

Министерство образования Республики Беларусь

**Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»**

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

**Отчет
по лабораторной работе N1**

Выполнил:

**Жолнерчик И. А.
Группа 121701**

Проверил:

Бутрин С. В.

Минск 2023

Вариант 1

В соответствии с вариантом задания нужно разработать программу, симулирующую взаимодействие пользователя с банкоматом «Беларусбанк». Для осуществления этой цели был разработан интерфейс командной строки, поддерживающий следующие команды:

- enter_pin
- get_card
- insert_card
- phone_payment
- view_balance
- withdraw_money

Пример реализации команды insert_card:

```
33 @start_atm.command('insert_card')
34 @click.option('-n', '--number', default=-1)
35 def insertCard(number):
36     if atm.isCardInserted():
37         print('Card inserted')
38         return
39     if number == -1:
40         currentNum = 1
41         for card in atm.getCards():
42             print(str(currentNum) + " - " + card.getNumber())
43             currentNum += 1
44     else:
45         try:
46             atm.selectCard(number)
47             print("Success")
48         except ValueError as error:
49             print("Error: ", error)
50     end_work()
```

Для взаимодействия пользователя с банкоматом была создана база данных json (database.json), которая делится на блок InfoBank, в котором описаны пользователи банка, и блок InfoATM, в котором содержится информации о банкомате, а именно: номер вставленной банковской карты (если таковая операция была осуществлена), количество купюр каждого номинала, статус введенного PIN.

Работоспособность программы осуществляется четырьмя файлами:

- ATM.py
- Bank.py

- BankAccount.py
- BanknoteVault.py

В файле ATM.py описаны функции, осуществляющие взаимодействие пользователя с банкоматом. Пример выполнения функции enterPin:

```
39 def enterPin(self, pin):
40     if not self.__insertedCard:
41         raise ValueError('card is not inserted')
42     if self.__insertedCard.getAttempts() >= 3 or self.__insertedCard.isLocked():
43         raise ValueError('card is locked')
44     if self.__isPinEntered:
45         raise ValueError('pin entered')
46     if self.__insertedCard.checkPIN(pin):
47         self.__isPinEntered = True
48         return True
49     else:
50         self.__insertedCard.increaseAttempts()
51         return False
```

Файл Bank.py осуществляет взаимодействие программы с базой данных в файле database.json, эти данные используются в файле BankAccount.py для обработки и последующего использования. Файл BanknoteVault.py необходим для работоспособности вывода наличных через команду withdraw_money, в нём осуществлена проверка наличия купюр каждого номинала в банкомате и последующий вывод наличных в соответствии с выбором пользователя.