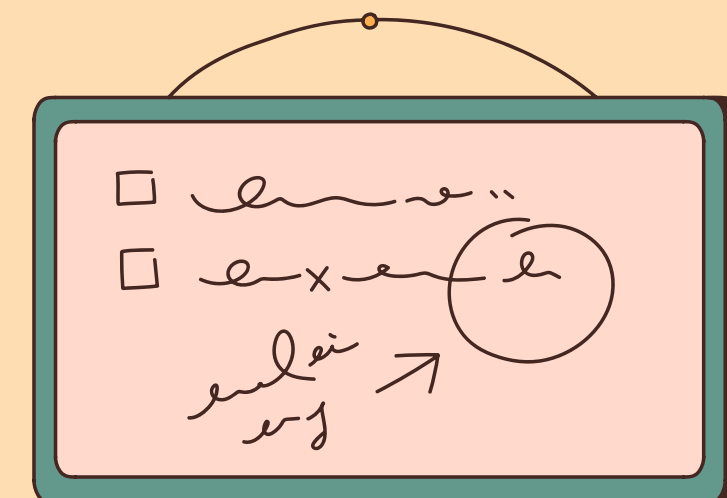
Про БД, UI, сервер

Демо



ПОСТАНОВКА ЗАДАЧІ

- Симуляція АТМ
- Без читача карт, приймача купюр, диспенсера
- Реалізація інтерфейсу
- Симуляція бази даних банку
- Змусити все працювати разом



ОСНОВНИЙ ФУНКЦІОНАЛ

- Авторизація та захист
- Інформація по рахунку
- Зняття готівки
- Поповнення рахунку готівкою
- Перекази
- Оплата спеціальних послуг

