

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего образования  
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий  
(наименование института полностью)

Кафедра «Прикладная математика и информатика»  
(наименование кафедры)

02.03.03 Математическое обеспечение и администрирование информационных систем  
(код и наименование направления подготовки / специальности)

Мобильные и сетевые технологии  
(направленность (профиль) / специализация)

## ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка follow-up приложения для клиентов банка»

Обучающийся

Н.А. Алекинов

(Инициалы Фамилия)

(личная подпись)

Руководитель

к.т.н., Т.Г. Султанов

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант

к.п.н., доцент, А.В. Егорова

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2023

## Аннотация

Тема бакалаврской работы – «Разработка follow-up приложения для клиентов банка».

Бакалаврская работа посвящена разработке гибридного приложения для контроля показаний бытовых счетчиков.

В ходе выполнения исследований по бакалаврской работе была проанализирована и выбрана технология разработки follow-up приложения, было разработано гибридное мобильное приложение, выполнена реализация и тестирование follow-up приложения.

В первой главе выполняется постановка задачи на разработку мобильного follow-up приложения, формулируются требования к мобильному приложению, и проводится анализ аналогов.

Во второй главе разрабатывается архитектура мобильного приложения, описываются сценарии использования, проводится логическое моделирование мобильного приложения и разрабатывается дизайн мобильного follow-up приложения.

Третья глава содержит выбор инструментов реализации мобильного приложения, программную реализацию требуемых функций и тестирование мобильного follow-up приложения.

В заключении представлены результаты выполнения выпускной квалификационной работы.

В приложении представлен фрагмент кода мобильного follow-up приложения.

Бакалаврская работа состоит из 50 страниц текста и содержит 14 рисунков, 7 таблиц и 22 источника.

## **Abstract**

The title of the bachelor's thesis is "Development of a follow-up application for bank clients".

The research is devoted to developing follow-up application for bank clients.

When doing a research, the follow-up application development technology was analyzed and selected, a hybrid mobile application was developed, the follow-up application was implemented and tested.

The introduction reveals the relevance of the research and gives a brief description of the work done.

The first chapter the subject area of the study is considered, the requirements for a follow-up application are formulated, and the analysis of analogues is carried out.

The second chapter the architecture of the follow-up application is being developed, usage scenarios and logical modeling of the follow-up application is carried out and the design of the follow-up application is developed.

The third chapter contains the selection of tools for implementing the follow-up application, software implementation of the required functions and testing of a follow-up application.

In conclusion, the conclusions of the entire work are drawn.

The bachelor's thesis consists of an introduction, three chapters, a conclusion, list of used literature and appendix with code listing.

The volume of the bachelor's thesis is 45 pages, it also contains 14 figures, 7 tables, a list of 22 references and 1 listing.

## Оглавление

Введение.....	4
Глава 1 Постановка задачи и формирование требования на разработку мобильного follow-up приложения для клиентов банка .....	7
1.1 Постановка задачи на разработку мобильного follow-up приложения для клиентов банка .....	7
1.2 Разработка требований к follow-up приложению .....	8
1.3 Обзор и сравнительный анализ аналогов мобильного follow-up приложения для клиентов банка .....	10
Глава 2 Проектирование мобильного follow-up приложения для клиентов банка .....	17
2.1 Выбор типа мобильного приложения .....	17
2.2 Логическое проектирование мобильного приложения.....	19
2.2 Выбор среды разработки мобильного приложения .....	23
Глава 3 Реализация и тестирование follow-up приложения для клиентов банка ...	28
3.1 Архитектура follow-up приложения для клиентов банка .....	28
3.2 Разработка модели данных follow-up приложения .....	29
3.3 Реализация follow-up приложения для клиентов банка .....	31
Заключение .....	37
Список используемой литературы .....	39
Приложение А Фрагмент кода follow-up приложения.....	42

## Введение

Одним из ключевых положений современной концепции управления взаимоотношениями с клиентами банка является применение механизма отслеживания (follow-up) клиентов.

Если потенциальный клиент просматривал какой-нибудь банковский продукт, но не совершил дальнейшего шага по воронке продаж, он получит триггерное письмо с благодарностью за посещение сайта банка и персональными рекомендациями [2].

Наличие системы отслеживания клиентов облегчит банку масштабирование бизнеса и надлежащее обучение новых сотрудников в будущем.

Для реализации механизма отслеживания клиентов используется специальное программное обеспечение – follow-up приложения.

При этом следует принять во внимание, что большинство просмотров банковских продуктов клиентами совершается через мобильные устройства.

Таким образом, актуальность темы заключается в обеспечении банка мобильным приложением для отслеживания клиентов.

Объектом исследования бакалаврской работы является система управления взаимоотношениями с клиентами банка.

Предмет исследования – мобильное follow-up приложение для клиентов банка.

Цель бакалаврской работы – разработка мобильного follow-up приложения для клиентов банка.

Для достижения цели работы необходимо выполнить следующие задачи:

- выполнить постановку задачи на разработку мобильного follow-up приложения для клиентов банка;
- спроектировать мобильное follow-up приложение для клиентов банка;
- выполнить реализацию и тестирование мобильного follow-up

приложения для клиентов банка.

Методы исследования – технологии разработки мобильных приложений, методы и технологии веб-дизайна.

Практическая значимость бакалаврской работы заключается в разработке мобильного follow-up приложения, которое может быть использовано для отслеживания клиентов банка.

Бакалаврская работа состоит из введения, трех глав, заключения, списка используемой литературы и приложения.

В первой главе выполняется постановка задачи на разработку мобильного follow-up приложения, формулируются требования к мобильному приложению, и проводится анализ аналогов.

Во второй главе разрабатывается архитектура мобильного приложения, проводится логическое моделирование мобильного приложения и разрабатывается дизайн мобильного follow-up приложения.

Третья глава содержит выбор инструментов реализации мобильного приложения, программную реализацию требуемых функций и тестирование мобильного follow-up приложения.

В заключении представлены результаты выполнения выпускной квалификационной работы.

В приложении представлен фрагмент кода мобильного follow-up приложения.

Бакалаврская работа состоит из 45 страниц текста и содержит 14 рисунков, 7 таблиц и 22 источника.

# **Глава 1 Постановка задачи и формирование требования на разработку мобильного follow-up приложения для клиентов банка**

## **1.1 Постановка задачи на разработку мобильного follow-up приложения для клиентов банка**

Система отслеживания клиентов (CFU, Customer Follow-Up System) – это набор процессов, которые предприятие, занимающееся продажами и оказанием услуг, и его сотрудники используют для отправки последующих сообщений как потенциальным, так и текущим клиентам в надежде либо привлечь их к себе, либо сохранить их бизнес в долгосрочной перспективе.

Допустим, кто-то взаимодействует с сайтом компании, предоставляя свой электронный адрес. Наличие системы отслеживания клиентов позволяет торговому представителю предприятия связаться с ними. Это ключевой шаг в любой маркетинговой деятельности [13].

Предприятие может отслеживать клиентов различными способами:

- телефонные звонки - это простой традиционный способ отслеживания клиентов;
- электронные письма;
- SMS-сообщения.

Как показывает практика, эффективным способом отслеживания клиентов в банковской сфере является активизация страницы с услугой и/или незаполненной формы, на которой остановился клиент при последнем посещении сайта банка.

Для решения данной задачи используются мобильные follow-up приложения.

Таким образом, целью настоящей работы является разработка мобильного follow-up приложения, которая позволит повысить эффективность процесс отслеживания клиентов банка.

## 1.2 Разработка требований к follow-up приложению

Для разработки требований к ИС НБДП используем методологию FURPS+.

Модель FURPS+ является одной из часто используемых моделей для классификации атрибутов качества программного обеспечения (ПО) [19].

Аббревиатура FURPS означает функциональность, удобство использования, надежность, производительность и возможность поддержки.

FURPS+ служит контрольным списком нескольких ключевых качеств, которые следует учитывать при определении требований.

Функциональность относится к возможностям и функциям приложения. Это основа того, чего хочет клиент.

Юзабилити (удобство использования) влияет на человека, который будет использовать программу.

Оно дает ответ является ли интерфейс ПО интуитивно понятным в использовании? Является ли документация точной и полной?

Чтобы определить требования к надежности, необходимо знать допустимое время простоя системы, предсказуемы ли сбои и как можно восстановить систему.

Требования к производительности задают время отклика приложения.

Они накладывают ограничения на системные ресурсы, которые ПО может использовать.

Возможность поддержки показывает, что приложение можно тестировать, расширять, обслуживать, устанавливать и настраивать.

В таблице 1 представлены основные требования к мобильному follow-up приложению для клиентов банка в методологии FURPS+.

Требования согласованы с руководителем маркетингового отдела банка.



Таблица 1 – Требования к мобильному follow-up приложению для клиентов банка

№	Требование	Статус	Полезность	Риск	Стабильность
<b>Functionality — Функциональные требования</b>					
1.	«Автоматизированное управление отслеживанием клиентов банка	одобренное	критическая	средний	низкая
<b>Usability— Требования к удобству использования</b>					
2.	Дружественный интуитивный интерфейс	одобренное	критическая	средний	низкая
<b>Reliability— Требования к надежности</b>					
3.	Допустимая частота/периодичность сбоев: 1 раз в 300 часов	одобренное	важная	средний	средняя
4.	Среднее время сбоев: 1 раб. день	одобренное	важная	средний	средняя
5.	Возможность восстановления системы после сбоев: 1 раб. день банка	одобренное	важная	средний	средняя
6.	Режим работы: 24/7/365	одобренное	важная	средний	средняя
<b>Performance — Требования к производительности</b>					
7.	Допустимое количество одновременно работающих пользователей: 50	предложенное	важная	средний	средняя
8.	Время реакции на возникновение аварийной ситуации: 10 с	предложенное	важная	средний	средняя
<b>Supportability — Требования к поддержке</b>					
9.	Время устранения критических проблем: в течение рабочего дня	предложенное	важная	средний	средняя
<b>Проектные ограничения</b>					
10.	Применение современных технологий разработки мобильных приложений	предложенное	критическая	средний	низкая
11.	Низкие затраты на разработку	предложенное	критическая	средний	низкая» [19]

Разработанный перечень требований является основой для проектирования мобильного follow-up приложению для клиентов банка.

