Функциональные характеристики ПО "Хайс"

1. Назначение ПО

Программное обеспечение предоставляет API для управления банковскими продуктами лиц, имеющих статус индивидуального предпринимателя на территории РФ. С помощью ПО можно осуществлять управление счётом организации (счёт ИП), а также счётом физического лица и сопутствующими банковскими продуктами.

ПО устанавливается на сервера банка партнёра с имеющейся банковской лицензией и интегрируется с его внутренними системами, такими как Автоматизированная Банковская Система (АБС), Процессинг, Финмониторинг и другими.

Далее ПО выступает в качестве промежуточного слоя между интерфейсом для клиентов и внутренними системами банка.

Клиенты банка используют функциональность ПО посредством отдельного интерфейса в виде мобильного приложения или веб-сайта, которые не входят в состав ПО. Интерфейс предоставляет пользователю визуальный метод отображения и управления финансовой информацией, который в свою очередь использует API ПО.

Перечень функциональности ПО

Регистрация/Аутентификация

- Регистрация нового пользователя в системе
- Аутентификация существующего в системе пользователя
- Механизм защиты от перебора паролей и смс-кодов

Процесс открытия расчётного счёта (ОРС)

- Поиск и выбор компании для открытия счёта в банке
- Загрузка документов об ИП и физическом лице в банк
- Отправка компании на проверку в службы безопасности банка
- Согласование встречи со специалистом банка для заключения договора и идентификации клиента
- Активация счёта ИП
- Активация счёта физического лица и карты, привязанной к этому счёту

Просмотр информации по счетам

- Получение списка счетов с балансом и реквизитами
- Получение списка операций по счёту
- Фильтрация и поиск операций по счёту

Платежи

- Отправка платежей по реквизитам
- Отправка платежей между своими счетами
- Повторение платежа
- Отправка платежа по УИН
- Отправка платежей с карты на карту

Чат со службой поддержки

- Отправка/получение сообщений

Управление картой

- Просмотр данных карты
- Установка/смена ПИН-кода карты
- Блокировка/разблокировка карты
- Перевыпуск карты

Выписки по счетам

- Генерация выписки по счёту ИП
- Генерация выписки по счёту физического лица

Документы для работы с контрагентами

- Выставление счёта в формате PDF
- Повторение и редактирование выставленных счетов
- Генерация акта по счёту в формате PDF

Кэшбэк

- Отображение накопленного кэшбэка
- Расчёт и выплата кэшбэка

2. Принципы взаимодействия с API

API работает по протоколу HTTP, тело запросов и ответов передается в формате JSON в соответствии с документацией к вызываемым методам.

Документация API методов выдаётся по запросу.

Методы API можно разделить на две группы: методы неавторизованной зоны и методы авторизованной зоны.

- В неавторизованной зоне находятся методы для прохождения регистрации и аутентификации, которые подтверждают личность пользователя и выдают ему временный токен доступа `access_token`.
- В авторизованной зоне пользователь действует от своего имени, передавая токен в заголовке Authorization HTTP запроса:

```
Authorization: Bearer ACCESS_TOKEN
```

В целях безопасности выданный токен работает в течение 5 минут, затем его необходимо перевыпустить.

Подробнее в разделе "Регистрация/Аутентификация".

3. Работа с файлами

ПО поддерживает хранилище типа S3 для хранения файлов пользователей. Работа с файлами происходит следующим образом:

- в API ПО передаются только идентификаторы файлов, вместо самих файлов
- Чтобы передать новый файл в API ПО, необходимо загрузить его в хранилище, а в API передать полученный от хранилища идентификатор
- Если в ответе от ПО фигурирует идентификатор файла, то соответствующий файл можно скачать из хранилища по идентификатору

Работа хранилища обеспечивается внешними инструментами, поэтому хранилище не входит в состав ПО.

На демо-стенде работа с файловым хранилищем имитируется: методы для скачивания файлов не предусмотрены, однако передача идентификаторов файлов поддерживается.

4. Тестирование API

Для тестирования API необходимо произвести установку ПО в соответствии с инструкциями из файла "Инструкция по установке ПО".

Далее необходимо проверить работоспособность каждой функциональности из "Переченя функциональности ПО". Их работоспособность проверяется посредством вызова API методов в определенной последовательности, которые формируют сценарии взаимодействия пользователя с банком.

