**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ**

**БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

**Факультет прикладной математики и информатики**

**Кафедра информационных систем управления**

**Отчет**

**По проектированию программных систем**

Выполнил студент группы № 13

*Веренич Владислав Николаевич*

# Минск 2022

**Вариант 4. АИС «Билетный киоск»**

Билетный киоск осуществляет продажу транспортных карт пассажирам городского транспорта. Используются 4 вида карт в зависимости от вида транспорта (автобус, трамвай, троллейбус, метро). В состав киоска входят следующие устройства: информационное табло, кнопки выбора вида транспорта, кнопки выбора ёмкости карт, хранилище денег и лоток для их выдачи, хранилище карт и лоток для их выдачи, принтер для печати чеков, сервисная консоль, линии связи со станцией обслуживания и сервером транспортной компании, устройства для чтения и записи данных на пластиковые билетные карты.

Загрузка карт в автомат осуществляется обслуживающим персоналом. Автомат следит за наличием карт. Если карты какого-либо вида заканчиваются, автомат отправляет сообщение об этом на станцию обслуживания по линии связи и информирует покупателей (зажигается красная лампочка рядом с кнопкой выбора данного вида транспорта).

Автомат принимает к оплате бумажные купюры и монеты. После ввода денег клиент выбирает вид транспорта нажатием на одну из четырёх кнопок. Затем клиент должен выбрать ёмкость карты, перемещаясь по списку при помощи кнопок "Вверх", "Вниз" и "Выбор". Ёмкости бывают: 1 поездка, 2 поездки, 5 поездок, 10 поездок, 20 поездок, 60 поездок, 1 день, 5 дней, 30 дней, 90 дней, 365 дней. После того как пассажир сделал свой выбор, выдача карты производится только в том случае, если имеются бланки выбранных карт, и если введённая сумма денег не меньше цены карты выбранной ёмкости. В этом случае автомат использует бланк карты из лотка, где лежат бланки нужного вида, и записывает данные о времени продажи карты и её ёмкости. Дата продажи карты и её ёмкость печатаются на чеке. Готовая карта выдаётся клиенту. Если введённая клиентом сумма превышает цену карты, вместе с картой выдаётся сдача. За один сеанс 4 клиент может купить не более одной карты.

У киоска имеется информационное табло, на котором высвечивается текущая сумма денег, принятых к оплате, и сообщения для клиентов, такие как: «введите деньги», «выберите вид транспорта», «выберите ёмкость карты», «введённой суммы недостаточно», «закончились бланки», «заберите карту», «заберите сдачу», «заберите деньги», «добавьте сумму ... руб. чтобы получить сдачу».

Купюры и монеты, имеющиеся в распоряжении киоска, хранятся в денежном ящике. Киоск хранит сведения о том, сколько купюр или монет какого номинала есть в ящике. Принимаются купюры по 5, 10, 20 и монеты по 1, 2, рублей, 10, 20, 50 копеек. Возможна ситуация, когда у киоска нет возможности выдать сдачу теми купюрами или монетами, что хранятся в ящике. В этом случае после нажатия на кнопку выдачи не производится выдача, а выводится сообщение, указывающее, минимальную сумму, которую надо добавить клиенту, чтобы киоск смог выдать сдачу. Например, пассажир покупает карту с 60 поездками на метро за 24 рубля, опуская 1 купюру по 20 рублей, 1 купюру по 5 рублей. Пусть в киоске есть только 2 монеты по 2 рубля, а других монет нет. Киоск мог бы выдать сдачу 2 рублевой монетой из денежного ящика. В такой ситуации автомат высвечивает сообщение «Добавьте сумму 1 руб. чтобы получить сдачу.».

При нажатии на кнопку "Возврат" клиенту возвращаются все принятые от него к оплате деньги. Возврат денег не производился после выдачи карты. Киоск должен корректно работать при нажатии на "Возврат" во время выдачи карты.

В течение рабочего дня киоск накапливает данные о проданных картах (тип карты, ёмкость, стоимость, время покупки). В конце рабочего дня эти сведения передаются серверу транспортной компании, после чего удаляются из памяти. Задание включает в себя разработку схемы базы данных о типах карт, ценах, проданных и пополненных картах.

В специальном отделении киоска, закрываемом замком, есть сервисная консоль, которая используется обслуживающим персоналом. С консоли производится управление доступом к ящику с деньгами для изъятия/добавления купюр и монет, управление доступом к лоткам для загрузки или замены бланков карт, а также ввод данных о текущих типах карт и тарифах за проезд в память автомата. Данные включают в себя цену карт всех видов и всех ёмкостей, а также сведения о том, сколько карт в каком лотке находится.

