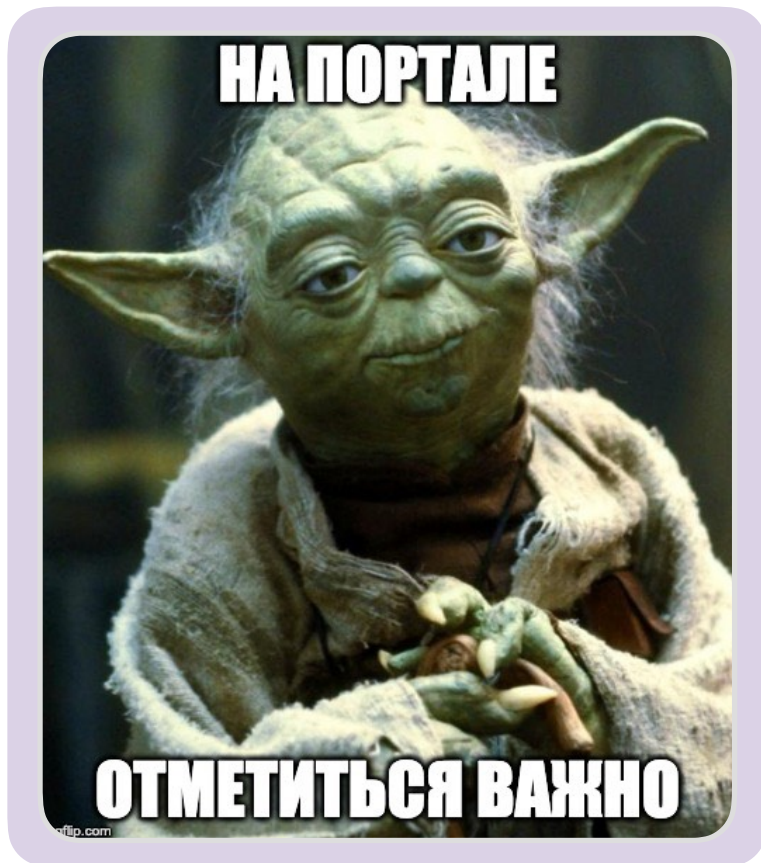


Лекция 1.

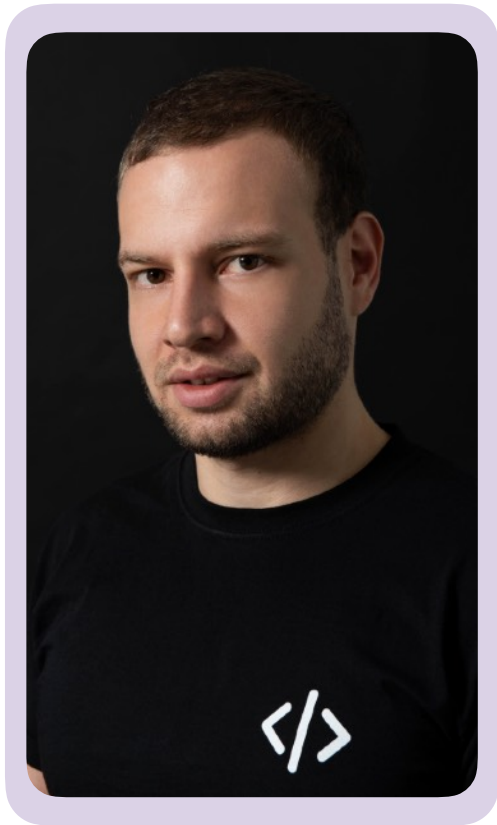
Не введение

Алексей Опалев



1. Отметиться на портале
2. Заполнить анкету
3. Оставить обратную связь

 **Класс!**



Алексей
Опалев
@a.opalev
Медиапроекты@Mail.RU

Фронтенд

Публичная часть веб-приложений, с которой непосредственно взаимодействует пользователь.

Во фронтенд входят: отображение пользовательского интерфейса, функционал, выполняющийся на стороне клиента, и обработка пользовательских запросов.

Фронтенд разработка

Работа по созданию публичной части веб-приложения, с которой непосредственно контактирует пользователь и функционала, который выполняется на стороне клиента.

nodejs

Среда для выполнения JS кода.
Нужен для работы с фреймворками/библиотеками (react, vue, express).

npm

Пакетный менеджер для nodejs модулей.

Генератор проектов

Программа/инструмент для генерации проекта.

Примеры генераторов:

`create-react-app`, `yeoman`, `slush`

Другие названия: "скаффолдер", "бутстраппер", "шаблон проекта".

Линтеры

Набор программ и инструментов для проверки кода на соответствие определенным правилам.

Основные: eslint, stylelint

CI/CD

Continious integration/continious delivery

Набор техник для реализации непрерывного цикла разработки и слияния нового кода в релизную версию продукта.