### 1.3 Обзор и сравнительный анализ аналогов мобильного follow-up приложения для клиентов банка

Рассмотрим архитектурные и функциональные особенности аналогов мобильного follow-up приложения для клиентов банка.

Рассмотрим мобильное приложение Customer Follow Up - Never loose a customer again! [9].

По утверждению разработчика, это очень простое, красивое и удобное приложение по небольшой цене позволит быстро записать всю информацию о телефонном звонке вашего клиента и предоставит вам возможность вернуться к клиенту в более удобное время.

Скриншоты приложения для iPhone показаны на рисунке 1.

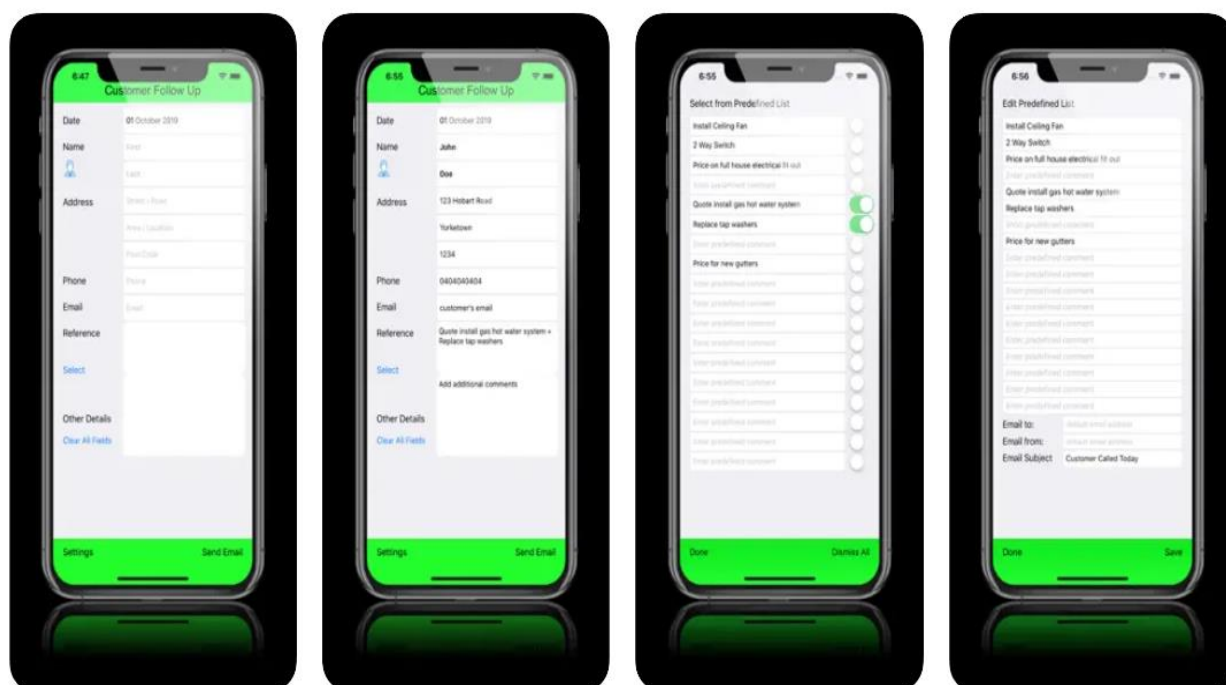


Рисунок 1 – Скриншоты приложения Customer Follow Up - Never loose a customer again! (iPhone)

Функции приложения:

- экономия времени;
- автоматическое предварительное заполнение имени клиента (если клиент есть в ваших контактах);
- predetermined по умолчанию ссылки на задания/общие заметки и т. д. (полностью настраиваемые);
- мгновенная очистка всех полей;
- сохранение введенных данных при закрытии приложения и др.

Приложение Follow up Chat по мнению его разработчиков – это универсальное решение всех проблем, связанных с клиентами и управлением подтверждениями Proof Manager [11].

Скриншот приложения показан на рисунке 2.

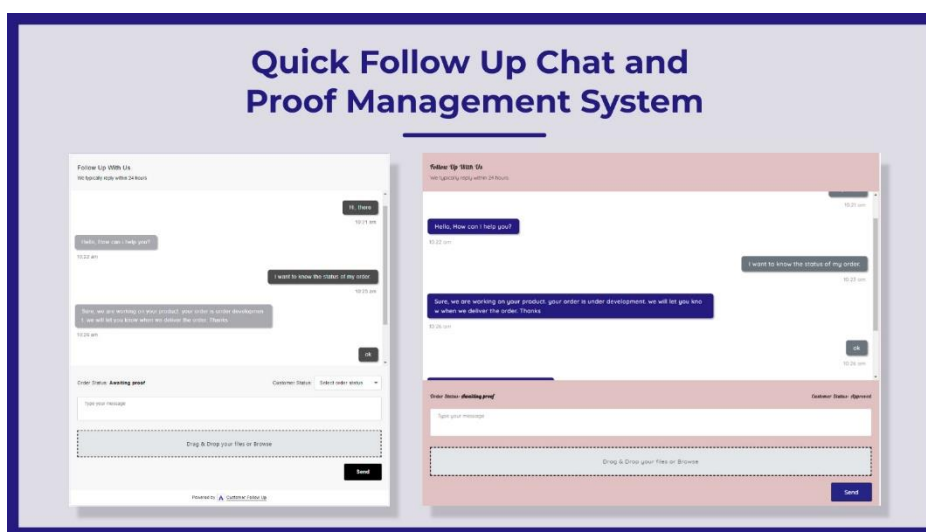


Рисунок 2 – Скриншот приложения Follow up Chat - Proof Manager

Пользователю предоставляется интерфейс в режиме реального времени для взаимодействия с клиентом для сбора информации, загрузки подтверждения заказа, включения контроля версий для всех файлов заказов и предоставления клиентам обновлений заказа в режиме реального времени с панели управления пользователя.

Приложение обеспечивает сквозное управление индивидуальными заказами от утверждения до окончательной доставки без необходимости рассылать холодные электронные письма.

Возможности:

- имеется виджет чата с клиентами в режиме реального времени для обмена информацией для индивидуальных заказов;
- позволяет отправлять несколько типов файлов (изображения, PDF-файлы и т. д.);
- настраиваемое уведомление клиентов по электронной почте для отправки статуса заказа;
- отправка счета клиенту для дополнительной оплаты за настройку и др.

Приложение представляет собой облачное решение, оптимизированное для работы с мобильными устройствами.

Приложение Customer Follow-Up - Odoo 12 App обеспечивает отслеживание платежей клиентов [10].

Скриншот приложения показан на рисунке 3.

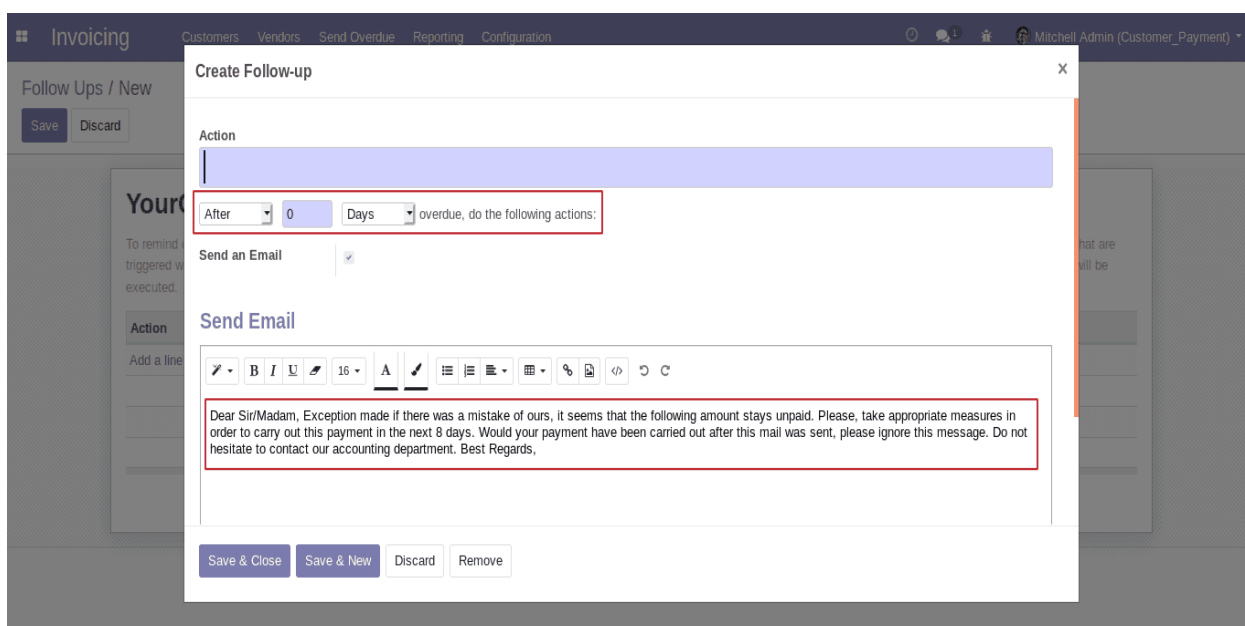


Рисунок 3 – Скриншот приложения Customer Follow-Up - Odoo 12 App

Приложение помогает отправлять соответствующие напоминания клиентам, чьи платежи задерживаются.

Приложение будет генерировать почту для каждого клиента, который получил неоплаченные счета. Неоплаченные счета можно просмотреть в партнерской форме. И, таким образом, легко распечатать его в виде письма или отправить письмо клиенту, приложив детали счета. Оба письма / email могут быть запущены с самой страницы. Для автоматической отправки электронных писем клиентам можно настроить уровни отслеживания, а также установить отдельную почту для каждого уровня и распечатать список счетов с общей суммой оплаты каждого клиента, используя опцию отправки просроченных. Приложение может эффективно использоваться для отслеживания счетов-фактур.

Приложение представляет собой облачное решение, оптимизированное для работы с мобильными устройствами.

Приложение Arptivo Mobile CRM App представляет собой мобильную CRM-систему, в которой реализованы опции отслеживания клиентов [8].

Приложение позволяет управлять повседневными рутинными задачами и действиями с помощью программного обеспечения для отслеживания активности.

Пользователи получают полный обзор своих заметок, электронной почты, управления документами, напоминаний, звонков, встреч и отслеживаний клиентов (Follow-Ups), что эффективно упрощает рабочий процесс.

