

РАЗРАБОТКА ОНЛАЙН-СЕРВИСА ДЛЯ СОЗДАНИЯ ЭЛЕКТРОННЫХ ИНТЕРАКТИВНЫХ УЧЕБНИКОВ

А. Ф. ПАНТЕЛЕЙКО, М. В. ДЕКАНОВА,
канд. физ.-мат. наук, доц. О. В. ГОЛУБЕВА
(Полоцкий государственный университет, Беларусь)

Аннотация. В докладе рассматриваются ключевые этапы разработки онлайн-сервиса для создания электронных интерактивных учебников: выбор и обоснование средств разработки, разбиение программной системы на структурные компоненты и связи, проектирование базы данных и архитектура онлайн-сервиса.

Ключевые слова. Электронный учебник, интерактивные задания, онлайн-сервис, онлайн обучение.

Введение. Электронный учебник – это специальное устройство либо программное обеспечение, используемое в образовательном процессе и заменяющее собой традиционный бумажный учебник [1].

Достоинствами электронного учебника являются: быстрое восприятие и поиск информации, экономичность, поддержка аудио, видео и анимации, интерактивность, удобство передачи и хранения.

В связи с рядом достоинств электронных учебников была поставлена задача проектирования и разработки онлайн сервиса для создания электронных интерактивных учебников.

Выбор и обоснование средств разработки. Для реализации веб-сервера выбрана программная платформа Node.js, которая является частью стека технологий, охватывающих все нужды веб-разработки основанных на JavaScript. К достоинствам данной платформы можно отнести следующие показатели:

- высокая производительность;
- один и тот же код можно использовать и на клиенте, и на сервере;
- большое количество дополнительных модулей;
- богатая стандартная библиотека;
- легкая масштабируемость;
- программное обеспечение с открытым исходным кодом.

Следующим шагом является выбор фреймворка для разработки на стороне сервера. Express – это минималистичный и гибкий веб-фреймворк для приложений Node.js, предоставляющий обширный набор функций для мобильных и веб-приложений [2]. Ключевыми достоинствами данного фреймворка являются:

- надежный механизм маршрутизации для обработки динамических URL-адресов;

- упрощение отладки кода;
- интегрируемость с наиболее популярными системами управления базами данных (СУБД).

Выбор СУБД представляет собой сложную многопараметрическую задачу и является одним из важных этапов при разработке клиент-серверных приложений.

В качестве СУБД для онлайн-сервиса была выбрана MongoDB, документно-ориентированная система управления базами данных, не требующая описания схемы таблиц [3]. Данную систему считают одним из классических примеров NoSQL-систем. MongoDB использует JSON-подобные документы и схему базы данных. Выбор СУБД был сделан на основании следующих достоинств:

- отсутствие схемы;
- легкая масштабируемость;
- для хранения данных используется внутренняя память;
- данные хранятся в виде JSON документов;
- поддерживаются динамические запросы документов.

Диаграмма компонентов. Диаграмма компонентов, представленная на рисунке 1, отображает разбиение программной системы на структурные компоненты и связи.

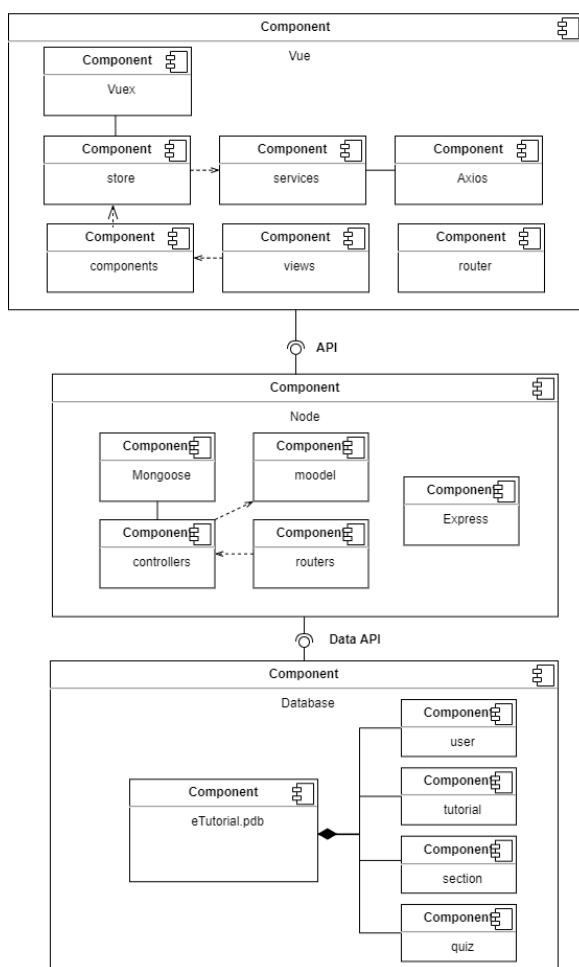


Рисунок 1. – Диаграмма компонентов

Программируемая система представляет собой три основных компонента: Vue, Node, Database.