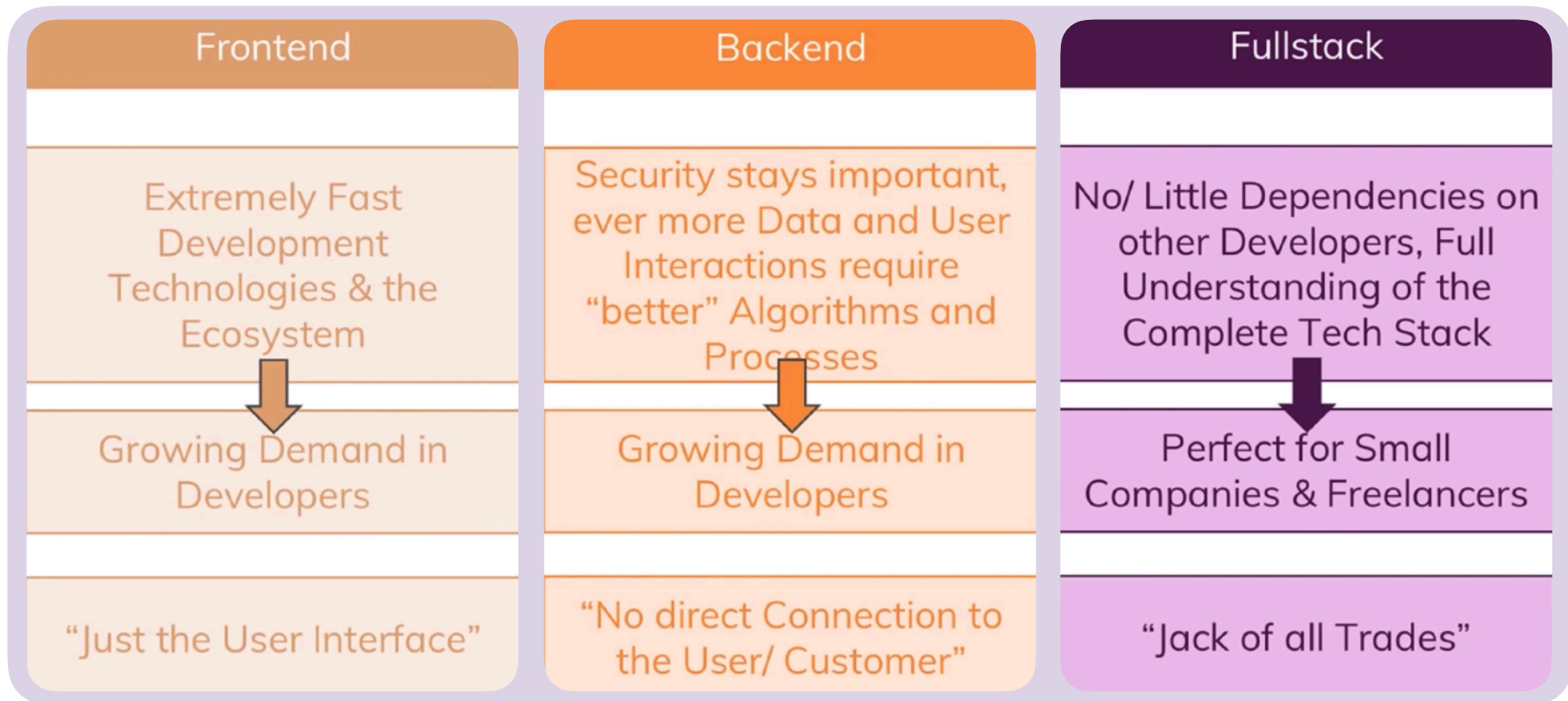
Docker

Инструмент для запуска приложений в изолированном окружении (в контейнере).

Travis

Популярный сервис для сборки проектов.
Легкая интеграция с сервисом github.

Аналоги: jenkins, gitlab-ci, circleci



- ☐ Сразу видно результат работы
- ☐ Очень быстрое развитие экосистемы
- ☐ Большое сообщество
- ☐ Низкий порог вхождения [много плохого кода :-(]



Веб-стандарты

Two Browsers Walked Into a Scrollbar



Заходят два браузера в скроллбар. Зак Лезерман показывает, как из навязчивого скроллбара сделать ненавязчивый, который не ломает дизайн сайта — <https://www.filamentgroup.com/lab/scrollbars/>



Веб-стандарты

Animating a Hue around the Color Wheel with Houdini



Анимлируем оттенки по цветовому колесу при помощи Houdini. Даниэл Уилсон об особенностях цветовой модели HSL и о том, как Houdini Properties and Values API решает проблему анимации цвета — <https://danielcwilson.com/blog/2019/09/houdini/>



Веб-стандарты

Automatically compress images on Pull Requests

Inbox Why dataviz does better for your career than TODOapps - , ~

Inbox Behind the curve ... of my bar donut chart 🤔 - , ~Swizec PS

Inbox Announcing my updated React for Data Visualization course

Inbox How pirates caused global warming - , ~Swizec PS: I'm anno

Inbox Why is JavaScript's best dataviz library so hard? - , ~Swizec

Inbox Dinosaurs, software engineers, and cholera. What do they h

Inbox The Big Mac index and Jest fetch testing - The Big Mac inde

Inbox Special live event 9am Pacific today – server-side dataviz v

Inbox Waiting for Godot with callbacks, promises, and async - , ~S

Inbox useReducer + useContext for easy global state without librar

Inbox useAuth – the simplest way to add authentication to your Re

Технологии

1. HTML
2. CSS
3. JavaScript
4. CSS пре, пост-процессоры (Sass, Stylus, PostCSS)
5. JavaScript библиотеки (lodash, underscore, ramda)
6. JavaScript frameworks (react, angular, vue, svelte)
7. Сборщики, менеджеры пакетов (npm, webpack)



С чем работать

1. UI
2. Написание JS логики и CSS стилей для UI компонентов
3. Формы и валидация пользовательского ввода
4. Общение с бэкендом
5. Реализация продвинутых UX стратегий (PWA, RTU)

Подготовка к работе в реальной команде на роль фронтенд разработчика.

- ☐ 10 занятий, 2 семинара
- ☐ ДЗ после каждого занятия
- ☐ Результат работы - pull request
- ☐ Экзамен [защита своего проекта, решений, кода]

Блоки лекций

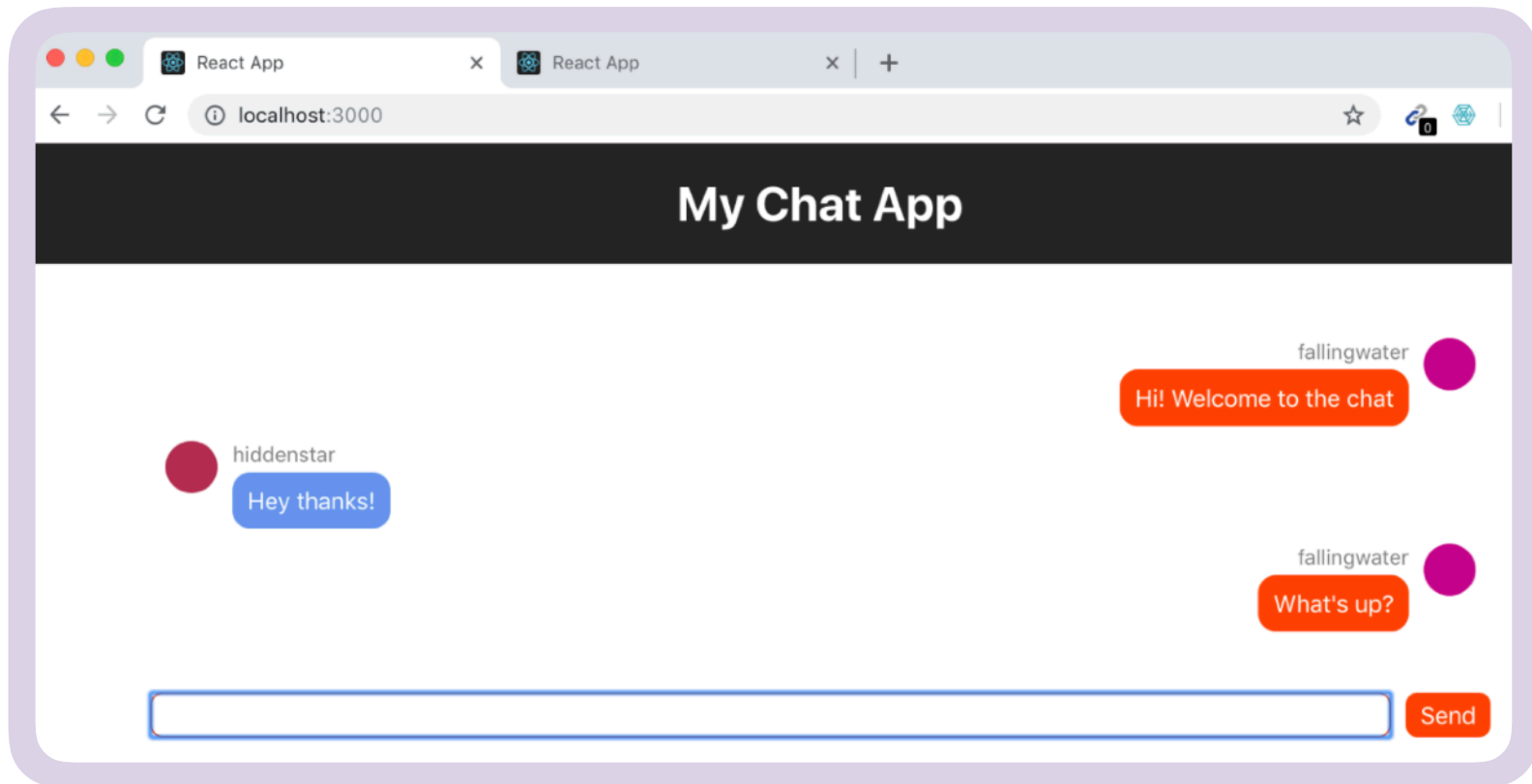
- HTML - CSS - JavaScript 
- React - Single Page Application - Forms 
- Server - Complex Interfaces - Authorization    

