

Я. И. Головки^{1✉}, Е. Ю. Воронкин¹

Разработка программного бота для ранжирования списков абитуриентов и информирования о вступительных испытаниях

¹Сибирский государственный университет геосистем и технологий,
г. Новосибирск, Российская Федерация
e-mail: golovko085@gmail.com

Аннотация. Разбирается процесс создания программного бота для автоматизации деятельности приемной комиссии СГУГиТ. Актуальность проекта следует из потребностей в автоматизированной информационной системе для уведомления абитуриентов о вступительных испытаниях, а также для просмотра абитуриентам их мест в конкурсных списках. Для разработки основной логической части использовался Golang, а для разработки интерфейса JavaScript, для хранения данных использовался SQLite, JWT использовался для механизма авторизации. Были предусмотрены возможности просмотра конкурсной позиции, подбора направлений по предметам ЕГЭ, подписка на уведомления об изменениях, также средства административной панели для создания, изменения, удаления специальностей, загрузки выгрузки конкурсных списков. Также был сделан дополнительный акцент на информационной безопасности системы. С помощью данной работы происходит демонстрация применения программных ботов в работе приемной комиссии.

Ключевые слова: веб-программирование, golang, javascript, sqlite, jwt

Y. I. Golovko^{1✉}, E. Yu. Voronkin¹

Development of a program bot to shuffle entrants lists and inform about entrance contests

¹Siberian State University of Geosystems and Technologies, Novosibirsk, Russian Federation
e-mail: golovko085@gmail.com

Abstract. The development of a software bot is described to automate the work of the admissions committee at Saratov State University of Geography and Technology. The project's relevance lies in the automated information system for receiving notifications to applicants about upcoming tests, as well as for viewing their places on the competition lists. Golang was used to develop the main logic, JavaScript was used for the interface, SQLite was used for data storage, and JWT was used for the authorization mechanism. There are included possibilities of browsing competition positions, choosing subjects based on Unified State Exam subjects, subscribing to change notifications, and administrative panel tools for working with specializations, as well as uploading and downloading competition lists. Additional emphasis was placed on information security. This project demonstrates the use of software bots in the work of the admissions committee.

Keywords: web programming, golang, javascript, sqlite, jwt

Введение

Современные образовательные учреждения нуждаются в эффективных цифровых инструментах для автоматизации процессов взаимодействия с абитуриентами [1, 2]. Приемные комиссии вузов ежедневно обрабатывают большое коли-

чество обращений, связанных с ранжированием конкурсных списков, информированием о вступительных испытаниях, а также предоставлением актуальных данных о поступлении. Использование традиционных способов информирования, таких как телефонные звонки, электронная почта и публикация информации на сайте, увеличивает нагрузку на сотрудников приемной комиссии и не всегда обеспечивает оперативное получение информации абитуриентами.

На данный момент значительная часть информации о ходе приемной кампании публикуется на официальном сайте СГУГиТ, однако данный способ не всегда является удобным для абитуриентов, поскольку требует постоянного самостоятельного отслеживания изменений. По этой причине было принято решение о разработке автоматизированного средства информирования, позволяющего упростить взаимодействие между абитуриентами и приемной комиссией.

Целью настоящей работы является разработка программного бота для ранжирования списка абитуриентов и информирования о вступительных испытаниях. Чтобы выполнить поставленную цель, было достигнуто выполнению определенных задачи:

- анализ существующих решений и сравнение с разрабатываемым проектом;
- проектирование архитектурной части основных модулей системы;
- разработка серверной программной части на языке программирования Golang;
- создание клиентской части на стеке технологий JavaScript, HTML, CSS;
- создание базы данных с использованием SQLite;
- авторизация через JWT;
- тестирование и включение проекта в рабочий процесс приемной комиссии.

Методы и материалы

Первым шагом необходимо спроектировать все необходимые диаграммы [3, 4].