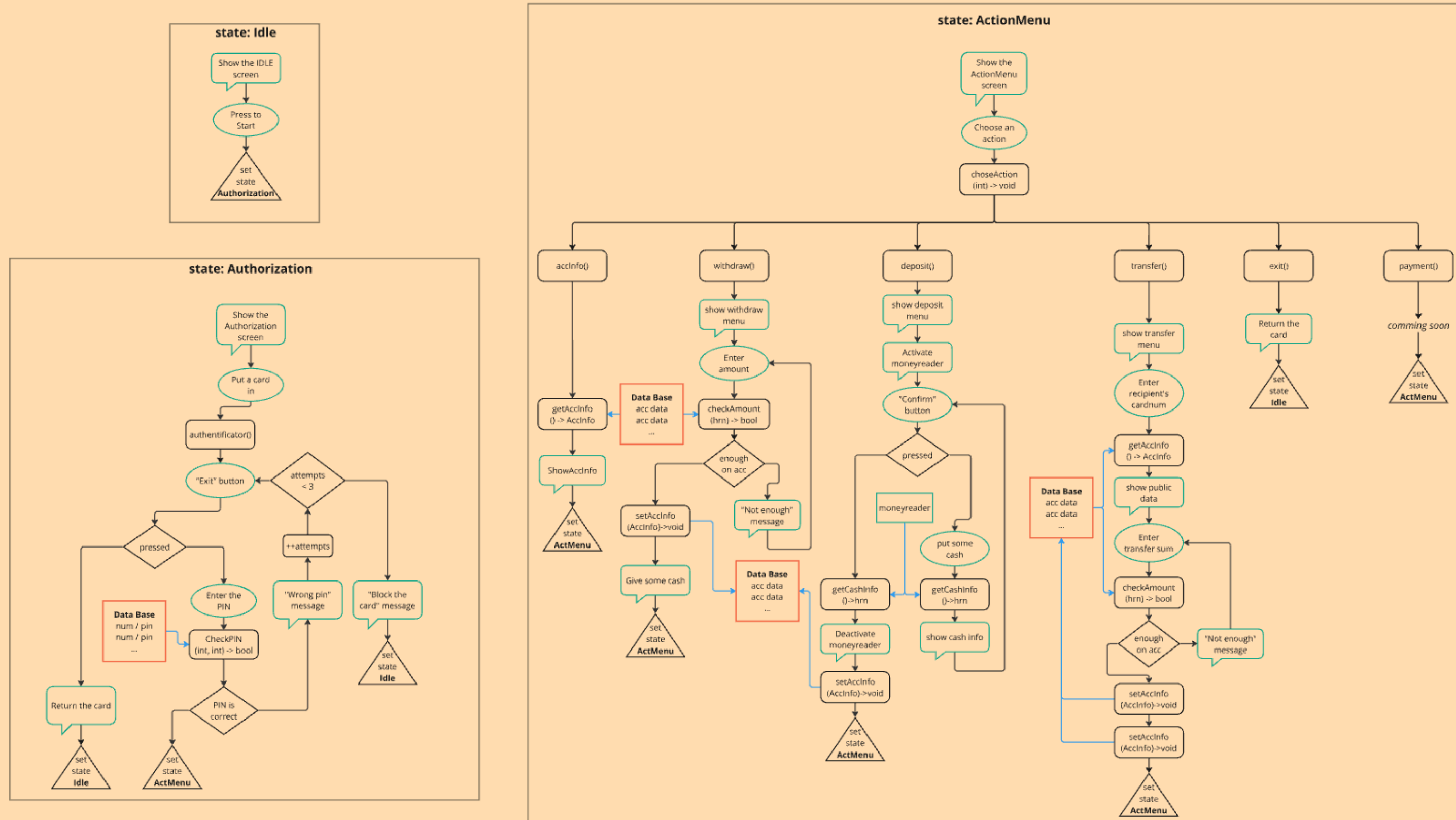


ДРУГОРЯДНИЙ ФУНКЦІОНАЛ

- Автоматичне закінчення сесії
- Кредитний рахунок
- Кредитний ліміт
- Обробка надлишку готівки
- Готівка в меню “Оплата”
- Стан maintenance



