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

### **Автор: Алексей Михайлов**

### **Дата: 21.01.2025**

### **Функциональные требования**

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

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Действующие лица или системы** | **Use Case** | **Описание** |
| UC1 | Новый клиент | Просмотр списка доступных депозитов на сайте | 1. Клиент открывает сайт 2. Видит список доступных депозитов с актуальными ставками. |
| UC2 | Новый клиент | Подача заявки на депозит на сайте | 1. Клиент выбирает интересующую ставку депозита на сайте 2. Оставляет свой номер телефона и ФИО 3. Заявка сохраняется в системе АБС 4. Система сообщает что клиенту перезвонят |
| UC3 | Менеджер бек-офиса | Обработка заявки на открытие депозита с сайта | 1. Менеджер выбирает заявку созданную на сайте 2. Менеджер видит информацию по ставке 3. Менеджер фиксирует какие особые условия можно предложить клиенту 4. Менеджер сохраняет решение по заявке 5. Система создает заявку в кол-центре для созвона с клиентом |
| UC4 | Менеджер кол-центра | Обработка заявки на депозит с сайта | 1. Менеджер открывает систему кол-центра 2. Менеджер видит заявку на открытие депозита с согласованными бек-офисом условиями 3. Менеджер звонит клиенту и согласует с ним условия 4. Менеджер приглашает клиента с отделение для идентификации 5. Менеджер закрывает заявку в системе кол-центра с результатом обсуждений (согласен ли клиент, какие условия принял) 6. Система кол-центра передает данные в систему АБС сохраняя новые данные в заявку открытия депозита |
| UC5 | Менеджер фронт-офиса | Прохождение идентификации в отделении | 1. Клиент приходит в отделение после приглашения от менеджера кол-центра 2. Менеджер в отделении проводит идентификацию клиента (проверку документов) 3. Менеджер находит заявку в системе АБС с ранее согласованными через кол-центр условиями 4. Менеджер получает подтверждение клиента что всё верно 5. Клиент подписывает документы с банком 6. Менеджер открывает депозит по согласованным ранее условиям в системе АБС 7. Клиент получает данные для входа в интернет-банк |
| UC6 | Клиент | Просмотр списка доступных депозитов в интернет-банке | 1. Клиент входит в интернет-банк 2. Клиент аутентифицируется 3. Клиент видит список доступных для него депозитов с актуальными ставками, а также персонализированные ставки |
| UC7 | Клиент | Подача заявки на депозит в интернет-банке | 1. Клиент выбирает интересующую ставку в интернет-банке, счет с которого списать сумму депозита и саму сумму депозита 2. Система отправляет код для подтверждения операции 3. Клиент подтверждает операцию через ввод смс кода 4. Заявка сохраняется в системе АБС для согласования |
| UC8 | Менеджер бек-офиса | Просмотр списка заявок на открытие депозита | 1. Менеджер заходит в систему АБС 2. Менеджер видит список заявок на открытие депозита |
| UC9 | Менеджер бек-офиса | Обработка заявки на открытие депозита с интернет-банка | 1. Менеджер выбирает заявку созданную в интернет-банке 2. Менеджер видит информацию по ставке, сумме депозита и счете, а также иную важную для решения информацию 3. Менеджер принимает решение по открытию депозита согласно информации по разрешенным ставкам депозитов из XLS-файлов 4. Менеджер принимает (или отклоняет) открытие депозита 5. Система АБС сохраняет инфомрацию про депозит и списывает деньги с счета 6. Система отправляет смс уведомление с результатом открытия депозита |

