

УДК 004.42+681.84

## ПРОГРАММНОЕ СРЕДСТВО ДЛЯ ОБМЕНА ГОЛОСОВЫМИ СООБЩЕНИЯМИ МЕЖДУ ПОЛЬЗОВАТЕЛЯМИ НА ОСНОВЕ HTTP-ИНТЕРФЕЙСА TELEGRAM BOT API

*Германович Д.О., студент гр.181073*

*Белорусский государственный университет информатики и радиоэлектроники,  
Институт информационных технологий,  
г. Минск, Республика Беларусь*

*Кунцевич О.Ю. – канд. пед. наук, доцент, доцент кафедры ИСиТ*

**Аннотация.** Рассмотрены вопросы разработки программного средства для социальной сети, позволяющего пользователям взаимодействовать через текстовые и аудиоформаты, управлять подписками и профилями, а также удалять свою учетную запись с сохранением конфиденциальности данных. Представлен концепт пользовательского интерфейса приложения, обоснован выбор средств разработки, таких как Quarkus и Kotlin, для обеспечения производительности и масштабируемости. Перечислены достоинства и недостатки применения разработки в виде Web-приложения. В качестве языка программирования выбран язык Kotlin. Отличительной чертой ПС от аналогов являются: поддержка аудиоформата в биографиях, удобный интерфейс взаимодействия через Telegram и гибкая система управления подписками.

**Ключевые слова.** Программное обеспечение, разработка приложений, аудиомессенджер, голосовые сообщения.

### **Введение.**

Аудиомессенджеры занимают важное место в современном цифровом пространстве, предлагая уникальный формат взаимодействия. Они позволяют передавать эмоции и интонации, что делает общение более живым и личным. Тем не менее, текущие разработки часто фокусируются на ограниченном наборе функций, что сужает возможности пользователей [1].

Существующие решения зачастую недостаточно адаптированы для гибкого управления профилем, эффективного использования аудиоконтента или обеспечения высокой вовлеченности пользователей. Это указывает на необходимость создания программного средства, которое сочетает удобство голосового общения с расширенными функциями управления контентом и взаимодействия между пользователями.

В современном мире мессенджеры стали неотъемлемой частью повседневного общения. Одним из самых популярных мессенджеров для создания чат-ботов является Telegram. Через Telegram пользователи могут обмениваться сообщениями, создавать группы, отправлять файлы и получать информацию из множества источников. Для разработчиков Telegram предлагает мощный API (интерфейс программирования приложений), который позволяет интегрировать мессенджер с различными сервисами.

Чат-бот представляет собой программу, которая имитирует общение с реальным человеком. Он может работать через текстовые или голосовые сообщения в мессенджерах, на вебсайтах, в мобильных приложениях или даже по телефону. Чат-боты играют важную роль в продвижении товаров и услуг, обеспечивая быструю и эффективную коммуникацию с клиентами. Они помогают ответить на вопросы пользователей, предоставляют рекомендации по выбору продуктов и предлагают персонализированные предложения. Одним из главных преимуществ чат-ботов является автоматизация, которая позволяет значительно сократить расходы и повысить эффективность бизнеса. Благодаря своей популярности и огромной аудитории Telegram предоставляет отличную платформу для создания ботов, которые могут достичь множества пользователей. Разработать бота можно на любом языке программирования, от PHP и Node.js до Java и Go [2].

Аудиомессенджер – это программное средство, позволяющее пользователям обмениваться голосовыми сообщениями, управлять профилями и подписками, а также получать уведомления. Разработка аудиомессенджера базируется на современном подходе, сочетающем удобство использования и техническую производительность [3].

Разработанное программное решение предоставляет следующие возможности:

- запись и отправка аудиосообщений;
- получение уведомлений о новых сообщениях;
- управление профилем и биографией, включая аудио- и текстовые описания;
- подписка на других пользователей и получение обновлений.

Аналоги имеют несколько ключевых недостатков, которые ограничивают возможности пользователей и снижают удобство использования существующих аудиомессенджеров.