АРХИТЕКТУРА

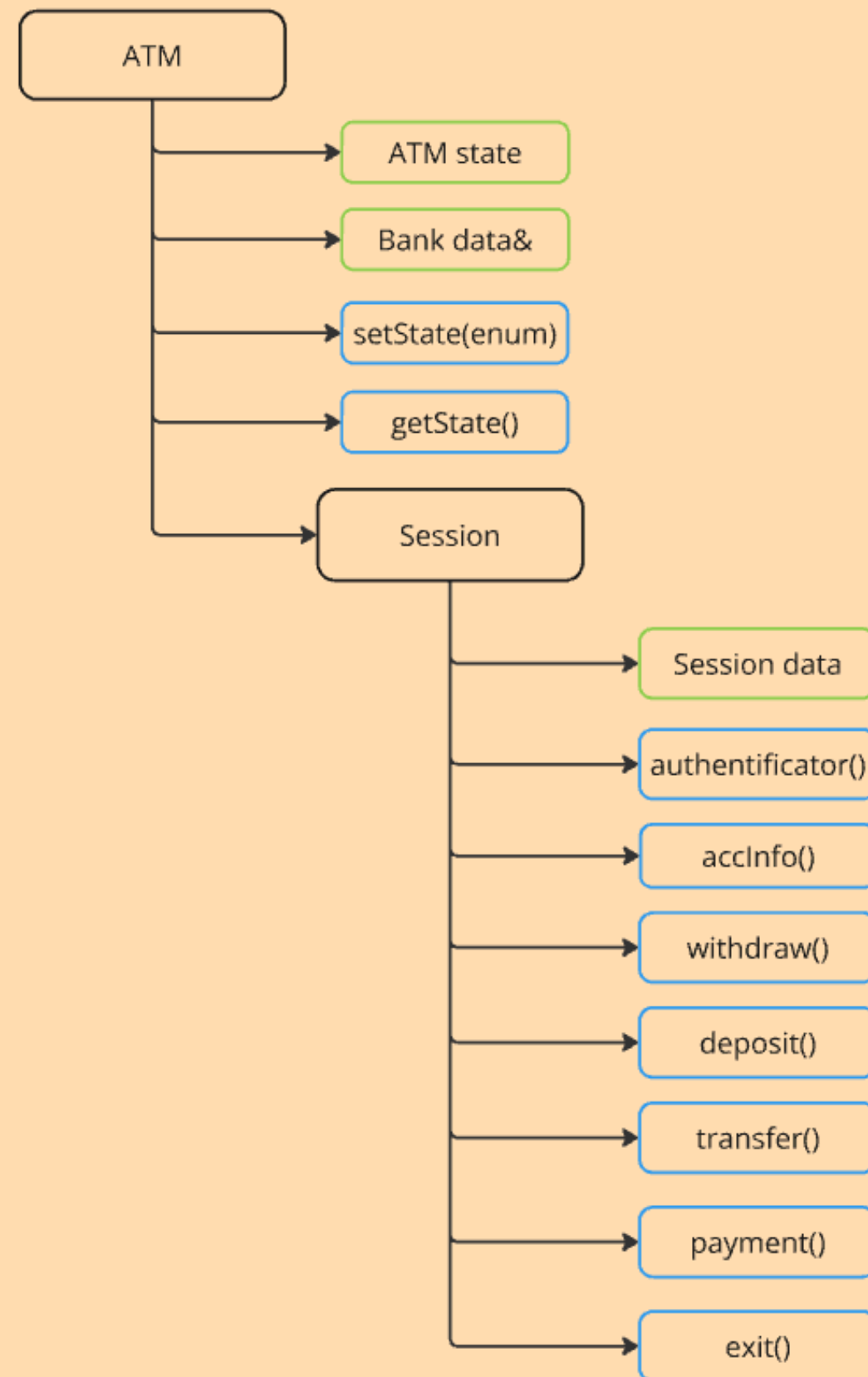


*Взято зі звіту Т1

БУЛО...