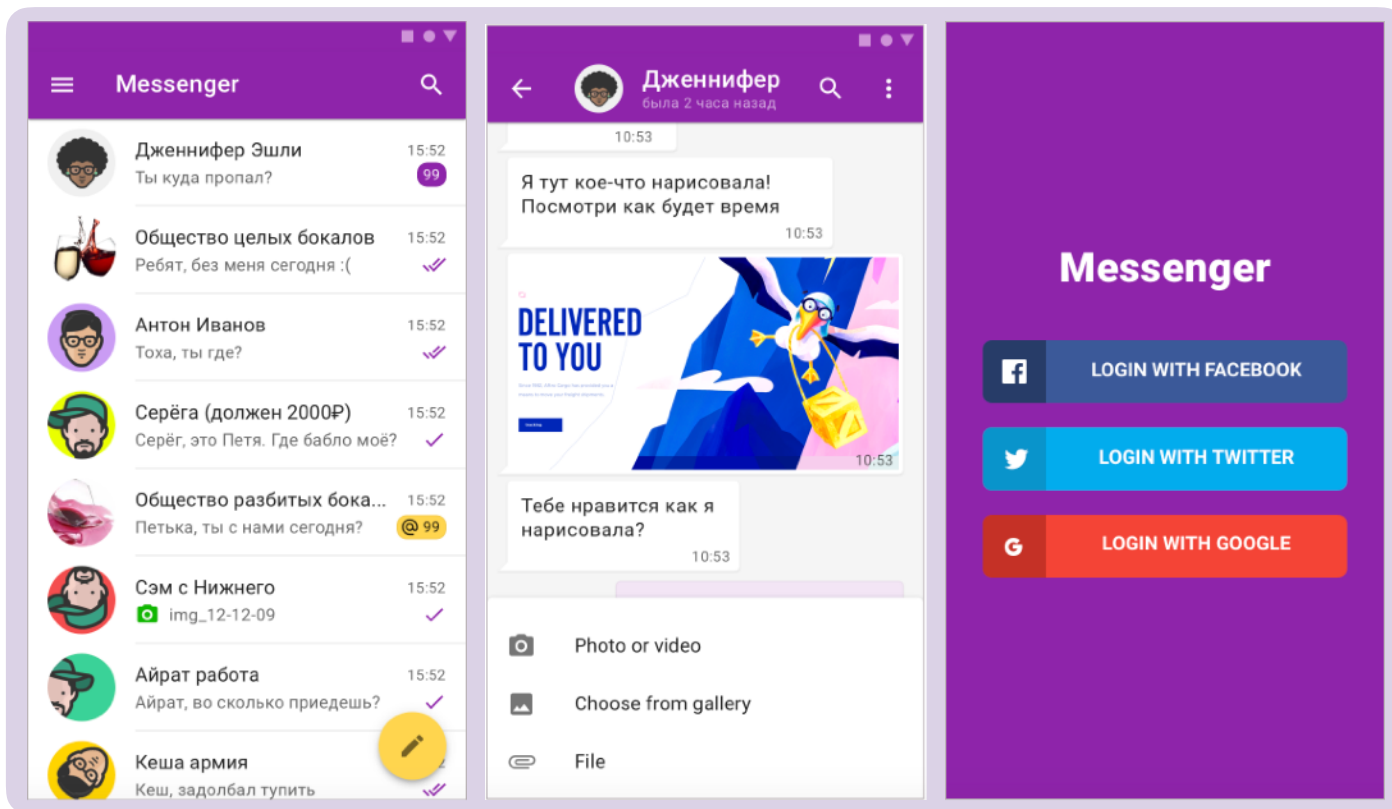
Что такое

1. devtools
2. javascript
3. async
4. transport
5. I/O
6. React/Angular/Vue
7. jQuery

Как происходит

1. Загрузка страницы
2. Отладка приложения
3. Деплой приложения
4. Коммуникация между клиентом и сервером





Оптимальный набор задач

- ☐ Много верстки
- ☐ Очень много событий, которые надо обработать
- ☐ Легко эволюционирует из простого приложения в сложное

Создаем проект

```
1.npm install -g yo  
2.npm install -g generator-track-mail  
3.mkdir project && cd $_  
4.yo track-mail
```

Создаем проект. Вариант 2

```
1.git clone git@github.com:track-mail-ru/atom-mail.git  
2.cd atom-mail  
3.rm -rf .git
```

Что происходит [mkdir]


ТЕХНОАТОМ 

a:a.opalev ~/atomnie \$ | ml | Vue | mail | awesome | Google | python | katas » | Другие закладки

Почта Картинки



Google

 Настроить

Что происходит [docker-compose]

```
dependencies yourself.
npm WARN eslint-plugin-jsx-a11y@5.1.1 requires a peer of eslint@^2.10.2 || ^3 || ^4 but none is installed. You must install peer dependencies yourself.
npm WARN stylelint-webpack-plugin@0.10.5 requires a peer of stylelint@^8.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN eslint-config-airbnb-base@11.3.2 requires a peer of eslint@^3.19.0 || ^4.5.0 but none is installed. You must install peer dependencies yourself.
npm WARN atom-mail-2019-a-opalev@0.1.0 No repository field.
npm WARN atom-mail-2019-a-opalev@0.1.0 No license field.

added 2376 packages from 921 contributors and audited 1794924 packages in 53.892s
found 0 vulnerabilities
```

New minor version of npm available! 6.10.1 → 6.11.3
Changelog: <https://github.com/npm/cli/releases/tag/v6.11.3>
Run `npm install -g npm` to update!