Ключевые недостатки:

- ограниченные возможности для асинхронного прослушивания и публикации аудиоконтента;

- отсутствие полноценной социальной сети, которая могла бы позволить пользователям легко взаимодействовать, комментировать записи, подписываться на интересующих их людей и поддерживать более широкие социальные связи;
- ориентация приложений на узкие форматы использования или на определенные социальные круги.

Разработанное программное средство (ПС) стремится устранить эти ограничения. Оно будет поддерживать публикацию записей для асинхронного прослушивания, предлагать развитые социальные функции для взаимодействия с пользователями.

**Основная часть.**

Определим назначение и возможности ПС:

1. Главной задачей разработанного приложения является облегчение обмена голосовыми сообщениями между пользователями через мессенджер Telegram. Приложение будет предоставлять пользователю удобный интерфейс для отправки и получения голосовых сообщений, делая процесс общения более быстрым и эффективным.

2. Приложение автоматизирует работу с голосовыми сообщениями, позволяя пользователям записывать, отправлять и прослушивать аудиосообщения с минимальными усилиями. Это обеспечит улучшенную обратную связь и оперативное решение запросов пользователей в реальном времени.

3. Аудиомессенджер интегрирован с Telegram Bot API, что позволит использовать все возможности мессенджера, включая создание групп, обмен файлами и информацию о пользователях, в сочетании с голосовой коммуникацией. Это обеспечит пользователям широкий спектр функций и удобный доступ к различным сервисам.

4. Приложение обеспечит обработку и хранение голосовых сообщений, предоставляя пользователям возможность прослушивать историю сообщений и управлять их хранением. Все данные будут храниться с учетом конфиденциальности информации.

5. ПС поддерживает возможность интеграции с внешними сервисами, что позволит использовать его для различных бизнес- и социальных целей, таких как голосовые уведомления, консультации или напоминания.

Для более глубокого понимания архитектуры разрабатываемого аудиомессенджера, представим его взаимодействие с внешними системами, как показано на рисунке 1.

