

ПРОЕКТ МУЛЬТИМЕДИЙНОГО УЧЕБНОГО ПОСОБИЯ ПО ДИСЦИПЛИНЕ «БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ ЧЕЛОВЕКА»

Т. А. Козлов

*Белорусский государственный университет, пр. Независимости, 4,
220030, г. Минск, Беларусь, fsc.kozlov@bsu.by
Научный руководитель — И. В. Пантюк, кандидат биологических наук, доцент*

В статье обсуждается проект мультимедийного учебного пособия по дисциплине «Безопасность жизнедеятельности человека». Дана классификация целевой аудитории. Представлены архитектура приложения, результаты разработки базы данных.

Ключевые слова: цифровые технологии; образование; мобильное приложение; безопасность жизнедеятельности; студенты.

Информационные технологии активно развиваются и повсеместно используются в различных сферах жизни: бизнесе, медицине, науке, образовании. Они позволяют автоматизировать процессы, улучшить эффективность работы, ускорить передачу информации и обеспечить доступ к большим объемам данных. Использование цифровых технологий в образовательном пространстве направлены на индивидуальные способности студентов, отличаются наглядностью, доступностью подачи учебного материала. Различные цифровые технологии в учебном процессе направлены на всестороннее развитие обучающихся, развитие у них практических навыков, повышение мотивации к обучению, а также упрощению и удобству работы преподавателей. Одной из приоритетных задач образовательного учреждения становится не только получение студентом теоретических знаний, но и развитие умения пользоваться информационными технологиями, самостоятельно добывать и анализировать информацию.

В Республике Беларусь в учреждениях образования функционируют 53 ресурсных центра с использованием информационно-коммуникационных технологий в учебно-воспитательном процессе. С 2018 г. реализуется проект Министерства образования Республики Беларусь «Цифровой университет», в который включены 33 учреждения высшего образования, в том числе и Белорусский государственный университет [1].

В соответствии с Кодексом Республики Беларусь об образовании от 13.01.2011 № 243-3; Правилами проведения аттестации студентов, курсантов, слушателей при освоении содержания образовательных программ высшего образования, утвержденными постановлением Министерства об-

разования Республики Беларусь от 29.05.2012 № 53; Инструкцией об общих требованиях к учебным планам, учебным программам, учебно-тематическим планам в Белорусском государственном университете Приказом Ректора БГУ от 29.04.2021 № 224-ОД утверждено Положение «Об использовании электронных средств обучения в БГУ» [2].

Внедрение информационно-коммуникативных технологий в образовательном пространстве формирует медиа-грамотность, цифровые компетенции, знания, навыки работы с большими объемами данных. С целью совершенствования качества образования нами разработан проект мультимедийного учебного пособия по дисциплине «Безопасность жизнедеятельности человека».

Создание мобильного приложения планируется на языке программирования Java с возможностью аутентификации пользователей, изучения дисциплины «Безопасность жизнедеятельности человека» с применением мультимедийных материалов, сохранения прогресса обучения студентов [3].

Мобильное приложение используется: для улучшения качества обучения студентов, для экономии очных академических часов программы обучения специалистов в высших учебных заведениях способом удаленного обучения, для обеспечения возможности повторения пройденного материала вне учебного плана.

Целевая аудитория представляет собой студентов высших учебных заведений, заинтересованных в получении высшего образования удобным для себя способом.

Классификация целевой аудитории может проводиться по следующим критериям:

1. Форма обучения: аудитория, получающая образование платно или за счет республиканского бюджета.
2. Способ получения образования: аудитория, получающая образование на очном или заочном отделении.
3. Тип специальности: аудитория, изучающая преимущественно гуманитарные или технические дисциплины.

Ниже приведены критерии анализа целевой аудитории для мультимедийного учебного пособия по безопасности жизнедеятельности человека:

1. Возраст, место проживания студента.
2. Частота и необходимость посещения учебного заведения.
3. Наличие рабочих отношений во время получения высшего образования.
4. Состояние здоровья студента в период обучения.