 Настроить

```
a:a.opalev ~/atomnie/proj $ do
```

Что происходит [open]

```
npm WARN stylelint-webpack-plugin@0.10.5 requires a peer of stylelint@^8.0.0 but none is installed. You must install peer de
pendencies yourself.
npm WARN eslint-config-airbnb-base@11.3.2 requires a peer of eslint@^3.19.0 || ^4.5.0 but none is installed. You must instal
l peer dependencies yourself.
npm WARN atom-mail-2019-a-opalev@0.1.0 No repository field.
npm WARN atom-mail-2019-a-opalev@0.1.0 No license field.

added 2376 packages from 921 contributors and audited 1794924 packages in 53.892s
found 0 vulnerabilities
```

New minor version of npm available! 6.10.1 → 6.11.3
Changelog: <https://github.com/npm/cli/releases/tag/v6.11.3>
Run `npm install -g npm` to update!

```
a:a.opalev ~/atomnie/proj $ docker-compose up -d
Creating network "proj_webnet" with the default driver
Creating proj_frontend_1 ... done
a:a.opalev ~/atomnie/proj $ |
```

 Настроить

От сложного

К сложному. Но чуть-чуть проще.

Веб приложение

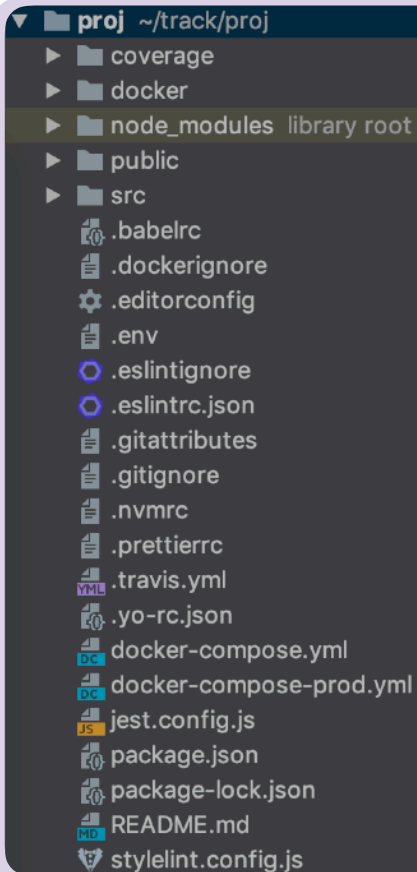
1. готовое приложение [react, redux, routes]
2. линтеры [eslint, stylelint, prettier]
3. docker
4. travis
5. git hooks
6. тесты [jest]



Atom Mail.Ru, 2019

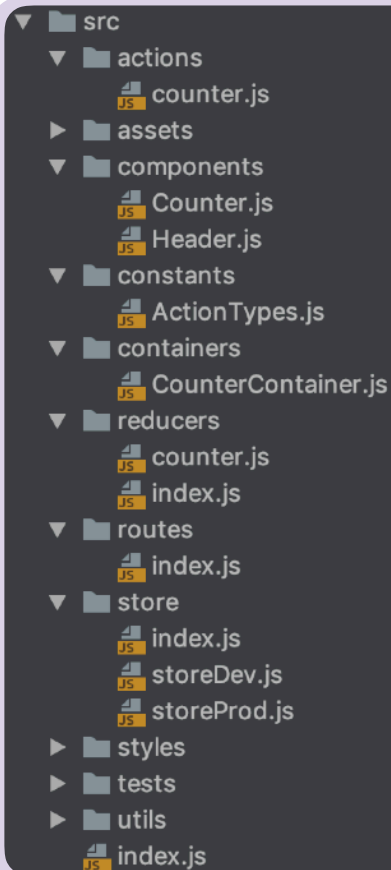
To get started, edit `src/routes/index.js` and save to reload.

Clicked: 0 times



▼ **proj** ~/track/proj

- ▶ coverage
- ▶ docker
- ▶ node_modules library root
- ▶ public
- ▶ src
 - .babelrc
 - .dockerignore
 - .editorconfig
 - .env
 - .eslintignore
 - .eslintrc.json
 - .gitattributes
 - .gitignore
 - .nvmrc
 - .prettierrc
 - .travis.yml
 - .yo-rc.json
 - docker-compose.yml
 - docker-compose-prod.yml
 - jest.config.js
 - package.json
 - package-lock.json
 - README.md
 - stylelint.config.js



▼ **src**

- ▼ actions
 - counter.js
- ▶ assets
- ▼ components
 - Counter.js
 - Header.js
- ▼ constants
 - ActionTypes.js
- ▼ containers
 - CounterContainer.js
- ▼ reducers
 - counter.js
 - index.js
- ▼ routes
 - index.js
- ▼ store
 - index.js
 - storeDev.js
 - storeProd.js
- ▶ styles
- ▶ tests
- ▶ utils
 - index.js

Со своим веб приложением

Залить в свой гитхаб. [Потребуется для выполнения дз]

Попытаться сломать приложение.

Попытаться починить приложение.

Разобраться в файлах конфигураций (АКА конфиги).

Популярная #3

Файлоориентированная.

```
api/  
  APIUtils.js  
  APIUtils.test.js  
  ProfileAPI.js  
  UserAPI.js  
components/  
  Avatar.js  
  Avatar.css  
  Feed.js  
  Feed.css  
  FeedStory.js  
  FeedStory.test.js  
  Profile.js  
  ProfileHeader.js  
  ProfileHeader.css
```

Популярная #2

Фичеориентированная.

```
common/  
  Avatar.js  
  Avatar.css  
  APIUtils.js  
  APIUtils.test.js  
feed/  
  index.js  
  Feed.js  
  Feed.css  
  FeedStory.js  
  FeedStory.test.js  
  FeedAPI.js  
profile/  
  index.js  
  Profile.js  
  ProfileHeader.js
```

Популярная #1

Атомный дизайн.