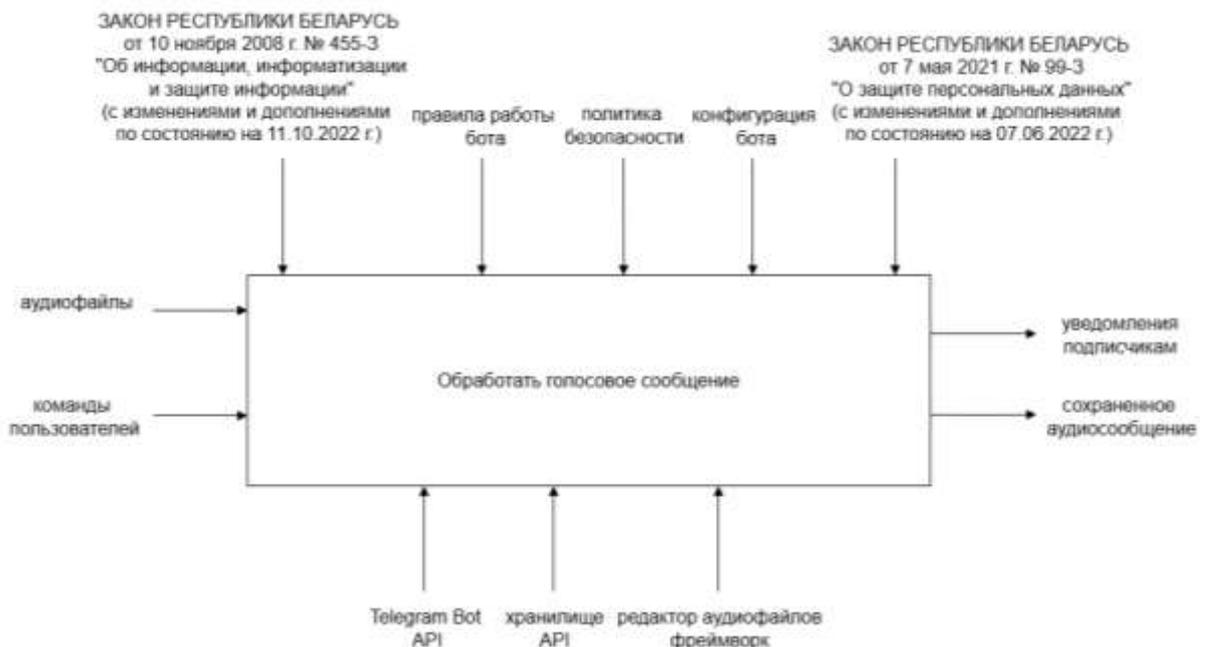


Рисунок 1 – Контекстная диаграмма A-0

Уникальность и отличия разработки от аналогичных приложений заключается, в частности, в следующих пунктах:

- интуитивно понятный интерфейс и минималистичный дизайн;
- возможность мгновенной регистрации с использованием существующего аккаунта Telegram;
- персонализированные настройки и функциональные возможности чат-бота в зависимости от потребностей пользователя;
- подключение различных внешних систем и сервисов, такие как базы данных, API, облачные хранилища;

- использование проверенных, современных технологий и библиотек;
- возможность отправки голосовых сообщений [4].

Для запуска аудиомесседжера необходимо его настроить. Конфигурационные файлы находятся в коренной директории проекта и называется «application.yaml» и «.env». Эти файлы содержат основные параметры, необходимые для работы приложения. Содержимое файла «application.yaml» представлено на рисунке 3.

```
telegram:
  bot:
    auth-token: '123:token' # OVERRIDE
    tgapi-port: 8081
  webhook:
    url: 'http://localhost:${quarkus.http.port}/api/v1/telegram/webhook'
    token: 'super-secret'
    max-connections: 80

storage:|
  local-object-storage: './local-object-storage'
  local-fs: './local-fs'

quarkus:
  rest-client:
    tg-bot-api:
      url: 'http://localhost:${telegram.bot.tgapi-port}/bot${telegram.bot.auth-token}'
    tg-file-api:
      url: 'http://localhost:${telegram.bot.tgapi-port}/bot${telegram.bot.auth-token}'
  s3:
    devservices:
      enabled: false
    aws:
      region: 'us-east-1'
  scheduler:
    enabled: true
  log:
    file:
      enable: true
      path: 'bewired_app.log'
      format: '%d{yyyy-MM-dd HH:mm:ss} %-5p [%c] (%t) %s%n'
      rotation:
        max-backup-index: 99
        max-file-size: '10M'
  resteasy-reactive:
    path: '/api'
  datasource:
    jdbc:
      driver: 'org.postgresql.Driver'
  hibernate-orm:
    dialect: 'org.hibernate.dialect.PostgreSQLDialect'
    log:
      sql: true
      sql-load-script: 'no-file'
    database:
      generation: 'none'
  swagger-ui:
    always-include: true
    theme: 'flattop'
  kotlin-serialization:
    json:
```

Рисунок 3 – Структура файла «application.yaml»

Этот файл используется для настройки параметров, относящихся к работе приложения:

1. Данные для подключения базы данных. Настраиваются параметры доступа к PostgreSQL, такие как URL базы данных, имя пользователя, пароль, а также дополнительные параметры Hibernate.
2. Параметры Telegram Bot API. Указывается токен аутентификации, порт API, URL вебхука, а также дополнительные настройки взаимодействия с Telegram.
3. Настройки хранилища. Указываются локальные пути к файлам и параметры интеграции с S3-совместимыми хранилищами для хранения аудиозаписей и других данных.
4. Логирование. Настраивается уровень логирования, путь к файлу лога, формат сообщений, а также параметры ротации логов.
5. REST-клиенты. Указываются URL API для работы с Telegram и файлового взаимодействия.

На диаграмме вариантов использования представлены актеры и их взаимодействие с ПС:

Гость – этот актер представляет пользователя, который ещё не прошёл процесс регистрации. Он может выполнять минимальный набор действий, таких как регистрация и выбор способа авторизации.

Пользователь – зарегистрированный пользователь, имеющий доступ к основным функциям системы, таким как запись, редактирование и удаление аудиосообщений, редактирование профиля и просмотр статистики [5].