АРХІТЕКТУРА

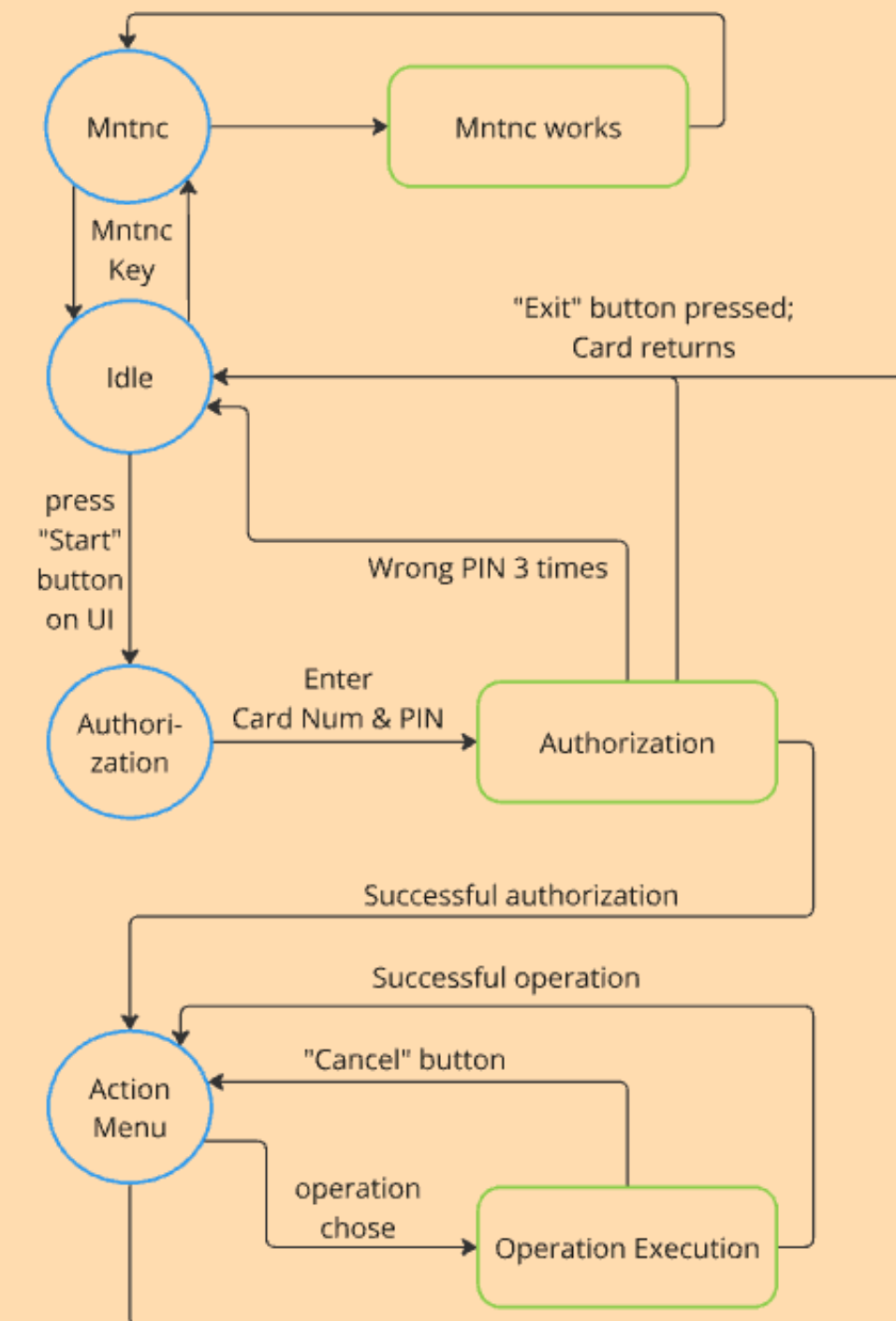
Ієрархія об'єктів



Узагальнена архітектура системи