Помимо позитивных сценариев (когда пользователь выполняет корректные действия), следует протестировать негативные сценарии (когда пользователь выполняет действия в неправильной последовательности, пытается получить доступ к информации, которая ему не принадлежит, итп).

Сценарии для тестирования представлены в соответствующих разделах данного файла.

Тестирование можно провести в автоматическом режиме или вызывая методы по одному, вручную.

Автоматическое тестирование

Сценарии перечисленные ниже, в разделах посвященных конкретным функциональностям, реализованы в виде автоматизированных тестов. С их помощью можно значительно ускорить тестирование.

При инициализации тесты запускают сервер ПО и заполняют таблицы Базы Данных строками, которые пригодятся в сценариях. Далее тесты отправляют HTTP запросы на сервер, имитируя поведение пользователя, и сверяют ответы сервера с ожидаемыми. Таким образом, сервер функционирует независимо от тестов.

Тесты расположены в папке tests/integration_tests в следующих файлах:

- `conftest.py` - инициализация тестов
- `test_opening.py` - сценарии разделов "Регистрация/Аутентификация" и "Процесс открытия расчётного счёта (ОРС)"
- `test_account.py` - сценарии разделов "Просмотр информации по счетам" и "Кэшбэк"
- `test_payments.py` - сценарии раздела "Платежи"
- `test_chat.py` - сценарии разделов "Чат со службой поддержки" и "Выписки по счетам" (выписки отправляются в чате)
- `test_card.py` - сценарии раздела "Управление картой"
- `test_invoices.py` - сценарии раздела "Документы для работы с контрагентами"

Тесты реализованы на языке Python с использованием пакета pytest. Для запуска тестов необходимо произвести установку ПО в соответствии с инструкциями из файла "Инструкция по установке ПО". Затем, в консоли предоставленного образа ПО произвести запуск тестов командами

```
cd /opt/demo-server
docker-compose exec app pytest
```

В результате запуска тестов будет указано количество пройденных тестов, предупреждений и ошибок при запуске тестов. Тесты считаются успешно пройденными, если пройдены все тесты без ошибок с любым количеством предупреждений. На момент написания этого файла имеется 56 тестов.

Наличие ошибки сигнализирует о неправильной установке сервера. В этому случае следует повторить установку и запустить автоматическое тестирование заново. Если это не поможет - обратиться за помощью в сервисную службу.

Во время работы сервер пишет логи. По этим логам можно отслеживать действия сервера:

```
cd /opt/demo-server
docker-compose logs
```

⚠️ Запуск тестов приводит к пересозданию таблиц в базе данных. После тестирования БД остается пустой.

Для возобновления работы сервера после запуска автоматического тестирования необходимо удалить текущую сборку и произвести сборку и запуск ПО заново:

```
cd /opt/demo-server
```

```
docker-compose rm -f -s -v && docker-compose up --build --force-recreate --detach
```

Ручное тестирование

Любую функциональность можно протестировать вручную, отправляя HTTP запросы по сценариям, перечисленным ниже, в разделах посвященных конкретным функциональностям.

HTTP запросы можно отправлять с любого компьютера, у которого есть сетевой доступ до сервера ПО, либо напрямую с той же машины, на которой запущен сервер ПО. Например:

```
curl -I http://localhost/ping
```

HTTP запросы можно отправлять с помощью любого инструмента для отправки HTTP запросов. Для упрощения работы с временным токеном доступа рекомендуется использовать HTTP клиенты со встроенными функциями авторизации, например Postman.

На демо стенде существует пользователь с предзаполненной информацией:

- Номер телефона: `+79000000002`
- Пароль: `Test12345`

5.1. Регистрация/Аутентификация

Аутентификация использует схему OAuth 2.0.

ПО поддерживает только 2х факторную Аутентификацию. В качестве логина используется номер телефона пользователя, в качестве первого фактора СМС код, в качестве второго - пароль.

Например:

- Номер телефона: `+79000000002`
- Пароль: `Test12345`

На демо стенде можно использовать следующие СМС коды:

- `1111` для успешного ввода кода
- `2222` для получения ответа "код протух"
- `3333` для получения ответа "превышено допустимое количество попыток ввода кода"
- любой другой код для получения ответа "код неверный"

При регистрации пользователь проходит только 1й фактор аутентификации, а затем задает пароль. При успешной установке пароля происходит аутентификация.