<http://bradfrost.com/blog/post/atomic-web-design>

```
└─ src/  
  └─ _settings/  
    └─ variables.css  
  └─ atoms  
    └─ button/  
      └─ index.js  
      └─ stories.js  
      └─ style.css  
    └─ *  
  └─ molecules/  
    └─ *  
  └─ organisms/  
    └─ *  
  └─ templates/  
    └─ default.css  
└─ pages/  
  └─ home/  
  └─ section/
```

Правильная

[move files around until they feel right](#) © Dan Abramov

Функции и тесты

В своем проекте выполнить `yo track-mail:functions``

Выполнить задачи в файлах:

`convertBytesToHuman.js correctSentence.js nonUniqueElements.js`

Написать тесты в соответствующих файлах `[.test.js]`

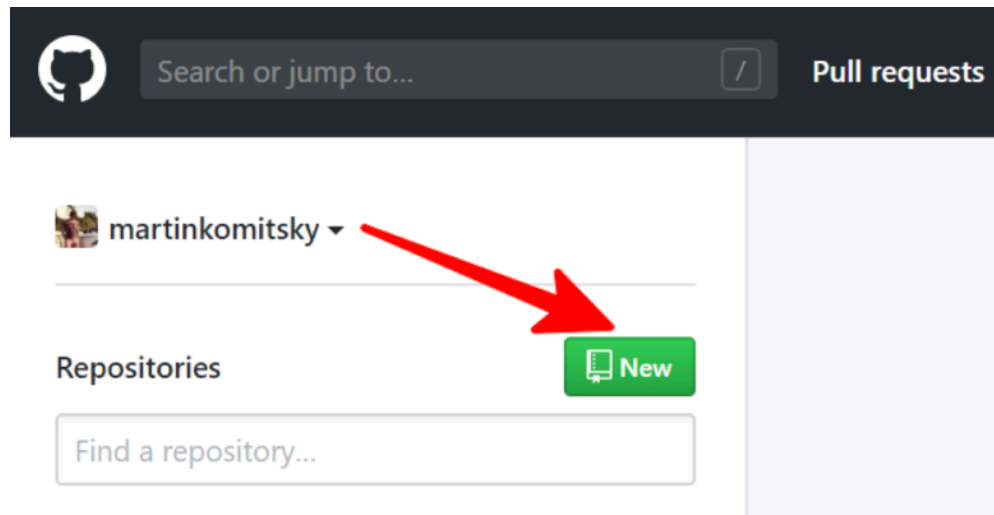
Запустить команду `docker-compose exec frontend npm run test`

Как сдавать дз

1. Выполнить локально задачи
2. Закоммитить изменения в ветке ``devel``
3. Запустить ветку в свой гитхаб
4. Поставить ПР с ``devel`` на ``master``
5. Запросить ревью ПР у меня [<https://github.com/chexex>]

Создаем репозиторий

1. Заходим на [GitHub](https://github.com)
2. Создаем новый репозиторий



Присваиваем имя репозиторию

Шаблон имени:

YYYY-HALF_YEAR-ATOM-Frontend-N-LAST_NAME

YYYY - год

HALF_YEAR - половина года. 1, если сейчас янв-июн, 2, если июл-дек

N - первая буква имени

LAST_NAME - фамилия


Пример: 2019-2-Atom-Frontend-A-Opalev

Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner

Repository name *

 martinkomitsky ▾


 /

2019-2-Track-Frontend-M-Komitsky ✓ 


Great repository names are short and memorable. Need inspiration? How about **potential-palm-tree**?

Description (optional)

Учебный проект


☒  Public

Anyone can see this repository. You choose who can commit.

☐  Private


You choose who can see and commit to this repository.


Skip this step if you're importing an existing repository.