Исходя из представленных критериев можно составить следующий перечень целевой аудитории разрабатываемого проекта:

1. Студенты заочного отделения, совмещающие образовательный процесс с подработкой или полноценной работой. Это основная целевая аудитория, которая будет использовать мультимедийное учебное пособие. Приложение позволит им обучаться вне дома и территории учебного заведения, экономя значительную часть временных ресурсов на поиск и систематизацию учебных материалов или на посещение очных занятий в университете.

2. Студенты очного отделения, не способные посещать учебное заведение по состоянию здоровья. Им требуется возможность поддерживать актуальный уровень знаний по дисциплине вне зависимости от обстоятельств.

3. Студенты очного отделения в период вспышек острых респираторных заболеваний. Они будут использовать мобильное приложение в качестве альтернативы очному получению образования с целью сокращения физического контакта и, как следствие, избегания распространения заболевания.

4. Студенты, получающие академическую стипендию. Приложение позволит отслеживать успеваемость студента в дисциплине для назначения стипендии объема, соответствующего уровню знаний студента.

В целом, большинство пользователей – это студенты в возрасте от 17 до 22 лет, получающие высшее образование. Они стремятся оптимизировать процесс получения образования и получить возможность совмещать работу с учебой.

Проектирование архитектуры приложения является ключевым аспектом в разработке любого приложения. От архитектуры зависит скорость разработки новой функциональности приложения, архитектура напрямую влияет на способность к ориентации в коде. Грамотно

спроектированная архитектура позволяет разбить большое приложение на модули, что увеличивает производительность приложения и позволяет его легко тестировать в различных сценариях. Общая архитектура приложения представлена на рисунке 1.

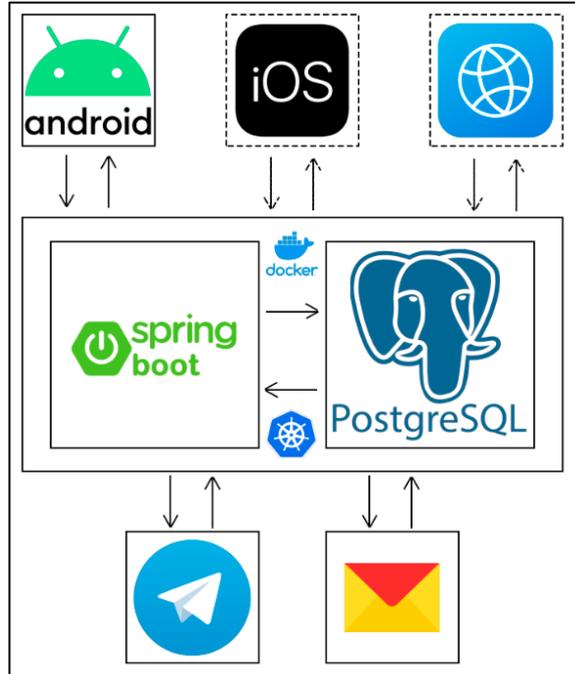


Рис. 1. Общая архитектура приложения

Платформа Docker оборачивает серверную часть приложения и базу данных в так называемые “контейнеры”, разворачивая и запуская их в собственной виртуальной среде, позволяя облегчить стартовую настройку приложения и доступ к нему по зарезервированным портам.

Надстройка над Docker, Kubernetes самостоятельно отслеживает нагрузку на каждый из контейнеров, распределяя ресурсы сервера оптимальным образом. В пиковой нагрузке Kubernetes может создавать дополнительные кластеры нагруженных контейнеров, дублируя их и распределяя нагрузку между ними.

Серверная часть приложения спроектирована по шаблону MVC (Model-View-Controller) [4] с использованием языка программирования Java и его фреймворка для разработки веб-приложений Spring Boot со встроенной реализацией шаблонов проектирования “внедрение зависимостей” и “инверсия управления”.

Клиентская часть, мобильное Android-приложение спроектировано по шаблону MVP (Model-View-Presenter) [5] с применением чистой архитектуры [6] и шаблона проектирования “внедрение зависимостей”. Схема чистой архитектуры представлена на рисунке 2.



