

Автор:

Хафизова А.А.

4 курс, группа 4407

КНИТУ-КАИ

Научный руководитель:

Мокшин В.В. доцент КНИТУ-КАИ

Обзор методов разработки сайта для бронирования номеров в гостиничном бизнесе, анализ и вывод

В этой статье рассматриваются различные методы разработки сайта для бронирования номеров в гостиничном бизнесе. Основное внимание уделяется методам программирования, дизайну интерфейса и методам веб-разработки. Будут рассмотрены их преимущества, недостатки и области применения.

Основные проблемы

При разработке сайта бронирования отеля могут возникнуть следующие проблемы:

1. Интеграция с информационной системой отеля: для эффективного функционирования сайта необходима хорошая интеграция с гостиничной информационной системой, чтобы обеспечить актуальность и точность информации о доступных номерах, ценах, специальных предложениях и т.д.
2. Безопасность данных: очень важно обеспечить высокий уровень безопасности для персональных данных клиентов, использующих сайт для бронирования. Необходимо защитить информацию о платежах, контактных данных и других личных сведениях от несанкционированного доступа.
3. Удобство использования: сайт должен быть интуитивно понятным и удобным для пользователей. Простой и понятный интерфейс, легкая навигация, быстрый поиск и понятная система бронирования помогут

привлечь и удержать клиентов.

4. Мобильная совместимость: учитывая растущую популярность мобильных устройств, важно создать адаптивный дизайн сайта, который будет отлично отображаться и функционировать на различных мобильных устройствах.
5. Отзывы и рейтинги: интеграция системы отзывов и рейтингов поможет повысить доверие потенциальных клиентов к вашему отелю. Однако, важно следить за качеством и подлинностью отзывов, чтобы избежать недостоверной информации.
6. Маркетинг и продвижение: создание сайта - это только первый шаг. Для привлечения клиентов необходимо провести качественную маркетинговую кампанию, оптимизировать сайт для поисковых систем (SEO), использовать рекламу в социальных сетях и другие инструменты продвижения.

После разработки сайта не стоит забывать также и о проблемах обслуживания сайта:

1. Невнимательность сотрудников: информация о бронировании может быть неверно интерпретирована или не учтена сотрудниками отеля.
2. Несоответствие тарифов: клиент может забронировать номер с завтраком, а по прибытии узнать, что завтрак не включён в стоимость.
3. Неактуальные фотографии: отель может предоставить старые или некачественные фотографии номеров, что может ввести клиента в заблуждение.
4. Некорректное описание номеров: клиент может забронировать номер с определёнными удобствами, а по прибытии обнаружить, что они отсутствуют.
5. Проблемы с онлайн оплатой: сложности с проведением платежей могут привести к задержкам в обработке бронирования.
6. Нестабильность работы сайта: технические проблемы на сайте отеля могут вызвать задержки в обработке бронирований и потерю информации.

Способы решения проблем, возникающих при создании ИС для бронирования номеров в отеле

Рассмотрим способы решения проблем, которые могут возникнуть при создании сайта для бронирования номеров в отеле:

1. Интеграция с информационной системой отеля:

- Разработка API для взаимодействия между сайтом и гостиничной информационной системой.
- Убедиться в корректной работе с базой данных отеля, чтобы информация о номерах, ценах и доступности была всегда актуальной.
- Тестирование интеграции для подтверждения ее надежности и эффективности.
- Выбор фреймворка, который максимально подходит нашим требованиям разработки

2. Безопасность данных:

- Использование SSL-шифрования для защиты данных, передаваемых на сайте.
- Регулярное обновление программного обеспечения для обеспечения безопасности.
- Регулярный аудит безопасности для выявления уязвимостей и их исправления.

3. Удобство использования:

- Разработка интуитивно понятного и простого интерфейса сайта.
- Обеспечение быстрого поиска и простой системы бронирования.
- Тестирование пользовательского опыта на различных устройствах для удобства использования.

4. Мобильная совместимость:

- Создание адаптивного дизайна сайта для отображения на мобильных устройствах.
- Проведение тестирования на различных устройствах и браузерах для

проверки корректной работы.