☒ **Initialize this repository with a README** 

This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾

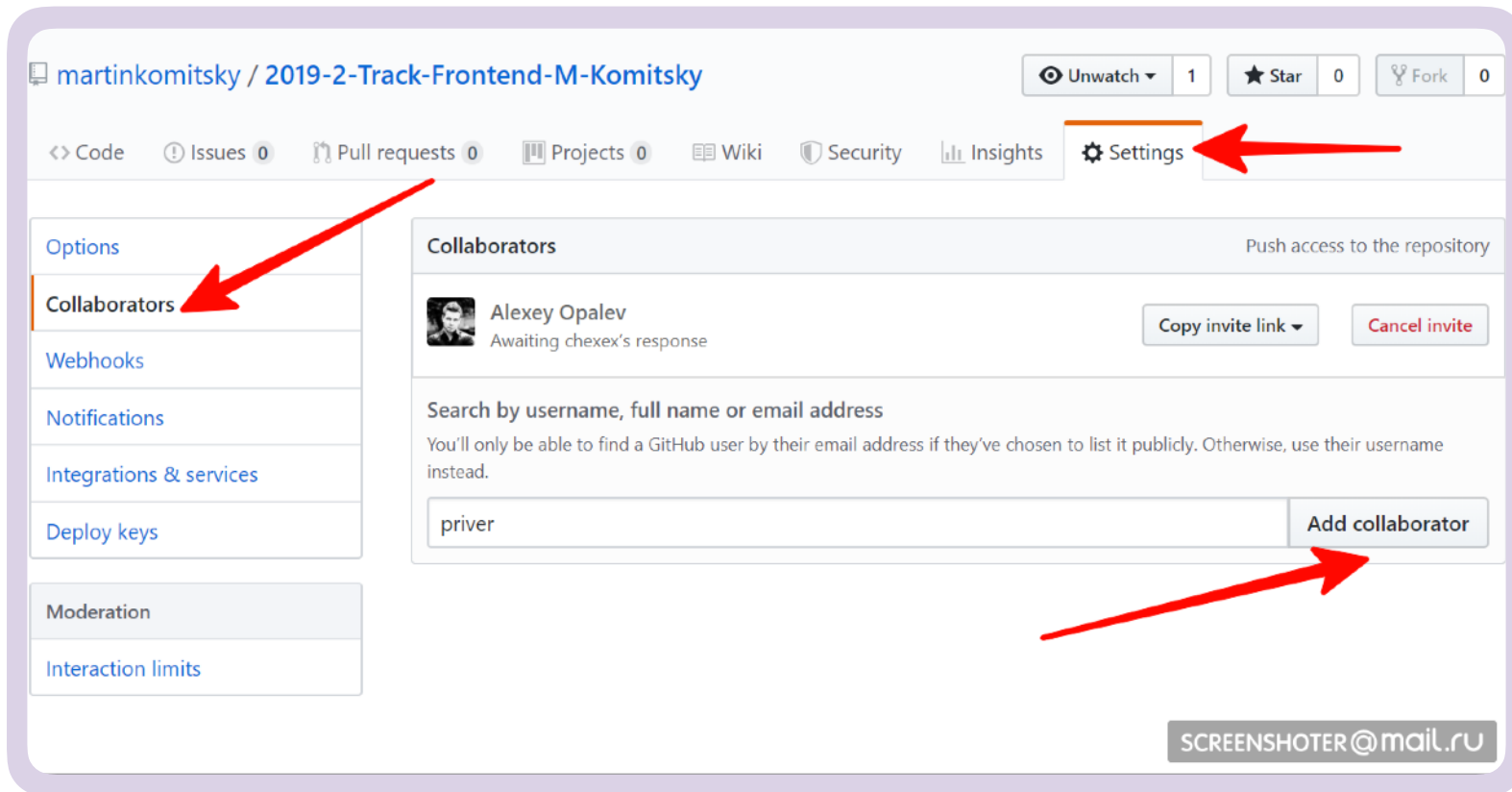
Add a license: None ▾ 

Create repository 

SCREENSHOTER@mail.ru

Добавить преподавателей

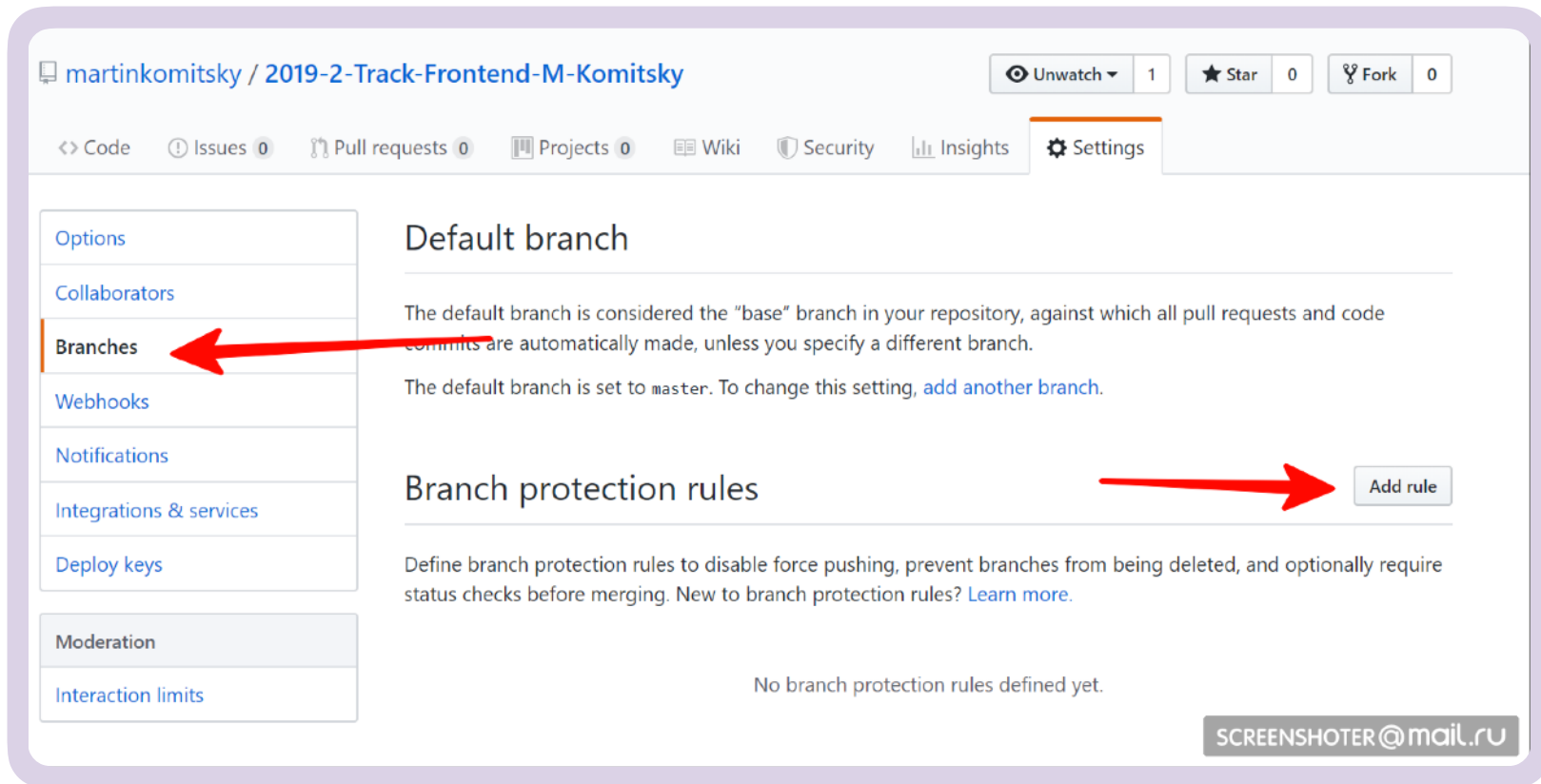
1. Заходим в настройки репозитория (`/settings/collaboration`)
2. Добавляем `martinkomitsky`, `chexex` и `priver` в `collaborators`



The screenshot shows the GitHub repository settings page for the repository `martinkomitsky / 2019-2-Track-Frontend-M-Komitsky`. The page has a top navigation bar with links for `<> Code`, `! Issues 0`, `Pull requests 0`, `Projects 0`, `Wiki`, `Security`, `Insights`, and `Settings`. The `Settings` tab is selected and highlighted with a red arrow. Below the navigation bar, the left sidebar contains a list of settings categories: `Options`, `Collaborators` (highlighted with a red arrow), `Webhooks`, `Notifications`, `Integrations & services`, `Deploy keys`, `Moderation`, and `Interaction limits`. The main content area is titled `Collaborators` and includes a sub-header `Push access to the repository`. It displays a list of collaborators, currently showing `Alexey Opalev` with the status `Awaiting chexex's response`. To the right of the collaborator's name are buttons for `Copy invite link` and `Cancel invite`. Below the list, there is a search bar with the placeholder text `Search by username, full name or email address` and a note: `You'll only be able to find a GitHub user by their email address if they've chosen to list it publicly. Otherwise, use their username instead.` The search bar contains the text `priver`. To the right of the search bar is a button labeled `Add collaborator`, which is pointed to by a red arrow. At the bottom right of the page, there is a watermark `SCREENSHOTER@mail.ru`.