Компонент Vue отвечает за отображение приложения пользователю. Информация, поступающая с сервера, хранится в компоненте store. Для организации хранения информации в store используется библиотека Vuex. Данные в данный компонент загружаются посредством методов из компонента services. Компонент services в свою очередь использует библиотеку Axios для выполнения запросов на сервер. В модуле components расположены Vue-компоненты, которые являются составными частями views. Компонент router отвечает за перемещение пользователя между страницами сервиса.

Компонент Node предоставляет компоненту Vue программный интерфейс приложения (API), который в данном случае используется для обращения компонента Vue к Node для получения информации. В компоненте model описаны схемы данных с помощью библиотеки Mongoose. Схемы используются для определения таких методов как: вставка, удаление и редактирование документов в базе данных. Данные методы реализованы в компоненте controllers, который используется для описания GET, POST, PUT и DELETE запросов. Веб-фреймворк Express устанавливает основные настройки сервера и обрабатывает запросы с разных URL-адресов.

Database – компонент, который определяет структуру базы данных. Компонент ETutorial.pdb используется в качестве базы данных типа файла. Составляющими компонента eTutorial.pdb являются файлы: user, tutorial, section и quiz.

База данных. MongoDB является документо-ориентированной системой, в которой центральным понятием является документ. Документ можно представить как объект, хранящий некоторую информацию. В некотором смысле он подобен строкам в реляционных СУБД, которые хранят информацию об отдельном элементе. Для каждого документа в MongoDB определен уникальный идентификатор, который называется `_id`. При добавлении документа в коллекцию данный идентификатор создается автоматически. Однако разработчик может сам явным образом задать идентификатор, а не полагаться на автоматически генерируемые, указав соответствующий ключ и его значение в документе [3].

Рассмотрим подробнее структуру документов, используемых в приложении. Основная информация о пользователях хранится в документе User. Наименование идентификаторов и их описание представлены в таблице.

Таблица 1 – Структура схема User

Идентификатор	Комментарий	Тип значений
id	Уникальный идентификатор	ObjectId
surname	Фамилия пользователя	String
name	Имя пользователя	String
password	Пароль для входа	String

Информация о содержании учебника представлена в таблице 2. Каждый учебник может содержать в себе некоторое количество параграфов (Sections). Структура схемы параграфов представлена в таблице 3.

Таблица 2 – Структура схема Tutorial

Идентификатор	Комментарий	Тип значений
id	Уникальный идентификатор	ObjectId
id_User	Идентификатор пользователя	ObjectId
name	Название учебника	String

Таблица 3 – Структура схема Sections

Идентификатор	Комментарий	Тип значений
id	Уникальный идентификатор	ObjectId
id_Tutorial	Идентификатор учебника	ObjectId
name	Название учебника	String
contentPath	Путь к содержимому страницы	String

Архитектура. Для реализации приложения была выбрана клиент-серверная архитектура. Клиентская часть реализует пользовательский интерфейс, формирует запросы к серверу и обрабатывает ответы от него.

Серверная часть реализует взаимодействие между базой данных и клиентской частью. Для обработки запросов к серверу были созданы контролеры для каждой сущности в базе данных.

Для маршрутизации была использована оболочка `express-promise-router` для фреймворка `Express`. Данная оболочка позволяет промежуточному программному обеспечению возвращать ответы.

Возможность хранения данных на стороне клиента повышает производительность приложения. Для того, чтобы каждый раз, когда пользователь нуждается в информации не обращаться к серверу, необходимо использовать локальное хранилище данных. В данном случае использовалась библиотека `Vuex 4` [4]. `Vuex` – паттерн управления состоянием, а также библиотека для приложений на `Vue.js`, которая служит централизованным хранилищем данных для всех компонентов приложения с правилами, гарантирующими, что состояние может быть изменено только предсказуемым образом.

Заключение. Разработана концепция пользовательского интерфейса приложения, спроектирована база данных. Онлайн-сервис предоставляет следующие возможности:

- просматривать уже существующие учебники;
- выполнять задания к параграфу;
- получать результат выполнения заданий;

- создавать новые учебники;
- редактировать учебники;
- разделять учебник на параграфы;
- добавлять к параграфам интерактивные задания;
- редактировать задания;
- удалять учебники.

Разработанный онлайн-сервис обеспечит удобство структурирования учебного материала и повысит эффективность организации учебного процесса.

ЛИТЕРАТУРА

1. Электронный учебник [Электронный ресурс]. – Режим доступа: <https://clck.ru/DRznP>. – Дата доступа:01.03.2022.
2. Express – быстрый, гибкий, минималистичный веб-фреймворк для приложений [Электронный ресурс]. – Режим доступа: <http://expressjs.com/ru/>. – Дата доступа:01.03.2022.
3. MongoDB [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/MongoDB>. – Дата доступа: 01.03.2022.
4. Что такое Vuex? [Электронный ресурс]. – Режим доступа: <https://spec-zone.ru/vuex~4/>. – Дата доступа:02.03.2022.