### **Нефункциональные требования**

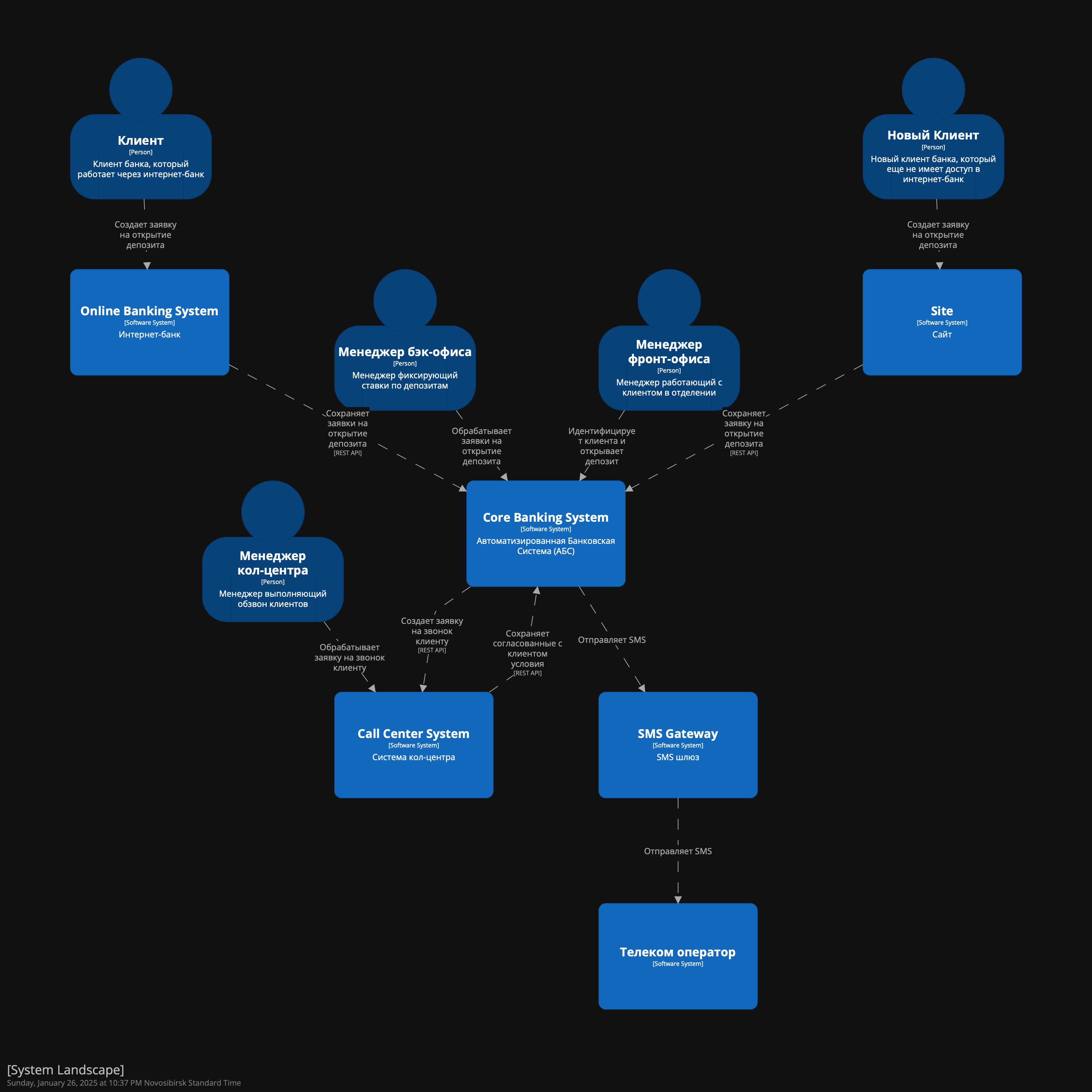
Опишите здесь нефункциональные требования и архитектурно-значимые требования.