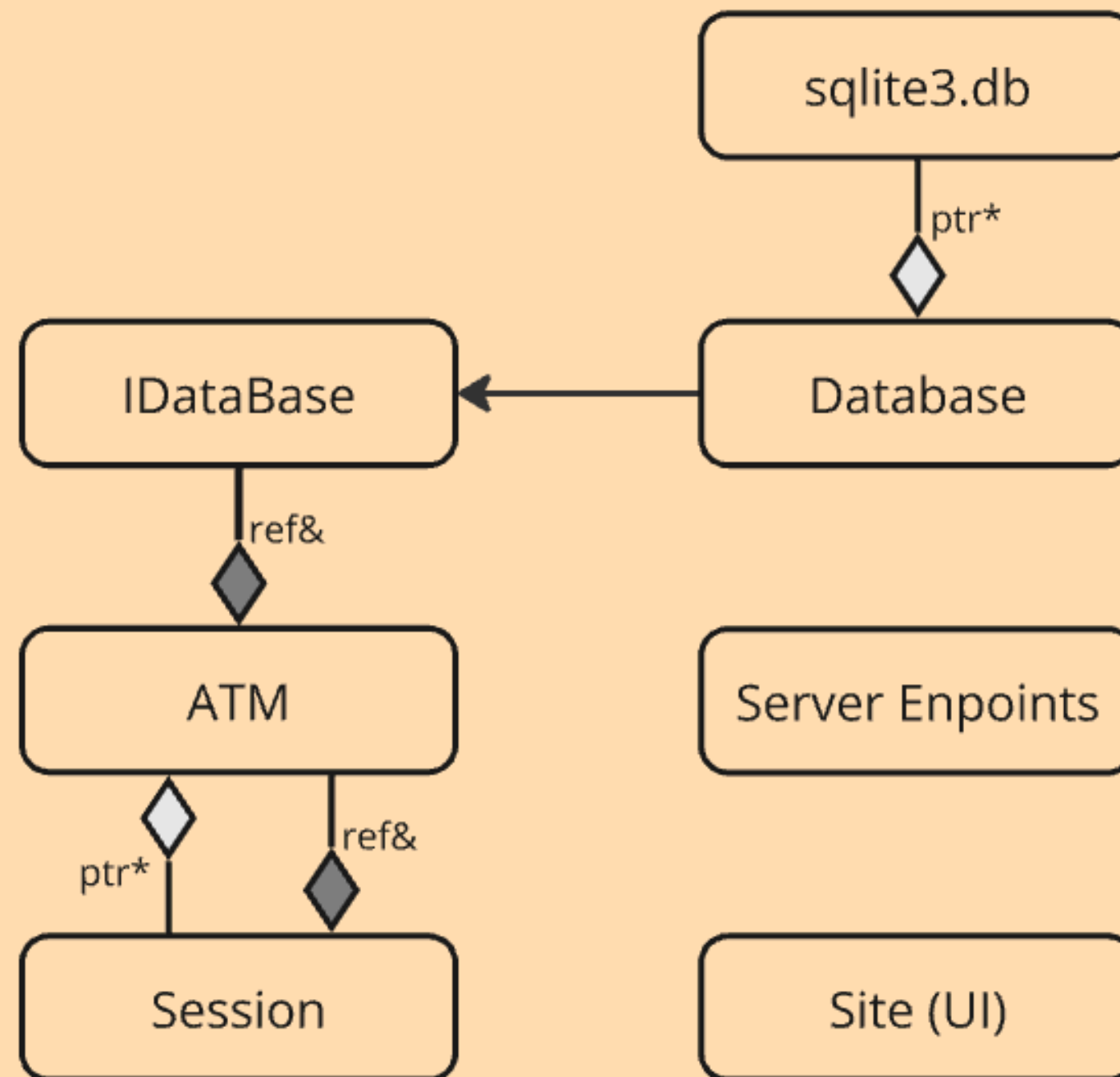
Динамічна модель



*Взято зі звіту Т1

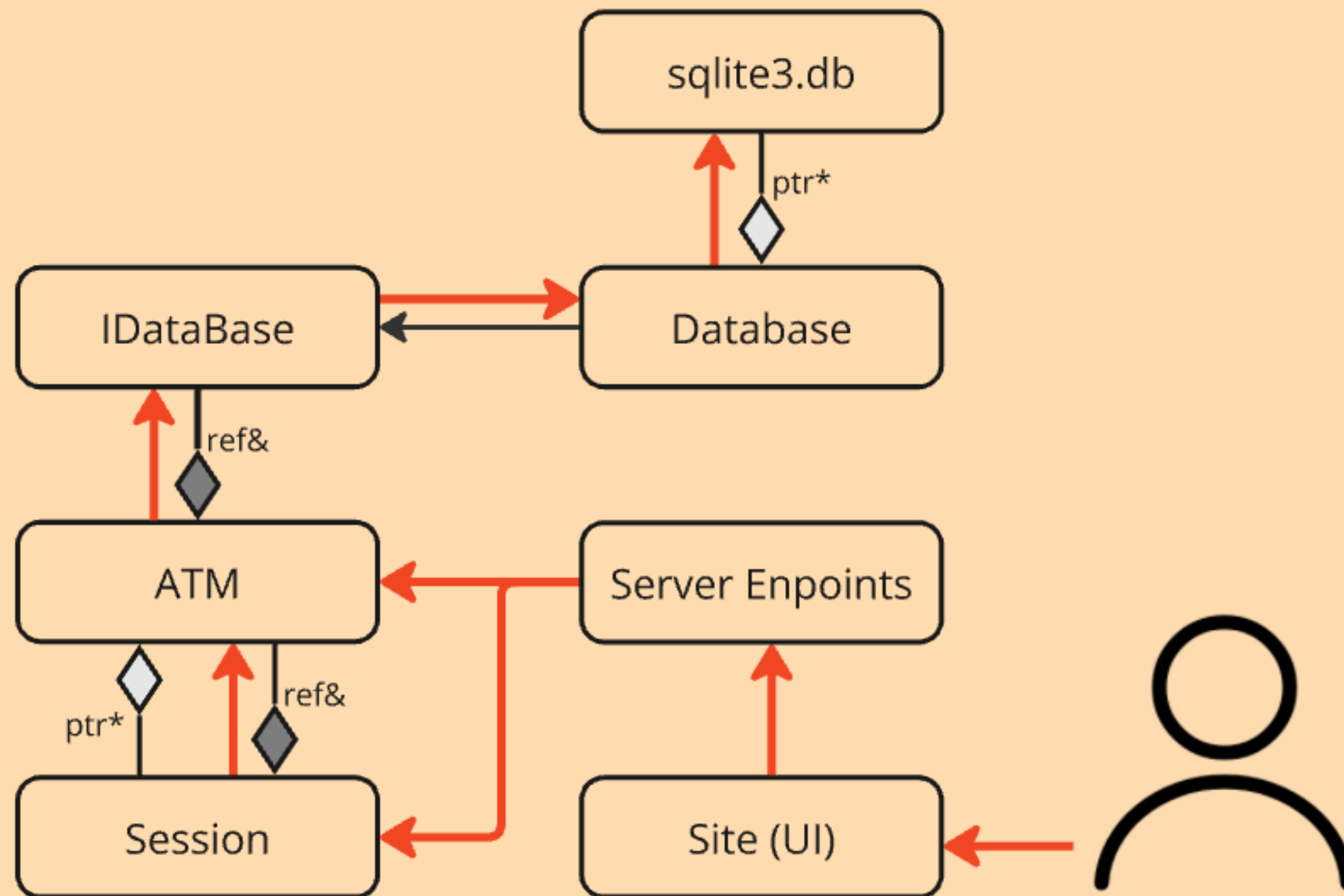
БУЛО...

