Выполнил: Большаков Илья Вячеславович, 298 группа.

**Отчет по практической работе №5**

**«Выбор средств реализации»**

**Цель работы:** выбрать средства реализации учебной практики.

Таблица 1 – Сравнение средств создания макета сайта

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Критерий | Скорость прототипировани я страницы (высокая, средняя, низкая) | Генераци я html кода и стилей (CSS) (да, нет, частично) | Необходимост ь знания HTML, CSS  (да, нет, частично) | Поддержка библиотек (указываютс  я  библиотеки, нет) | Визуализаци я (только макет, макет и стили, полностью готовый код) |
| NinjaMoc  k | Высокая | нет | нет | нет | макет |
| Figma | Высокая | частично | нет | нет | макет и стили |
| Photoshop | Высокая | нет | нет | нет | макет и стили |

NinjaMock был выбран, так как он позволяет быстро создавать простые прототипы сайтов. В сервисе присутствует возможность визуализации переходов между страницами, совестная работа над проектом.

Скорость прототипирования достаточно высокая, так как используются готовые графические элементы для создания прототипы. В то же время, в сервисе не предусмотрена генерация кода. Явным преимуществом с NinjaMock является отсутствие необходимости знаний в области HTML, CSS и методов верстки сайта. Стоит отметить, что в сервисе используются графические элементы, отображающие элементы библиотеки bootstrap, однако это не является полноценной поддержкой.

Figma была выбрана, так как очень удобно создавать макеты сайтов. В нем есть много инструментов, множество шрифтов, имеется прототип для мобильных приложений, удобный интерфейс.

Скорость прототипирования высокая, так как есть возможность продолжить создание дизайна на основе прототипа. Гинерация кода (готовый код) имеется при помощи плагина. С самого начала поддерживает генерацию CSS стилей и кода для мобильных устройств. Figma не использует элементы библиотеки.

Photoshop удобен для создания картинок, графического дизайна.

Скорость прототипирования высокая, средняя скорость создания прототипа, высокая детализация прототипа, прототипы имеют эстетичный вид, высокая скорость внесения изменений без повторной отрисовки прототипа, полная доступность для всех участников разработки проекта.

Таблица 2 – Сравнение баз данных

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Критерий | MySQ  L | Maria  DB | Red  is | Mongo  DB | Neo  4j | Cassand  ra | PostgreS  QL | SQLi  te | memcach  ed | ClickHo use |
| Вид базы данных (реляционна я (NoSQL, «ключзначение», документна  я, графовая, колоночная) | Реляционная | Реляционная | NoSQL  «ключ- значение» | Документно-ориентированная | Графовая | NoSQL | Реляционная | Реляционная | NoSQL  «ключ- значение» | Колоночная |

Разрыв страницы

Продолжение на следующей странице

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Для каких данных используетс  я | Числовые данные, дата и время, и данные типа строка | Строковые, числовые, дату/время и типы данных больших объектов | Cтроки, [списки](https://ru.wikipedia.org/wiki/%D0%A1%D0%BF%D0%B8%D1%81%D0%BE%D0%BA_(%D0%B8%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), [множества](https://ru.wikipedia.org/wiki/%D0%9C%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%BE_(%D1%82%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85)), [хеш-таблицы](https://ru.wikipedia.org/wiki/%D0%A5%D0%B5%D1%88-%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86%D0%B0), упорядоченные множества. | Целые числа, булевы значения, строки, массивы, объекты, дата и время, ID объекта, JavaScript код, бинарные данные | Целое число, дата и время, булевы значения, ID узлов, список ,  карта | Строки,  Большие целые числа,  Список , множества | Числовые, символьные, логические, дата и время, бинарные | Целые числа, булевы значения, строки | Строковые, числовые, дату/время и типы данных больших объектов | Числовые данные, дата и время, и данные типа строка, карты |
| Максимальн ый размер базы данных | Числовые данные, дата и время, и данные типа строка, карты | 4 ГБ | Максимальный размер ключа - 512 MB | Документ – 16 мб | Размер оперативной памяти | Нет ограничения | 32 Тбайт | 140 Тбайт | Размер оперативной памяти | Нет ограничения |
| Максимальн ый размер кластера | 128 МБ | 128 МБ | Длина от 0 до 256 символов | 64 МБ | - | - | 1,6 Тбайт | - | - | - |

**3. Выбор библиотек для реализации бэкенда.**

В качестве основного языка был выбран NodeJS, его преимуществами из-за которых он был выбран: Скорость работы

В качестве фреймворка для NodeJS был выбран ExpressJS, за его Middleware и простоту

Для авторизации будет использоваться JsonWebToken так как это один из самых безопасных способов на данный момент, а для шифрования паролей – bcrypt, был выбран за надежность и простоту в работе

Как база данных выбрана MongoDB

Для работы с бд из NodeJS будет использоваться Mongoose, как ORM

Supervisor – как утилита командной строки для автоматического перезапуска сервера

**Список используемых источников:**

Сравнение современных СУБД . — Текст : электронный // Drach : [сайт]. — URL: https://drach.pro/blog/hi-tech/item/145-db-comparison (дата обращения: 27.05.2021).

NodeJS. — Текст : электронный // NodeJS : [сайт]. — URL: https://nodejs.org/ru/ (дата обращения: 27.05.2021).

ExpressJS. — Текст : электронный // ExpressJS : [сайт]. — URL: https://expressjs.com/ru/ (дата обращения: 27.05.2021).

Bcrypt. — Текст : электронный // NPM : [сайт]. — URL: https://www.npmjs.com/package/bcrypt (дата обращения: 27.05.2021).

Jsonwebtoken. — Текст : электронный // NPM : [сайт]. — URL: https://www.npmjs.com/package/jsonwebtoken (дата обращения: 27.05.2021).

**Вывод:**

Были определены основной язык, библиотеки и фреймворки для него, также проведено сравнение современных СУБД и средств для графического дизайна