При успешном прохождении аутентификации пользователь получает 2 токена: `refresh_token` и `access_token` :

- `access_token` работает в течение 5 минут, затем становится недействительным
- `refresh_token` активен в течение 180 дней после последнего применения, затем становится недействительным, а также отзывается в случае совершения выхода из приложения (logout)
- для перевыпуска `access_token` используется `refresh_token`
- для обращения к методам авторизованной зоны используется `access_token` (см. "Принципы взаимодействия с API")

Сценарий регистрации

Основной сценарий:

1. Ввод номера телефона и запроса СМС-кода
2. Подтверждение кода и получение токена для установки пароля
3. Установка пароля и получение `refresh_token + access_token`

Негативные сценарии:

- Ввести неверный номер телефона
- Ввести неверный код при подтверждении кода
- Установить невалидный пароль или передать неправильный токен для установки пароля

Сценарий аутентификации

Основной сценарий:

1. Ввод номера телефона и запроса СМС-кода
2. Подтверждение кода и получение токена для ввода пароля
3. Ввод и подтверждение пароля и получение `refresh_token + access_token`
4. Применение `refresh_token` для обновления `access_token`
5. Применение `access_token` для вызова любого метода авторизованной зоны
6. Логаут

Негативные сценарии:

- Ввести неверный номер телефона
- Ввести неверный код при подтверждении кода
- Ввести невалидный пароль или передать неправильный токен для ввода пароля
- Применить невалидный или протухший `refresh_token` для обновления `access_token`
- Применить невалидный или протухший `access_token` для вызова любого метода авторизованной зоны
- Применить `refresh_token` после логаута

Сценарий защиты от перебора паролей и смс-кодов

Перебор смс-кодов:

1. Ввод номера телефона и запроса СМС-кода по сценарию регистрации или аутентификации
2. Ввести неверный смс-код более 10 раз и получить ошибку

Перебор паролей:

1. Ввод номера телефона и запроса СМС-кода по сценарию регистрации или аутентификации
2. Ввести верный смс-код и получить токен для ввода пароля
3. Ввести неверный пароль более 10 раз и получить ошибку

5.2. Процесс открытия расчётного счёта (ОРС)

После успешной регистрации клиент приступает к процессу ОРС.

На первом шаге клиенту необходимо выбрать организацию и отправить её на проверку банка.

После успешного проверки клиенту открывается возможность загрузки документов и назначения встречи для подтверждения личности. Можно провести очную встречу или онлайн (видео-встречу).

Для назначения очной встречи необходимо выбрать город и адрес для проведения встречи. Клиенту будут предложены доступные слоты времени для проведения встречи.

Также клиент может отправить запрос на видео-встречу, после чего с ним свяжется сотрудник банка.

После проведения встречи и обмена документами, банк активирует счёт ИП. С этого момента клиент может использовать его для проведения банковских операций и получений услуг банка. На демо стенде, ввиду отсутствия сотрудников банка, счёт ИП активируется сразу после прохождения встречи и загрузки документов.

После активации счёта ИП клиент может активировать счёт физического лица (ФЛ), выпустив виртуальную карту.

Сценарий открытия расчётного счёта

Сценарий активации счёта ИП

1. Поиск компании по названию или ИНН
2. Отправка компании на проверку в службы безопасности банка
3. Назначение встречи (доступны 2 варианта)
    i. Назначение видео-встречи:
        a. Вызов метода для назначения видео-встречи
    ii. Назначение очной встречи:
        a. Выбор города из списка предложенных
        b. Указание адреса проведения встречи
        c. Выбор доступного слота времени для проведения встречи
4. Загрузка документов, необходимых для активации счёта
5. Активация счёта ИП на демо стенде происходит автоматически после назначения встречи и загрузки документов

Сценарий активации счёта ФЛ

1. Установка секретного слова
2. Ввод имени, эмбоссируемого на карте
3. Установка пин-кода (Метод описан в разделе `Управление картой`)

5.3. Просмотр информации по счетам

ПО поддерживает 2 типа компаний:

- `individual` - компания ИП
- `personal` - компания ФЛ

У каждой компании может быть не более одного активного счёта.

Клиент может просматривать актуальную информацию по своим компаниям и счетам, включая реквизиты, баланс и данные привязанной карты. А также запрашивать ленту операций по каждому из счетов по отдельности, применяя различные фильтры.