5. Отзывы и рейтинги:

- Модерация отзывов и контроль за их достоверностью.
- Интеграция системы рейтингов для удобства выбора клиентов.
- Поощрение доверия клиентов через отзывы оставленные другими гостями.

6. Маркетинг и продвижение:

- Проведение маркетинговых исследований для определения целевой аудитории.
- Использование SEO-стратегии для оптимизации сайта под поисковые системы.
- Реклама в социальных сетях и другие каналы продвижения для привлечения посетителей на сайт.

Перспективные направления и недостатки существующих

Для создания сайта для бронирования номеров в отеле, существует несколько популярных бэкэнд фреймворков, которые предлагают широкий выбор инструментов и функциональность для разработки:

1. ASP.NET Core: ASP.NET Core является полностью переписанной и современной версией фреймворка ASP.NET, который поддерживает кроссплатформенную разработку. ASP.NET Core обладает широким функционалом, высокой производительностью и хорошей масштабируемостью, что делает его отличным выбором для создания сайтов с бэкендом на языке C#. Из минусов можно выделить зависимость от Microsoft: Поскольку ASP.NET Core разрабатывается Microsoft, некоторые разработчики могут предпочесть более независимые фреймворки
2. Node.js: Node.js является популярным фреймворком для разработки серверных приложений на JavaScript. Он основан на движке V8 (тот же, что используется в браузере Google Chrome) и обладает высокой производительностью. Node.js подходит для создания быстрых и

масштабируемых веб-приложений.

Так же он имеет ряд минусов:

- Единичный поток: По умолчанию Node.js работает в однопоточном режиме, что может привести к блокировкам и медленной обработке запросов.
- Большое количество пакетов: Иногда выбор нужного пакета может быть непростым из-за огромного количества существующих пакетов в экосистеме Node.js.

3. Django: Django - фреймворк на языке Python, который предоставляет мощные инструменты для разработки веб-приложений. Он обладает удобной ORM, встроенной админ-панелью, защитой от некоторых видов атак, что делает его привлекательным выбором для создания сайтов с бэкендом на Python.

Его минусы:

- Тяжеловесность: Django предлагает много функциональности из коробки, что может привести к избыточности в простых проектах.
- Жесткие структурные принципы: Использование стандартных шаблонов Django и механизмов ORM может ограничивать свободу действий в сложных проектах.

4. Ruby on Rails: Ruby on Rails - фреймворк на языке Ruby, который известен своей простотой и удобством разработки. Rails предлагает множество готовых решений для типичных задач, таких как маршрутизация, ORM, аутентификация и авторизация. Минусы Ruby:

- Производительность: Ruby известен своей медленной производительностью по сравнению с некоторыми другими языками программирования. Это связано с интерпретируемым характером Ruby, который может привести к более долгому времени выполнения кода.
- Масштабируемость: Хотя Ruby хорошо подходит для быстрой разработки прототипов и небольших проектов, его масштабируемость может быть проблемой для крупных и высоконагруженных приложений. Некоторые разработчики сталкиваются с трудностями при масштабировании Ruby-

приложений.

- Зависимость от гемов (gems): Ruby экосистема использует много гемов (библиотек), и хотя это может быть преимуществом в виде большого разнообразия функциональности, некоторые гемы могут оказаться неактуальными или поддерживаться плохо.

5. Spring Boot: Spring Boot - фреймворк на языке Java, который предоставляет широкие возможности для создания веб-приложений. Spring Boot позволяет быстро создавать приложения с использованием готовых модулей (starter packs), что упрощает процесс разработки. Минусы Spring Boot:

- Сложность конфигурации: Spring Boot может иметь довольно высокий порог входа для новичков из-за своей сложности конфигурации. Настройка Spring Boot-приложения может потребовать понимания множества концепций и функций, что может быть вызовом для некоторых разработчиков.
- Объемность: Spring Boot предоставляет множество инструментов и функциональности из коробки, что делает его довольно объемным фреймворком. В результате, приложения, построенные на Spring Boot, могут иметь больший размер и потреблять больше ресурсов.
- Сложность отладки: Иногда отладка Spring Boot-приложений может быть сложной из-за сложной структуры и разветвленных зависимостей, особенно в крупных проектах. Это может затруднить обнаружение и устранение ошибок в приложении