Скриншот опции отслеживания клиентов Arptivo Mobile CRM App показан на рисунке 4.

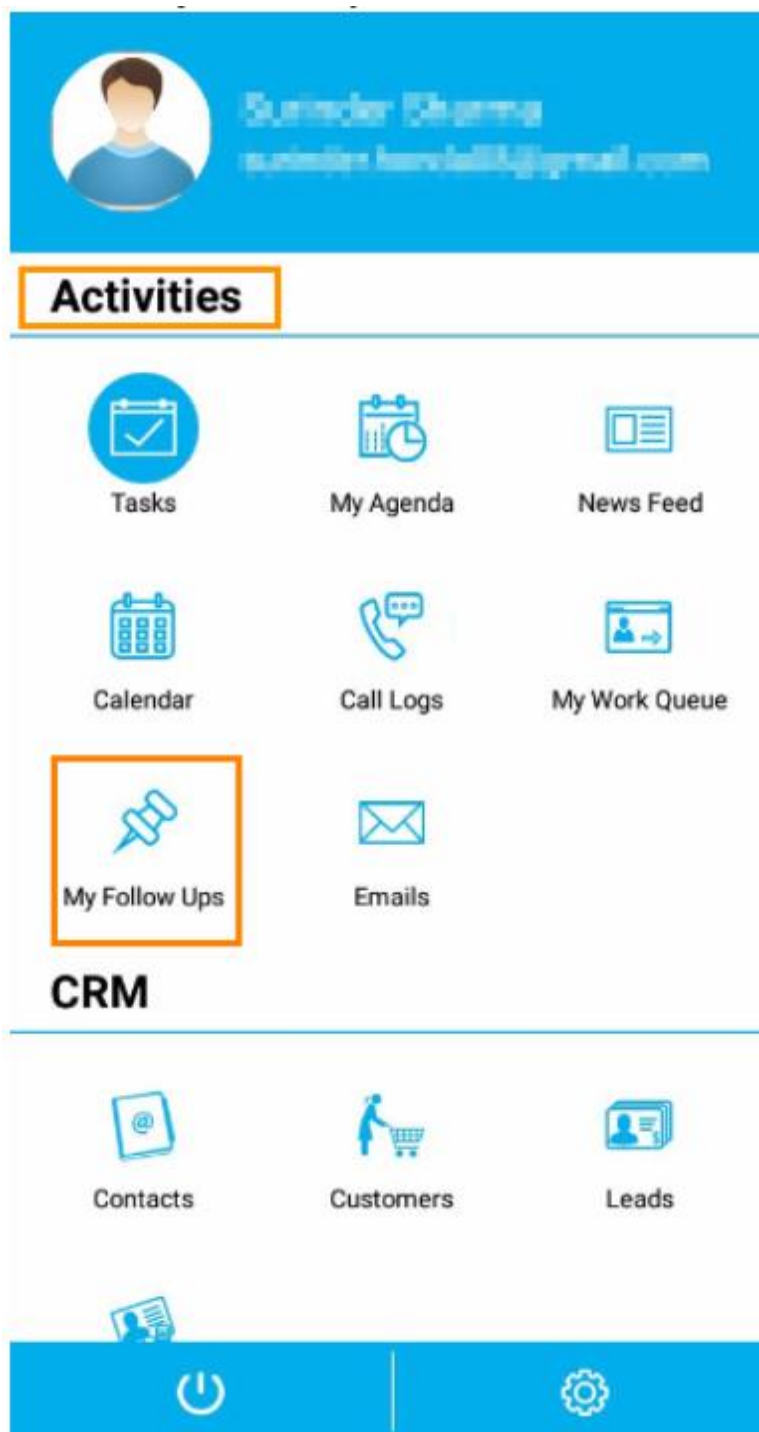


Рисунок 4 – Скриншот опции отслеживания клиентов Arptivo Mobile CRM App

Для сравнительного анализа рассмотренных аналогов используем таблицу 2.

Сравнительный характеристик аналогов приложения представлен значениями от 0 до 5, где 0 – полное несоответствие требованиям, 5 – полное соответствие требованиям.

Таблица 2 – Сравнительный анализ аналогов мобильного follow-up приложения для отслеживания клиентов банка

Характеристика/балл (0-5)	Customer Follow Up - Never loose a customer again!	Follow up Chat - Proof Manager	Customer Follow-Up - Odoo 12 App	Apptivo Mobile CRM App
Автоматизированное управление отслеживанием клиентов банка	3	2	3	3
Дружественный интуитивный интерфейс	4	3	3	4
Применение современных технологий разработки мобильных приложений	5	3	3	5
Низкие затраты на разработку	4	2	2	3
Итого	15	10	11	15

Как показал сравнительный анализ, ни одно из рассмотренных приложений не отвечает полностью заданным требованиям.

Таким образом, разработка мобильного follow-up приложения для отслеживания клиентов банка актуальна и представляет практический интерес.

#### Выводы к главе 1

Результаты проделанной работы позволили сделать следующие выводы:

- одним из ключевых положений современной концепции управления

взаимоотношениями с клиентами банка является применение механизма отслеживания (follow-up) клиентов. Наличие в банке системы отслеживания клиентов облегчит банку масштабирование бизнеса и надлежащее обучение новых сотрудников в будущем;

- как показывает практика, эффективным способом отслеживания клиентов в банковской сфере является активизация страницы с услугой и/или незаполненной формы, на которой остановился клиент при последнем посещении сайта банка. Для решения данной задачи используются мобильные follow-up приложения;
- разработанный в методологии FURPS+ перечень требований является основой для проектирования мобильного follow-up приложения для клиентов банка;
- как показал сравнительный анализ, ни одно из рассмотренных приложений не отвечает полностью заданным требованиям.

Таким образом, разработка мобильного follow-up приложения для отслеживания клиентов банка актуальна и представляет практический интерес.



## **Глава 2 Проектирование мобильного follow-up приложения для клиентов банка**

### **2.1 Выбор типа мобильного приложения**

«Выбор нужного решения для мобильного приложения зависит от его целевой аудитории, платформ, на которых оно будет установлено, особенностей и функциональности, которые вы планируете включить в приложение. Важен ресурс времени и средств, которые вы готовы потратить на разработку приложения, а также уровень навыков команды разработчиков.

Существует три основных типа приложений:

- нативные приложения;
- мобильные веб-приложения;
- гибридные приложения» [4].

Рассмотрим архитектурные особенности вышеперечисленных типов мобильных приложений [16].

Нативные приложения находятся на мобильном устройстве, а доступ к ним осуществляется с помощью значков на главном экране устройства.

Нативные приложения устанавливаются через магазин приложений (например, Google Play или Apple App Store). Они разрабатываются специально для одной платформы и могут в полной мере использовать все возможности устройства.

Они могут использовать камеру, GPS, акселерометр, компас, список контактов и так далее. Они также могут включать жесты (либо стандартные жесты операционной системы, либо новые жесты, определяемые приложением). Нативные приложения могут использовать систему уведомлений устройства и работать в автономном режиме.

«Мобильные веб-приложения - это не настоящие приложения; на самом деле это веб-сайты, которые во многих отношениях выглядят как нативные приложения, но не реализованы как таковые.

Они запускаются браузером и обычно написаны на языке HTML5.

Пользователи сначала получают к ним доступ, как к любой веб-странице: они переходят по специальному URL, а затем имеют возможность установить их на главном экране, создав закладку на этой странице.

Веб-приложения стали по-настоящему популярными, когда появился HTML5 и пользователи поняли, что они могут получить функциональность, подобную нативной, в браузере.

Сегодня, когда все больше и больше сайтов используют язык HTML5, различие между веб-приложениями и обычными веб-страницами стало размытым» [16].

Гибридные приложения - это частично нативные приложения и частично веб-приложения.

Как и нативные приложения, они доступны в App Store и могут использовать множество доступных функций устройства.

Как и веб-приложения, они опираются на HTML, отображаемый в браузере, с оговоркой, что браузер встроен в приложение.

Часто компании создают гибридные приложения в качестве оболочки для существующей веб-страницы; таким образом, они надеются получить присутствие в App Store, не затрачивая значительных усилий на разработку другого приложения.

«Гибридные приложения также популярны, потому что они позволяют разрабатывать на разных платформах и, таким образом, значительно снижают затраты на разработку: то есть одни и те же компоненты HTML-кода можно повторно использовать в разных мобильных операционных системах» [3].

Для сравнительного анализа типов мобильных приложений используем таблицу 3 [1].

Таблица 2 – Сравнительный анализ типов мобильных приложений

Тип мобильного приложения	Преимущества	Недостатки
Нативное приложение	Высокая производительность, простота интеграции	Высокая стоимость разработки и обслуживания, ограничение одной платформой
Мобильное веб-приложение	Удобство, персонализация, работа в автономном режиме	Плохая совместимость, слабая поддержка и обслуживание
Гибридное приложение	Низкая стоимость разработки и обслуживания, повторное использование кодовой базы	Возможные проблемы с интеграцией, низкая производительность и надежность

По результатам сравнительного анализа для мобильного follow-up приложения для клиентов банка выбран тип гибридного приложения.

## 2.2 Логическое проектирование мобильного приложения

На стадии логического проектирования мобильного приложения используется объектно-ориентированный подход и язык UML.

В качестве средства моделирования использовано CASE-средство Rational Rose [14].

Для описания функциональных требований мобильного follow-up приложения для клиентов банка разработана диаграмма вариантов использования.

Единственным актором является Клиент банка.

Описание вариантов использования мобильного приложения представлено в таблицах 3-5.

Таблица 3 – Описание прецедента: Аутентификация

«Прецедент: Аутентификация
ID: 1
Краткое описание: Аутентификация клиента
Главный актер: Клиент банка
Второстепенный актер: нет
Предусловие: нет
Основной поток: Клиент банка проходит аутентификацию
Постусловие: нет
Альтернативные потоки: нет» [20]

Таблица 4 – Описание прецедента: Согласие на переход

«Прецедент: Согласие на переход
ID: 2
Краткое описание: Согласие клиента перейти на последнюю незаполненную форму
Главный актер: Клиент банка
Второстепенный актер: нет
Предусловие: успешная аутентификация, получение предложения и переход
Основной поток: Клиент банка соглашается перейти на последнюю незаполненную форму
Постусловие: нет
Альтернативные потоки: нет» [20]

Таблица 5 – Описание прецедента: Активизация последней незаполненной формы

«Прецедент: Активизация последней незаполненной формы
ID: 2
Краткое описание: Активизация последней незаполненной формы
Главный актер: Клиент банка
Второстепенный актер: нет
Предусловие: успешная аутентификация, согласие на переход
Основной поток: Клиент банка активизирует последнюю незаполненную форму
Постусловие: нет
Альтернативные потоки: нет» [20]

Диаграмма вариантов использования мобильного приложения показана на рисунке 5.