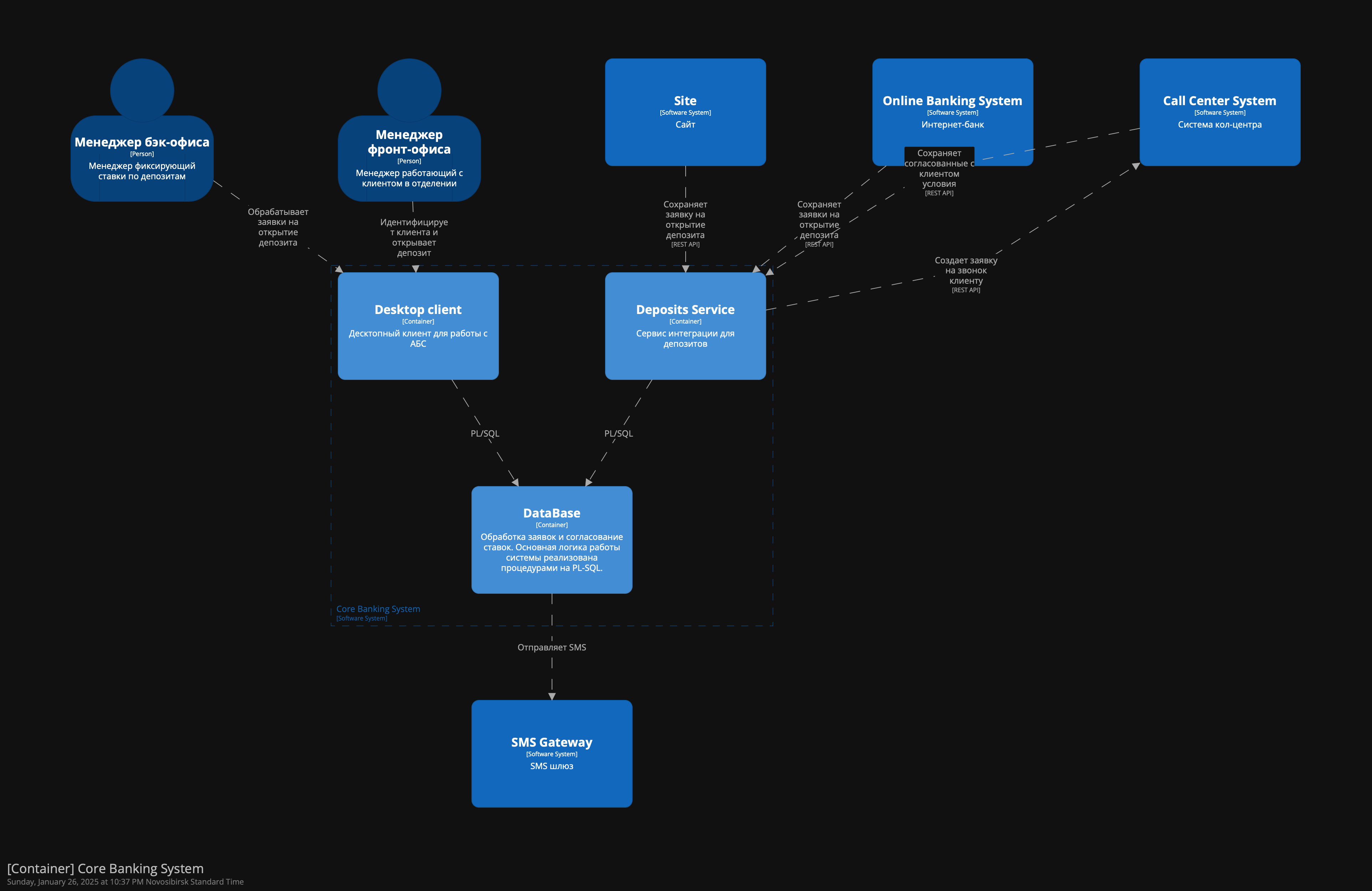
|  |  |
| --- | --- |
| **№** | **Требование** |
| 1 | Сервисы должны быть доступны 24/7 с доступностью не менее 99,9%. |
| 2 | Реализовать механизм переключения на резервный ЦОД в случае сбоя. |
| 3 | Время отклика всех операций не более 200 мс. |
| 4 | Минимизировать прямую работу интернет-банка с АБС. |
| 5 | Оптимизировать работу с базами данных MS SQL и Oracle. |
| 6 | Шифрование трафика между клиентом, сайтом, интернет-банком и АБС. |
| 7 | Использовать существующий СМС-шлюз телеком-оператора для подтверждения операций. |
| 8 | Документировать архитектуру и бизнес-процессы. |
| 9 | Использовать существующие технологии банка. |

### **Решение**

Приведите диаграммы контекста и контейнеров в модели C4. Опишите там основные компоненты и интеграции всех элементов решения.

Также опишите, какой логикой вы руководствовались в ходе принятия решений и выбора технологий. Не забывайте, что необходимо учесть все функциональные и нефункциональные требования.



