



ПОДХОДЫ ИСПОЛЬЗУЕМЫЕ ПРИ РАЗРАБОТКЕ ВЕБ-ПРИЛОЖЕНИЙ

Каримов И.С.

Магистрант, Бухарский государственный университет

Научный руководитель: Касимов Ф.Ф.

к.ф.-м.н. доцент. Бухарский государственный университет

Аннотация. *Разработка веб-приложений является важной частью современной цифровой трансформации бизнеса. Веб-приложения становятся неотъемлемой частью повседневной жизни, обеспечивая пользователей доступом к различным сервисам и функционалу через интернет. Чтобы создать эффективное, безопасное и масштабируемое приложение, разработчики используют различные подходы и методологии, которые помогают справляться с уникальными требованиями и вызовами каждого проекта. В этой статье рассмотрены основные подходы, применяемые в разработке веб-приложений, а также их особенности, преимущества и недостатки. Понимание этих подходов позволяет выбрать оптимальное решение для создания качественных веб-приложений, которые могут эффективно удовлетворять потребности пользователей и бизнеса.*

Ключевые слова: *разработка веб-приложений, микро-сервисная архитектура, модульная архитектура, облачные технологии, API-ориентированное развитие, масштабируемость, гибкость разработки*

Введение. Веб-приложения - в основном, это динамичные приложения, они постоянно меняются и развиваются.

Это все результат развития Интернета и рыночной среды, частью которой являются эти приложения. По мере развития или совершенствования веб-приложений они также могут изменять категорию, к которой они принадлежат. Сама эволюция веб-приложений — это не только изменение категории веб приложения, это вопрос восприятия и мотивации веб-разработки, а также общего изменения в использовании веб-технологий. Сама по себе веб-разработка, требует специальных навыков. К таким навыкам относятся знания в области взаимодействия человека и компьютера, а также навыки в проектировании информационной архитектуры. Мотивация, сильно зависит от первоначальной цели применения и использования веб-приложения, ожиданий клиентов, конкурентной среды и развитию самого Интернета.



Изменение Интернета, изменяет и технологии, которые используются в этой среде. Что в свою очередь отражается и на разрабатываемых веб-приложениях [1].

Такое положение вещей, привело к тому, что существующие методики разработки настольных приложений не подходят для разработки веб-приложений. Основное отличие, это высокая скорость развертывания и моментальная доступность. Такая «гонка за рынком», ставит высокие требования к разработке веб-приложений, постепенно сокращая время, отведенное на разработку. Еще одной важной особенностью, является огромный рынок, ведь каждый веб-браузер, установленный на персональном компьютере это потенциальный пользователь создаваемого веб-приложения. Также влияют как было сказано выше, развитие технологий, вспомогательных инструментов, программных библиотек и каркасов. Появилось большое разнообразие различных инструментов. Но все же в этом большом наборе можно выделить так называемую архитектуру или модели современных веб-приложений [2].

Так, например, в настоящее время разработка веб-приложений может быть сгруппирована в два вида разработки. Первое это традиционное веб-приложение или как их еще называют (многостраничные приложения), а второе это так называемые SPA (одностраничное приложение). Основное отличие состоит в том, что, когда пользователь переходит или как говорят осуществляет навигацию на другой пункт меню, страницу, браузеру требуется время для получения нового HTML-документа с сервера. Внутренняя обработка сервера также может занять некоторое время. В наше время, устройства конечного потребителя будь то компьютер, планшет или мобильный телефон постоянно развиваются. Современные устройства обладают большей мощностью обработки информации и более большим объемом памяти. Благодаря таким улучшенным характеристикам большую долю логики и обработки приложения можно передать конечному устройству потребителя. Это освободит сервер от использования большого количества ресурсов для каждого клиента и большого количества запросов и обращений к серверу. Поэтому отталкиваясь от этой концепции, к ней лучше всего подходит так называемая одностраничная модель приложения (SPA – single page application). В так называемых одностраничных приложениях все содержимое приложения загружается сразу и поэтому начальная загрузка страницы обычно длиннее, но последние изменения страницы происходят мгновенно. Это достигается за счет



манипуляций моделью HTML документа или по-другому манипуляцией на DOM деревом.

DOM означает объектную модель документа и является абстракцией структурированного текста. Для веб-разработчиков этот текст является HTML кодом, а DOM просто называется HTML DOM. Элементы HTML становятся узлами в DOM. Поэтому одностраничные приложения динамически получают доступ и обновляют содержимое, структуру и стиль документа. С помощью объектной модели документа программисты могут создавать документы, перемещаться по их структуре и добавлять, изменять или удалять элементы и содержимое. DOM представляет документ как узлы и объекты. Таким образом, языки программирования могут подключаться к странице. DOM – это объектно-ориентированное представление веб-страницы, которое может быть изменено с помощью языка сценариев, такого как JavaScript. Все, что найдено в HTML документе, можно получить, изменить, удалить или добавить с помощью объектной модели документа.

И по мере развития одностраничных приложений, все больше и больше обработки стало сосредоточено на стороне клиента. Поэтому такие языки программирования как JavaScript тоже стали развиваться. Все это породило появление большого количества инструментов, программных каркасов, методик и т.д. Это все увеличивает сложность разработки и понимание того какой результат необходимо достичь.