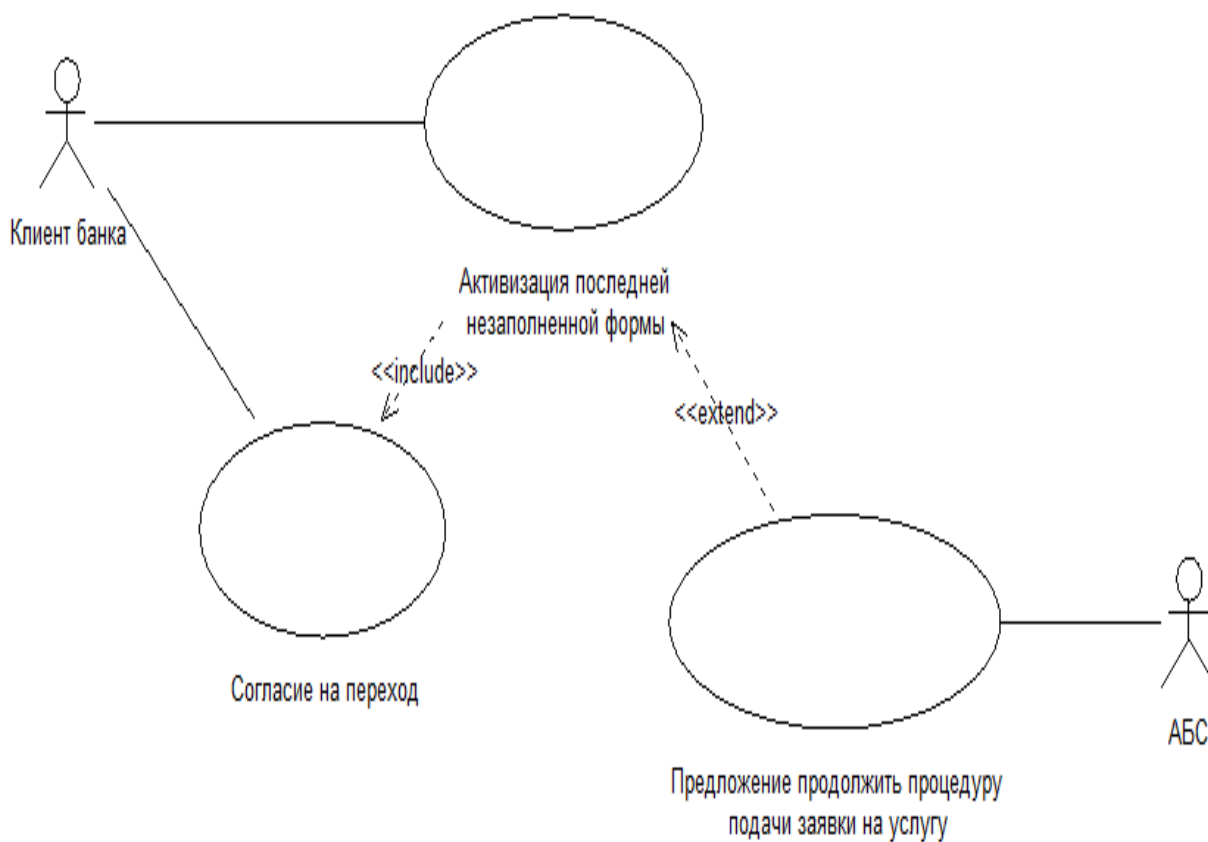


Рисунок 5 – Диаграмма вариантов использования мобильного follow-up приложения для клиентов банка

Диаграмма вариантов использования отражает функциональный аспект проектируемой мобильного follow-up приложения для клиентов банка.

Для построения объектной модели мобильного follow-up приложения разработана диаграмма классов, показанная на рисунке 6.

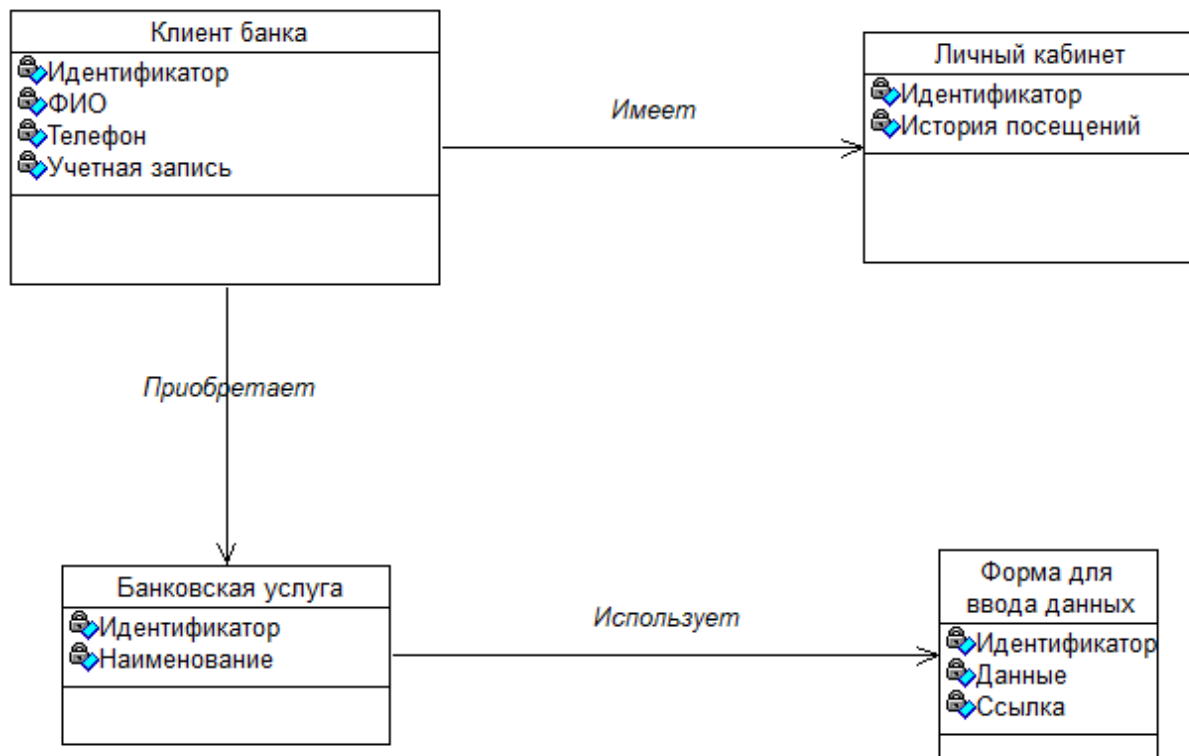


Рисунок 5 – Диаграмма классов мобильного follow-up приложения для клиентов банка

В таблице 6 представлена спецификация классов мобильного follow-up приложения для клиентов банка.

Таблица 6 – Спецификация диаграммы классов мобильного follow-up приложения для клиентов банка

Класс	Описание
«Клиент банка»	Класс объектов, моделирующих на логическом уровне клиентов банка
Личный кабинет	Класс объектов, моделирующих на логическом уровне личные кабинеты клиентов банка
Банковская услуга	Класс объектов, моделирующих на логическом уровне услуги банка
Форма для ввода данных	Класс объектов, моделирующих на логическом уровне формы для ввода данных заявок на услугу» [20]

Представленные классы объектов являются основой для формирования

программных классов мобильного follow-up приложения для клиентов банка.

## **2.2 Выбор среды разработки мобильного приложения**

В качестве прототипа мобильного приложения разработаем гибридное мобильное приложение для операционной системы (ОС) Android.

Рассмотрим и сравним возможности популярных сред разработки мобильных приложений для ОС Android [5]:

- Eclipse;
- IntelliJ Idea;
- Android Studio.

Далее представлены характеристики каждой из указанных сред разработки.

### **2.2.1 Среда разработки Eclipse**

Eclipse – это бесплатная интегрированная среда разработки (IDE), разработанная некоммерческой компанией Eclipse Foundation.

Согласно индексу PYPL, среда разработки Eclipse с открытым исходным кодом является второй по популярности в мире [12].

Eclipse начиналась как среда разработки Java, но с тех пор выросла для поддержки множества различных языков программирования. В течение нескольких лет версия Eclipse с подключаемым модулем Android (ADT) была рекомендуемой средой разработки для платформы Android.

Преимущества Eclipse:

- интерфейс переведен на грамотный русский язык (документация прилагается);
- отлично «гоняет» на компьютерах с низкой производительностью;
- имеет дополнительные функции (для серверной работы и анализа базы данных);
- может подключаться к модулям;

– может работать в групповом режиме (когда проект создают несколько человек одновременно).

Хотя Google прекратил поддержку ADT, многие разработчики мобильных приложений продолжают использовать Eclipse для создания приложений для Android (рисунок 6).

