**Создание сайта-агрегатора отелей для животных**

**Тема для разработки веб-сайта**: сайт-агрегатор отелей для животных.

**Заказчиком** является владелец онлайн-сервиса Туту.ру, который хочет занять нишу агрегатора отелей для животных. Он имеет довольно ясные представления о продукте, который хочет получить и, следовательно, дает значительную часть требований для проекта.

**Требования к проекту**

* Наличие системы аккаунтов, а также разных уровней доступа
* Наличие таких критериев для отелей, как: цена, рейтинг и отзывы, наличие разного рода услуг
* Подключение к картографической службе для того, чтобы увидеть нужный отель на карте, либо посмотреть, какие отели для животных находятся ближе всего к пользователю
* Обеспечение безопасности персональных данных пользователей, а также защиту от DDoS-атак на сайт

**Существующими аналогами** на сервисы являются: **Туту.ру**, Суточно.ру, Яндекс.Путешествия, Booking.com.

**Предполагаемая нагрузка на сайт**:

**Сильной стороной** проекта является то, что существует сервис (Туту.ру), который будет играть роль опоры в разработке веб-сайта, так как заказчиком является владелец этого сервиса-опоры, который желает, чтобы веб-сайты были похожи друг на друга и обладали общими функциями. **Слабой** же **стороной** проекта является то, что точно рассчитать нагрузку на веб-сайт не представляется возможным, следовательно, существует вероятность необходимости смены серверов.

Роли в нашей команде:

1. Разработчик backend – разработчик «задней» стороны сайта. Занимается безопасностью приложения

2. Разработчик frontend – разрабатывает необходимые скрипты на JavaScript.

3. Верстальщик – верстает макеты на html и css.

4. Дизайнер – создает макеты страницы по т.з.

5. Архитектор – разработка архитектуры веб-приложения.

6. Планировщик БД – проектирование и создание баз данных.

7. Тех. писатель – составление документации.

8. Специалист по контролю качества – проверяет качество продукции.

9. Системный аналитик – работает в паре с бизнес аналитиком, но в отличие от него обращает внимание на то, как система должна работать в различных вариантах использования.

10. Тестировщик приложения – тестирует базовые функции приложения.

11. Тестировщик безопасности – тестирует безопасность приложения.

12. бизнес аналитик – договаривается с заказчиком, передает требования менеджеру, анализирует похожие сайты.

13. менеджер проекта – координирует команду, создает т.з.

14. релиз менеджер – осветляет заказчику сроки выхода продукта.

**Модели, подходящие для данного проекта:**

1. **V-образная модель.** Модель основана на объединении фазы тестирования с каждой соответствующей стадией разработки. Разработка каждого шага напрямую связана с этапом тестирования. Следующая фаза начинается только после завершения предыдущей. Каждый этап разработки, напрямую связан с тестированием этого этапа.

Данная модель позволяет минимизировать риски, что позволяет выявить отклонения в проекте и риски на ранних стадиях. Но нашему проекту она не подходит, так как является прогнозирующей и смену серверов не предусматривает.

1. **Инкрементная модель.** Модель основана на разработке конечного программного продукта отдельными сборками или приращениями. Весь цикл разработки разбивается на более лёгкие и быстрые этапы. При этом каждый из этапов включает все фазы жизненного цикла.

С помощью инкрементной модели можно быстро занять нишу сайта-агрегатора отелей для животных, а дальше, ориентируясь на фидбек от пользователей, добавлять новые функции. Но минусом является то же самое – невозможность смены серверов, так как модель является прогнозирующей

1. **Спиральная модель.** Модель, которая сочетает в себе как проектирование, так и постадийное прототипирование с целью сочетания преимуществ восходящей и нисходящей концепции. Отличительной особенностью этой модели является особое внимание рискам, влияющим на организацию жизненного цикла.

Данная модель является адаптивной и предполагает возможность изменения требований на каждой итерации, так что она подходит больше всего для проекта, ведь с ее помощью возможно сменить сервера в ходе разработки.

Этапы создания продукта:

1. Анализ требований

- Сбор и анализ требований

- Определение ограничений системы

- Разработка спецификаций требований

2. Проектирование

- Архитектурное проектирование

- Детальное проектирование

- Проектирование интерфейсов

3. Разработка

- Кодирование

- Модульное тестирование

- Интеграционное тестирование

4. Тестирование

- Системное тестирование

- Приемочное тестирование

- Нагрузочное тестирование

5. Внедрение

- Подготовка документации

- Обучение пользователей

- Установка и настройка системы

6. Поддержка и сопровождение

- Обработка инцидентов и запросов на изменения

- Обновление и улучшение системы

- Мониторинг производительности

7. Оценка и управление рисками

- Идентификация рисков

- Оценка рисков

- Разработка стратегий управления рисками

Эти виды деятельности помогают организовать процесс разработки программного обеспечения и обеспечить его высокое качество на всех этапах жизненного цикла.