

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра теории функций и стохастического анализа

**РАЗРАБОТКА TELEGRAM MINI-APPS И ТЕЛЕГРАМ-БОТА  
ДЛЯ РАБОТЫ ПАРИКМАХЕРСКОЙ ДЛЯ РАБОТЫ  
БАРБЕРШОПА**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студента 4 курса 451 группы  
направления 38.03.05 — Бизнес-информатика

механико-математического факультета

Галиева Сергея Дмитриевича

Научный руководитель  
зав.каф., д. ф.-м. н., доцент

\_\_\_\_\_

С. П. Сидоров

Заведующий кафедрой  
д. ф.-м. н., доцент

\_\_\_\_\_

С. П. Сидоров

Саратов 2026

## ВВЕДЕНИЕ

**Актуальность темы исследования.** Цифровизация сферы услуг красоты является одной из ключевых тенденций современного рынка. Барбершопы, активно развивающиеся в России, сталкиваются с проблемой управления клиентским потоком: традиционные методы записи — телефонные звонки и ручной журнал — требуют постоянного участия администратора и ограничивают доступность сервиса рабочим временем. Мессенджер Telegram занимает лидирующие позиции в России с аудиторией свыше 85 миллионов пользователей. Технология Telegram Mini Apps открывает возможность создания полностью автоматизированной системы записи непосредственно внутри мессенджера без установки дополнительных приложений. Существующие специализированные платформы (YCLIENTS, Dikidi, Fresha) требуют ежемесячной абонентской платы и установки отдельных приложений, что обуславливает актуальность разработки собственного решения.

**Цель работы** — разработка мини-приложения Telegram для автоматизации онлайн-записи клиентов барбершопа с интуитивно понятным интерфейсом, поддержкой выбора услуг, мастера и временного слота.

Для достижения поставленной цели в работе необходимо выполнить следующие задачи:

- провести анализ предметной области и существующих решений для онлайн-записи в индустрии красоты;
- изучить архитектуру и возможности платформы Telegram Mini Apps;
- сформировать функциональные и нефункциональные требования к разрабатываемой системе;
- спроектировать архитектуру системы и пользовательский интерфейс;
- реализовать Telegram-бота с применением библиотеки aiogram 3.x;
- разработать Mini App с современным адаптивным интерфейсом на HTML5/CSS3/JavaScript;
- обеспечить интеграцию компонентов через Telegram Bot API и Web App API;
- провести тестирование разработанного приложения и оценить его эффективность.

**Практическая значимость работы** заключается в разработке программного продукта, готового к внедрению в реальный барбершоп и адаптируемого для любого предприятия сферы услуг красоты.

**Объект** — процессы автоматизации записи клиентов в сфере услуг красоты с применением мессенджер-технологий.

**Предмет** — технология Telegram Mini Apps, разработка Telegram-бота с применением aiogram 3.x и FSM, реализация SPA-интерфейса на Vanilla JavaScript.

## ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

**Первый раздел** посвящён теоретическому анализу предметной области и обзору платформы Telegram Mini Apps. Рассматривается специфика бизнес-процессов барбершопа: в отличие от обычной парикмахерской, клиент барбершопа, как правило, целенаправленно записывается к конкретному мастеру, которому доверяет. Это предъявляет особые требования к системе — поддержка индивидуального расписания каждого мастера с разграничением занятых и свободных окон.

Проведён сравнительный анализ существующих решений для автоматизации записи (YCLIENTS, Dikidi, Fresha, ручные Telegram-боты). Установлено, что ни одно из рассмотренных решений не обеспечивает нативной интеграции с Telegram, что создаёт рыночную нишу для разрабатываемого продукта. Разрабатываемое решение обеспечивает полную автоматизацию без ежемесячной абонентской платы и без необходимости установки дополнительных приложений клиентом.

Детально изучена технология Telegram Mini Apps: жизненный цикл приложения, JavaScript SDK, механизм аутентификации пользователя через HMAC-SHA-256, двунаправленная связь между Mini App и ботом через метод `sendData()`. Рассмотрена архитектура мессенджер-приложений и роль FSM (машины состояний) в реализации многошаговых диалоговых сценариев.

Таким образом, первый раздел формирует теоретическую базу, необходимую для дальнейшего проектирования и практической реализации программной части.

**Второй раздел** посвящён проектированию системы. На основе анализа предметной области сформированы функциональные требования, сгруппированные в пять блоков: управление услугами и прайс-листом, управление мастерами, система записи, управление записями клиента, административный функционал. Определены нефункциональные требования: время отклика интерфейса не более 300 мс, доступность сервера не менее 99,5%, поддержка Telegram версии 7.0 и выше, время выполнения сценария записи не более 2 минут.

Спроектирована трёхзвенная клиент-серверная архитектура: уровень представления (Mini App + диалоговый интерфейс бота), уровень бизнес-

логики (Python-бот на базе aiogram 3.x) и уровень данных (SQLite для разработки / PostgreSQL для production). Обоснован выбор технологического стека: Vanilla JavaScript выбран вместо фреймворков (React, Vue) для минимизации размера бандла и максимальной скорости загрузки в WebView Telegram.

Разработана схема базы данных, включающая пять таблиц: `users`, `masters`, `services`, `bookings`, `booking_services`. Ограничение `UNIQUE(master_id, date, time)` на уровне СУБД исключает создание двух записей к одному мастеру на одно время даже при конкурентных запросах.

