Выполнил: Россь Виталий Владиславович, 286 группа.

**Отчет по практической работе № 5**

**«Выбор средств реализации»**

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

Решение задач.

1. **Произвести обзор инструментов для создания макетов сайтов**,не менее трех.

Примерами средств могут быть как онлайн инструменты, так и приложения. К примеру: Figma, Photoshop, Axure, NinjaMock, Moqups, Sketch (только для Mac). Так же может быть рассмотрен Bootstrapstudio (https://bootstrapstudio.io/) или аналогичные инструменты.

Провести оценку этих средств, заполнив таблицу 1 и привести обоснование каждого значения критерия (см. «Пример обоснования»).

Таблица 1 – Сравнение средств создания макета сайта

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Критерий | Скорость прототипирования (высокая, средняя, низкая) | Генерация html кода и стилей (CSS) (да, нет, частично) | Необходимость знания HTML, CSS (да, нет, частично) | Поддержка библиотек (указываются библиотеки, нет) | Визуализация (только макет, макет и стили, полностью готовый код) |
| NinjaMock | высокая | нет | нет | нет | макет |
| Figma | высокая | да | да | Team Libraries | макет и стили |
| Moqups | высокая | нет | нет | нет | макет |

**Обоснование каждого значения критерия.**

Figma был выбран, так как он позволяет быстро создавать простые прототипы сайтов. В сервисе присутствует возможность визуализации переходов между страницами, совместная работа над проектом. Скорость прототипирования достаточно высокая, так как используются готовые графические элементы для создания прототипа. В то же время, в сервисе не предусмотрена генерация кода. Явным преимуществом Figma является отсутствие необходимости знаний в области HTML, CSS и методов верстки сайта. Moqups и NinjaMock являются похожими средствами создания прототипов сайтов. Преимущество Moqups перед NinjaMock состоит в расширенном прототипировании интерфейса (белая доска, приглашение гостей, комментарии, оптимизация под мобильные устройства).

1. **Выбрать средства реализации фронтенда.** Определить список необходимыхбиблиотек или языков для верстки. В этом разделе так же могут быть указаны методологии (к примеру, БЭМ).

**Список необходимых библиотек или языков для верстки:**

* CSS фреймворк – Bootstrap
* JS-библиотека для реализации анимации и других функций на стороне клиента – Jquery
* визуализация данных - Chart.js

Краткое описание основных преимуществ и недостатков **выбранных** средств реализации:

Bootstrap (также известен как Twitter Bootstrap) — свободный 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-интерфейса, включая JavaScript-расширения. Bootstrap использует современные наработки в области CSS и HTML.

Библиотека JQuery – одна из самых популярных библиотек JavaScript, которые можно использовать на сайте, чтобы добавлять интересные и полезные эффекты. Существует около 50 библиотек JQuery, функций которых хватит для реализации большинства задач.

Chart.js — простая и в то же время довольно гибкая JavaScript библиотека для визуализации данных. Она представляет собой прекрасное базовое решение для тех, кому не требуется большое разнообразие типов графиков и индивидуальных настроек, но кому достаточно, чтобы графики выглядели аккуратно, наглядно и информативно.

1. **Выбор базы данных.**

* качестве базы данных рекомендуется использовать MySql. Однако, вы можете использовать и другие базы данных. Для понимания, какая из баз данных может лучше вам подходит, заполните таблицу 2.

Таблица 2 – Сравнение баз данных.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Критерий | MySQL | MariaDB | Redis | MongoDB | Neo4j | Cassandra | PostgreSQL | SQLite | memcached | ClickHouse |
| Вид базы данных (реляционная, MySQL, «ключ-значение», документная, графовая, колоночная) | реляционная | реляционная | реляционная | реляционная | реляционная | реляционная | реляционная | реляционная | реляционная | реляционная |
| Для каких данных используется | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных | Все типы данных |
| Максимальный размер базы данных | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. |
| Максимальный размер кластера | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. | Не огр. |

1. **Выбор языка реализации и фреймворка (если необходимо) для реализации**

**бэкенда.**

Вы можете выбрать **любой язык** для реализации. Необходимо привести обоснование, почему вы выбрали тот или иной язык. Объем обоснования выбора языка и фреймворков должен быть не менее одной страницы и содержать ссылки на источники, оформленные по ГОСТ.