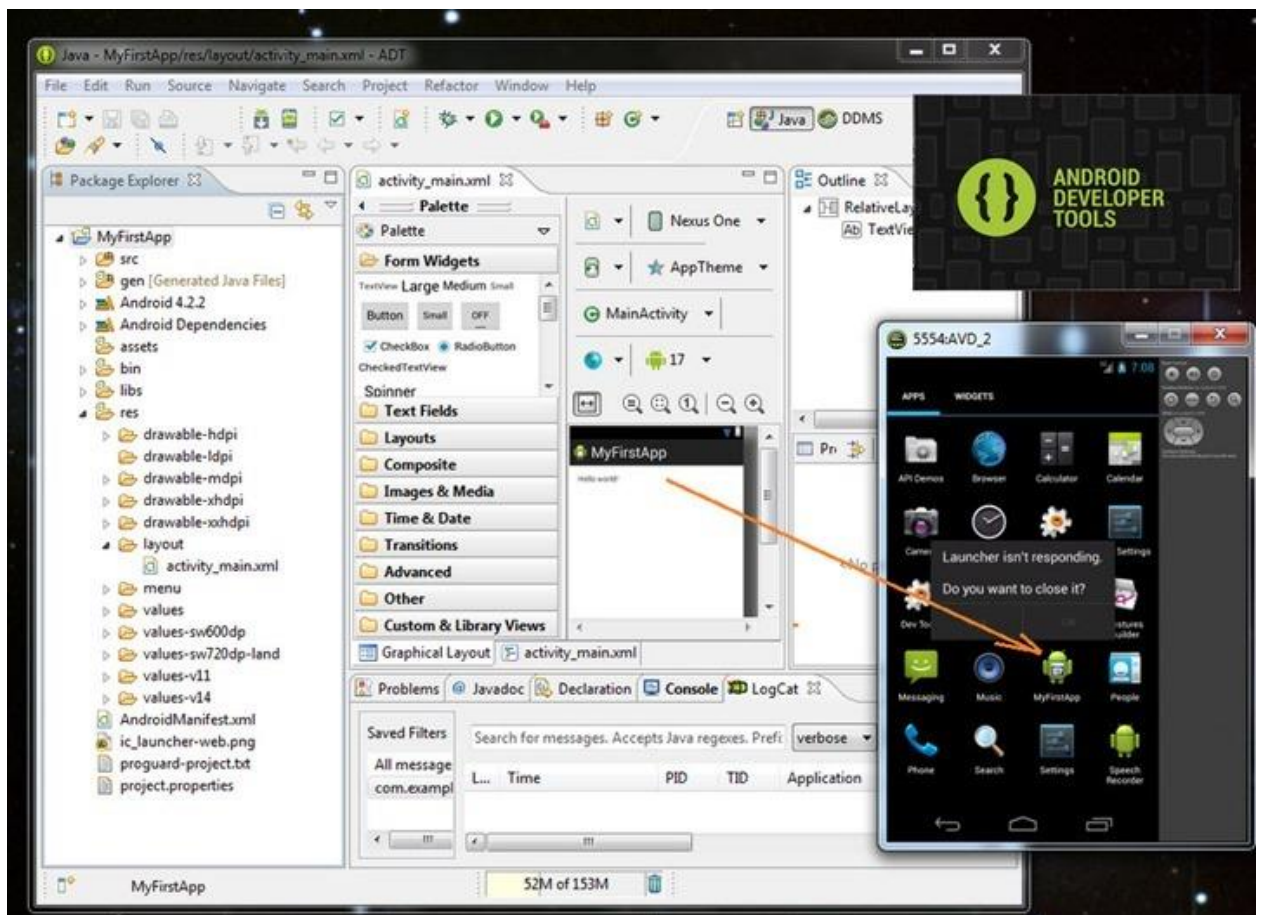


Рисунок 6 – Скриншот среды разработки мобильного приложения Eclipse IDE

Eclipse поставляется в нескольких версиях, в том числе для Java, Java EE, C/C++ и PHP.

### 2.2.2 Среда разработки IntelliJ IDEA

«IntelliJ IDEA – это ведущая, а по мнению многих отраслевых экспертов и Java-разработчиков, просто лучшая на сегодняшний день среда быстрой разработки приложений на языке Java.



IntelliJ IDEA представляет собой высокотехнологичный комплекс тесно интегрированных инструментов программирования, включая интеллектуальный редактор исходных текстов с развитыми средствами автоматизации, мощные инструменты рефакторинга кода, встроенную поддержку технологий J2EE, механизмы интеграции со средой тестирования Ant/JUnit и системами управления версиями, уникальный инструмент оптимизации и проверки кода Code Inspection, а также инновационный визуальный конструктор графических интерфейсов (рисунок 7)» [15].

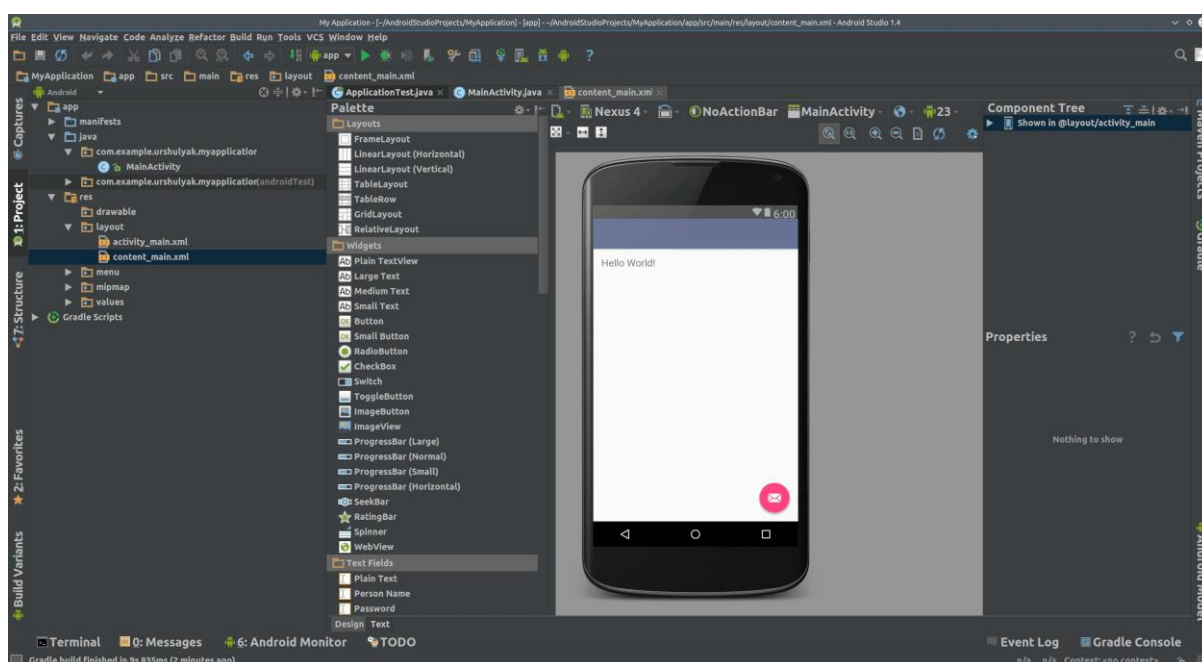


Рисунок 7 – Скриншот среды разработки мобильного приложения IntelliJ IDEA

Начиная с версии 9.0, среда доступна в двух редакциях: Community Edition (бесплатная) и Ultimate Edition.

Бесплатная версия поддерживает Java, Kotlin, Groovy и Scala; Android; Maven, Gradle и SBT; работает с системами контроля версий Git, SVN, Mercurial и CVS.

### 2.2.3 Среда разработки Android Studio

Android Studio — это официальная интегрированная среда разработки

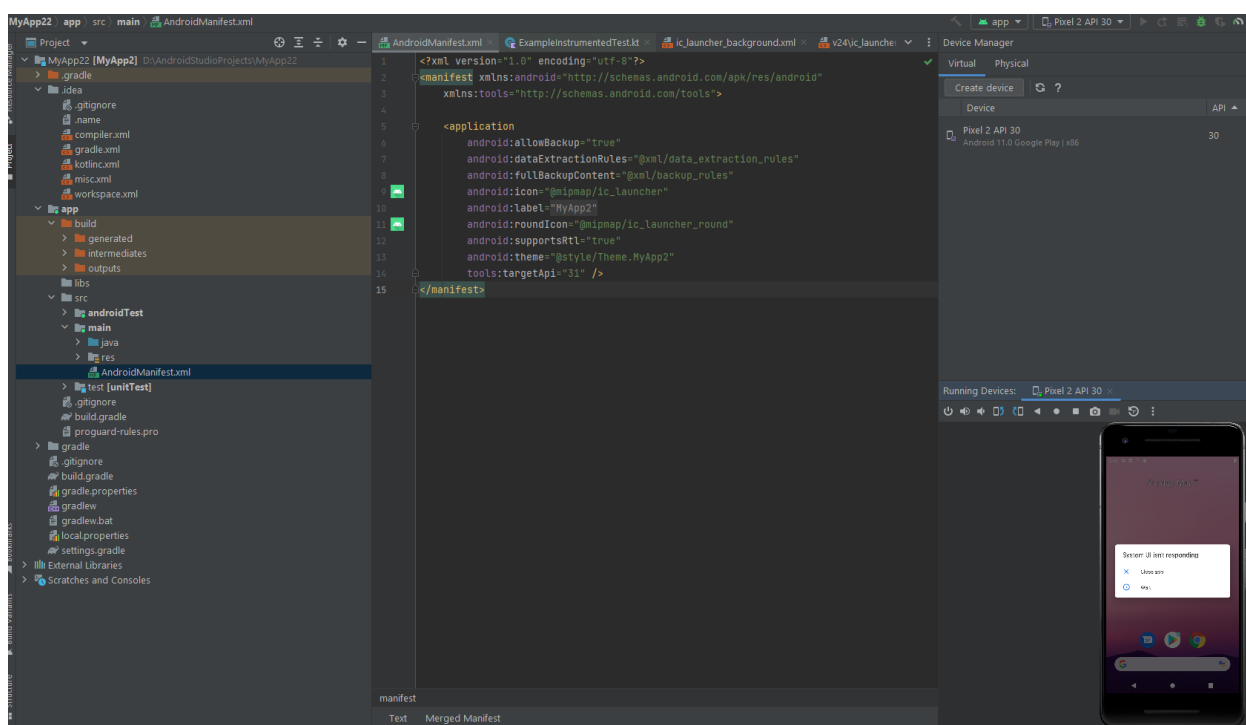
для разработки приложений для Android.

«Среда Android Studio основана на IntelliJ IDEA, интегрированной среде разработки программного обеспечения на языке Java, и включает инструменты редактирования кода и средства разработки.

Для поддержки разработки приложений в операционной системе Android среда Android Studio использует систему сборки на основе Gradle, эмулятор Android, шаблоны кода и интеграцию с GitHub.

Каждый проект в Android Studio имеет одну или несколько модальностей с исходным кодом и файлами ресурсов. Эти модальности включают модули приложений Android, модули библиотеки и модули Google App Engine.

Android Studio использует функцию «Применить изменения», чтобы передать изменения кода и ресурсов в работающее приложение. Редактор кода помогает разработчику в написании кода и предлагает завершение кода, проверку и анализ (рисунок 8)» [6].



