

УДК 007.51:007.3



М.А. Морозов, А.А. Суконников
Вологодский государственный университет

РАЗРАБОТКА СТРУКТУРЫ ВИРТУАЛЬНОГО ПРОСТРАНСТВА ОРГАНИЗАЦИИ

В статье определены необходимые инструменты для разработки структуры виртуального пространства с использованием метки сканируемого устройства (NFC). Спроектированы два уровня front-end разработки, а back-end реализуется в виде CRUD-интерфейса средствами REST. В качестве базы данных выбирается MongoDB.

Виртуальное пространство, инструменты, метка сканируемого устройства, база данных.

Виртуальная реальность является полноценным техническим миром, признаки которой передаются человеку через его органы чувств: обоняние, осязание, слух, зрение и так далее. Как правило передача подобной информации осуществляется одновременно сразу на несколько органов чувств, а не на один.

В свою очередь виртуальная организация является усовершенствованной архитектурой традиционных организаций, где используются в меньшей степени физические активы, менее развита физическая структура, большее доверие коммуникационным технологиям, высокая гибкость и отсутствие границ – масштабируемость [1].

Таким образом, виртуальное пространство является неким фундаментом. Оно по сравнению с традиционным предоставляет улучшенные возможности для управления и ведения бизнеса, также решает те некоторые проблемы, которые могли возникнуть на физическом уровне в организации. С его помощью появляется возможности:

- частично избавиться от большинства материальных активов;
- усовершенствовать модель динамической связи как между людьми в организации, так и между компонентами самой системы;
- избавление от тесного физического контакта;
- возможность объединения разных отраслей [2].

Стоит отметить, что и виртуальная реальность, и виртуальная организация являются сложными полноценными системами, где взаимодействуют множество различных компонентов. Одним из главных таких компонентов как раз является виртуальное пространство, которому и посвящена данная работа [3, 4].

Разрабатывается виртуальное пространство в виде веб-приложения, которое будет предоставлять информацию пользователю о требуемом им устройстве или системе в виде 3D-модели при помощи поднесения смартфона к Near Field Communication (NFC) метке сканируемого устройства.

Для этого понадобятся следующие инструменты:

1. Visual Studio Code.

Это редактор исходного кода, который разработан Microsoft. Позиционирует себя как редактор для кроссплатформенной веб-разработки. Если затронуть плюсы, то он абсолютно бесплатный, имеет свою экосистему расширений, намного быстрее своего конкурента Atom, интеграция с GitHub – веб-сервисом, предназначенным для совместной разработки проектов.

2. Yarn.

Это пакетный менеджер. С его помощью будут загружать нужные библиотеки в проект. Самый главный плюс – это его скорость. Он быстрее своего аналога под названием NPM. Это единственное их отличие.

3. Node.JS.

Это программная платформа, превращающая JS из узкоспециализированного языка в язык общего назначения. Данный инструмент понадобится для подключения внешних библиотек и организации серверной части.

4. JavaScript.

Это всем известный язык программирования. Выбор на него пал только по причине раннего использования и опыта. Стоит лишь отметить, что отсутствие данного языка при разработке веб-приложений – невозможно.

5. JSX.

Это расширения языка JavaScript. Следуя этому синтаксису, явно описывается то, что и как хочется увидеть на странице. Основная его задача – предоставить нам ясный и понятный синтаксис для определения древовидной структуры элементов и их атрибутов с любым уровнем вложенности [3].

6. React.JS.

Это JavaScript-библиотека, на которой и будет построен весь пользовательский интерфейс. Его основная цель, которую он уже достиг давно – высокая

скорость, масштабируемость и простота. Также данную библиотеку легко совмещать с любыми другими библиотеками.

7. MongoDB.

MongoDB имеет новый подход к построению баз данных, где отсутствуют таблицы, схемы, запросы SQL, внешние ключи, которые характерны объектно-реляционным базам данных. MongoDB имеет документ-ориентированную модель данных, из-за чего она работает быстрее и обладает лучшей масштабируемостью. В MongoDB применяется формат BSON. Данный формат позволяет работать с данными быстрее: быстрее выполняется поиск и обработка.

