### **Название задачи:** Открытие депозитов онлайн

### **Автор: Author**

### **Дата: 24.12.2024**

### **Функциональные требования**

Опишите здесь верхнеуровневые Use Cases. Их нужно оформить в виде таблицы с пошаговым описанием:

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Действующие лица или системы** | **Use Case** | **Описание** |
| 1 | Клиент | Клиент может просматривать список доступных вариантов для депозита на сайте и в интернет-банке | 1. Клиент заходит на сайт (или в интернет-банк); 2. Клиент видит список доступных для него вариантов депозитов; |
| 2 | Клиент | Клиент может оставить заявку на депозит через сайт или через интернет-банк | Клиент:   1. заходит на сайт (или в интернет-банк); 2. выбирает какой-либо из доступных депозитов; 3. оставляет заявку на депозит, которая поступает в систему банка; |
| 3 | Менеджер банка | Менеджер банка может использовать систему кол-центра для изучения заявки на новый депозит | 1. Клиент оставил заявку на депозит через сайт или через интернет-банк; 2. Менеджер банка увидел заявку клиента через систему кол-центра; |
| 4 | АБС | В АБС необходимо вести информацию по депозитным ставкам | 1. Клиент заходит на сайт (или в интернет-банк); 2. Клиент видит список доступных для него вариантов депозитов; 3. Информация по ставкам поступает на основе данных из АБС |
| 5 | Клиент | Клиент должен получать СМС-уведомления после подтверждения размера ставки и открытия депозита | 1. Клиент оставил заявку на депозит через сайт или через интернет-банк; 2. Менеджер банка оформил заявку; 3. Заявка прошла через все этапы согласования; 4. После одобрения заявки и открытия депозита пользователь получил СМС-уведомление |

### **Нефункциональные требования**

Опишите здесь нефункциональные требования и архитектурно-значимые требования.

|  |  |
| --- | --- |
| **№** | **Требование** |
| 1 | Доступность сервисов - 99,9%; |
| 2 | Все сервисы должны работать 24/7; |
| 3 | Возможность горизонтального масштабирования всех компонентов; |
| 4 | Базы данных: MS SQL и Oracle; |
| 5 | Микросервисная архитектура для новых сервисов, связанных с депозитами |
| 6 | Для новых сервисов интернет-банк должен взаимодействовать с АБС асинхронно |

### **Решение**

1. Сервис депозитов – это отдельный сервис с отдельным API – не связанный с существующим backend'ом интернет-банка. Такое решение проще масштабировать;
2. Одним из требований к новой системе – сервисы депозитов интернет-банка не должны взаимодействовать с АБС напрямую. Поэтому взаимодействие осуществляется через Kafka. В доменах интернет-банка и АБС развернуты дополнительные сервисы, которые обеспечивают гарантию доставки сообщений и позволяют дополнительно обрабатывать заявки;
3. В качестве дополнительной БД для сервиса депозитов используется MySQL, так как в банке есть опыт поддержки такой БД;
4. Доступность сервисов будет обеспечиваться за счет failover-стратегии типа ACTIVE-ACTIVE. В банке уже есть дополнительный ЦОД, поэтому на нем можно подготовить инфраструктуру на случай сбоя в работе в основном ЦОД.

### **Альтернативы**

1. Альтернативных решением является решение на стороне домена АБС. Вместо того, чтобы завязывать новый бизнес-процесс на БД АБС можно было бы использовать отдельную БД и выстраивать процесс относительно отдельной БД.

**Недостатки, ограничения, риски**

1. Завязывание новой логики на старую АБС является одним из ключевых недостатков. При возрастании нагрузки масштабироваться будет сложнее. Уже сейчас АБС может масштабироваться только вертикально, соответственно, в случае большого притока клиентов процесс обработки заявки на депозиты может затянуться.
2. В случае долгого процесса обработки заявок появляется риск потери клиентов;
3. Также, данное решение изначально имеет потолок масштабирования данных – связанных с невозможностью горизонтального масштабирования БД АБС.