Для правильного понимания взаимодействия различных пользователей, как админ панели, так и программного бота, была создана диаграмма прецедентов [5, 6] (рис. 1).

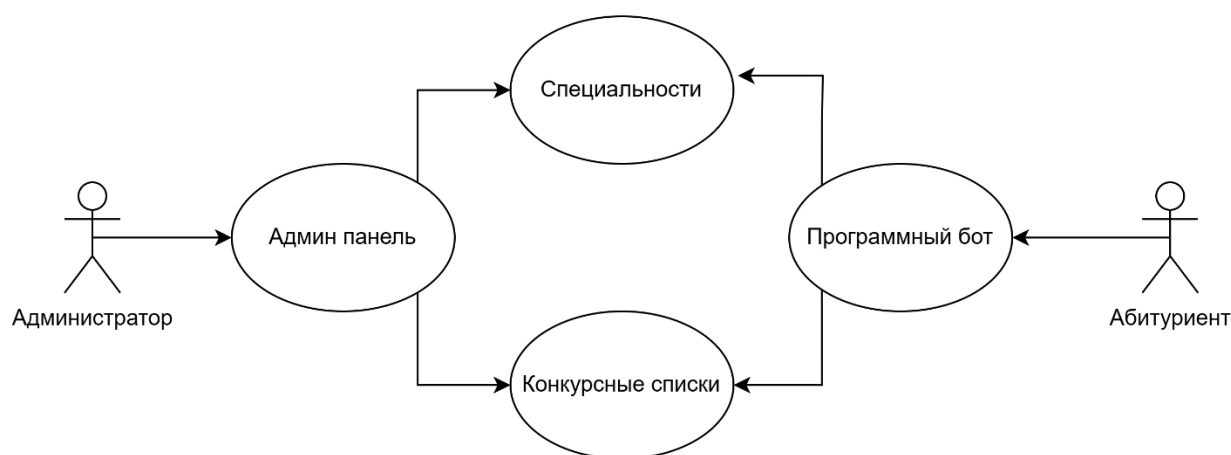


Рис. 1. Диаграмма прецедентов

Затем была сделана схема архитектуры проекта, она позволяет понять то, как части системы, в данном случае Telegram-бота и админ панели, будут взаимодействовать между собой [7, 8] (рис. 2).



Рис. 2. Архитектурная схема

Разработка

Далее был определен набор технологий проекта.

Для серверной части был выбран админ-панели Golang [9, 10]. Интерфейс был разработан с помощью HTML, CSS и JavaScript [11, 12].

Для разработки Telegram бота, также использовался Golang [13, 14].

Для хранения данных админ панели и бота используется SQLite и JSON, но для хранения конкурсных списков использовался XML [15, 16].

Далее был определен набор команд для бота, «Просмотр специальности для поступления» (рис. 3), «Посмотреть место в рейтинге» (рис. 4), «Подписаться на изменения» (рис. 5) [17, 18].

После был определен набор функциональных требований к админ-панели, Авторизация, выход, удаление, изменение, добавление специальности, выгрузка и загрузка конкурсных списков, смена языка и световой темы (рис. 6) [19, 20].

Далее был разработан определенный функционал.