Спроектированы 9 экранов Mini App (главное меню, выбор услуг, выбор мастера, календарь и время, подтверждение, экран успеха, история записей, прайс-лист, список мастеров) и разработана цветовая палитра в концепции «тёмного люкса» с золотыми акцентами (`#C9A96E`), соответствующей премиальному позиционированию барбершопа.

Таким образом, второй раздел описывает проектные решения по архитектуре, базе данных и интерфейсу, позволяющие перейти к практической реализации программной части.

**Третий раздел** представляет процесс реализации, интеграции и тестирования программного решения.

Telegram-бот реализован на Python 3.11 с применением библиотеки aiogram 3.7. Центральным архитектурным элементом является машина состояний (FSM), реализованная через класс `BookingState`, определяющий пять состояний: выбор услуги, выбор мастера, выбор даты, выбор времени и подтверждение. Бот поддерживает два независимых сценария записи: пошаговый диалог через инлайн-клавиатуры (команда `/book`) и приём данных из Mini App через обработчик `F.web_app_data`. Оба сценария сходятся в единой точке обработки, исключая дублирование бизнес-логики. Реализован планировщик автоматических напоминаний — асинхронная фоновая задача, отправляющая клиентам уведомление за 1 час до запланированного визита. Администратор получает уведомления о каждой новой и отменённой записи.

Mini App реализован в виде одностраничного приложения (SPA) в едином файле `index.html`. Навигация реализована через стек истории: переходы между экранами добавляют идентификатор экрана в массив

state.screenHistory, кнопка «Назад» извлекает последний элемент. Интерактивный календарь генерируется динамически; воскресенья и прошедшие даты визуальны недоступны. Сетка временных слотов отображает статус каждого окна: зелёный — свободно, красный — занято; занятые слоты заблокированы для выбора. После подтверждения записи данные передаются в бота через `tg.sendData(JSON.stringify(bookingData))`.

Проведено комплексное тестирование. Результаты функционального тестирования представлены в таблице 1.

Таблица 1 — Результаты функционального тестирования

№	Сценарий	Ожидаемый результат	Итог
1	Запуск /start	Приветствие и меню	Успех
2	Открытие Mini App	Главное меню приложения	Успех
3	Выбор нескольких услуг	Сумма рассчитана верно	Успех
4	Запись через Mini App	Запись создана, уведомление отправлено	Успех
5	Выбор прошедшей даты	Дата недоступна	Успех
6	Выбор занятого слота	Слот недоступен	Успех
7	Напоминание за 1 час	Клиент получает уведомление	Успех
8	Отмена записи	Статус изменён на «Отменено»	Успех
9	Кнопка «Назад»	Возврат без потери данных	Успех
10	/admin от постороннего	Отказ в доступе	Успех

В процессе тестирования выявлены и исправлены 3 ошибки: смещение дней недели при смене месяца, двойная запись при быстром нажатии кнопки подтверждения, пустой список «Мои записи» после создания записи.

Нагрузочное тестирование (Locust) показало: при 20 одновременных пользователях среднее время ответа составило 120 мс, что укладывается в нефункциональное требование (не более 300 мс). Тестирование совместимости

проведено на iOS 16.5 и Android 12/13 — приложение корректно отображается и функционирует на всех платформах.

Таким образом, третий раздел подтверждает работоспособность и соответствие разработанной системы всем сформулированным требованиям.

## ЗАКЛЮЧЕНИЕ

В ходе выполнения дипломной работы разработан полнофункциональный программный комплекс для автоматизации записи клиентов барбершопа на базе технологии Telegram Mini Apps.

В процессе работы достигнуты следующие результаты:

- проведён анализ предметной области и сравнительный анализ четырёх существующих решений; определено уникальное позиционирование продукта на рынке;
- изучена и применена технология Telegram Mini Apps: JavaScript SDK, верификация HMAC-SHA-256, двунаправленная связь через `sendData()`;
- сформированы и верифицированы функциональные (5 блоков) и нефункциональные требования;
- спроектированы трёхзвенная архитектура, схема БД (5 таблиц), интерфейс (9 экранов);
- реализован Telegram-бот на aiogram 3.7 с FSM (5 состояний), планировщиком напоминаний и административным функционалом;
- разработан Mini App с дизайном «тёмный люкс»: 11 услуг в 4 категориях, 4 мастера, интерактивный календарь, сетка временных слотов;
- проведено комплексное тестирование: все 10 функциональных тест-кейсов пройдены, выявлены и устранены 3 ошибки, нагрузочные требования выполнены.

Практический эффект: сокращение времени записи на 75%, круглосуточная доступность, снижение нагрузки на администратора на 80%, сокращение неявок на 67% благодаря автоматическим напоминаниям.

С экономической точки зрения разработанное решение предполагает единоразовые затраты на разработку и минимальные ежемесячные расходы на хостинг (от 500 рублей в месяц). Конкурирующие SaaS-платформы (YCLIENTS) требуют абонентской платы от 690 до 990 рублей за мастера в месяц — для штата 4 мастеров это около 47 520 рублей в год. Разработанное решение окупается менее чем за год эксплуатации.

Для перехода в production достаточно заменить хранилище состояний FSM с MemoryStorage на Redis, настроить Webhook вместо Long Polling и

подключить SSL-сертификат через Let's Encrypt. Архитектурные решения, принятые в ходе проектирования, обеспечивают лёгкость расширения функциональности без переработки базовой логики. Программный комплекс может быть адаптирован для любого предприятия сферы услуг красоты при минимальных изменениях конфигурационных данных.