Сценарий просмотра информации по счетам

Основной сценарий:

1. Запросить список компаний
2. Выбрать счёт компании с типом `individual` , запросить список операций
3. Применить различные фильтры при запросе списка операций
4. Выбрать счёт компании с типом `personal` , запросить список операций
5. Применить различные фильтры при запросе списка операций
6. Применить пагинацию (используя параметры `limit` для ограничения числа операций и `offset` для сдвига)

5.4. Платежи

ПО поддерживает создание и отправку платежей:

- по реквизитам,
- между счетами,
- по УИН,
- по номеру карты,
- а также повтор уже созданных платежей.

На демо-стенде все платежи проходят первичную валидацию и считаются проведенными автоматически.

Сценарии по работе с платежами

Сценарий отправки платежа по реквизитам:

1. Отправить платеж по реквизитам
2. Запросить ленту по счёту, проверить появилась ли операция

Сценарий отправки платежа между счетами:

1. Отправить платеж между своими счетами
2. Запросить ленту операций по счёту списания, проверить появилась ли операция
3. Запросить ленту операций по счёту зачисления, проверить появилась ли операция

Сценарий повтора платежа:

1. Запросить ленту, выбрать платеж с флагом `repeatable : True`
2. Повторить выбранный платеж
3. Запросить ленту операций, проверить появилась ли операция

Негативный сценарий повтора платежа:

1. Запросить ленту, выбрать платеж с флагом `repeatable : False`
2. Повторить выбранный платеж
3. Убедиться в получении ошибки

Сценарий отправки платежа по УИН:

1. Отправить платеж по УИН
2. Получить успешный ответ. На демо-стенде не предусмотрена отправка платежа

Негативный сценарий отправки платежа по УИН:

1. Отправить платеж по УИН, указав УИН неверного формата
2. Убедиться в получении ошибки

Сценарий отправки платежей с карты на карту:

1. Отправить платеж по номеру карты
2. Запросить ленту операций, проверить появилась ли операция

5.5. Чат со службой поддержки

Чат поддерживает обмен текстовыми сообщениями и файлами между клиентами и сотрудниками службы поддержки банка.

Чат предоставляет методы по отправке сообщений от имени клиента и по получению истории сообщений для отображения диалога.

Система работы сотрудников службы поддержки не включена в ПО, поэтому на демо-стенде отправка ответов от имени оператора службы поддержки не предусмотрена. Однако можно отправить сообщение от банка клиенту, запросив выписку (см. раздел "Выписки по счетам"). Также в истории сообщений клиента с номером телефона +79000000000 сообщения от операторов созданы в демонстрационных целях.

Сценарии тестирования

Основной сценарий

1. Запросить историю чата
2. Отправить сообщение в чат от имени клиента
3. Запросить историю чата, убедиться что отправленное сообщение появилось в списке

Негативный сценарий запроса истории чата

1. Запросить историю чата, убедиться в получении ответа
2. Запросить историю чата, применив фильтра before_id, указав значение больше максимального идентификатор, убедиться в получении пустой истории сообщений
3. Запросить историю чата, применив фильтра after_id, указав значение меньше максимального идентификатор, убедиться в получении пустой истории сообщений

5.6. Управление картой

По требованиям эмитентов карт данные карты должны храниться в защищённом контуре банка, и запрашиваться из защищённого контура банка соответствующего требованиям PCI DSS. Передача номеров карт и прочих данных карты через ПО запрещена. На демо-стенде взаимодействие с этим контуром эмулируется.

Для авторизации запроса в защищённый контур, необходим специальный токен, логика его проверки так же эмулируется.

Общий алгоритм взаимодействия с защищённым контуром:

1. запросить у ПО кратковременный (30 секунд) одноразовый токен для совершения запроса в защищённый контур,
2. ПО выпустит токен и сообщит его защищённому контуру,
3. применить токен в запросе к защищённому контуру в течение 30 секунд.

Действия доступные в защищённом контуре:

- просмотр данных карты
- перевыпуск карты
- блокировка карты
- разблокировка карты
- установка ПИН-кода карты.

