**ГОСТ 19.505-79**

1. **Назначение программы**

Сервис подбора банковских карт предназначен для нахождения оптимального банковского продукта (дебетовая карта), используя сведения о расходах пользователя за определенный период времени, а также используемых банковских картах.

1. **Условия выполнения программы**

Для корректной работы программы программы со стороны пользователя необходимо наличие современного браузера с поддержкой html5, js-es6.

Для работы серверной части нужны следующие характеристики.

-Поддержка 64-битного ядра и процессора для виртуализации;

-Поддержка виртуализации KVM;

-Не менее 4 ГБ оперативной памяти.

1. **Выполнение программы**

**Сценарий**  **“Регистрация пользователя”**

**Предусловия:** пользователь заходит на веб сервис и он на нём не зарегистрирован

**Пользовательская история:** как пользователь я хочу зарегистрироваться на веб сервисе

1. Пользователь инициирует открытие формы регистрации
2. Пользователь вводит данные
3. Пользователь нажимает кнопку “создать аккаунт”

**Альтернативный сценарий №1.**

После выполнения шага 3 пользователем, система не может сохранить данные из-за того, что введенный пользователем пароль слишком короткий.

1. Пользователь получает сообщение рядом с полем для ввода пароля: “Минимум 6 символов”.
2. Пользователь вводит подходящий пароль.
3. Возвращение на шаг 3 основного сценария.

**Альтернативный сценарий №2.**

После выполнения шага 3 пользователем, система не может сохранить данные из-за того, что введенный пользователем никнейм уже занят.

1. Пользователь получает сообщение в поле для ввода никнейма: “Этот никнейм уже занят”.
2. Пользователь вводит подходящий никнейм.
3. Возвращение на шаг 3 основного сценария.

**Альтернативный сценарий №3.**

После выполнения шага 3 пользователем, система не может сохранить данные из-за того, что введенный пользователем номер уже занят.

1. Пользователь получает сообщение в поле для ввода номера: “Этот номер уже занят”.
2. Пользователь вводит подходящий номер.
3. Возвращение на шаг 3 основного сценария.

**Альтернативный сценарий №4.**

После выполнения шага 3 пользователем, система не может сохранить данные из-за того, что введенный пароль не совпадает с повторно введенным паролем.

1. Пользователь получает сообщение в поле для ввода номера: “Пароли не совпадают”.
2. Пользователь вводит подходящий пароль.
3. Возвращение на шаг 3 основного сценария.

**Альтернативный сценарий №5.**

На любом этапе пользователь может выйти из формы регистрации, для этого:

1. Пользователь нажимает кнопку “Вернуться”.

**Сценарий**  **“Авторизация пользователя”**

**Предусловия:** пользователь заходит на веб сервис, ранее он был зарегистрирован на нём.

**Пользовательская история:** как пользователь я хочу авторизоваться на веб сервисе.

1. Пользователь заходит на веб сервис
2. Пользователь заполняет поля формы и нажимает кнопку “Вход”

**Альтернативный сценарий №1.**

На шаге 2 основного сценария, система понимает, что введённые пользователем данные не верные.

1. Пользователь получает уведомление “Почта/пароль не совпадают”.
2. Пользователь вводит верные данные.
3. Возвращение на шаг 2 основного сценария.

**Альтернативный сценарий №2.**

На шаге 2 основного сценария пользователь понимает, что забыл ранее не был зарегистрирован в веб сервисе.

1. Пользователь нажимает “Регистрация”.

**Сценарий**  **“Авторизация пользователя с ролью “Оператор””**

**Предусловия:** пользователь был ранее зарегистрирован на веб сервисе и ему присвоена роль “Оператор”.

**Пользовательская история:** как пользователь, которому присвоена роль “Оператор”, я хочу авторизоваться на веб сервисе.

1. Пользователь заходит на веб сервис
2. Пользователь заполняет поля формы и нажимает кнопку “Вход”.

**Сценарий**  **“Взаимодействие с реестрами”.**

**Предусловия:** пользователь, авторизован с ролью “Оператор”.

**Пользовательская история:** как пользователь, которому присвоена роль “Оператор”, я хочу провести провести взаимодействие с одним из реестров: “Банки”, “Категории”, “Карты” или “Магазины”.

1. Пользователь открывает выбранный реестр.
2. Пользователь вводит в поисковую строку название записи.

**Альтернативный сценарий №1.**

После выполнения шага 1 основного сценария пользователь желает добавить новую запись.

1. Пользователь нажимает “Добавить”.

**Альтернативный сценарий №2.**

После выполнения шага 1 основного сценария пользователь желает изменить существующую запись.

1. Пользователь выбирает запись и нажимает “Изменить”.

**Альтернативный сценарий №3.**

После выполнения шага 1 основного сценария пользователь желает удалить существующую запись.

1. Пользователь выбирает запись и нажимает “Удалить”.

**Сценарий “Взаимодействие со списком операций пользователя”.**

**Предусловия:** пользователь авторизован.

**Пользовательская история:** как пользователь веб сервиса я хочу провести взаимодействие со списком своих операций.

1. Пользователь инициирует открытие списка операций.
2. Пользователь нажимает кнопку “Добавить”.

**Альтернативный сценарий №1.**

После выполнения шага 1 основного сценария пользователь решает удалить имеющуюся запись.

1. Пользователь ставит “галочку” слева от выбранной записи и нажимает кнопку “Удалить”.
2. Пользователь подтверждает удаление.

**Альтернативный сценарий №2.**

После выполнения шага 1 основного сценария пользователь решает загрузить список операций из файла с банковской выпиской.

1. Пользователь нажимает кнопку “Загрузить”
2. Пользователь нажимает кнопку “Выбрать файл”.
3. Пользователь выбирает файл и нажимает “Загрузить”.

**Сценарий “Подбор банковских карт”.**

**Предусловия:** пользователь авторизован.

**Пользовательская история:** как пользователь веб сервиса я хочу подобрать банковские карты на основе своих банковских операций.

1. Пользователь нажимает кнопку “Подобрать карты”.

**Альтернативный сценарий №1.**

Если ранее для пользователя уже были подобраны карты, то после шага 1 основного сценария.

1. Пользователь нажимает “Подобрать”.

**Глоссарий**

**MCC** - код, представляет собой 4-значный номер, классифицирующий вид деятельности торговой точки в операции оплаты по банковским картам в торгово-сервисном предприятии при электронной передаче информации в рамках транзакции за предоставляемые товары или услуги.

**Виртуализация KVM** - технология аппаратной виртуализации, позволяющая создать на хост-машине полный виртуальный аналог физического сервера.

**Html5** - язык для структурирования и представления содержимого интернет страниц. Современный стандарт, доступен во всех современных браузерах.

**Js-es6** - ECMAScript 2015. Версия языка JavaScript, которая вышла в 2015.

**Тест кейс** - это форма записи проверки, которую проводит тестировщик.