Рис. 2. Схема чистой архитектуры

Проект предусматривает возможность реализации клиентской части на различные платформы, включая iOS и веб. Отображаемые пользователю лекции и вопросы в тестировании хранятся на сервере в виде html-страниц, которые могут быть отрисованы большим количеством платформ с применением встроенных средств.

В базе данных хранятся лишь данные для генерации пути к этим файлам.

Проект предусматривает интеграцию с API (Application Programming Interface) Телеграмм и Яндекс Почты. В специальный чат-бот телеграмм сервер отправляет пользовательские сообщения об ошибках. Яндекс Почта используется для отправки кодов подтверждения на почту пользователя.

Для разработки структуры базы данных необходимо определиться с ключевыми сущностями, с которыми будет осуществляться взаимодействие.

Требуется определить нормальную форму, которой будет соответствовать база данных, необходимую для соблюдения баланса избыточности и производительности. Для приложения была выбрана база данных PostgreSQL.

Ключевые сущности базы данных:

1. "Пользователь" – эта сущность отражает информацию о студенте, включая его логин и пароль для авторизации в приложении, электронный адрес почты для подтверждения своей личности, имя, фамилию, курс и факультет.

2. Существует возможность задавать пользовательский аватар. Полученный в запросе на сохранение аватара, набор байтов сохраняется на жестком диске сервера, а в базу данных заносится только путь к аватару, оптимизируя ее по занимаемой памяти и скорости операций чтения.

3. "Лекция" – эта сущность хранит информацию о лекциях, прикрепленных к модулям курса "Безопасность жизнедеятельности человека", включая тему лекции и ее порядковый номер как элемента модуля. Путь к html-странице лекции генерируется на основе порядковых номеров модуля и лекции, обеспечивая уникальность имен файлов.

4. "Тест" – эта сущность хранит информацию о тестах, прикрепленных к модулям курса "Безопасность жизнедеятельности человека", включая тему теста, его порядковый номер как элемента модуля и количество вопросов, на которые должен ответить студент при прохождении этого теста.

5. "Вопрос" – эта сущность содержит информацию о вопросах, прикрепленных к тестам, включая порядковый номер вопроса, используемый при генерации пути к html-странице вопроса, тип вопроса, например, с единственным вариантом ответа или свободным вводом, "стоимость" правильного ответа.

6. "Разработчик" – эта сущность определяет учетную запись разработчика, использующего чат-бот телеграмм для получения оповещений о проблемах, с которыми столкнулись пользователи и включает такую информацию как электронный адрес разработчика и состояние его учетной записи.

Полученная схема базы данных учитывает как сущности клиентской части приложения, так и сущности, необходимые для разработчиков-тестировщиков, получающих сообщения пользователей в чат-бот телеграмм.

Спроектированная база данных соответствует второй нормальной форме, так как каждая сущность имеет свой числовой идентификатор, а уникальные поля отмечены модификатором UNIQUE.

Схема базы данных представлена на рисунке 3.

В будущем базу данных можно привести к третьей нормальной форме, путем отделения таких полей как faculty, course таблицы users к отдельной таблице students, так как не каждый пользователь является студентом (существуют также роли преподавателя и администратора) и имеет факультет и курс, на котором обучается.