Сценарии тестирования

Сценарий перевыпуска карты

1. Запросить токен для взаимодействия с методами карт
2. Запросить данные карты
3. Запросить код для перевыпуска карты
4. Перевыпустить карту
5. Запросить новый токен для взаимодействия с методами карт
6. Запросить данные карты, убедиться что они обновились

Негативный сценарий перевыпуска карты

1. Перевыпустить карту не передав токен, убедиться в получении ошибки
2. Перевыпустить карту передав неверный токен, убедиться в получении ошибки

Сценарий блокировки, разблокировки карты

1. Запросить токен для взаимодействия с методами карт
2. Вызвать метод для блокировки карты, убедиться в получении успешного ответа
3. Вызвать метод для блокировки карты, убедиться в получении неуспешного ответа
4. Вызвать метод для разблокировки карты, убедиться в получении успешного ответа
5. Вызвать метод для разблокировки карты, убедиться в получении неуспешного ответа

Негативный сценарий блокировки, разблокировки карты

1. Запросить токен для взаимодействия с методами карт
2. Вызвать метод для блокировки карты, убедиться в получении успешного ответа
3. Вызвать метод для блокировки карты повторно, убедиться в получении неуспешного ответа

4. Вызвать метод для разблокировки карты, убедиться в получении успешного ответа
5. Вызвать метод для разблокировки карты повторно, убедиться в получении неуспешного ответа
6. Вызвать метод для блокировки карты не передав токен, убедиться в получении ошибки
7. Вызвать метод для блокировки карты передав неверный токен, убедиться в получении ошибки

Сценарий установки ПИН-кода

1. Запросить токен для взаимодействия с методами карт
2. Вызвать метод для установки ПИН-кода, убедиться в получении успешного ответа

Негативный сценарий установки ПИН-кода

1. Вызвать метод для установки ПИН-кода не передав токен, убедиться в получении ошибки
2. Вызвать метод для установки ПИН-кода передав неверный токен, убедиться в получении ошибки

5.7. Выписки по счетам

Для запроса на генерацию выписки по счёту необходимо отправить запрос с указанием периода, за который необходима выписка.

По готовности выписка приходит в сообщении в чате.

Сценарий генерации выписки

Основной сценарий:

1. Запросить выписку по счёту
2. Запросить чат, убедиться в получении сообщения с выпиской (см. раздел Чат)

5.8. Документы для работы с контрагентами

Клиенту доступен функционал генерации документов для взаимодействия с контрагентами, а также их редактирования.

Чтобы выставить счёт, необходимо передать список проданных товаров и оказанных услуг, а также реквизиты контрагента.

По этим данным будут сгенерированы 2 `pdf` файла: счёт и акт, при редактировании оба файла обновляются.

Сценарии работы с документами для контрагентов

Сценарий выставления счёта:

1. Выставить счёт
2. Запросить выставленный счёт и акт, убедиться в корректности данных

Сценарий редактирования счёта:

1. Выставить счёт
2. Запросить выставленный счёт и акт, убедиться в корректности данных
3. Отредактировать счёт
4. Запросить отредактированные счёт и акт, убедиться в корректности данных

Негативный сценарий редактирования счёта: 3. Отредактировать счёт с несуществующим идентификатором, убедиться в получении ошибки

5.9. Кэшбэк

Кэшбэк - это бонус, начисляемый за совершение клиентом покупок по карте.

Расчёт и выплата кэшбэка

Кэшбэк начисляется на баланс накопленного кэшбэка автоматически при совершении пользователем покупки в торговой точке с кодом мерчанта из утвержденного списка по правилам описанным в условиях программы лояльности «КШБК»:

https://modulbank.ru/fs/files/Hice_Cashback.pdf

Выплата происходит автоматически в конце расчётного периода по счёту. Пользователь получает на счёт входящую операцию, равную сумме накопленного к текущему моменту кэшбэка, баланс кэшбэка при этом сбрасывается.

Отображение накопленного кэшбэка

Кэшбэк по каждой операции передаётся вместе с данными этой операции (см. Получение списка операций по счёту в разделе "Просмотр информации по счетам").

Баланс накопленного к текущему моменту кэшбэка передаётся в ответе на запрос со списком счетов (см. Получение списка счетов с балансом и реквизитами в разделе "Просмотр информации по счетам").

6. Техническая поддержка ПО (сервисная служба)

Вопросы возникающие в ходе работы с ПО следует направлять в службу поддержки по адресу tech-support@hicebank.ru.

Обращения рассматриваются в сроки, установленные в договоре с банком-партнёром. Неполадки в работе ПО устраняются в сроки, установленные в договоре с банком-партнёром.

Подробности о технической поддержке и сопровождении ПО описаны в файле "Описание процессов ПО "Хайс"".