Framework Usage Statistics

Statistics for websites using Framework technologies

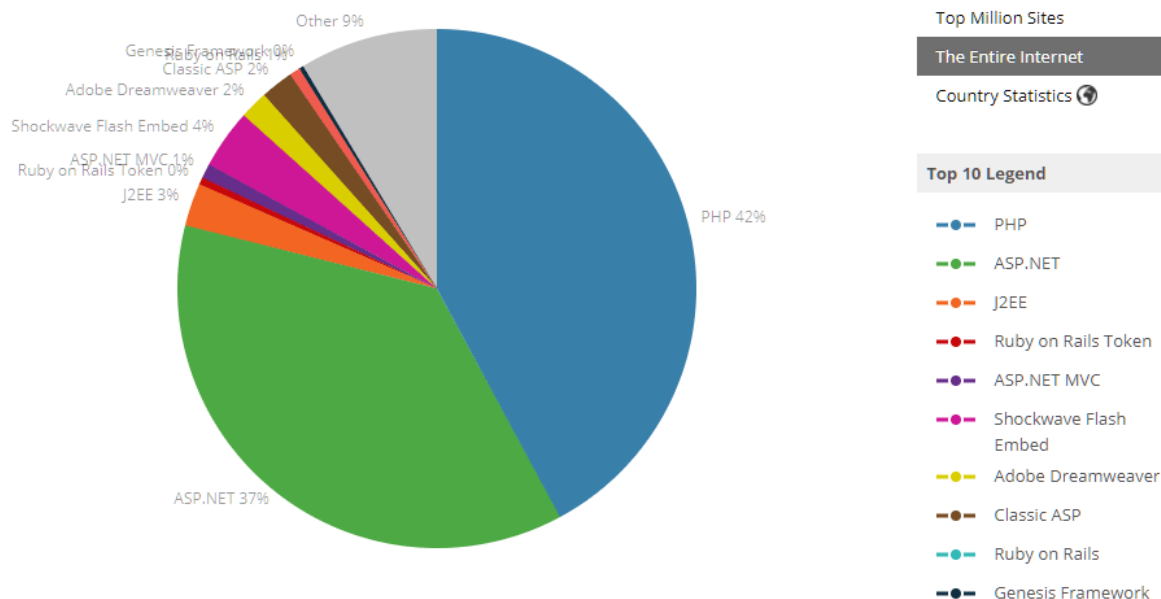


Рис.1. Статистика использования фреймворков

Каждый из этих фреймворков имеет свои особенности и преимущества, и выбор зависит от потребностей проекта, языка программирования, с которым вы работаете, и вашего уровня опыта. ASP.NET Core, безусловно, является отличным выбором благодаря своей гибкости, производительности и современным функциональным возможностям.

Для написания фронтенд части веб-приложений есть несколько основных языков программирования, которые широко используются в индустрии разработки. Вот некоторые из лучших языков программирования для разработки фронтенда:

1. JavaScript: JavaScript является одним из самых популярных языков программирования для фронтенда. Он широко используется для создания интерактивных веб-сайтов, анимаций, игр и многого другого. JavaScript также широко поддерживается браузерами, что делает его незаменимым инструментом для веб-разработки.
2. ReactJS: ReactJS - это библиотека JavaScript, разработанная компанией Facebook, которая позволяет создавать пользовательские интерфейсы.

ReactJS использует концепцию компонентов, что делает разработку веб-приложений более удобной и эффективной.

3. Vue.js: Vue.js - это прогрессивный фреймворк JavaScript, который также используется для создания пользовательских интерфейсов. Vue.js обладает простым синтаксисом, а также хорошо масштабируется, что делает его популярным выбором среди разработчиков.
4. Angular: Angular - это фреймворк JavaScript, разработанный компанией Google, который предоставляет мощные инструменты для создания одностраничных приложений. Angular также использует концепцию компонентов и обеспечивает хорошую производительность.