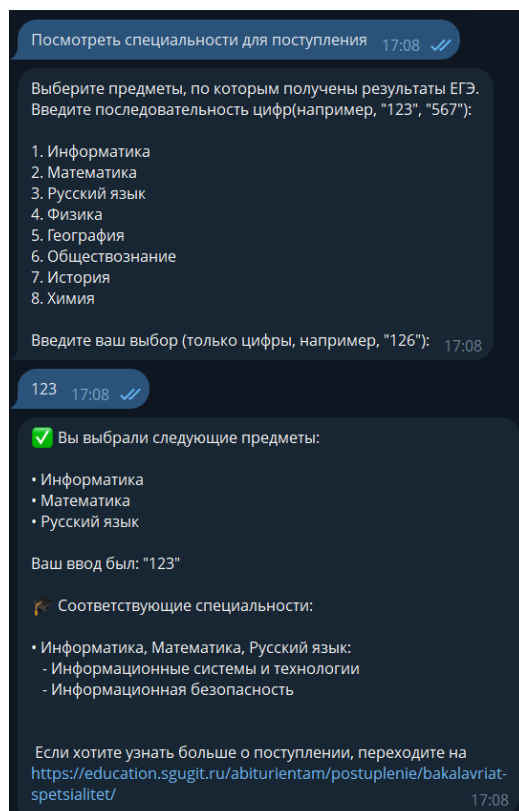


Рис. 3. Просмотр специальности для поступления

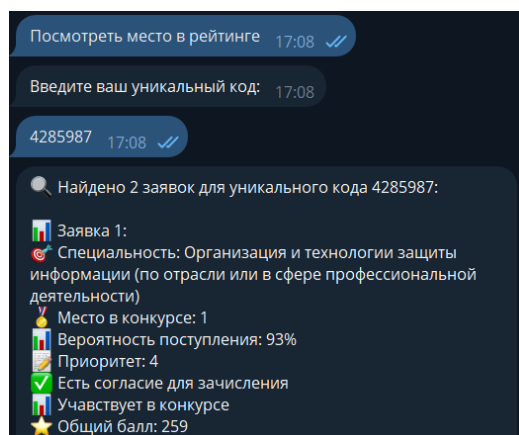


Рис. 4. Посмотреть место в рейтинге

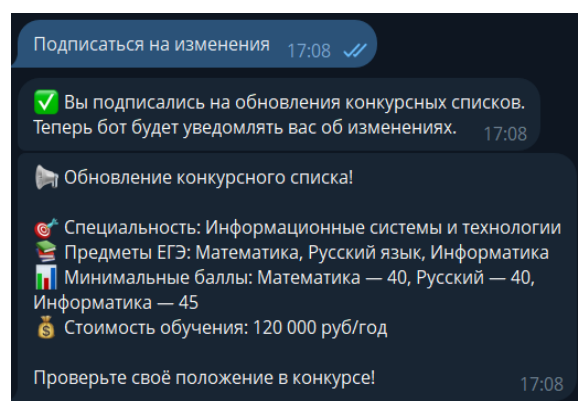


Рис. 5. Подписаться на изменения

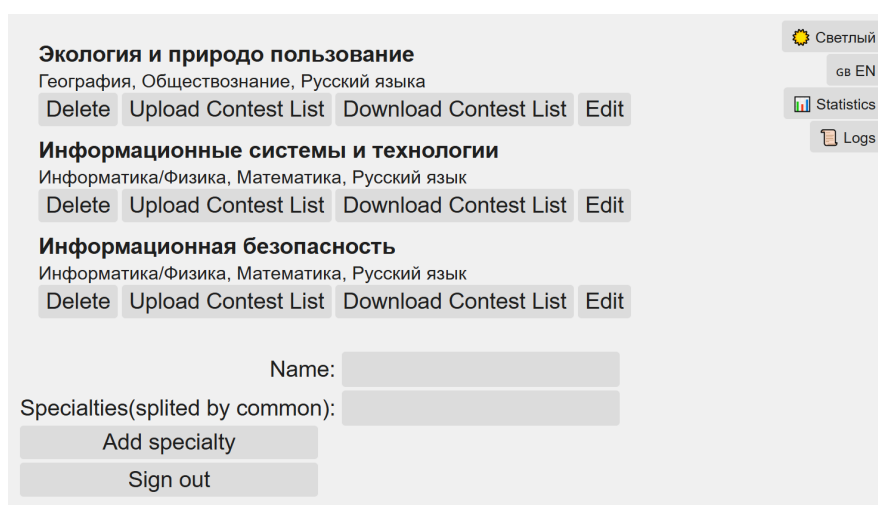


Рис. 6. Работа админ-панели

Заключение

В данной работе продемонстрированы проектирование и разработка системы для автоматизированной работы с абитуриентами. В отличие от традиционных способов информирования через сайт и телефонные обращения, разработанное решение позволяет получать необходимую информацию в автоматическом режиме через Telegram.

