



Критерии акселератор: Фронтенд-разработчик



рейтинг

↗ Related to Докуме... Empty

↗ Документы поток... Empty



Направление

Фронтенд

Обязательные критерии

HTML2-B22. Процесс сборки личного проекта настроен с помощью `Guip`.

HTML2-B23. Все зависимости проекта должны быть указаны в файле `package.json`.

Команда `npm ci` должна установить всё необходимое для того, чтобы сборка проекта работала.

HTML2-B24. Сборка проекта запускается командами `build` и `start`:

- `npm run build` — запускает сборку проекта;
- `npm run start` — запускает сборку проекта `build`, а после запускает `server`.

Также при сборке не должно возникать никаких ошибок.

HTML2-B11. При наполнении контентом (как в макете) элементы каждой версии страницы (мобильной, планшетной и десктопной) соответствуют макету.

► Текст критерия

HTML1-TEST-01. Кроссбраузерность.

- ▶ Текст критерия

HTML1-TEST-03. Размеры страницы.

- ▶ Текст критерия

HTML1-TEST-05. Шрифты.

- ▶ Текст критерия

HTML1-TEST-08. Взаимодействие.

При взаимодействии с элементами (наведение, нажатие) ни сам элемент, ни окружающие его блоки не меняют своего положения, если такое поведение не предусмотрено макетом.

HTML1-TEST-07. Состояние элементов соответствует стайлгайду

- ▶ Текст критерия

HTML1-HTML-04. Работоспособные формы.

- ▶ Текст критерия

HTML2-B19. Логотип должен адаптироваться

Логотип должен адаптироваться к мобильному, планшетному и десктопному состоянию проекта.

JS1-B1. Поведение интерактивных элементов соответствует техническому заданию проекта

JS1-B21. Код является кроссбраузерным и не вызывает ошибок в разных браузерах и разных операционных системах.

HTML1-HTML-07. Валидация HTML.

- ▶ Текст критерия

HTML1-PROJ-01. Кодгайд вёрстки

Код должен быть написан с соблюдением требований кодгайда Академии.

Соблюдать правило сортировки CSS-свойств необязательно.

HTML1-HTML-05. Подписи полей форм.

- ▶ Текст критерия

HTML1-CSS-01. Подключение стилей.

- ▶ Текст критерия

HTML1-CSS-04. ID в селекторах.

- ▶ Текст критерия

HTML1-CSS-05. Использование `!important`.

- ▶ Текст критерия

HTML2-B5. Отсутствуют типовые ошибки в разметке по методологии

- ▶ Текст критерия

HTML2-D1. У всех векторных изображений размер прописан в теге ``, у встроенных SVG-изображений размер прописан в теге `<svg>`.

- ▶ Текст критерия

HTML2-B15. В разметке есть правильный выюпорт тег

Адаптивность должна работать на планшетах и мобильных устройствах. То есть не только при изменении ширины окна браузера.

HTML2-D11. Родительский селектор `&` используется только для псевдоэлементов, псевдоклассов и модификаторов.

Использование `&` для комбинации селекторов не допускается в именах блоков и элементов. Комбинировать можно только псевдоэлементы, псевдоклассы и модификаторы блоков и элементов.

- ▶ Примеры

HTML2-D12. Не используются расширения (`extend`).

- ▶ Текст критерия

JS1-B11. Код соответствует стилю оформления, принятому на проекте.

JS1-B3. Названия переменных, параметров, свойств и методов начинаются со строчной буквы и записываются в нотации camelCase

JS1-B13. Версии используемых зависимостей зафиксированы в `package.json`.

Дополнительные критерии

HTML1-TEST-04. Переполнение.

- ▶ Текст критерия

HTML1-IMG-01. Контентная и декоративная графика

- ▶ Текст критерия

HTML1-HTML-03. Кнопки, ссылки и интерактивные элементы

- ▶ Текст критерия

HTML2-B26. Папка `build` со всем её содержимым не должна попадать в Гитхаб.

Это значит, что папка `build` должна находиться в файле `.gitignore`.

HTML1-CONT-01. Текстовые переносы.

- ▶ Текст критерия

HTML1-HTML-08. Фавиконки.

- ▶ Текст критерия

HTML1-PROJ-02. Именованение.

- ▶ Текст критерия

HTML1-PROJ-03. Название папок и файлов.

- ▶ Текст критерия

HTML1-PROJ-04. Закомментированный код.

- ▶ Текст критерия

HTML1-PROJ-05. Неиспользуемые файлы.

- ▶ Текст критерия

HTML1-HTML-01. Ориентиры.

- ▶ Текст критерия

HTML1-HTML-02. Заголовки.

- ▶ Текст критерия

HTML1-HTML-06. Лишние элементы.

- ▶ Текст критерия

HTML1-CSS-02. Глобальный селектор

- ▶ Текст критерия

HTML1-CSS-06. Использование шрифтов

- ▶ Текст критерия

HTML1-IMG-02. Форматы графики

- ▶ Текст критерия

HTML1-ALLY-01. Описание графики

- ▶ Текст критерия

HTML1-ALLY-02. Подписи интерактивных элементов

- ▶ Текст критерия

HTML1-ALLY-03. Видимое состояние элементов

- ▶ Текст критерия

HTML1-ALLY-04. Области интерактивных элементов

- ▶ Текст критерия

HTML1-ALLY-05. Фоновый цвет у фоновых картинок

- ▶ Текст критерия

HTML1-ALLY-06. Порядок фокуса.