## Рисунок 8 – Скриншот среды разработки мобильного приложения Android Studio

Приложения, созданные в Android Studio, затем компилируются в формат APK для отправки в Google Play Store.

Android Studio и Software Development Kit можно загрузить непосредственно с сайта Google.

Для выбора среды разработки используем таблицу 7.

Таблица 7 – Сравнение сред для разработки мобильного приложения

Среда разработки	Eclipse	IntelliJ IDEA	Android Studio
Поддержка двух и более языков программирования	+	+	+
Дружественный локализованный интерфейс	+	-	-
Большое сообщество разработчиков	-	+	+
Предпочтение разработчика	-	-	+
Итого	2	2	3

Исходя из проведенного сравнения, выбираем в качестве среды разработки мобильного follow-up приложения для клиентов банка Android Studio.

### Выводы по главе 2

Результаты проделанной работы позволили сделать нижеследующие выводы.

Тип «гибридное приложение» является наиболее предпочтительным для реализации follow-up приложения для клиентов банка.

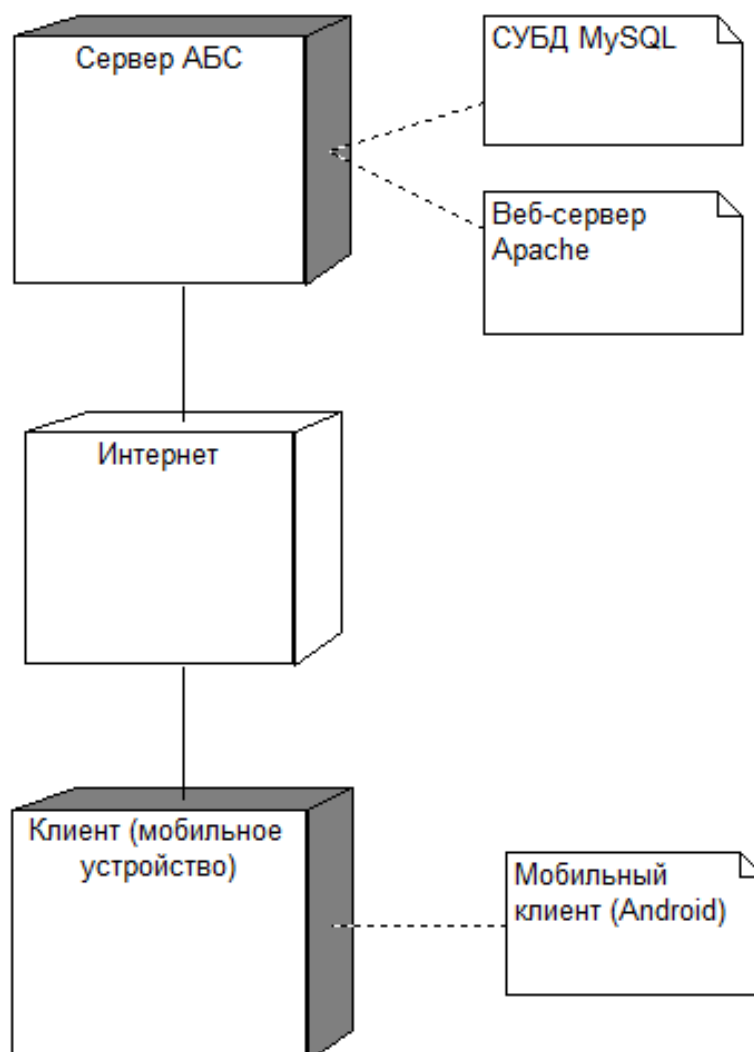
Характеристики среды Android Studio обусловили ее выбор в качестве средства для разработки мобильного follow-up приложения для клиентов банка.

## Глава 3 Реализация и тестирование follow-up приложения для клиентов банка

### 3.1 Архитектура follow-up приложения для клиентов банка

Архитектура follow-up приложения для клиентов банка построена по трехзвенной модели архитектуры «клиент-сервер».

Архитектура follow-up приложения для клиентов банка показана на рисунке 9.



## Рисунок 9 – Архитектура follow-up приложения для клиентов банка

Как показано на рисунке 9, на мобильном устройстве устанавливается мобильный клиент follow-up приложения.

Функции веб-сервера выполняет программа Apache HTTP Server [7].

Функции сервера баз данных выполняет СУБД MySQL.

Обе программы размещены на сервере автоматизированной банковской системе (АБС).

### 3.2 Разработка модели данных follow-up приложения

MySQL — самая популярная в мире база данных с открытым исходным кодом. По данным DB-Engines, MySQL занимает второе место по популярности после Oracle Database. MySQL поддерживает многие из наиболее популярных приложений, включая Facebook, Twitter, Netflix, Uber, Airbnb, Shopify и Booking.com.

Поскольку исходный код MySQL является открытым, он включает в себя множество функций, разработанных в тесном сотрудничестве с пользователями на протяжении более 25 лет.

«MySQL — это система управления реляционными базами данных.

Реляционная база данных хранит данные в отдельных таблицах, а не хранит все данные в одном большом хранилище. Структура базы данных организована в физические файлы, оптимизированные для скорости.

Логическая модель данных с такими объектами, как таблицы данных, представления, строки и столбцы, предлагает гибкую среду программирования» [22].

Следует отметить, что среда разработки Android Studio успешно взаимодействует с СУБД MySQL [17].

Для разработки модели данных приложения используем программу MySQL Workbench.

«MySQL Workbench — это унифицированный визуальный инструмент для архитекторов баз данных, разработчиков и администраторов баз данных. MySQL Workbench обеспечивает моделирование данных, разработку SQL и комплексные инструменты администрирования для настройки сервера, администрирования пользователей, резервного копирования и многого другого.

MySQL Workbench доступен в Windows, Linux и Mac OS X.

MySQL Workbench упрощает проектирование и обслуживание базы данных, автоматизирует трудоемкие и подверженные ошибкам задачи, а также улучшает взаимодействие между группами администраторов баз данных и разработчиков.

Это позволяет архитекторам данных визуализировать требования, общаться с заинтересованными сторонами и решать проблемы проектирования до того, как будут сделаны значительные затраты времени и ресурсов.

Это позволяет проектировать базы данных на основе моделей, что является наиболее эффективной методологией для создания надежных и высокопроизводительных баз данных, обеспечивая при этом гибкость, необходимую для реагирования на меняющиеся бизнес-требования.

Утилиты проверки моделей и схем реализуют передовые стандарты моделирования данных, а также стандарты физического проектирования, специфичные для MySQL, поэтому при построении новых диаграмм ER или создании физических баз данных MySQL не допускаются ошибки» [18].

На рисунке 10 показана модель данных follow-up приложения для клиентов банка.

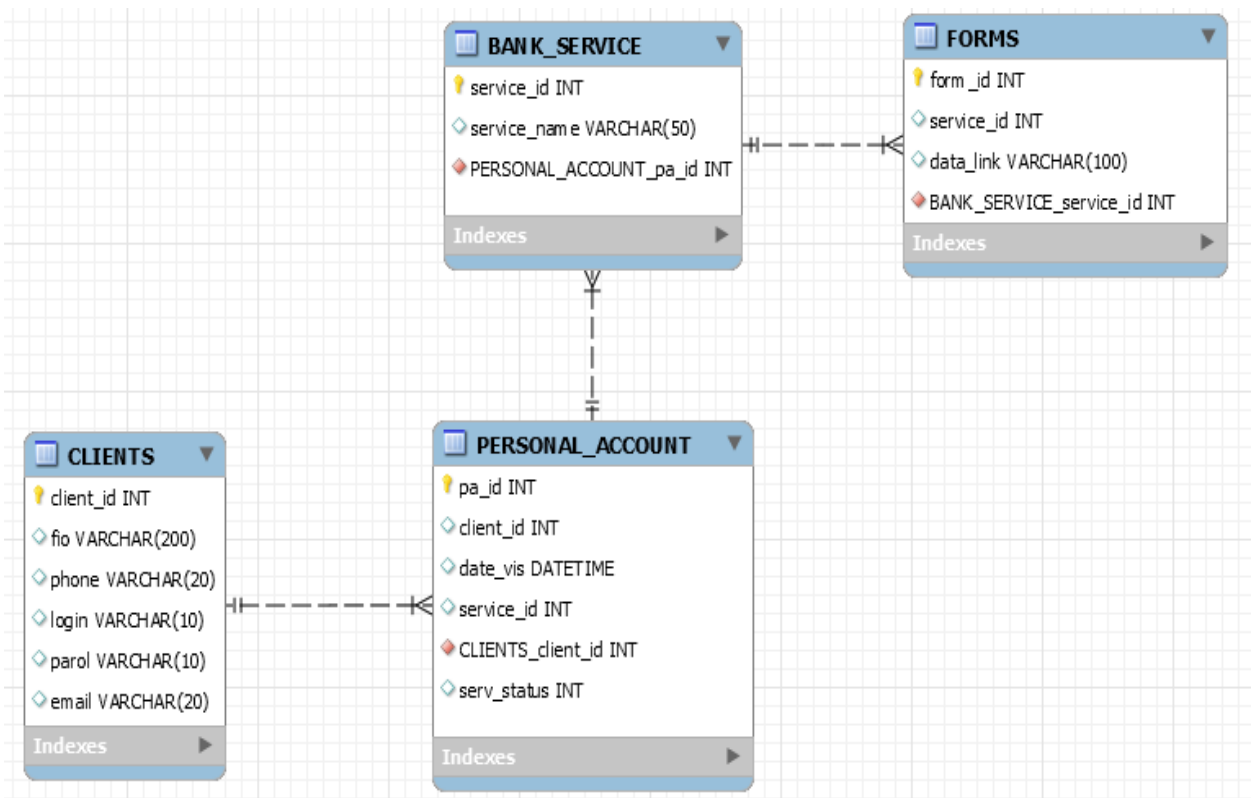


Рисунок 10 – Модель данных follow-up приложения для клиентов банка

Так как среда Workbench позволяет разрабатывать модели данных конкретно для СУБД MySQL, данная модель содержит элементы логической и физической модели данных приложения.

### 3.3 Реализация follow-up приложения для клиентов банка

Алгоритм работы follow-up приложения для клиентов банка в виде диаграммы деятельности с плавательными дорожками показан на рисунке 11.