Основные сценарии использования:

1. Регистрация. Пользователь, начинающий работу с системой, проходит процесс регистрации, который может включать выбор способа авторизации. После успешной регистрации гость становится полноценным пользователем системы.

2. Запись аудиосообщения. Основной сценарий использования аудиомессенджера. Пользователь может записать новое аудиосообщение, а также удалить или редактировать его. Есть возможность расширения сценария удаления аудиосообщения через подтверждение удаления.

3. Редактирование профиля. Сценарий, при котором пользователь может изменять информацию о себе, такие как текстовые данные, аудиоинформацию и уведомления. Этот сценарий включает действия, направленные на редактирование конкретных аспектов профиля, например, редактирование текстовой информации или аудиосообщений.

4. Просмотр статистики. Пользователь может просматривать статистику по подписчикам и просмотрам его аудиосообщений, что помогает анализировать популярность и активность его контента. Этот сценарий использования включает статистические данные о количестве подписчиков и количестве просмотров аудиосообщений. Включение этой функции в систему важно с точки зрения мотивации пользователей к созданию большего количества качественного контента.

5. Подписка на пользователя. В этом сценарии реализована возможность отправки и получения заявок на подписку. Пользователь может просматривать заявки на подписку, принимать или отклонять их. Этот сценарий расширяется возможностью разрешить или отказать в подписке.

6. Внесение изменений в настройки. Пользователь может изменить язык системы, часовой пояс, а также редактировать уведомления, что расширяет возможности персонализации приложения.

#### **Заключение.**

Разработанное ПС для обмена голосовыми сообщениями в Telegram будет отображать удобный и интуитивно понятный интерфейс для пользователей, включает несколько модулей, таких как регистрация, отправка и получение голосовых сообщений, управление настройками и интеграция с внешними сервисами. Приложение выполняет функции быстрого обмена аудиосообщениями, автоматической обработки запросов, а также персонализированных уведомлений и рекомендаций для пользователей. Главное преимущество перед аналогами заключается в интеграции с Telegram API для простоты использования, быстрой регистрации через Telegram и высокой гибкости, что позволяет адаптировать функциональность под индивидуальные потребности пользователей.

#### **Список использованных источников:**

1. Харрис С. Мобильные приложения для обмена голосовыми сообщениями. / С. Харрис. – СПб.: БХВ-Петербург, 2019. – 256 с.: ил.
2. Уиллсон Р. Программирование Telegram ботов. Руководство по использованию Telegram API. / Р. Уиллсон. – М.: Вильямс, 2020. – 352 с.: ил.
3. Браун К. Голосовые мессенджеры: Технологии и будущее. / К. Браун. – М.: Вильямс, 2022. – 512 с.: ил.
4. Дэвидсон С. Использование Telegram API для создания ботов и интеграций. / С. Дэвидсон. – 2-е изд. – СПб.: Питер, 2022. – 464 с.: ил.
5. Ли А. Разработка аудиомессенджеров на платформе Android. / А. Ли. – М.: Эксмо, 2018. – 448 с.: ил.

UDC 004.42+681.84

## **SOFTWARE FOR EXCHANGING VOICE MESSAGES BETWEEN USERS BASED ON THE HTTP INTERFACE OF THE TELEGRAM BOT API**

*Germanovich D.O., student*

*Institute of Information Technologies of the Belarusian State University of Informatics and Radioelectronics,  
Minsk, Republic of Belarus*

*Kuntsevich O.Yu. – Candidate of Pedagogic Sciences, Associate Professor*

**Annotation.** This article discusses the development of software for a social networking application with dating features, allowing users to interact via text and audio formats, manage subscriptions and profiles, as well as deleting their accounts while maintaining data privacy. The concept of the application's user interface is presented, and the choice of development tools, such as Quarkus and Kotlin, is justified to ensure performance and scalability. The advantages and disadvantages of implementing the solution as a web application are listed. Kotlin was chosen as the programming language. The distinguishing feature of the software compared to similar solutions is the support for audio format in biographies, a convenient interface for interaction via Telegram, and a flexible subscription management system.

**Keywords.** Software, application development, audio messenger, voice messages.