Порядок фокуса на интерактивных элементах **не должен** изменяться стилями.

- ▶ Примеры

HTML2-B4. В разметке отсутствует дублирование кода для одного и того же элемента, с помощью которого элемент отображается в разных местах страницы на разных версиях: мобильной, десктопной, планшетной

Этот критерий не касается элементов, которые скрываются или показываются в разных версиях.

- ▶ Пример

HTML2-B9. Нестандартные шрифты подключены локально

- ▶ Текст критерия

HTML2-B12. Использован CSS-препроцессор.

Писать CSS-код можно только с помощью CSS-препроцессора — Sass.

Стилевой код, написанный без препроцессора, является ошибкой

HTML2-B13. Код стилей должен быть разбит на несколько частей

- ▶ Текст критерия

HTML2-B14. Выполнена вёрстка трёх состояний каждой страницы: мобильной, планшетной и десктопной

- ▶ Текст критерия

HTML2-B17. Выполнена ретинизация растровой графики

- ▶ Текст критерия

HTML2-B25. В результате сборки должна получаться папка **build** со всеми необходимыми файлами

- ▶ Текст критерия

HTML2-Д21. Используются изображения в формате WebP

- ▶ Текст критерия

HTML2-Д22. Использован векторный спрайт

- ▶ Текст критерия

HTML2-Д23. Произведена оптимизация загрузки шрифтов

Шрифты предзагружаются через `link rel="preload"`.

Используется подходящее значение `font-display` в описании `@font-face`

JS1-Б2. При выполнении кода не возникает необработанных ошибок.

Дополнительные критерии «Лиги А.»

Л-1. Элементы которые в техническом задании помечены как контентная область — стилизуются по каскаду

 Стилизация: по классу или по тегу?

Л-2. `nth-child` без `n` в любом формате запрещены, вместо `nth-child1` используем `first-child`

▼ Пример

Неправильно: поля формы невозможно будет поменять местами или добавить новые.

```
.form-feedback__form { ul{ margin-left: 3px; margin-bottom: 40px; }
li { opacity: 1; margin-bottom: 49px; &:nth-of-type(2) {
display:none; } &:nth-of-type(3) { margin-right: 0; } &:last-of-
type { width: 100%; margin-right: 0; } } }
```

Неправильно: `nth-child` используется для задания размеров `img`, но если элементы поменяют местами, то у будут неправильные размеры. Если в разметку еще добавят несколько таких элементов, то у в них не будет задана нужная ширина.

```
.project-effect__item img { margin-right: 30px; @media (min-width:
$desktop-width + 1){ margin-right: calc-vw(30); } } .project-
effect__item:nth-child(1) img { @media (min-width: $desktop-width +
1) { width:calc-vw(138); } } .project-effect__item:nth-child(2) img
{ @media (min-width: $desktop-width + 1) { width:calc-vw(122); } }
.project-effect__item:nth-child(3) img { @media (min-width:
$desktop-width + 1) { width:calc-vw(86.5); } }
```

Неправильно: `nth-child` использован для стилизации карточек. Если добавить четвертую карточку, у нее не будет нужных стилей. Плюс дублирование кода.

```
&:nth-child(1){... } &:nth-child(2){ background-color: $soft-peach;
@include mobiile { padding-top: 30px; padding-right: 26px; padding-
bottom: 10px; padding-left: 26px; } } &:nth-child(3){ background-
color: $soft-peach; @include tablet { margin-bottom: 0; } @include
mobiile { padding-top: 30px; padding-right: 26px; padding-bottom:
10px; padding-left: 26px; } } h3 { min height: size(80px);
```


Неправильно: `nth-of-type` использован для позиционирования псевдоэлемента в конце элемента списка. Во-первых, количество текста в строке может измениться, а псевдоэлемент с такими стилями останется на месте. Во-вторых, может добавиться еще несколько элементов списка, а стилей для псевдоэлемента нет. В-третьих, стили заданы за каскадом от конкретного блока `.step--forth` (подразумевалось `fourth`, т.е. четвертый шаг), что не позволяет использовать их в каком-либо другом блоке.

```
.step--forth .step__item:nth-of-type(1)::before { right: 64px;
@include mobile { right: 51px; } } .step--forth .step__item:nth-of-
type(2)::before { right: 25px; @include mobile { right: 23px; } }
.step--forth .step__item:nth-of-type(3)::before { right: 3px;
@include mobile { right: 10px; } } .step--forth .step__item:nth-of-
type(4)::before { right: -2px; @include mobile { right: 7px; } }
.step--forth .step__item:nth-of-type(5)::before { right: 6px;
@include mobile {
```

Л-3. Вёрстка сделана по принципу desktop-first

▼ Подробнее

Если не указано обратного, то по умолчанию мы делаем вёрстку начиная с десктопа.

По медиавыражениям видно, что сначала был сверстан десктопный интерфейс, а от него уже делается планшетный и мобильный. Общие стили применяются к десктопным разрешениям, а когда ширина экрана