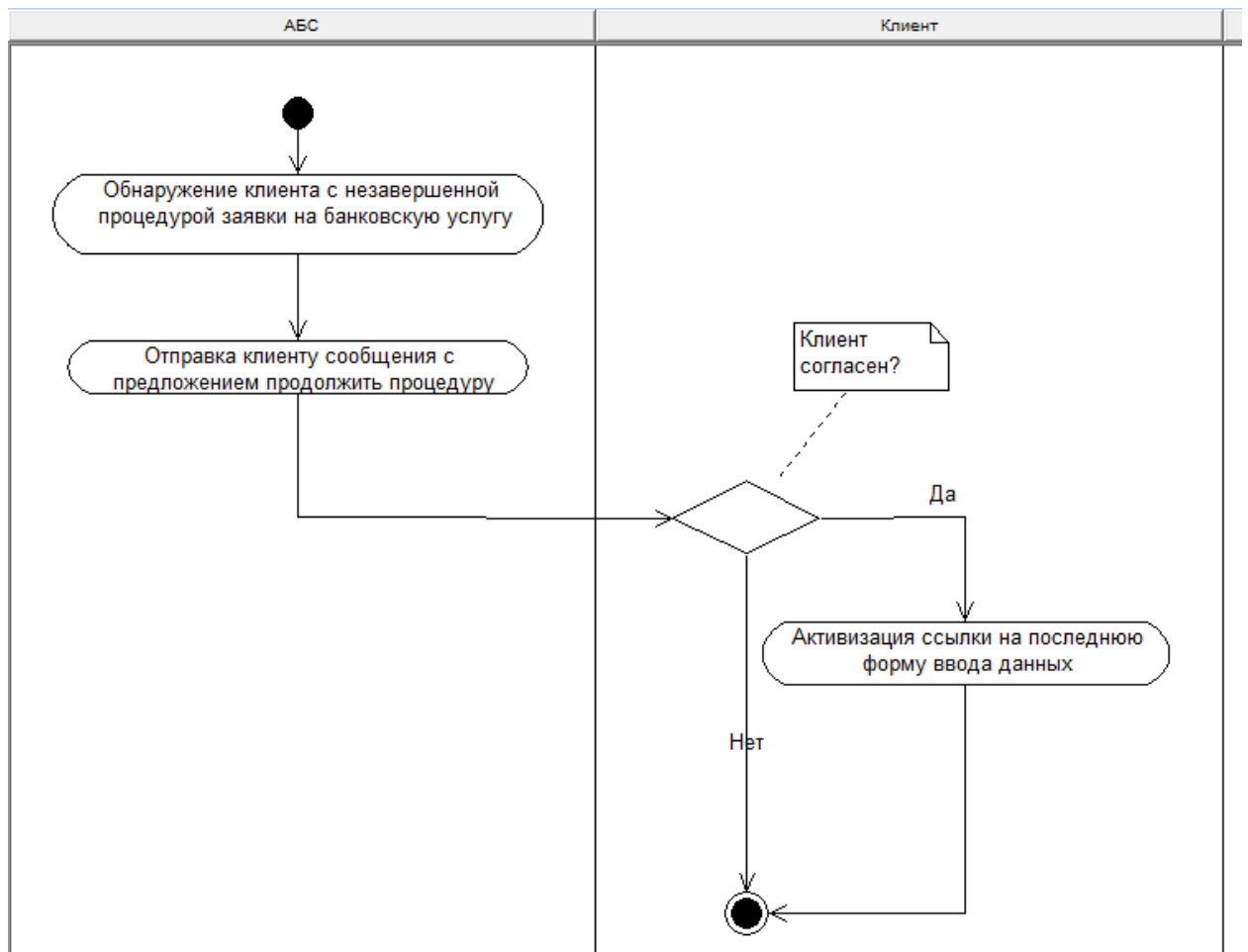


Рисунок 11 – Диаграмма деятельности follow-up приложения для клиентов банка

Работа follow-up приложения организована следующим образом:

Шаг 1. АБС обнаруживает клиента с незавершенной процедурой заявки на банковскую услугу, например, на предоставление кредита.

Шаг 2. АБС отправляет клиенту сообщение по СМС или электронной почте с предложением продолжить процедуру оформления заявки.

Шаг 3. Клиент получает сообщение и рассматривает его. Он может отказаться или принять предложение банка. Если клиент согласен с предложением, то он активизирует ссылку на последнюю форму ввода данных заявки, которая содержится в сообщении, с помощью follow-up приложения.



На рисунках 12-14 показаны скриншоты follow-up приложения.

OnlineApp

### Заявка на кредит 100% завершено

Фамилия Имя Отчество  
**Степанов Юрий Дмитриевич**

Дата рождения  
**12.08.1989**

Мобильный телефон  
**+7 (999) 999-99-99**

Персональный ежемесячный доход  
**100 000**

Серия и номер паспорта  
**3333 333333**

Желаемая сумма кредита  
**1 000 000**

Электронная почта  
**info@mail.ru**

Нажимая кнопку «Продолжить», я соглашаюсь с правилами заполнения заявки, а также даю согласие на обработку персональных данных

Оформить

Рисунок 12 – Скриншот формы заявки на кредит follow-up приложения для клиентов банка

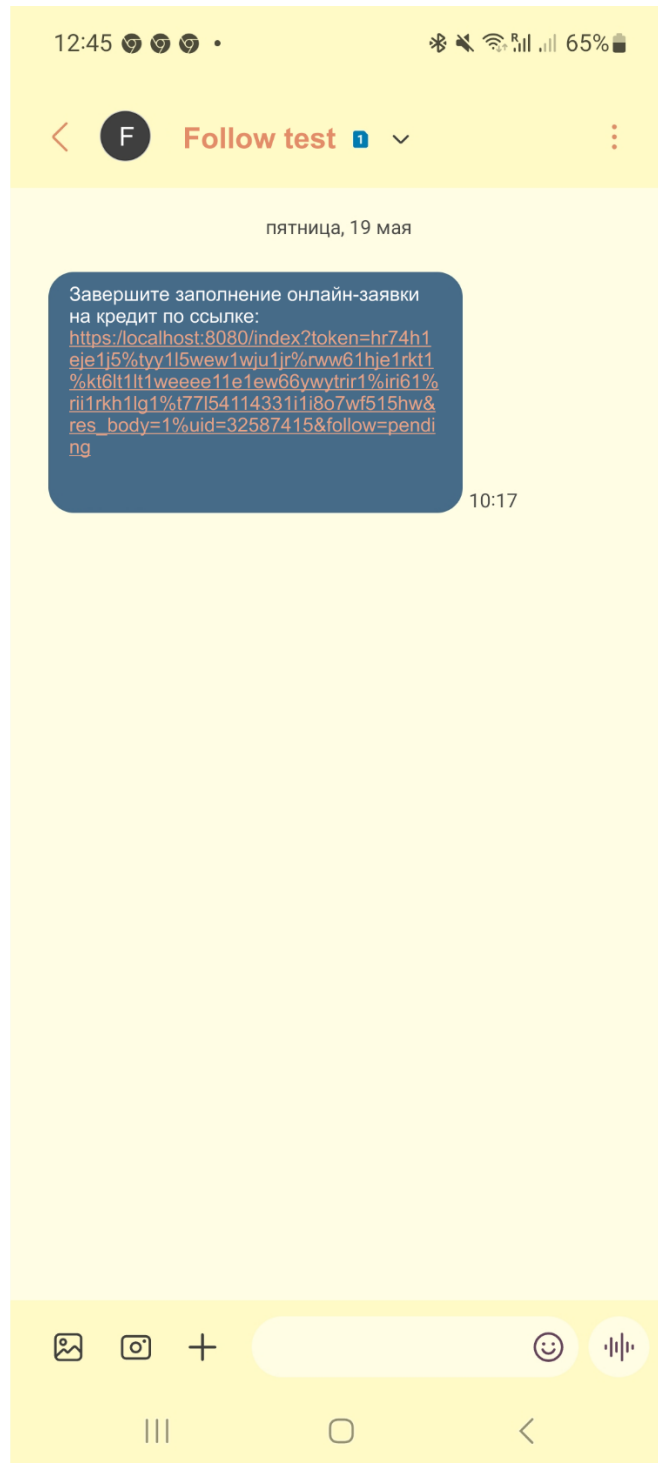


Рисунок 13 – Скриншот sms-сообщения клиенту с предложением завершить процесс оформления заявки на кредит

OnlineApp ☰

### Заявка на кредит 100% завершено

Фамилия Имя Отчество  
**Степанов Юрий Дмитриевич** ✓

Дата рождения  
**12.08.1989** ✓

Мобильный телефон  
**+7 (999) 999-99-99** ✓

Персональный ежемесячный доход  
**100 000** ✓

Серия и номер паспорта  
**3333 333333** ✓

Желаемая сумма кредита  
**1 000 000** ✓

Электронная почта  
**info@mail.ru**

**Введите смс-код**  
Мы отправили код подтверждения на номер +7 (999) 999-99-99  
Пожалуйста, введите код

2 5 7 |

**Отправить СМС-код повторно**

Рисунок 14 – Скриншот формы заявки на кредит follow-up приложения для клиентов банка с окном для ввода подтверждающего смс-кода

Фрагмент кода follow-up приложения представлен в Приложении А.

Функциональное тестирование follow-up приложения на эмуляторе среды Android Studio.

Как показали результаты тестирования, follow-up приложение позволяет клиенту без повторного входа в АБС подать заявку на требуемую услугу, что обеспечит повышение лояльности клиента банка.

### Выводы по главе 3

Результаты проделанной работы позволили сделать следующие выводы:

- архитектура follow-up приложения для клиентов банка построена по трехзвенной модели архитектуры «клиент-сервер»;
- follow-up приложение выполняет функции мобильного клиента;
- так как среда Workbench позволяет разрабатывать модели данных конкретно для СУБД MySQL, данная модель содержит элементы логической и физической модели данных приложения.

Как показали результаты тестирования, follow-up приложение позволяет клиенту без повторного входа в АБС подать заявку на требуемую услугу, что обеспечит повышение лояльности клиента банка.

## Заключение

Наличие системы отслеживания клиентов облегчит банку масштабирование бизнеса и надлежащее обучение новых сотрудников в будущем.