Проблема как кажется автору данной работы лежит в следующем. Например, если взять и проанализировать существующие так называемые классические методы разработки веб-приложений [3].

1. Каскадная модель или «водопад».

Это одна из самых старых методик разработки программного обеспечения. Эта методика подразумевает прохождение всех этапов. Каждый из которых в свою очередь должен завершиться полностью до начала, следующего этапа (Рисунок 1).

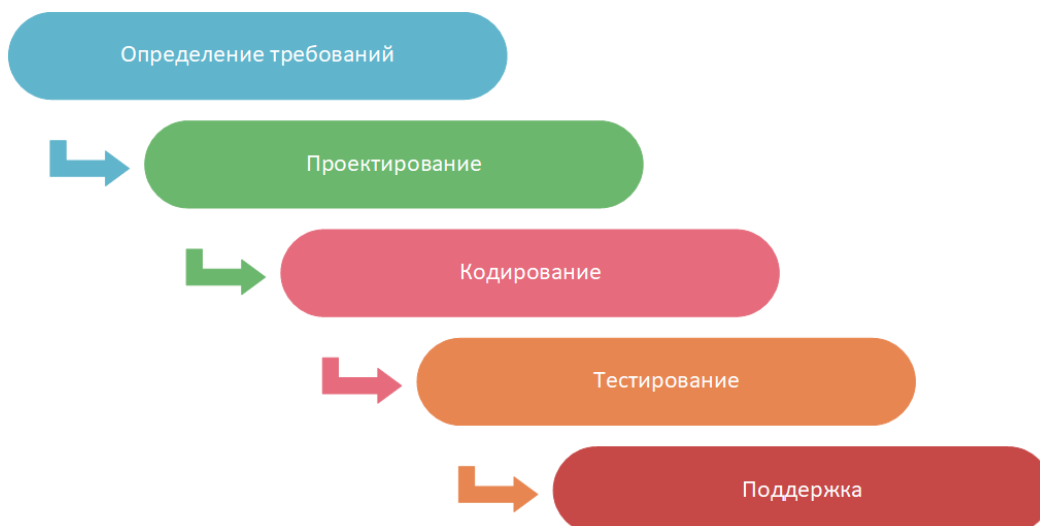


Рисунок 1 - Каскадная модель разработки информационных систем

2. Итеративная модель.

При использовании итеративной модели в начале жизненного цикла нет нужды использовать какие-либо требования, вместо этого создается начальная версия продукта [4]. Которая в свою очередь и является точкой отсчета, т.е. служит исходным материалом для определения требований (Рисунок 2).



Рисунок 2 – Итеративная модель

3. Инкрементная модель.

Модель используется для поэтапной сборки программного обеспечения. Она имеет несколько циклов разработки, где они и составляют общий жизненный цикл информационной системы [5]. Сам цикл разделен на мелкие модули. В свою очередь каждый модуль проходит всю цепочку, анализ,



проектирование, реализация, тестирование. Сам процесс продолжается пока не будет создана вся система (Рисунок 3).



Рисунок 3 – Инкрементная модель

Самыми современными и известными методологиями проектирования являются WebML и WSDM. WebML – это одновременно метод разработки и язык Web Modeling Language с помощью которого производится проектирование и разработка; WSDM – это один из первых методов разработки веб-приложений Web Site Design Method. WebML – это визуальный язык для определения структуры контента веб-приложения, а также организации и представления такого контента в гипертексте.

Заключение. В разработке веб-приложений важно выбрать подход, который соответствует специфике проекта и требованиям бизнеса. Классическая модель Waterfall, гибкие методологии Agile, микросервисная архитектура, модульные подходы и использование облачных технологий — все эти методы имеют свои преимущества и недостатки. В зависимости от сложности проекта, сроков и необходимости масштабирования, разработчики могут комбинировать различные подходы для достижения наилучшего результата. В условиях постоянных изменений в технологиях и потребностях пользователей, гибкость, масштабируемость и безопасность становятся основными приоритетами при создании веб-приложений. Таким образом, правильный выбор подхода является ключом к успешной разработке,



обеспечению высокого качества и удовлетворению требований конечных пользователей.

Список использованной литературы

1. Использование метода композиционного планирования эксперимента для описания технологических процессов: метод. указания / сост. А.Н.Гайдадин, С.А.Ефремова; ВолгГТУ. – Волгоград, 2008. – 16 с.
2. Жумаев Ж., Опокина Н.А. Решение математических задач в пакетах математических программ Maxima и Mathcad// Электронное учебное пособие. Казань. 2021. 227 с. https://repository.kpfu.ru/?p_id=253062
3. Хайдаров А., Жумаев Ж., Шафиев Т.Р. Основы математического моделирования// Учебник. Бухара. «Дурдона», 2022. 216 с.
4. Жумаев Ж., Мирзаева Ш.У. Совершенствование процесса экстракции лакричного корня в среде CO₂// Монография. Изд-ва «Дурдона». 2023. 144 с.
5. Агалаков С.А., Эконометрические модели [Текст] : учебное пособие / С. А. Агалаков ; М-во образования и науки Российской Федерации, Федеральное гос. бюджетное образовательное учреждение высш. проф. образования Омский гос. ун-т им. Ф. М. Достоевского. - Омск : Изд-во Омского гос. ун-та им. Ф. М. Достоевского, 2015. - 140 с.