АРХИТЕКТУРА



...СТАЛО

АРХИТЕКТУРА



APXITEKTYPA

- `class ATM {`
- `public:`
 - `class Session; enum class State; enum class Result;`
 - `ATM(IDataBase& db);`
 - `~ATM();`
 - `ATM::Result start();`
 - `ATM::Result authenticator(const string& cardNum, const string& pin);`
 - `Session* const getSession() { return _currentSession; };`
 - `ATM::Result endSession();`
- `private:`
 - `void setState(State state) { _currentState = state; }`
 - `State getState();`
 - `bool checkInDB(const string& cardNum);`
 - `nlohmann::json getAccInfo(const string& cardNum);`
 - `double getAccBalance(const string& cardNum);`
 - `IDataBase& _bank;`
 - `Session* _currentSession = nullptr;`
 - `State _currentState;`
- `};`

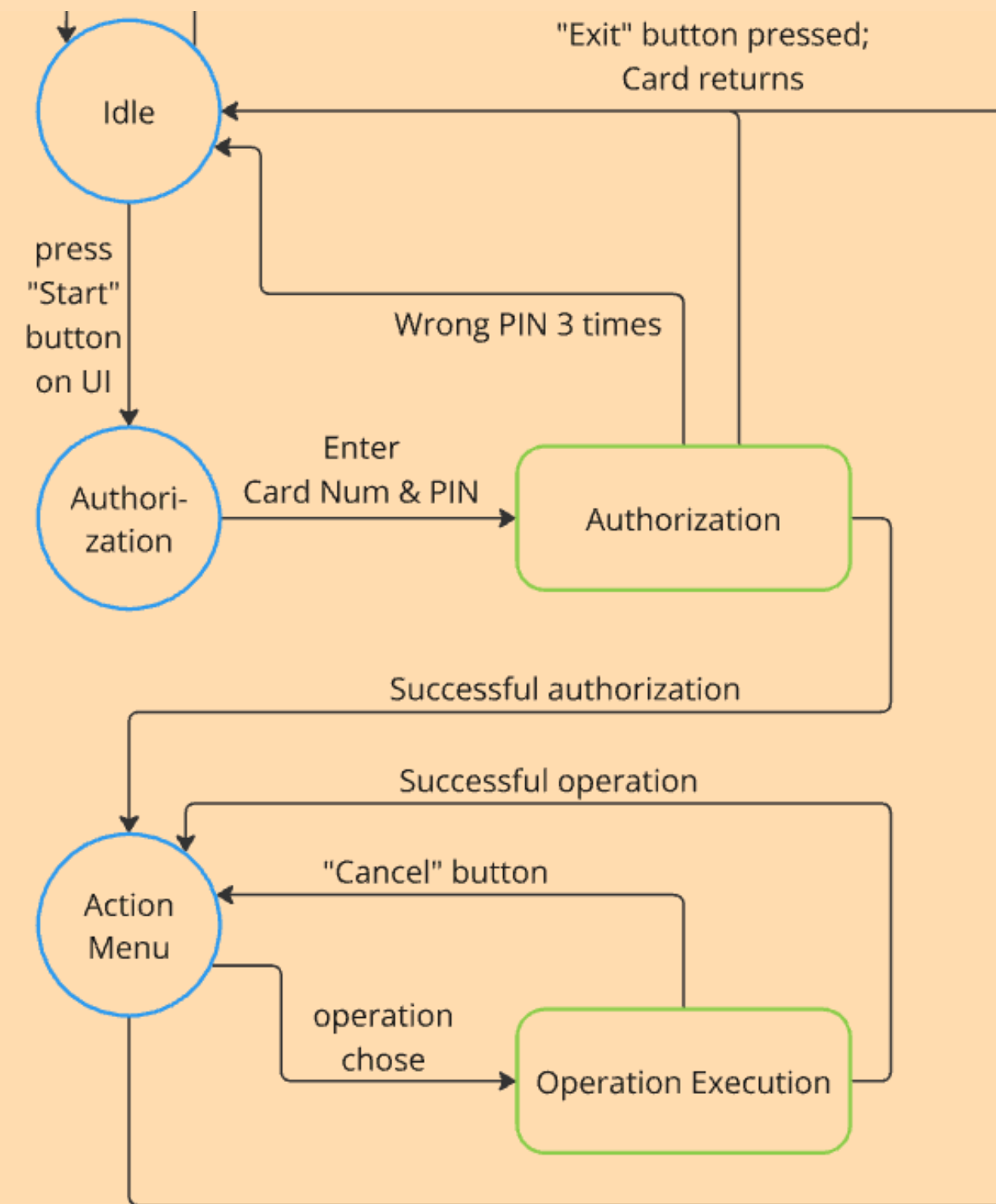
APXITEKTYPA

- `class ATM::Session {`
 - `friend class ATM;`
- `public:`
 - `~Session() {};`
 - `nlohmann::json accInfo() { return _atm.getAccInfo(_cardNum); };`
 - `double accBalance() { return _atm.getAccBalance(_cardNum); };`
 - `ATM::Result withdraw(double);`
 - `ATM::Result deposit(double);`
 - `ATM::Result transfer(const string&, double);`
 - `ATM::Result paymentMenu(const string&, const string&, double);`
- `private:`
 - `Session(ATM& myatm, const std::string& info);`
 - `ATM& _atm;`
 - `const string _cardNum;`
- `};`

APXITEKTYPA

- `class IDatabase {`
 - `friend class ATM;`
- `public:`
 - `virtual ~IDatabase() = default;`
- `private:`
 - `virtual bool insertCard(const string& cardNumber, const string& pin, double balance) = 0;`
 - `virtual bool removeCard(const string& cardNumber) = 0;`
 - `virtual double getCardBalance(const string& cardNumber) = 0;`
