**Выполнение домашнего задания к Занятию № 3.**

**USER STORY:** Я, как пользователь, хочу быстро повторять платежи или переводы денег в мобильном приложении банка, чтобы экономить время, используя историю выполненных ранее операций.

AS-IS. В настоящее время клиенты банка, желающие повторно осуществить платеж или перевод денег через мобильное приложение, должны вручную вводить все данные платежа: получателя, сумму, детали платежа и т.д. Это не только занимает значительное время, но и увеличивает риск ошибок при вводе данных.

TO-BE. Желаемое улучшение – чтобы пользователи могли быстро повторять платежи или переводы денег, выбирая предыдущую транзакцию из истории платежей в мобильном приложении банка. Это сделает процесс более эффективным, экономя время клиентов и снижая вероятность ошибок, так как не потребуется повторный ввод всех платёжных реквизитов и получателя.

**Нефункциональные требования.**

1. Данная доработка должны быть реализована на всех версиях операционных систем Android и IOS.
2. Интерфейс окна “повторная отправка платежа” должен быть идентичен интерфейсу окна обычного платежа.
3. Подтверждение должно быть реализовано посредством двухфакторной аутентификации клиента.
4. Платеж или перевод должен быть исполнен в течении 1 секунды с момента его отправки.
5. Система должна обеспечить успешное выполнение повторных платежей или переводов с вероятностью не менее 99,8% в любой момент времени.
6. В случае сбоев или потери связи, должна быть предусмотрена возможность автоматического возобновления операции без потери данных.
7. Данный сервис пользователь может использовать 24/7 без каких-либо существенных перерывов на обслуживание
8. Информация о выполненном платеже и переводе должна обновляться в реальном времени
9. В случае непредвиденного выхода из приложения или сбоя, информация о текущем состоянии незавершенной операции должна сохраняться для возможности быстрого восстановления.
10. Система должна включать легкодоступные инструкции или подсказки для данной доработки.

**Функциональные требования.**

1. Система должна предоставлять функцию просмотра истории платежей по карте или счёту.
2. Пользователь должен иметь возможность выбрать интересующую его отправленную операцию из истории для повторного её совершения.
3. После выбора операции из истории, приложение должно автоматически заполнить данные платежа (получателя, сумму, дату, назначение и т.д.) для проверки и подтверждения пользователем.
4. Пользователь должен иметь возможность изменить сумму и назначение операции.
5. После отправки платежа или перевода, он попадает в историю платежей приложения, где должен отображаться его статус.
6. У пользователя должна быть возможность отменить выполнение повторной отправки операции, перед её отправкой.
7. Пользователь должен иметь возможность создать шаблон платежа, для дальнейшего его использования
8. Система должна спрашивать подтверждение перед повторным совершением транзакции, чтобы предупредить случайные переводы.
9. Система должна иметь возможность использовать голосовые команды для начала процесса повторного платежа для удобства и скорости.
10. У пользователя должна быть возможность изменить источник списания средств для выполнения повторной операции.

**«Плохие» требования**

*Недвусмысленность:* Данная доработка должна быть реализована на системах Android и IOS, но не старше 11 версии.

*Полнота*: Платеж должен быть исполнен быстро, с момента его отправки.

*Корректность:* Система должна обрабатывать неограниченное количество операций в секунду

*Непротиворечивость*: Данная доработка должна быть реализована на всех системах Android и IOS, но не старше 11 версии.

*Проверяемость:* Окно пользователя “ повторная отправка платежа” должно быть удобным.

**Трассировка требований**

**МП – мобильное приложение.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Реализовано**  **Новое** | **В мобильном приложении отображены все счета и карты** | **Через МП пользователь должен совершать переводы и платежи** | **Возможность оформить продукт через МП** | **В МП должны быть Push - уведомления** | **В МП**  **Реализовать**  **новостную**  **витрину** |
| **Добавить оплату ЖКХ** |  | x |  | x | x |
| **Добавить возможность повторить платёж** | x | x |  | x | x |
| **Добавить диаграмму расходов** |  | x | x |  | x |
| **Добавить карту банкоматов и офисов** |  |  |  |  | x |
| **Реализовать систему кэшбеков** | x | x | x | x | x |

Исходя из проведения трассировки можно сделать вывод о том, что увеличится нагрузка на модуль переводов и транзакций, поэтому необходимо предусмотреть фактор масштабируемости и нагрузки на систему. Также увеличится нагрузка на систему отправки Push - уведомлений. Данную ситуацию необходимо обговорить с архитекторами и отделом интеграции.

Далее мы видим, что все новые требования должны попасть на новостную витрину. Значит в каждой доработке будет участвовать дизайн офис. И необходимо данную ситуацию также согласовать с ними.

По приоритетам мы видим, что реализовать Карту с банкоматами и офисами нам легче всего и данное требование требует согласование только со стороны дизайн офиса. Поэтому данную задачу можно взять в работу, пока мы согласовываем остальные задачи с архитектором и отделом интеграции.

После, в работу мы возьмем “**Добавить оплату ЖКХ”**, т.к. данная доработка важна для бизнеса (получение дохода с комиссии), и мы видим по таблице, что она связанна с 3-мя выполненными бизнес-требованиями.

Далее берём: “**добавить диаграмму расходов”.** Онатакже связана с 3-мя требованиями и не является трудоемкой.

**“Добавить возможность повторить платёж”** идет после диаграммы расходов**.**

И затем уже можно начинать **реализовывать систему кэшбеков.** т.к. это целая подсистема, которая требует больших трудозатрат, интеграций с множеством магазинов и т.д. И по схеме мы видим, что данное требование пересекается со всеми реализованными требованиями.