Создание правила для веток

1. Заходим в настройки веток (branches)
2. Создаем новое правило



The screenshot shows the GitHub repository settings for 'martinkomitsky / 2019-2-Track-Frontend-M-Komitsky'. The 'Settings' tab is selected, and the 'Branches' sub-tab is highlighted in the left sidebar. The 'Default branch' section explains that the default branch is the 'base' branch, and the current default is 'master'. The 'Branch protection rules' section is empty, with a red arrow pointing to the 'Add rule' button. The bottom right corner contains the text 'SCREENSHOTER@mail.ru'.

martinkomitsky / 2019-2-Track-Frontend-M-Komitsky

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

Options
Collaborators
Branches
Webhooks
Notifications
Integrations & services
Deploy keys

Moderation
Interaction limits

Default branch

The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

The default branch is set to `master`. To change this setting, [add another branch](#).

Branch protection rules

Define branch protection rules to disable force pushing, prevent branches from being deleted, and optionally require status checks before merging. New to branch protection rules? [Learn more](#).

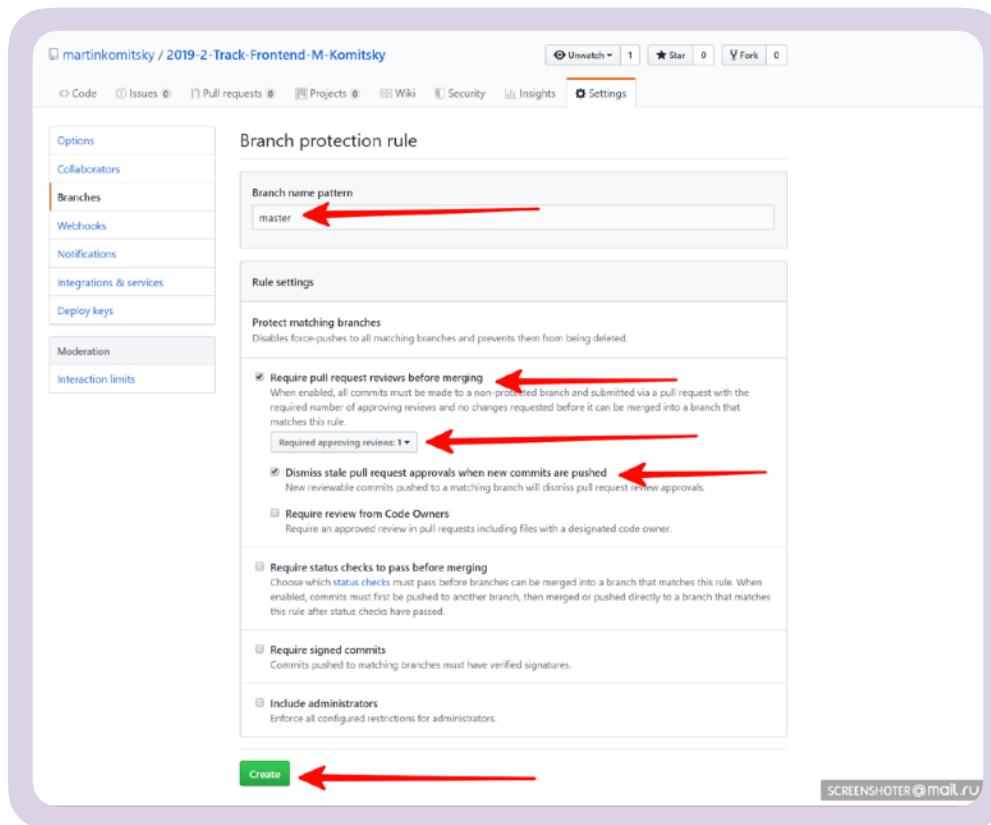
No branch protection rules defined yet.

[Add rule](#)

SCREENSHOTER@mail.ru

Настройка правила для веток

1. Защищаем ветку `master` от пуша
2. Требуем обязательный апрув от одного ревьюера для мержа ПР



The screenshot shows the GitHub interface for configuring a branch protection rule. The repository is 'martinkomitsky / 2019-2-Track-Frontend-M-Komitsky'. The 'Settings' tab is selected, and the 'Branch protection rule' section is active. The 'Branch name pattern' is set to 'master'. Under 'Rule settings', the 'Protect matching branches' section is expanded, showing several checked options: 'Require pull request reviews before merging' (with a dropdown set to '1'), 'Dismiss stale pull request approvals when new commits are pushed', 'Require review from Code Owners', 'Require status checks to pass before merging', 'Require signed commits', and 'Include administrators'. A green 'Create' button is at the bottom. Red arrows highlight the 'master' pattern, the 'Require pull request reviews before merging' option and its dropdown, the 'Dismiss stale pull request approvals...' option, and the 'Create' button.

martinkomitsky / 2019-2-Track-Frontend-M-Komitsky

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Projects Wiki Security Insights Settings

Options
Collaborators
Branches
Webhooks
Notifications
Integrations & services
Deploy keys

Moderation
Interaction limits

Branch protection rule

Branch name pattern

master

Rule settings

Protect matching branches
Disables force-pushes to all matching branches and prevents them from being deleted.

- ☒ **Require pull request reviews before merging**
When enabled, all commits must be made to a non-protected branch and submitted via a pull request with the required number of approving reviews and no changes requested before it can be merged into a branch that matches this rule.
Required approving reviews: 1
- ☒ **Dismiss stale pull request approvals when new commits are pushed**
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.
- ☐ **Require review from Code Owners**
Require an approved review in pull requests including files with a designated code owner.
- ☐ **Require status checks to pass before merging**
Choose which status checks must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.
- ☐ **Require signed commits**
Commits pushed to matching branches must have verified signatures.
- ☐ **Include administrators**
Enforce all configured restrictions for administrators.