Необходимо осуществлять следующую обработку данных:

- итоговая сумма на текущую дату;

- список проданных карт и их количество (тип, название, сумма )текущую дату;

**В ходе выполнения задания должна быть разработана схема базы данных системы. Для БД:**

Денежная единица - код, номинал, название (монеты или купюры), руб.\коп., количество;

Карты проданные - тип карты, ёмкость, стоимость, время покупки.

Карты в наличии - тип карты, ёмкость, стоимость, количество.

Unit testing

Class TrayMoneyOutput

member function getChange()

| Входные данные | Ожидаемые данные | Действительные данные | Результат |
| --- | --- | --- | --- |
| input sum: 100  price : 100 | 0 | 0 | accept |
| input sum: 100  price : 90 | 10 | 10 | accept |
| input sum: 100  price : 80 | 20 | 20 | accept |

Class Moneyvault

member function getDenominationCount()

Assume this is the vult:

10 penny: 10

20 penny: 30

50 penny: 40

1 ruble: 10

2 ruble: 2

| Входные данные | Ожидаемые данные | Действительные данные | Результат |
| --- | --- | --- | --- |
| getDenominationCount(denomination\_10penny) | 10 | 10 | accept |
| getDenominationCount(denomination\_20penny) | 30 | 30 | accept |
| getDenominationCount(denomination\_50penny) | 40 | 40 | accept |

Integration testing

Функциональные требования:

1. Возможность отмены покупки билета
2. Возможность выбирать транспорта
3. Возможность выбирать емкость
4. Возможность входить в систему под разными пользователями

Нефункциональные требования:

1. Возможность использования системы несколькими пользователями без падения приложения

System testing

class LoginForm

Let’s say admin login and password are the following:

login : verenich

password: vladIsTheBest

| Входные данные | Ожидаемые данные | Действительные данные | Результат |
| --- | --- | --- | --- |
| login : verenich  password: vladIsTheBest | Welcome! | Welcome | accept |
| login : verenich  password: vladIsTheBest | incorrect login or password | incorrect login or password | accept |
| login : verenich  password: vladIsTheBest | incorrect login or password | incorrect login or password | accept |

Acceptance testing

| Действие | **Ожидаемый результат** | **Результат теста** |
| --- | --- | --- |
| Открыть страницу "Вход в систему" | - Окно "Вход в систему" открыто  - Название окна - Вход в систему  - Логотип компании отображается в правом верхнем углу  - На форме 2 поля - Имя и Пароль  - Кнопка Вход доступна  - Ссылка "забыл пароль" - доступна |  |
| Открыть страницу “Купить билет” | - Окно "Купить билет" открыто  - Название окна - Покупка билета  - Логотип компании отображается в правом верхнем углу  - На форме 2 ComboBox - Транспорт, Емкость, 2 кнопки “Подтвердить”, “Отменить покупку” | accept |
| Открыть страницу “Пополнить хранилище карт”(для менеджера по снабжению) | - Окно "Пополнить хранилище карт" открыто  - Название окна - Пополнение хранилища  - Логотип компании отображается в правом верхнем углу  - На форме счетчики кол-ва карт всех типов емкостей.  1 кнопка “Закончить пополнение”. | accept |
| Открыть страницу “Установить тарифы”(для администратора) | - Окно "Установить тарифы" открыто  - Название окна - Тарифы  - Логотип компании отображается в правом верхнем углу  - На форме располагается таблица с 2 ComboBox(для транспорта и для Емкости) при выборе комбинации транспорта и емкости значение в лейбле обновляется и показывается текущая стоимость. Имеется 2 кнопки - “Изменить”, “Подтвердить” | accept |

Scenario testing

1. Проверка функциональности системы при входе в систему под разными пользователями(администратор, менеджер по снабжению)
2. Пополнение слота для карт(менеджер по снабжению)
3. Установка тарифов для проездных документов(администратор)

***Список литературы:***

1. Технологии программирования: Учебник/В.А. Камаев, В.В. Костерин. — 2-е изд., перераб. и доп. — М.: Высш. шк., 2006. - 454 с: ил.

2. Коберн. Современные методы описания функциональных требований. (2002).djvu

3. Леоненков А. Самоучитель UML (2-е издание, 2004).pdf

4. Гради Буч, Айвар Джекобсон, Джеймс Рамбо. UML — основы, 3-е издание

5. Рамбо Джеймс. UML 2: Объектно-ориентированное моделирование и разработка

**6.** А.М. Вендров. Объектно-ориентированный анализ и проектирование с использованием языка UML и Rational Rose (umlpracticum2004.pdf)