Учреждение образования

“БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ”

Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №2

по курсу “ППвИС”

Выполнил студент группы 821701: Залесский А.А.

Проверила Орлова А.С.

**МИНСК**

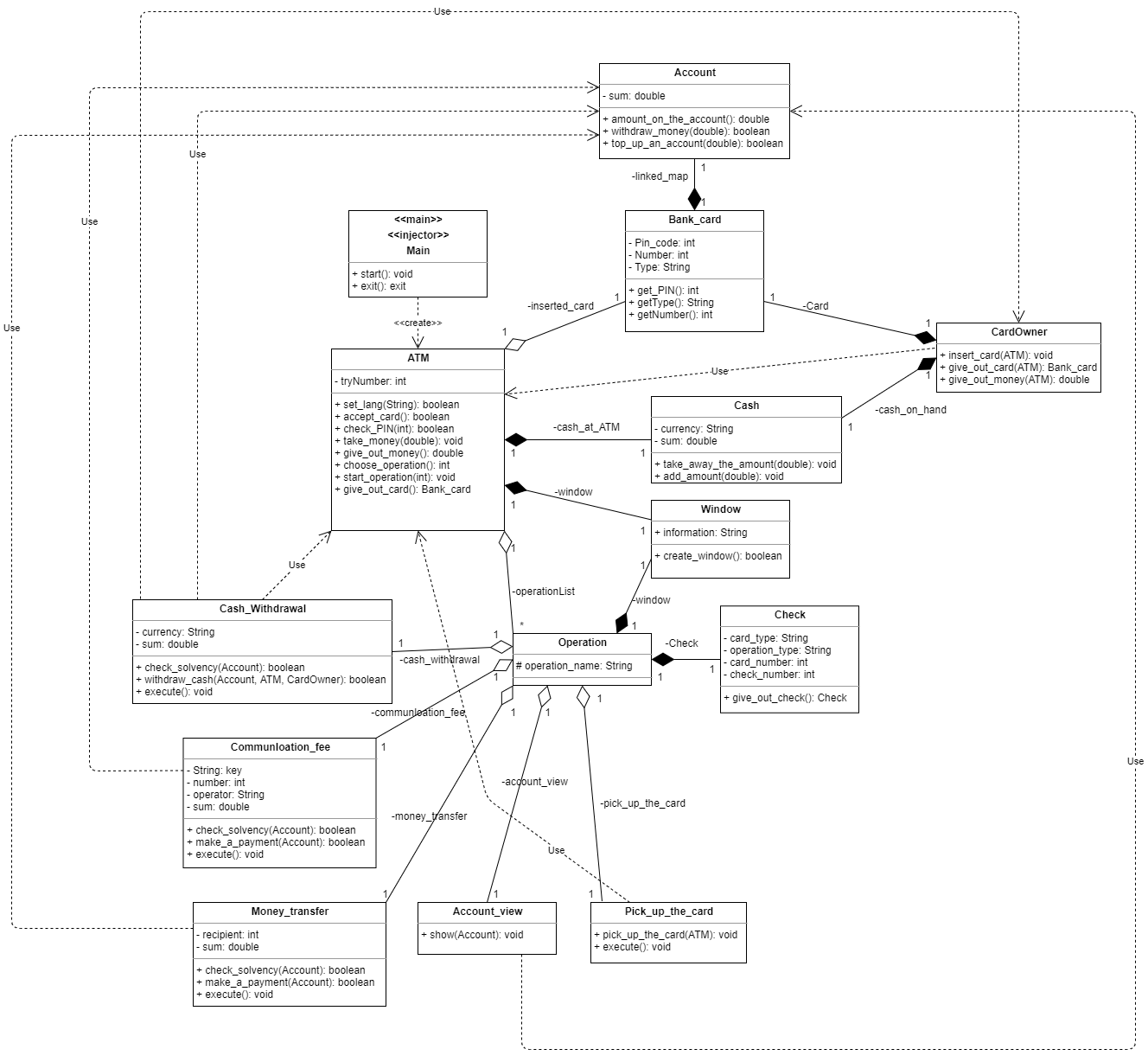
**2020**

**Тема: Модель Банкомата**

**Список ошибок:**

1. В ТЗ рассматривается банкомат со стороны программного обеспечения, хотя по сути мы рассматриваем непосредственно саму программную систему, ее внутреннюю логику.
2. В данном варианте рассматривается симуляция работы с банкоматом, в диаграмме деятельности передаваемые данные, как например PIN-код и многое другое располагаются на границе между пользователем и банкоматом, хотя эти данные по сути не являются чем-то материальным и принадлежащим пользователю, нужно расположить эти данные на стороне банкомата.
3. В диаграммах вариантов использования и деятельности пользователь должен иметь конкретное название, например, владелец карты.
4. На диаграмме классов нет идентификатора у класса со стереотипами <<main>> и <<injector>>.
5. Инжектор создает только ATM, непонятно, кто создает классы communication\_fee, money\_transfer, account\_view, pick\_up\_the\_card, cardowner.
6. У агрегаций от класса operation нет множителей и идентификаторов переменных. То же самое с композицией между ATM и Window.
7. Не понятна логика метода give\_out\_check. Метод возвращает объект класса Check, т. е. класс чек создает чек, но как этот метод узнает о данных, которые должны печататься. Они являются полями этого же класса. Скорее всего подразумевается, что это метод вызывается другим классом, т. к. он public, но тогда метод не может узнать о данных. Если ставить и сеттерами, то тогда не понятно, зачем этот метод тогда вообще нужен.
8. Не понятно, зачем методу take\_money класса cardowner на вход подается double. По идее, ATM должно быть достаточно.
9. Не понятна логика класса Window, как метод create\_window() должен знать, какое окно создавать.
10. Между ATM и Bank\_card есть зависимость Use, у ATM нет ни одного метода, который бы использовал Bank\_card, аналогично с ATM и Operation.
11. Нет класса(-ов) со стереотипом <<UI>>. Не понятно, как без него реализовывать рабочее GUI.
12. Не понятно, как ATM проверяет 4 попытки ввода PIN-кода. Добавлено поле -tryNumber: int для подсчета 4 попыток.
13. Ошибки в диаграммах классов развития, нет никаких абстракций, нарушаются принципы ООП, SOLID.

**Диаграмма классов физического уровня:**



**Вывод:**

Были внесены некоторые изменения в исходную диаграмму классов, однако из-за ошибок, как например отсутствие класса(ов) со стереотипом <<UI>> и непонятная логика некоторых методов, данный программный комплекс не удалось заставить работать по задумке автора.

Был также реализован пользовательский интерфейс программы, однако подвязать его к программе не является возможным из-за той же ошибки с UI.