 - `virtual nlohmann::json getCardDetails(const string& cardNumber) = 0;`
 - `virtual void addMoney(const string& cardNumber, double amount) = 0;`
 - `virtual void removeMoney(const string& cardNumber, double amount) = 0;`
 - `virtual bool isCardValid(const string& cardNumber) = 0;`
 - `virtual bool isPinCorrect(const string& cardNumber, const string& pin) = 0;`
- `};`

APXITEKTYP A



БЕЗПЕКА СИСТЕМИ

- Немає доступу до БД не через Сесію
- Немає доступу до сесії без авторизації
- Авторизація має подвійну перевірку синтаксису
- Сума має потрібну перевірку синтаксису



БАЗА ДАНИХ

- Слугує міні прикладом реальної банківської бази даних
- Реалізовано за допомогою c++ sqlite3 бібліотеки
- Використано Tiger стиль тестування валідності програми



КОРИСТУВАЦЬКИЙ ІНТЕРФЕЙС (UI)

- UI написаний з використанням html, css, javascript
- Інтерфейс складається з трьох сторінок:
вступна сторінка -> авторизація -> основне меню
- Зв'язок з сервером забезпечується за допомогою
запиту на endpoints, що реалізовані на c++
- Інтерфейс надає зрозумілий дизайн, покликаний
зімітувати роботу банкомату



ЛОКАЛЬНИЙ СЕРВЕР

- Реалізовано за допомогою с++ бібліотеки Crow
- Працює на локальному порті 8000 (localhost:8000)
- Містить ендпоінти:
 - / - початковий ендпоінт
 - /authorization - ендпоінт з формою авторизації
 - /main - ендпоінт оперуванням картою
 - /authorization/send - опрацювання даних
 - /exit - закриття сесії банкомату
 - /acc_info - отримання даних карти
 - /withdraw, /transfer_to_card, /top_up_the_card, /payments - опрацювання грошових запитів користувача

ЧАС ДЛЯ ДЕМО!