[Back-end](#) [Front-end](#) [Data processing](#) [Desktop](#) [System](#) [QA automation](#) [Full Stack](#) [IoT](#) [GameDev](#)

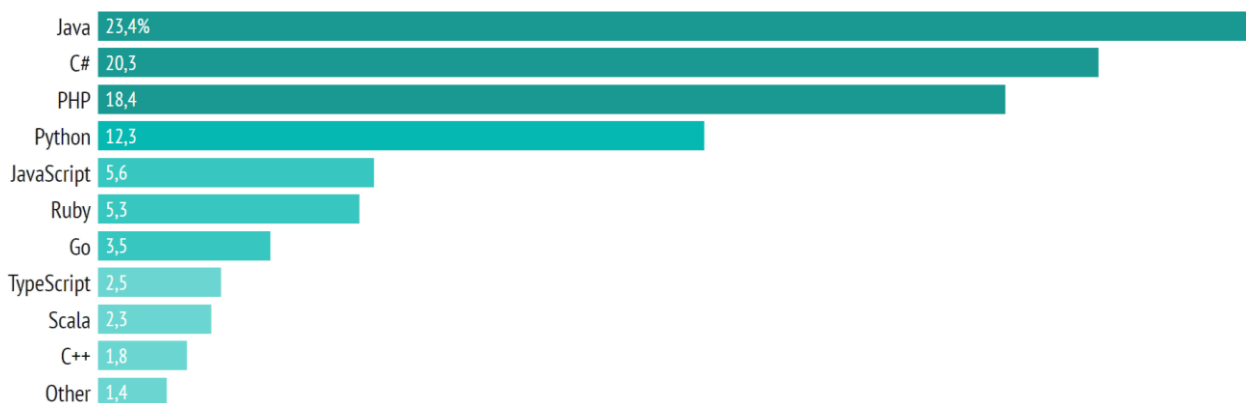


Рис.2. Рейтинг языков программирования в back-end.

Каждый из этих языков и фреймворков имеет свои особенности и преимущества, и выбор зависит от потребностей вашего проекта, личных предпочтений и уровня опыта в разработке фронтенда. JavaScript, безусловно, остается одним из самых важных и широко используемых языков программирования для фронтенд разработки.

ЗАКЛЮЧЕНИЕ

В ходе анализа предметной области по теме я поняла, что создание сайта для бронирования отеля - это задача, требующая внимательного подхода и выбора правильных инструментов. От выбора фреймворков до среды разработки, каждый шаг играет ключевую роль в успехе проекта.

Важно учитывать потребности пользователей и обеспечивать им удобство при бронировании номеров, а также гибкость и безопасность при обработке данных. Подходящая среда разработки способствует ускорению процесса разработки, обеспечивает легкость масштабирования и поддержки проекта в будущем.

Для написания моего проекта по созданию сайта для бронирования номеров в отеле я выбрала связку ASP.Net Core + JavaScript. Эта комбинация позволяет достичь высокой производительности, обеспечивает широкие возможности для разработки как на серверной, так и на клиентской стороне, а также обеспечивает удобство в разработке и поддержке проекта.

В конечном итоге, создание успешного сайта для бронирования отеля требует не только технических знаний, но и понимания потребностей клиентов и правильного подбора инструментов для их удовлетворения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Алексеев А. Н. Проектирование и разработка веб-приложений и веб-сервисов. — С.-Пб.: БХВ-Петербург, 2017.
2. Байрамкулова Ж. А. Разработка веб-порталов с использованием Java Enterprise Edition. — М.: ДМК Пресс, 2019.
3. 10 Best Web Development Frameworks // hackr.io URL: <https://hackr.io/blog/web-development-frameworks> (дата обращения: 12.03.2024).
4. Павлов А. В. Frontend-разработка для веб-программистов. — М.: ДМК Пресс, 2019.
5. Сверивская И. Н. Программирование на языке JavaScript. — М.: Лань, 2018.
6. Фронтенд или бэкенд: по какому пути в разработке пойти // Практикум URL: <https://practicum.yandex.ru/blog/chem-otlichaetsya-backend-i-frontend/> (дата обращения: 16.03.2024).