Для реализации механизма отслеживания клиентов используется специальное программное обеспечение – follow-up приложения.

При этом следует принять во внимание, что большинство просмотров банковских продуктов клиентами совершается через мобильные устройства.

Актуальность темы 3 бакалаврской работы заключается в обеспечении банка мобильным приложением для отслеживания клиентов.

Для поставленной в бакалаврской работе цели были решены следующие задачи:

- выполнена постановка задачи на разработку мобильного follow-up приложения для клиентов банка. Как показывает практика, эффективным способом отслеживания клиентов в банковской сфере является активизация страницы с услугой и/или незаполненной формы, на которой остановился клиент при последнем посещении сайта банка. Для решения данной задачи используются мобильные follow-up приложения. В методологии FURPS+ разработан перечень требований, который является основой для проектирования мобильного follow-up приложения для клиентов банка. Как показал сравнительный анализ, ни одно из рассмотренных приложений не отвечает полностью заданным требованиям. Таким образом, разработка мобильного follow-up приложения для отслеживания клиентов банка актуальна и представляет практический интерес;
- спроектировано мобильное follow-up приложение для клиентов банка. Как показал анализ, тип «гибридное приложение» является наиболее предпочтительным для реализации follow-up приложения

для клиентов банка. Произведен выбор среды разработки follow-up приложения. Характеристики среды Android Studio обусловили ее выбор в качестве средства для разработки мобильного follow-up приложения для клиентов банка;

- выполнены реализация и тестирование мобильного follow-up приложения для клиентов банка. Архитектура follow-up приложения для клиентов банка построена по трехзвенной модели архитектуры «клиент-сервер». В данной архитектуре follow-up приложение выполняет функции мобильного клиента. Так как среда Workbench позволяет разрабатывать модели данных конкретно для СУБД MySQL, данная модель содержит элементы логической и физической модели данных приложения.

Как показали результаты тестирования, follow-up приложение позволяет клиенту без повторного входа в АБС подать заявку на требуемую услугу, что обеспечит повышение лояльности клиента банка.

## Список используемой литературы

1. Кроссплатформенность VS нативность: как разрабатывать мобильные приложения [Электронный ресурс]. URL: <https://vc.ru/u/830518-cifrovye-privychki/367064-krossplatformennost-vs-nativnost-kak-razrabatyvat-mobilnye-prilozheniya> (дата обращения: 05.04.2023).
2. Персонализация предложений в мобильном приложении и интернет-магазине: кейс ВсеИнструменты.ру [Электронный ресурс]. URL: <https://vc.ru/marketing/201512-personalizaciya-predlozheniy-v-mobilnom-prilozhenii-i-internet-magazine-keys-vseinstrumenty-ru> (дата обращения: 05.04.2023).
3. Плюсы и минусы мобильных сайтов и приложений [Электронный ресурс]. URL: <https://artismedia.by/blog/plyusy-i-minusy-mobilnyh-sajtov-i-prilozhenij/> (дата обращения: 05.04.2023).
4. Подробное руководство по созданию архитектуры мобильных приложений [Электронный ресурс]. URL: <https://scand.com/ru/company/blog/the-ultimate-guide-to-mobile-application-architecture/> (дата обращения: 05.04.2023).
5. Топ-3 среды разработки для Android [Электронный ресурс]. URL: <https://app-android.ru/blog/environment-develop-android> (дата обращения: 05.04.2023).
6. Android Studio [Электронный ресурс]. URL: <https://www.techtarget.com/searchmobilecomputing/definition/Android-Studio> (дата обращения: 05.04.2023).
7. Apache HTTP Server [Электронный ресурс]. URL : <https://httpd.apache.org/> (дата обращения: 05.04.2023).
8. Apptivo Mobile CRM App [Электронный ресурс]. URL: <https://www.apptivo.com/answers/questions/11082/how-do-i-create-follow-up-in-mobile-crm-app> (дата обращения: 05.04.2023).

9. Customer Follow Up - Never loose a customer again! [Электронный ресурс]. URL: <https://apps.apple.com/us/app/customer-follow-up/id1481743509> (дата обращения: 05.04.2023).

10. Customer Follow-Up - Odoo 12 App [Электронный ресурс]. URL: <https://www.cybrosys.com/blog/customer-follow-ups-odoo-12-app> (дата обращения: 05.04.2023).

11. Follow up Chat - Proof Manager [Электронный ресурс]. URL: <https://apps.shopify.com/customer-follow-up> (дата обращения: 05.04.2023).

12. Harvey C. Eclipse IDE for Android Development Overview [Электронный ресурс]. URL: <https://www.developer.com/mobile/android/eclipse-ide-android-development/> (дата обращения: 05.04.2023).

13. How To Use a Customer Follow-Up System [Электронный ресурс]. URL: <https://www.getapp.com/resources/how-to-use-a-customer-follow-up-system/> (дата обращения: 05.04.2023).

14. IBM Rational Rose Enterprise 7.0.0.4 [Электронный ресурс]. URL: <https://www.ibm.com/support/pages/ibm-rational-rose-enterprise-7004-fix-pack-4-7000> (дата обращения: 05.04.2023).

15. JetBrains IntelliJ IDEA [Электронный ресурс]. URL: <https://www.jetbrains.com/> (дата обращения: 05.04.2023).

16. Mobile: Native Apps, Web Apps, and Hybrid Apps [Электронный ресурс]. URL: <https://www.nngroup.com/articles/mobile-native-apps/> (дата обращения: 05.04.2023).

17. MySQL and Android Studio Example [Электронный ресурс]. URL: <https://people.utm.my/shaharil/mysql-and-phpmyadmin-xampp-2/> (дата обращения: 05.04.2023).

18. MySQL Workbench [Электронный ресурс]. URL: <https://www.mysql.com/products/workbench/> (дата обращения: 05.04.2023).



19. Software Requirements [Электронный ресурс]. URL: <http://beervolume.com/oop/2020/software-requirements/> (дата обращения: 05.04.2023).

20. UML 2.ru – Сообщество Аналитиков [Электронный ресурс]. URL: <https://www.uml2.ru/> (дата обращения: 05.04.2023).

21. Visual Paradigm [Электронный ресурс]. URL: <https://online.visual-paradigm.com/> (дата обращения: 05.04.2023).

22. What is MySQL [Электронный ресурс]. URL: <https://www.oracle.com/mysql/what-is-mysql/> (дата обращения: 05.04.2023).

## Приложение А

### Фрагмент кода follow-up приложения

```
<!DOCTYPE html>
<html>
<head>
<title>Online App </title>
<meta http-equiv=Content-Type content="text/html;charset=UTF-8">
<meta name=viewport content="width=device-width,initial-scale=1,maximum-
scale=1.0,user-scalable=no">
<meta http-equiv=X-UA-Compatible content="IE=edge">
<meta name="robots" content="index" />
<link rel="icon" href="/img/favicon.ico" type="image/x-icon"/>
<link rel="stylesheet"
href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;500;700&disp
lay=swap"/>
<link href="/local/myhost/generic/css/build-chunker-0c196e1b.38b5aed7.css"
rel="prefetch">
<link href="/local/myhost/generic/css/build-chunker-307189cd5.75ed0865.css"
rel="prefetch">
<link href="/local/myhost/generic/css/build-chunker-49e28620.c81a5ea6.css"
rel="prefetch">
<link href="/local/myhost/generic/css/build-chunker-4ff4b79f.15514372.css"
rel="prefetch">
<link href="/local/myhost/generic/css/build-chunker-5bc77636.be702b58.css"
rel="prefetch">
<link href="/local/myhost/generic/css/build-chunker-54774709.29fde5ae.css"
rel="prefetch">
```

<link href="/local/myhost/generic/css/build-chunker-7b38b17d.dc08fd44.css"  
rel="prefetch">

#### Продолжение Приложения А

<link href="/local/myhost/generic/css/build-chunker-20zde94.b2d8a160.css"  
rel="prefetch">

<link href="/local/myhost/generic/css/build-chunker-987c5234.f803934d.css"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-0c196e1b.88ebc061.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-2d20327c.f2252cef.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-2d21e7e1.7b82a57a.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-36189cd5.064d4638.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-49e28550.280db348.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-4ff4b79f.3cff9b2a.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-5bc54636.b3a94796.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-64744909.ba72868f.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-7a38b17d.3699aae8.js"  
rel="prefetch">

<link href="/local/myhost/generic/js/build-chunker-920bde94.3eadf02c.js"  
rel="prefetch">

```
<link href="/local/myhost/generic/js/build-chunker-987c5274.1e78a0a1.js"
rel="prefetch">
```

```
<link href="/local/myhost/generic/css/build-chunker-vendors.5ec44d29.css"
rel="stylesheet">
```

#### Продолжение Приложения А

```
<link href="/local/myhost/generic/css/app.7128fff7.css" rel="stylesheet">
```

```
<script>
```

```
sessionStorage.setItem("generic_token");
```

```
const indexUrlBuilder = [
```

```
  "/index/credit/"
```

```
];
```

```
const allocationFull = [
```

```
  "/index/follow/"
```

```
]
```

```
let allocationFull = param.indexOf(u);
```

```
let firstPageReally = indexUrlBuilder.indexOf(u);
```

```
let me = window.location.pathname;
```

```
function getCookie(name) {
```

```
  let matches = document.cookie.match(new RegExp(
```

```
    "(?:^|; )" + name.replace(/[\. $?*\{\}\(\)\[\]\|\^\+]/g, "\\$1') + "=(^[^;]*)"
```

```
  ));
```

```
  return matches ? decodeURIComponent(matches[1]) : undefined; }
```

```
if(firstPageReally !== -1) {
```

```
  // проверяю в локальном хранилище, что это не первая страница по объектам
  из реквеста
```

```
  sessionStorage.setItem("generic", "03br774-44i8-1fg2-81e1-141733d0err5");
```

```
  sessionStorage.setItem("generic_form_id", "credit");
```

```
sessionStorage.setItem("generic_curl_find", window.location.search);  
sessionStorage.setItem("generic_curl_section", 2);  
sessionStorage.setItem("generic_drop", 1);  
sessionStorage.setItem("generic_canal", 9);
```

Продолжение Приложения А

```
if (!esia && !backToFirst && !dp) {  
  sessionStorage.setItem("vuex", "");  
  }  
}  
</script>
```