**Язык программирования**

Выберем язык программирования для написания серверной части веб приложения. Сервер принимает запросы клиентов и возвращается им некоторую информацию. Рассмотрим популярные серверные кроссплатформенные языки, ориентированные на веб разработку:

* PHP

Один из наиболее популярных скриптовых языков применяемый для разработки веб-приложений. Язык и его интерпретатор разрабатываются группой энтузиастов в рамках проекта с открытым исходным кодом.

Основные преимущества:

* + легкость изучения;
  + быстрота разработки;
  + простая поддержка;
  + язык активно развивается.

Недостатки:

* + отсутствует многопоточность;
  + большое количество глобальных функций;
  + бессистемное именование функций.
* Java

Объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Приложения Java обычно транслируются в специальный байт-код, поэтому они могут работать на любой виртуальной Java-машине вне зависимости от компьютерной архитектуры.

Преимущества:

* + мощные стандартные библиотеки;
  + разнообразные инструменты сборки;
  + удобный механизм многопоточности;
  + автоматическая сборка мусора;
  + надежность.

Недостатки:

* + ресурсоемкость;
  + время разработки;
  + код программ многословен.

Оба языка имеют большое количество библиотек, большие сообщества и хорошую документацию. Java имеет строгую типизацию, компилятор позволяет выявлять большинство ошибок на этапе сборки приложения. За счет этого повышается надежность разрабатываемых программ и поэтому будем использовать Java.

**Система управления базами данных**

Система управления базой данных (СУБД) представляет собой программное обеспечение, которое управляет всем доступом к базе данных.

Основные требования к СУДБ:

* реляционная модель данных;
* поддержка клиент-серверной архитектуры взаимодействия;
* низкая стоимость;
* кроссплатформенность.

Рассмотрим наиболее популярные свободные и кроссплатформенные реляционные СУБД:

* MySQL

MySQL является свободной система управления базами данных (СУБД). Разработку и поддержку MySQL осуществляет корпорация Oracle.

В качестве преимуществ можно отметить:

* + скорость работы;
  + кэш запросов, за счет которого ускоряется обработка повторяющихся запросов;
  + поддержка различных типов таблиц.

Недостатки:

* + отсутствует оптимизация для подзапросов;
  + команды работы со структурой базы данных не участвуют в транзакциях.
* MariaDB

Данная СУБД — это ответвление СУБД MySQL, разрабатываемое сообществом.

MariaDB относительно MySQL поставляется с большим количеством типов таблиц, имеет оптимизацию подзапросов

* PostgreSQL - система управления базами данных с открытым исходным кодом.

Разработка PostgreSQL является общественным проектом и не управляется какой-либо компанией.

Преимущества:

* последовательности – генераторы уникальных последовательностей чисел;
* большое количество стандартных типов данных;
* механизм транзакций, поддерживающий команды работы со структурой базы данных;
* операция сборки мусора VACUUM;
* хорошая документация.

Недостатки:

* отсутствует кэш запросов;
* уступает MySQL и PostgreSQL в скорости работы

Все рассмотренные СУБД являются кроссплатформенными и имеют API для популярных языков программирования. Выберем PostgreSQL, так как эта СУБД имеет преимущество над MySQL и MariaDB: в PostgreSQL операции, модифицирующие записи UPDATE и DELETE, не удаляют на самом деле кортеж, а помечают его удаленным (в случае обновления – старая версия кортежа помечается как удаленная, а новая версия вставляется). Такой подход позволяет избежать падения производительности при удалении элементов и ускорить обновление, но это приводит к тому, что в ходе модификации таблицы в ней появляется неиспользуемое пространство (старые удаленные кортежи). Это пространство не может быть повторно использовано без предварительной подготовки. Для решения данной проблемы используется команда сборки мусора – VACUUM.

**Сервер приложений**

Сервер приложений – программа промежуточного слоем между клиентом и сервером, которая предоставляет среду для развертывания, управления и выполнения Java приложений.

Рассмотрим сервера приложений с открытым исходным кодом, поддерживающие Java EE 7:

* Glassfish

Сервер приложений с открытым исходным кодом, реализующий спецификации Java EE, принадлежит корпорации Oracle. С 2013 года Oracle прекратили оказание услуг коммерческой техподдержки. Пользователям коммерческих версий было рекомендовано перейти на закрытый сервер приложений WebLogic Server.