8. Redux.

В планах данное веб-приложение разрабатывается в виде Single Page Application (SPA). У подобных структур имеется такое понятие как состояние хранилища данных. Чтобы управлять таким хранилищем как раз и используется инструмент Redux. Один из плюсов этой библиотеки – это возможность использования FLUX-архитектуры, которая будет рассмотрена в следующих пунктах.

9. Vectary.

Современная разработка, целью которой является предоставление инструментов пользователю для моделирования. Все происходит в браузере, никаких скачиваний не требуется. Имеется бесплатная версия с предоставлением огромного множества функционала. Данный инструмент имеет много различных форматов моделей, что способствует беспрепятственному импорту/экспорту готовых моделей.

10. Blender.

Данный инструмент позволяет работать с компьютерной графикой. Одни из главных преимуществ данной программы – кроссплатформенность и возможность установления дополнительного ПО.

11. Verge3D.

То самое дополнительное ПО, которое было отмечено выше. Оно позволит реализовать аннотации к моделям более простым путем. Данный сервис был выбран в пользу своей доступности, а именно без каких-либо ограничений загружать в него собственные 3D-модели и делиться ссылками на них. Подобного (бесплатного пользования) не имеется ни у Sketchfab, ни у CGTrader, где есть ограничения на загружаемые модели и время бесплатного пользования.

12. NFC-метка.

Данный инструмент позволит взаимодействовать и получать информацию об объекте в разы быстрее. Он превосходит своего оппонента (QR-код) в том, что ему не требуется определенный уровень освещения. Кроме этого, на смартфон не нужно устанавливать определенное программное обеспечение. Современные телефоны имеют функционал бесконтактного сканирования, а чтобы сканировать такие метки, компании своевременно выпускают обновления на смартфоны.

Кроме этого, был подключен инструмент express, предназначенный для обработки запросов. С его помощью создается API.

Разработан итоговый алгоритм работы всего виртуального пространства с учетом использования NFC-меток. NFC-метка – чип, который способен работать без питания и использоваться для передачи записанных на него данных на смартфон. Под данными в разрабатываемом виртуальном пространстве будет пониматься определенный маршрут (URL) на ту или иную модель. Запрограммировать любую подобную метку можно при помощи инструмента под названием NFC Tools [4, 5].

Стоит отметить, что запись с метки также можно редактировать и удалять. Таким образом, она является многократно. К минусам можно отнести размер хранения информации – предел 150–888 байт, а также радиус считывания не более 10 см.

Главное преимущество меток над всем известным QR-кодом в том, что для сканирования не требуется определенный уровень освещения в помещении, а также фокусировки камеры. Кроме этого, возможность считывания информации с меток на текущий момент имеется на всех смартфонах операционной системы Андроид и начиная с 14-ой версии IOS и не требует установки дополнительного ПО.

При разработке виртуального пространства в качестве веб-приложения будет использоваться в совокупности несколько архитектурных решений, которые позволят в разы увеличить скорость и работоспособность пространства, а также масштабировать его без каких-либо трудностей [5].

Первое, на что стоит обратить внимание, так это сама концепция программирования с использованием Virtual Document Object Model (далее – VDOM и DOM). Суть этого подхода заключается в том, чтобы хранить виртуальное представление пользовательского интерфейса в памяти и в последующем синхронизировать его с истинным DOM. Другими словами, имеется структура элементов (объектов) – это истинный DOM. Если в эту структуру внести изменения (добавить новый объект), то это изменение фиксируется в Virtual DOM, затем программным способом (согласование) синхронизируется с истинным DOM и добавляет только один этот новый элемент, а не перерисовывает всю структуру целиком. Формируется новый DOM. Схема такого подхода представлена на рисунке 1.

Такой подход без проблем можно реализовать с помощью библиотеки React. Следовательно, мы получаем первый архитектурный уровень, который связан напрямую с пользовательским интерфейсом (UI-уровень).