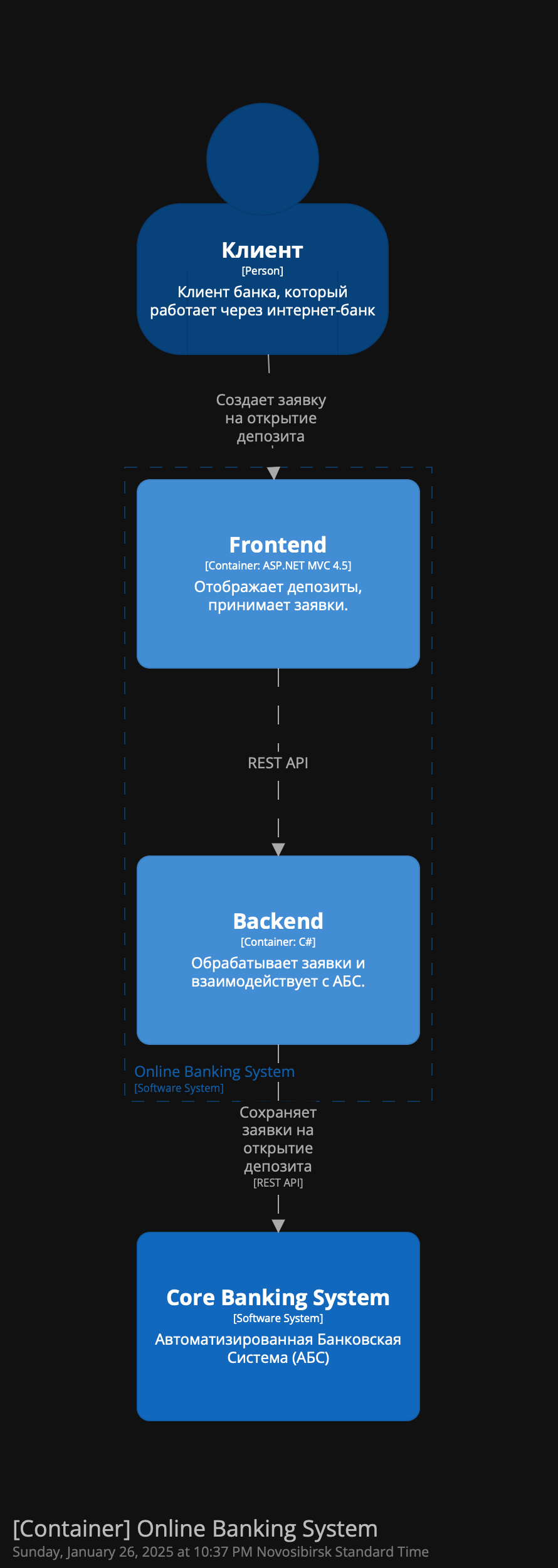
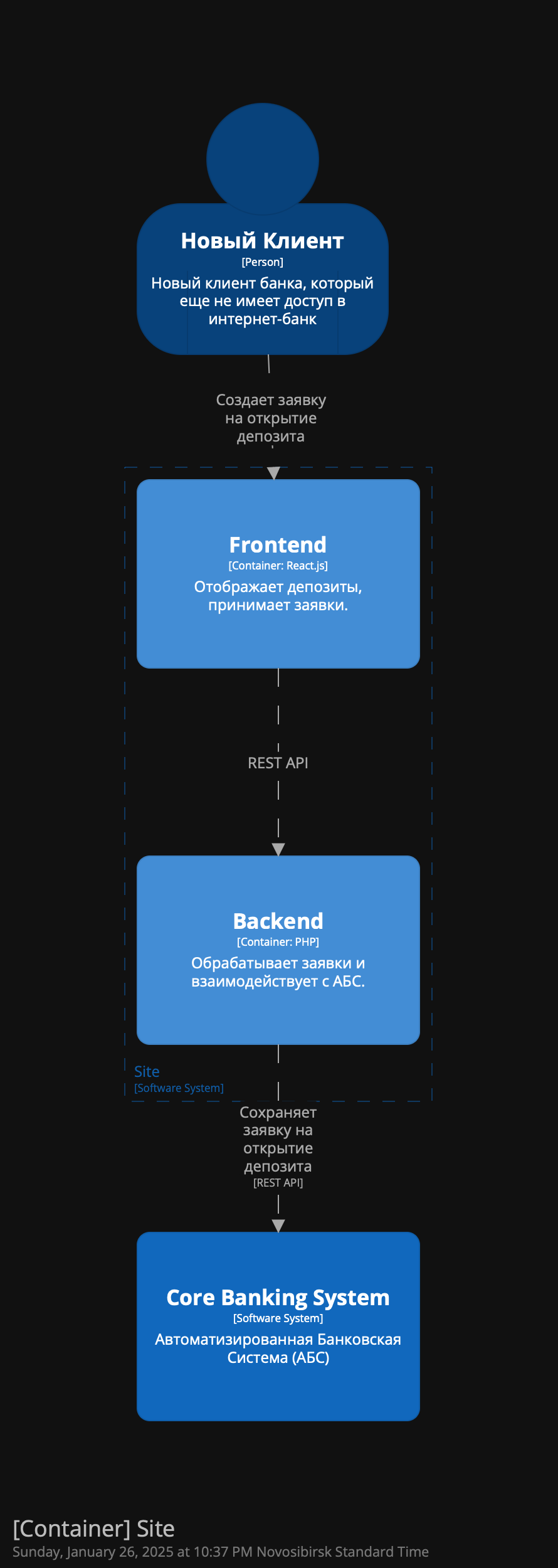