* WildFly

Данный сервер приложений - развиваемый сообществом проект, на основе которого создаётся сертифицированный для Java EE коммерческий продукт JBoss Enterprise Application Platform. Коммерческая версия JBoss предоставляет такие возможности, как пятилетний жизненный цикл поддержки, системы кэширования и кластеризации, устойчивость к веб-атакам и собственную среду разработки.

Выберем WildFly, потому что у него большее сообщество, к нему чаще выходят обновления и он имеет режим Server Suspend Mode. В режиме Server Suspend Mode сервер перестает принимать новые запросы, обрабатывает оставшиеся в очереди запросы и выключается, для проведения необходимых работы по обновлению программного обеспечения.

**Java фреймворк**

Фреймворк (от англ. framework — каркас, структура) - программная платформа, определяющая структуру программной системы; программное обеспечение, облегчающее разработку и объединение разных компонентов большого программного проекта.

Рассмотрим наиболее популярные фреймворки:

* Struts

Фреймворк с открытым исходным кодом для разработки веб-приложений на Java

Достоинства:

* + легок в изучении;
  + помощь в валидации форм;
  + простая работа с событиями на стороне сервера.

Недостатки:

* + сложно реализовывать не предусмотренную авторами функциональность;
  + плохая документация;
  + проблемы с совместимостью версий.
* Java Server Faces (JSF)

Фреймворк для веб-приложений, написанный на Java. Он служит для того, чтобы облегчать разработку пользовательских интерфейсов для Java-приложений. В отличие от прочих фреймворков, которые управляются запросами, подход JSF основывается на использовании компонентов. Фреймворк предоставляет набор стандартных элементов для построения пользовательских интерфейсов обработку клиентских событий на сервере.

Достоинства:

* + большое количество элементов пользовательского интерфейса;
  + экономия времени на верстке - html и javascript код генерируются фреймворком;
  + простая работа с событиями на стороне сервера.

Недостатки:

* + сложно реализовывать не предусмотренную авторами функциональность;
  + проблемы с совместимостью версий.
* Spring Framework

**У**ниверсальный фреймворк с открытым исходным кодом для Java-платформы. Он имеет довольно широкую функциональность и активно используется при разработке сложных бизнес-приложений. **Spring** Framework может быть рассмотрен как коллекция меньших фреймворков. Большинство этих фреймворков может работать независимо друг от друга.

Достоинства:

* хорошая документация;
* большое сообщество;
* гибкость - содержит несколько модулей, которые не зависят друг от друга.

Недостатки:

* сложен для изучения;
* не предоставляет инструментов для создания пользовательского интерфейса;

Будем использовать Spring Framework, так как он состоит из компонентов, которые не зависят друг от друга. Этот подход позволяет выбрать только нужные компоненты и не подключать к нашему приложению лишние библиотеки, которые не будут использоваться.

**CSS фреймворк**

CSS-фреймворк помогает упростить работу по верстке, увеличить скорость разработки пользовательских интерфейсов и решить проблемы совместимости различных версий браузеров.

Рассмотрим самые популярные css-фреймворки, которые позволяют создавать отзывчивый дизайн:

* Bootstrap

Набор инструментов для создания сайтов и веб-приложений. Включает в себя HTML и CSS шаблоны оформления для типографики, веб-форм, кнопок, меток, блоков навигации и прочих компонентов веб-интерфейсов, включая JavaScript расширения.

Достоинства:

* + содержит большое количество готовых компонентов;
  + есть онлайн настройщик;
  + имеет много различных дополнений.

Недостатки:

* + плохая документация для препроцессоров;
  + большой вес.
* Foundation

Фреймворк предоставляет

Достоинства:

* + хорошая документация для препроцессоров;
  + имеет несколько уникальных компонентов.

Недостатки:

* + нет онлайн настройщика.
* Pure

Легкий модульный фреймворк, который предлагает только основу для проекта и некоторые основные компоненты пользовательского интерфейса.

Достоинства:

* + маленький вес.

Недостатки:

* + нет дополнений;
  + нет иконок.

Все фреймворки имеют хорошую поддержку большого количества браузеров. Bootstrap и Foundation хорошо подходят для создания отзывчивого дизайна. Так как Bootstrap содержит больше готовых компонентов и для него существует большое количество дополнений будем использовать его.