Теперь перейдем к разработке Business Logic Layer (в дальнейшем – BLL). Данный уровень позволяет сформировать логику и алгоритмы, которые будут использоваться только для данных, а не для интерфейса. Данный архитектурный подход реализуется посредством инструмента Redux. Он позволяет оптимально управлять хранилищем данных.

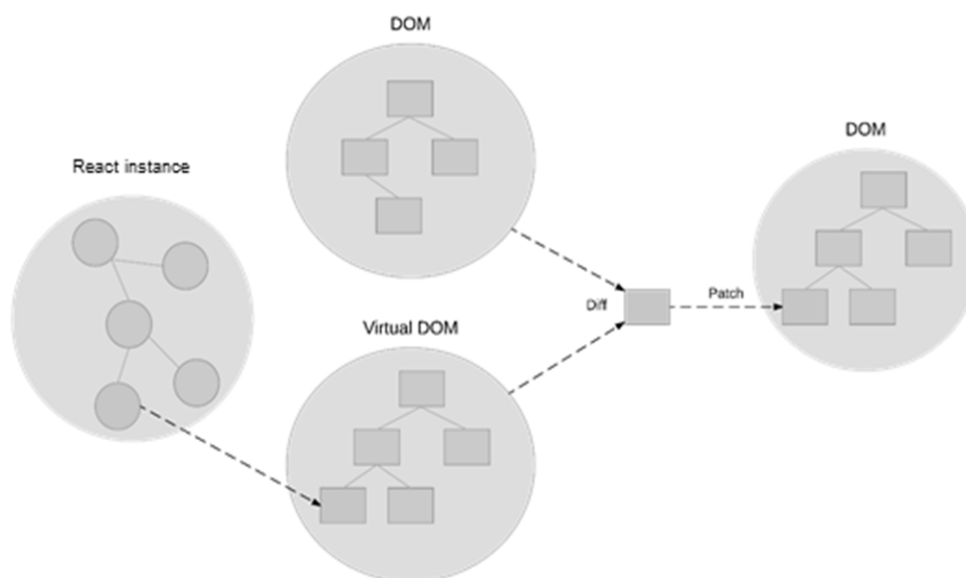


Рис. 1. Обновление DOM с помощью Virtual DOM

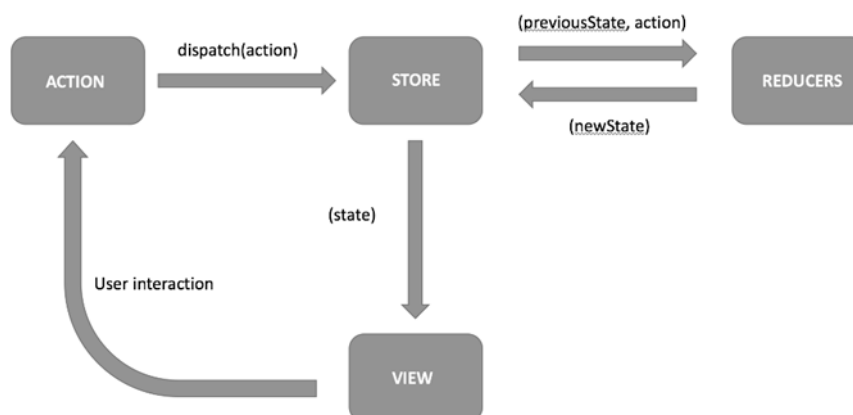


Рис. 2. Business Logic Layer

Идея разработки заключается в следующем: имеется глобальное хранилище данных (объект), которое имеет какое-то начальное состояние (initial state). И данное хранилище при помощи встроенного метода «подписывается» на изменения, которые происходят при формировании действия (action). Под действием понимается любое изменение пользовательского интерфейса, совершаемое пользователем, например нажатие на кнопку вызова диалогового окна. Action представляет из себя объект с обязательным свойством type.