По диаграмме контекста можно увидеть что с сайта и интернет-банка идут обращения к системе АБС, но на уровне контейнеров можно увидеть что в АБС появляется новая подсистема Deposits Service, которую выделим в отдельный микросервис, для реализации логики интеграционной депозитов. Процесс открытия депозита всё равно потребует работу с АБС и его логикой, а значит наши системы должны самостоятельно уметь обращаться к бизнес логике сделанной на PL/SQL. Сам сервис будет реализован с помощью .NET Framework, так как текущая команда IT подразделения больше всего опыта имеет в нём, PL/SQL и Delphi.

В реализации Deposits Service помимо самого сервиса на .NET применим кеширование с помощью redis. В кеше будем хранить текущие ставки, глобальные для всех клиентов. А также короткий кеш с персональными ставками, так как мы знаем что чаще чем раз в день ставки меняться не смогут (расчёт ставок делается раз в день).

Также будет добавлена очередь Kafka, чтобы дать пользователям большую отзывчивость при подаче заявок (мы сохраним заявку в очередь а ответ юзеру дадим сразу).

Для повышения пропускной способности новый сервис можно горизонтально масштабировать.

По диаграммам контейнеров Интернет-банка и сайта можно увидеть что изменения относительно текущей реализации не большие - по REST API бекенды должны обращаться к новому сервису в АБС.

### **Альтернативы**

Опишите здесь наиболее важные альтернативные решения.

Для взаимодействия интернет банка и сайта с АБС можно было бы обращаться напрямую вызовами PL/SQL. Однако пускать напрямую бекенды интернет банка и сайта в PL/SQL будет чрезчур, это потребует и более сложную настройку со стороны "клиента" и слишком много деталей реализации станет видно для стороны "клиента". Поэтому всё взаимодействие будет по REST API с новым микросервисом, а он уже будет в нужные моменты обращаться к процедурам в БД.

Также выделенный микросервис четче ограничивает ответственность, позволяя не раздувать и дальше АБС, а забирая на себя новую логику. Такой первый шаг в сторону микросервисов.

**Недостатки, ограничения, риски**

Подробно опишите здесь недостатки, ограничения и риски выбранного решения.

Риски:

* Задержки в интеграции нового микросервиса с устаревшими компонентами (PL/SQL).
* Возможные узкие места производительности в PL/SQL или при взаимодействии с Oracle.
* Ограниченные возможности вертикального масштабирования для АБС.

Ограничения:

* Существующие ограничения инфраструктуры банка (старые версии технологий).
* Невозможность полного перехода на микросервисную архитектуру из-за зависимости от монолита.