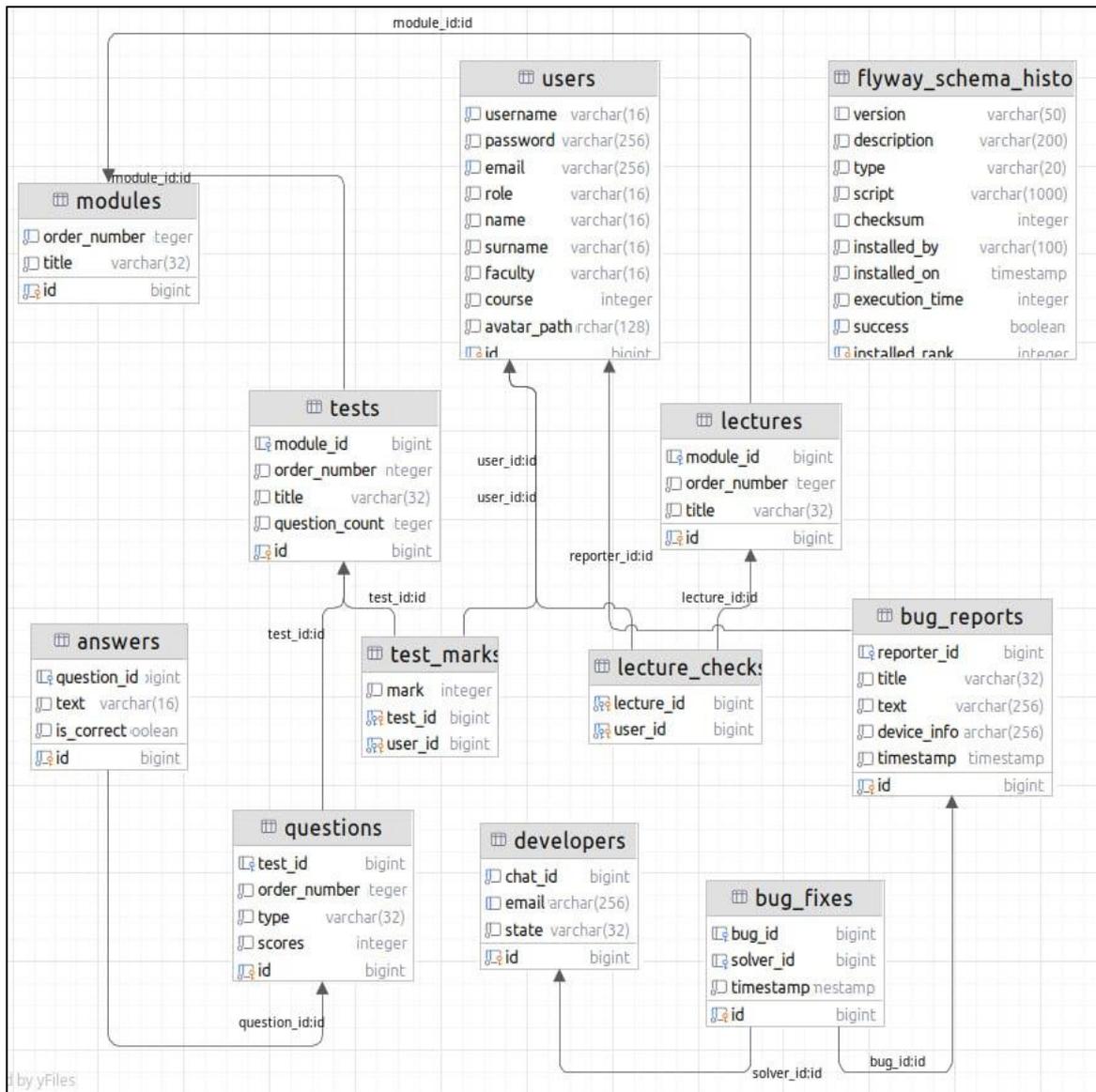


Рис. 3. Схема базы данных

Для реализации проекта были выбраны следующие инструментальные средства:

– Spring Boot 3.3.0 – фреймворк для языка программирования Java с открытым исходным кодом, упрощающий разработку веб-приложений, реализующий шаблоны проектирования “внедрение зависимостей” и “инверсия управления” [7].

– Swagger UI 5.17.12 – библиотека, позволяющая настроить документацию к маршрутам серверной части приложения и отобразить ее в виде генерируемой html-страницы.

– База данных PostgreSQL 16.3, поддерживающая индексирование таблиц, транзакции, не имеющая ограничений по количеству записей в ней [8].

– Hibernate 7.0 – библиотека, предназначенная для решения задач объектно-реляционного отражения [9].

– Flyway 10.13.0 – инструмент миграции баз данных [10].

– Docker 24.0 – платформа для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации [11].

– Kubernetes 1.30.1 – открытое программное обеспечение для оркестровки контейнеризированных приложений – автоматизации их развёртывания, масштабирования и координации в кластере [12].

