Министерство образования

Учреждение образования

«Белорусский государственный университет информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра инженерной психологии и эргономики

Отчет по лабораторной работе №1

«Использование языка программирования Kotlin»

Выполнил: Наджмиддинзода У.Х.

Студент группы 310902

Проверил: Усенко Ф. В.

Минск 2024

Цель работы – Выполнить разработку приложения с использованием языка программирования Kotlin

Индивидуальное задание

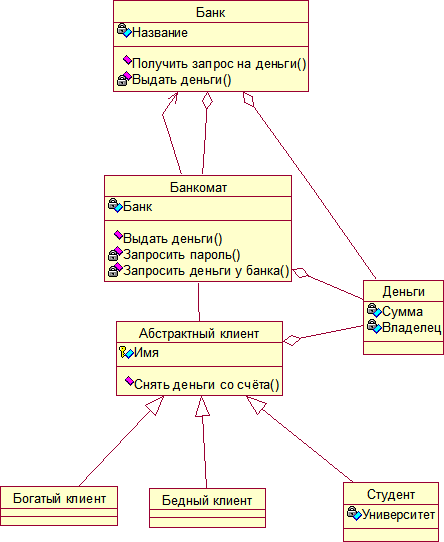
1. Добавить класс Банк и организовать его взаимодействие с классом Банкомат. В частности реализовать запрос денег Банкоматом из Банка. 

Рисунок 1 – Диаграмма классов

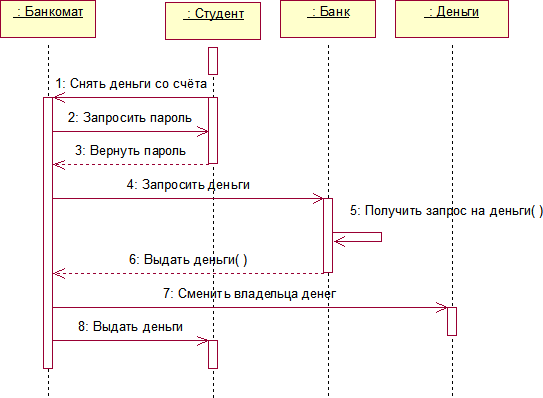


Рисунок 2 – Диаграмма последовательности

Листинг кода:

package files

class Money(private var amount: Double, private var owner: String) {

protected fun changeOwner(newOwner: String) {

owner = newOwner

println("Владелец денег изменен на: $newOwner")

}

fun getAmount(): Double {

return amount

}

fun getOwner(): String {

return owner

}

fun deduct(amountToDeduct: Double) {

if (amountToDeduct <= amount) {

amount -= amountToDeduct

} else {

throw IllegalArgumentException("Недостаточно средств!")

}

}

}

abstract class Client(protected val name: String, private val password: String) {

fun findName(): String {

return name

}

fun authorize(inputPassword: String): Boolean {

return if (inputPassword == password) {

println("Пароль верен.")

true

} else {

println("Неверный пароль.")

false

}

}

}

class Student(name: String, password: String, private val university: String) : Client(name, password)

class Bank {

private fun receiveRequest(amount: Double): Boolean {

println("Банк получил запрос на сумму $amount")

return amount <= 1000

}

protected fun provideMoney(amount: Double, owner: String): Money {

println("Банк выдает сумму $amount владельцу: $owner")

return Money(amount, owner)

}

fun processMoneyRequest(amount: Double, owner: String): Money? {

return if (receiveRequest(amount)) {

provideMoney(amount, owner)

} else {

println("Запрос отклонен: превышен лимит.")

null

}

}

}

class ATM(private val bank: Bank) {

private fun requestMoneyInternal(client: Student, amount: Double): Money? {

println("Банкомат запрашивает сумму $amount от имени клиента: ${client.findName()}")

val money = bank.processMoneyRequest(amount, client.findName())

if (money != null) {

println("Банкомат выдаёт деньги клиенту: ${money.getAmount()}")

} else {

println("Ошибка: банк отклонил запрос.")

}

return money

}

fun withdraw(client: Student, amount: Double, password: String): Money? {

println("Клиент пытается снять $amount через банкомат.")

if (client.authorize(password)) {

return requestMoneyInternal(client, amount)

} else {

println("Ошибка: неверный пароль.")

return null

}

}

}

package files

fun main() {

val bank = Bank()

val atm = ATM(bank)

val student = Student("Иван Иванов", "1234", "Университет")

atm.withdraw(student, 500.0, "1234")

atm.withdraw(student, 500.0, "wrong\_pass")

atm.withdraw(student, 1500.0, "1234")

}

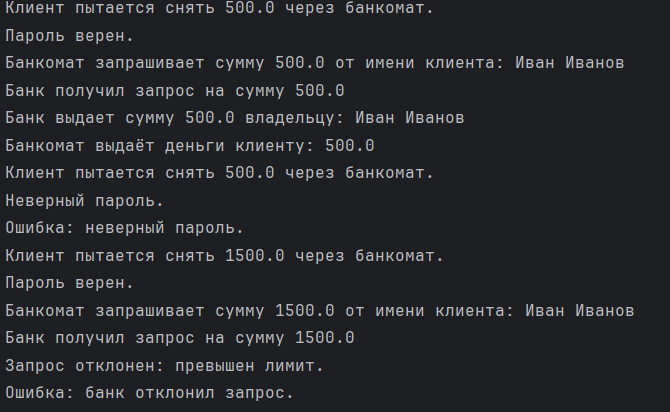


Рисунок 1 – Результат выполнения программы

Вывод: в ходе выполнения лабораторной работы я разработал приложение, используя язык программирования Kotlin.