Разработка системы велась с использованием языка программирования Golang, базы данных SQLite, а также технологий JavaScript, HTML и CSS для реализации административной панели. Для обеспечения безопасной авторизации был реализован механизм JWT-аутентификации. В процессе разработки были спроектированы основные компоненты системы, структура базы данных и пользовательские сценарии взаимодействия.

Функциональные возможности Telegram-бота включают просмотр доступных специальностей с учетом выбранных предметов ЕГЭ, просмотр положения абитуриента в конкурсных списках, а также подписку на уведомления об изменениях конкурсной ситуации. Административная панель предоставляет сотрудникам приемной

комиссии возможности авторизации и выхода из системы, добавления, изменения и удаления специальностей, загрузки и выгрузки конкурсных списков, а также смены языка интерфейса и световой темы.

Итогом работы стала разработка полноценной системы автоматизации взаимодействия абитуриентов с приемной комиссией, позволяющей упростить получение информации о поступлении, снизить нагрузку на сотрудников и повысить удобство сопровождения приемной кампании.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Разработка автоматизированных систем информирования в высших учебных заведениях // Вестник высшей школы. – 2024. – № 1. – с. 5–11.
2. Автоматизация процессов в приемной комиссии вуза с использованием чат-ботов // Цифровизация образования. – 2024. – № 3. – с. 15–22.
3. Проектирование информационных систем с использованием UML // Открытые системы. – 2023. – № 2. – с. 40–45.
4. Фаулер, М. UML. Основы / М. Фаулер. – Санкт-Петербург : Символ-Плюс, 2020. – 192 с.
5. Применение диаграмм прецедентов для моделирования бизнес-процессов // Компьютерные науки и технологии. – 2025. – № 1. – с. 12–18.
6. Разработка архитектуры распределенных веб-приложений // Программная инженерия. – 2024. – № 7. – с. 34–41.
7. Проектирование клиент-серверных систем взаимодействия // Вестник компьютерных технологий. – 2023. – № 4. – с. 22–29.
8. Особенности построения архитектуры современных чат-ботов // Информационные технологии в управлении. – 2025. – № 2. – с. 50–57.
9. Донован, А. А. Язык программирования Go / А. А. Донован, Б. У. Керниган. – Санкт-Петербург : Питер, 2018. – 400 с.
10. Боднер, Дж. Go. Идиомы и паттерны проектирования / Дж. Боднер. – Санкт-Петербург : Питер, 2022. – 464 с.
11. Разработка пользовательских интерфейсов на JavaScript, HTML и CSS // Системный администратор. – 2024. – № 5. – с. 60–65.
12. Современные веб-технологии в создании административных панелей // Мир ПК. – 2023. – № 9. – с. 18–24.
13. Реализация серверной логики на языке Golang // Программирование и ИТ. – 2025. – № 3. – с. 27–33.
14. Опыт создания Telegram-ботов на языке Go // Технологии программирования. – 2024. – № 11. – с. 45–51.
15. SQLite Documentation // SQLite (дата обращения: 11.05.2026).
16. Использование форматов XML и JSON для обмена данными в информационных системах // Молодой ученый. – 2023. – № 42. – с. 89–94.
17. Разработка интерактивных интерфейсов для мессенджеров // Информационное общество. – 2024. – № 6. – с. 71–77.
18. Проектирование систем оповещения пользователей в реальном времени // Научно-технический вестник. – 2025. – № 4. – с. 103–110.
19. Безопасность веб-приложений и механизмы аутентификации // Защита информации. – 2024. – № 2. – с. 14–21.
20. JWT Introduction // JWT.io (дата обращения: 11.05.2026).

© Я. И. Головкин, Е. Ю. Воронкин, 2026