– Android SDK API 34 – набор инструментов для разработки под операционную систему Android.

– Jetpack Navigation 2.8.0 – библиотека семейства Jetpack, позволяющая настроить навигацию между экранами, представленными фрагментами пользовательского интерфейса [13].

– Retrofit 2.11.0 – библиотека, упрощающая работу с сетью при разработке Android-приложения [14].

– Dagger 2.51.1 – библиотека, служащая для реализации шаблона проектирования “внедрение зависимостей” в контексте Android-приложения [15].

– Lombok 1.18 – библиотека, сокращающая бойлерплейт-код [16].

Таким образом, реализация данного проекта позволит студентам различных специальностей обучаться дисциплине “Безопасность жизнедеятельности человека” удаленно, без участия преподавателя и эффективно усваивать учебный материал.

Внедрение в учебный процесс мобильного приложения позволит студентам в соответствии с учебной программой овладеть основными способами и средствами получения, хранения, переработки информации, навыками работы с компьютером как средством управления информацией, самостоятельно приобретать с помощью информационных технологий и использовать в практической деятельности новые знания и умения, в том числе в новых областях знаний, непосредственно не связанных со сферой профессиональной деятельности.

Библиографические ссылки

1. В центре внимания – цифровой университет [Электронный ресурс]. – Режим доступа: <https://edu.gov.by/news/v-tsentre-vnimaniiya--tsifrovizatsiya-obrazovaniya> Дата доступа: 10.03.2024.
2. Приказ Ректора БГУ от 29.04.2021 № 224-ОД Об использовании электронных средств обучения в БГУ [Электронный ресурс]. – Режим доступа: https://bsu.by/upload/All_units/Polozhenie_ob_ispolzovanii_elektronnyh_sredstv_obucheniya_29_04_2021.pdf– Дата доступа: 10.04.2024.
3. Документация Java [Электронный ресурс] Режим доступа: <https://docs.oracle.com/en/java/>. Дата доступа: 14.05.2024.
4. MVC для веб-разработки [Электронный ресурс] Режим доступа: <https://habr.com/ru/articles/181772/>. Дата доступа: 14.05.2024.
5. MVP для разработки Android-приложения [Электронный ресурс] Режим доступа: <https://habr.com/ru/companies/productstar/articles/508892/>. Дата доступа: 14.05.2024.
6. Чистая архитектура Android-приложения [Электронный ресурс] Режим доступа: <https://habr.com/ru/articles/766762/>. Дата доступа: 14.05.2024.
7. Документация Spring Boot [Электронный ресурс] Режим доступа: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/>. Дата доступа: 14.05.2024.
8. Документация PostgreSQL [Электронный ресурс] Режим доступа: <https://postgrespro.ru/docs/postgresql>. Дата доступа: 14.05.2024.
9. Документация Hibernate [Электронный ресурс] Режим доступа: <https://hibernate.org/orm/documentation/6.5/>. Дата доступа: 14.05.2024.
10. Документация Flyway [Электронный ресурс] Режим доступа: <https://documentation.red-gate.com/flyway>. Дата доступа: 14.05.2024.
11. Документация Docker [Электронный ресурс] Режим доступа: <https://docs.docker.com/guides/>. Дата доступа: 14.05.2024.
12. Документация Kubernetes [Электронный ресурс] Режим доступа: <https://kubernetes.io/ru/docs/home/>. Дата доступа: 14.05.2024.
13. Документация Jetpack Navigation [Электронный ресурс] Режим доступа: <https://developer.android.com/guide/navigation>. Дата доступа: 14.05.2024.
14. Документация Retrofit [Электронный ресурс] Режим доступа: <https://square.github.io/retrofit/>. Дата доступа: 14.05.2024.
15. Уроки по применению Dagger 2 [Электронный ресурс] Режим доступа: <https://habr.com/ru/articles/466725/>. Дата доступа: 14.05.2024.
16. Документация Lombok [Электронный ресурс] Режим доступа: <https://project-lombok.org/features/>. Дата доступа: 14.05.2024.