Данное действие фиксируется и отправляется (dispatch) через хранилище в Reducer. Функция Reducer заключается в том, чтобы по свойству type определить из всех внесенных в него действий (actions) нужный action и внести соответствующие изменения в хранилище. Другими словами, изменить его состояние. Новое состояние фиксируется в хранилище (store) и отображается на интерфейсе. Следовательно, мы получаем второй архитектурный уровень, который связан напрямую с управлением данных

(BLL-уровень). Данный архитектурный подход представлен на рисунке 2.

На данном этапе мы спроектировали два уровня, которые будут взаимодействовать друг с другом. Такое взаимодействие еще называют FLUX-архитектурой. Все это относится к front-end-разработке. Теперь приступим к архитектурным решениям back-end.

Но стоит еще отметить, что при реализации подобной архитектуры с такими уровнями принято отдельно реализовать DAL-уровень (Data Access Layer), где отправляются запросы на сервер. Таким образом, мы сохраним принцип единственной ответственности – каждый уровень выполняет конкретно только свою работу и не занимается тем, что ему не свойственно. Запросы будут реализованы при помощи инструмента axios.

Приложение разрабатывается с нуля, следовательно готовые решения (CMS) использоваться не будут. Поэтому приступим к реализации собственного ручного API, организации сервера и формированию базы данных.

При проектировании back-end наиболее распространенной практикой является реализация CRUD-интерфейса средствами REST. REST в свою очередь является одним из существующих методов для работы с объектами на сервере. Благодаря ему мы можем работать с одним URL используя не только встроенные методы GET и POST, но и также PUT и DELETE. Аббревиатура CRUD расшифровывается как Create (POST), Read (GET), Update (PUT), Delete и позволяет разделить программный код на отдельные части с своей определенной функцией.

В основном для создания сервера будет использоваться инструмент Express. Главные преимущества использования данного фреймворка в отличие от простого Node.js заключаются в том, что он предоставляет ряд готовых решений по работе с cookie и CORS-политикой (Cross Origin Resource Sharing), позволяющей совместное использование ресурсов между разными источниками.

Также ранее упоминалась библиотека Mongoose. Она позволит определять схемы со строго типизированными данными. Как только схема определена, Mongoose дает возможность создать Model (Модель), основанную на этой только что созданной схеме. Позже модель синхронизируется с документом базы данных с помощью определения схемы модели. Схематично это представлено на рисунке 3.

Из структуры выше можно сделать вывод, что используемая база данных будет MongoDB, которая является NoSQL. Структура ее довольно проста – вся база состоит из коллекций. Каждая коллекция способна хранить огромное количество документов, а документ, в свою очередь, имеет поля для данных. Документ – это объект, который также может иметь вложенные объекты. Планируемая структура базы данных представлена на рисунке 4.