**Причины выбора php в качестве языка реализации.**

Я выбрал php в качестве языка реализации, так как он очень распространен в сфере веб-дизайна (php был создан примерно 40 лет назад. За это время с его помощью было написано множество сайтов и программ). Также, он обладает следующими преимуществами.

*Простота*. Код, написанный на php, легок для чтения и понимания. Благодаря этому в случае возникновения проблем разработчик может сосредоточиться именно на них, не отвлекаясь при этом на сложности языка.

*Бесплатность*. php это бесплатная и открытая технология. Разработчики могут ее распространять, копировать и изменять. Как следствие этого, php быстро развивается и имеет хорошее сообщество. Что касается последнего, оно поддерживает начинающих разработчиков, делясь с ними знаниями.

*Совместимость*. php предлагает совместимость с различными платформами. Это избавляет разработчиков, пишущих на php, от многих проблем, обычных для других языков.

*Объектно-ориентированность*. php поддерживает как процедурно-ориентированное, так и объектно-ориентированное программирование. Первое позволяет применять разные части кода повторно.

*Обилие библиотек.* Для php создано очень много библиотек. Благодаря этому разработчикам проще работать с документацией, базами данных и браузерами, проводить модульное тестирование и т. д. Кроме того, php может использоваться для самых разных задач, например, для разработки десктопных и веб-приложений, сложных вычислительных систем, игр, интернета вещей и многого другого.

*Сообщество php.* Код на php хорошо читается и понятен даже разработчикам, никогда не работавшим с этим языком. В результате сообщество php непрерывно растет. Кроме того, этот язык преобладает в сфере науки, так что среди его пользователей много профессоров и ученых.

Вывод. Таким образом, языку php присущи скорость работы, наличие документации на русском языке, сообщество языка, небольшое использование памяти, низкая степень сложности развертывания на сервере.

**Так же вам необходимо выбрать фреймворк для реализации или обосновать отказ от его использования.**

**Причины выбора фреймворка Yii2 для реализации.**

Yii2 это современный и широконаправленный PHP фреймворк, для разработки веб и консольных приложений. Он призывает к чистому написанию кода, без лишних связанностей, принципу DRY (don’t repeat yourself) направленный на снижение повторения кода, событийно-ориентированное программирование, когда выполнение подпрограммы определяется исходя из события запущенного пользователем, соглашение по конфигурации, по которому если класс соответствует соглашению наименованию, тогда он не нуждается в дополнительной конфигурации, а так же модульная структура приложения. Для фреймворка уже разработано много библиотек, виджетов и расширений как основными, так и сторонними разработчиками. Это все позволяет быстро разрабатывать качественные приложения.

Изучать и осваивать Yii2 поможет онлайн документация и [wiki](http://www.yiiframework.com/wiki/?tag=yii2), есть [документация на русском](https://github.com/yiisoft/yii2/tree/master/docs/guide-ru). Многие вопросы находят свои ответы на [stackoverflow](http://stackoverflow.com/questions/tagged/yii2), а также возможность задать вопросы в соответствующем разделе ([issue tracker](https://github.com/yiisoft/yii2/issues" \t "_blank)) на GitHub. Помимо этого есть [форум на официальном сайте](http://www.yiiframework.com/forum/index.php/forum/42-general-discussions-for-yii-20/), [чат](http://www.yiiframework.com/chat/), активное [сообщество в facebook](https://www.facebook.com/groups/yiitalk/).

Сразу из коробки реализована отложенная загрузка (lazy loading), это позволяет загружать и исполнять только те компоненты, которые нужны для выполнения запроса. Можно увеличить производительность за счет применения различных техник кеширования.

Это отличный фреймворк как для небольших проектов, которые легко можно начать, взяв за основу базовый шаблон так и для масштабных проектов, можно использовать продвинутый (advanced) шаблон. Хотя, нужно отметить, что выбор шаблона это совершенно не обязательный пункт, можно скачать фреймворк и без шаблона, выстроив необходимую структуру. Для большинства проектов Yii2 будет также удобен за счет возможности разделения функциональности на модули.

1. **Оформить отчет по работе. Отчет должен быть оформлен по ГОСТ 7.32-2017**
2. Зафиксировать отчет в репозитории с названием коммита «Выбор средств реализации»
3. Поставить отметку («+») о выполнении задания в таблице https://docs.google.com/spreadsheets/d/1zY1FVLFKwMr5mjItVXQQ7BKncezWVFRS Cf4oxBCDyig/edit?usp=sharing