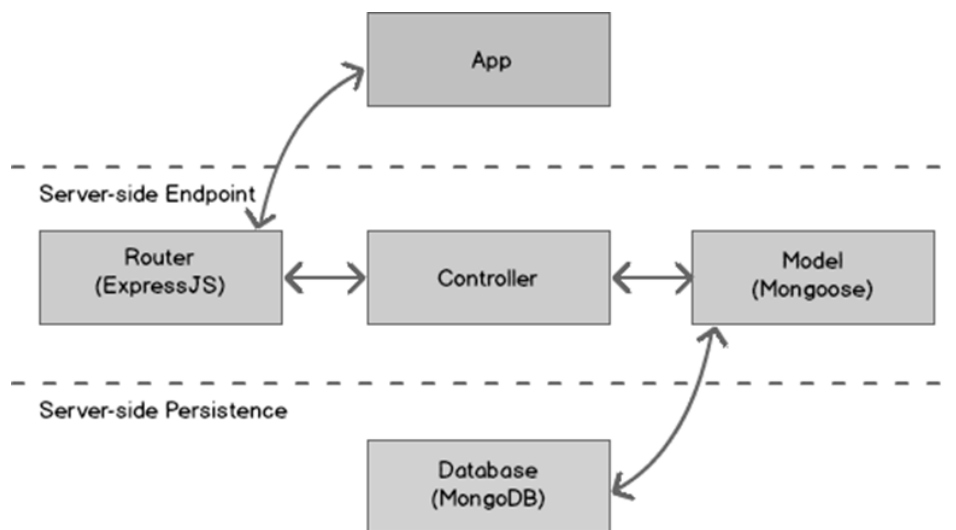


Рис. 3. Схематичная структура back-end

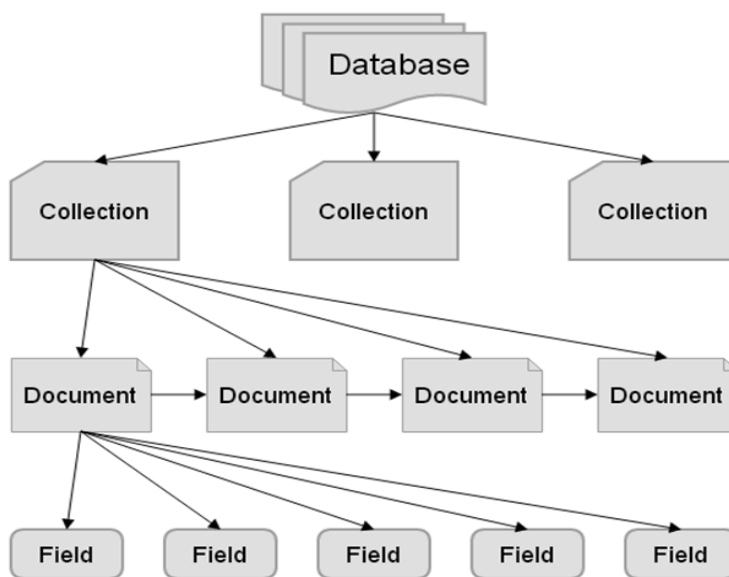


Рис. 4. Структура базы данных

В статье проведено исследование на тему разработки виртуального пространства для организации. Для достижения данной цели было выполнено:

- рассмотрение аналогов, предоставляющих 3D-модели, с выявлением сильных и слабых сторон;
- проведение подробного анализа предметной области с указанием классификаций пространств и его формирования;
- детальный подбор инструментов разработки для проектируемого алгоритма и структуры виртуального пространства;
- создание данного виртуального пространства с разбивкой на уровни и вложенность.

Литература

1. Малкольм, У. Виртуальные организации. Новая форма ведения бизнеса в XXI веке / У. Малкольм, М. Витцель. – Москва : Издательство Добрая книга, 2005. – 424 с.
2. Афанасьев, М. Я. Организация единого информационного пространства виртуального предприятия /

М. Я. Афанасьев, А. А. Грибовский // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий. – 2011. – С. 113–117.

3. Суконщиков, А. А. Сравнение инструментов разработки React, Vue и Angular / А. А. Суконщиков, М. А. Морозов // Информационные технологии в образовании, науке и технике : материалы I Международной научно-практической конференции «Наука-практике». – Барановичи : БарГУ, 2020. – С. 44–46.

4. Суконщиков, А. А. Принцип построения виртуального визуализатора / А. А. Суконщиков, М. А. Морозов // Информационные системы и технологии : материалы XIV Ежегодной научной сессии аспирантов и молодых ученых. – Вологда : ВоГУ, 2020. – С. 107–110.

5. Суконщиков, А. А. Разработка обобщенного алгоритма виртуального пространства для организации / А. А. Суконщиков, М. А. Морозов // Технические науки : материалы сетевого издания «Научные исследования XXI века». – 2021. – С. 95–98.

М.А. Morozov, A.A. Sukonshchikov
Vologda State University

DEVELOPMENT OF ORGANIZATION VIRTUAL SPACE STRUCTURE

The article discusses the construction of the structure of the virtual space of an organization. The necessary tools for developing the structure of the virtual space using a scannable device tag (NFC) have been identified. Two levels of front-end development have been designed, and the back-end is implemented as a CRUD interface using REST. MongoDB is selected as a database.

Virtual space, tools, scanned device label, database.