Create

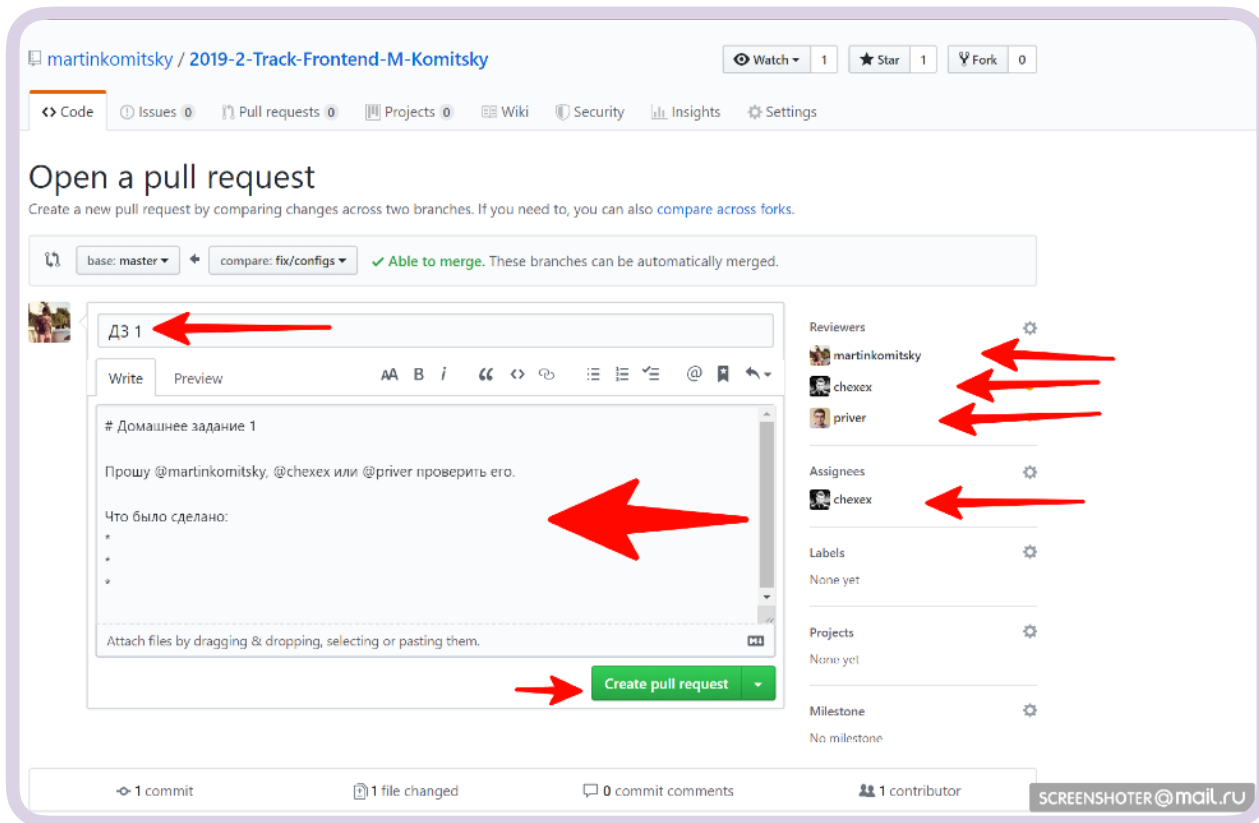
SCREENSHOTER@mail.ru

Создаем шаблон для пулл реквестов

1. Создаем файл с именем **pull_request_template.md** в корне проекта
2. Содержимое должно быть следующим:
 1. # Домашнее задание №
 2. Прошу @martinkomitsky, @schexex или @priver проверить его.
 3. Что было сделано:
 4. *
 5. *

Сдаем ДЗ на проверку

1. После выполнения домашнего задания, создаем пр в ветку `master`
2. Добавляем `martinkomitsky`, `chexex` и `priver` в поле `reviewers`
3. Добавляем того, кто выдал домашнее задание (лектора конкретной лекции) в поле `assignee`
4. В теме **обязательно** пишем номер ДЗ, в описании опционально пишем то, что сделано



martinkomitsky / 2019-2-Track-Frontend-M-Komitsky Watch 1 Star 1 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Security Insights Settings

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

base: master compare: fix/configs ✓ Able to merge. These branches can be automatically merged.

ДЗ 1

Write Preview

Домашнее задание 1

Прошу @martinkomitsky, @chexex или @priver проверить его.

Что было сделано:

- *
- *
- *

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Reviewers

- martinkomitsky
- chexex
- priver

Assignees

- chexex

Labels

None yet

Projects

None yet

Milestone

No milestone

1 commit 1 file changed 0 commit comments 1 contributor

SCREENSHOTER@mail.ru

Ожидаем проверки и вносим правки

1. Жмем на большую зеленую кнопку и ждем комментариев
2. Если все выполнено корректно - выполняется мерж
3. Если есть недочеты - будут оставлены замечания, которые надо исправить и запустить в текущий PR
4. Мы видим правки, повторяем процесс ревью
5. Пересоздавать PR не нужно

Сдача ДЗ

1. Для всех ДЗ один репозиторий
2. В одном PR сдается только **одно** ДЗ
3. Каждое ДЗ делается в отдельной ветке, чтобы избежать возможных конфликтов при мерже
4. Каждая ветка, из которой делается PR, должна быть синхронизирована с master

1. Для того, чтобы успешно сдать домашнее задание без применения штрафов (снятие 40% баллов) - нужно вовремя заблаговременно создать PR
2. Срок сдачи каждого задания - 2 недели с момента его выдачи на лекции, но не позднее 23:59:59 дня перед лекцией
3. Если вы уложились с решением ДЗ в 2 недели, то получаете к максимальным 10 баллам еще дополнительных 2 за "скорость"
4. Если вы не уложились в 2 недели, но сдали ДЗ на семинаре, то вы получаете максимально 10 баллов
5. Если вы не сдали ДЗ на семинаре текущего модуля, то применяется штраф 40% от максимальных 10 баллов, что делает максимальную оценку после штрафа равной 6
6. Максимальной оценкой оценивается решение, которое было выполнено правильно с первого раза
7. Если были допущены сильные недочеты, то баллы снижаются в зависимости от тяжести положения

//

<https://github.com/track-mail-ru/homework>

Просьба

1. Заполнить профиль реальными именами и фото на портале/в тг
2. Следить за блогом на портале
3. Задавать вопросы в чат
4. Зазубрить инструкцию по сдаче ДЗ
5. Понимать, что жесткие требования обусловлены желанием максимально быстро вогнуть в атмосферу реальной разработки, которая будет на стажировке

GH-Pages

1. Зайти на <https://travis-ci.org>
2. Подключить к своему github
3. Перейти в travis в настройки своего репозитория
4. Добавить в Environment Variables repo_token и токен
5. Зайти на <https://github.com/settings/tokens/new>
6. В строке Token description указать "repo_token"
7. В Select scopes выбрать "repo"
8. В package.json задать ключ "homepage" (см.далее)
9. В package.json в "scripts" обновить ключ "test" (см.далее)

Token description

repo_token

What's this token for?

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

- | | |
|---|--------------------------------------|
| <input checked="" type="checkbox"/> repo | Full control of private repositories |
| <input checked="" type="checkbox"/> repo:status | Access commit status |
| <input checked="" type="checkbox"/> repo_deployment | Access deployment status |
| <input checked="" type="checkbox"/> public_repo | Access public repositories |
| <input checked="" type="checkbox"/> repo:invite | Access repository invitations |

```
1. // package.json
2. {
3.   "name": "cra-caddy-docker",
4.   "homepage": "https://chexex.github.io/cra-caddy-docker/",
5.   ...
6.   "scripts": {
7.     ...
8.     "test": "CI=true react-scripts test --env=jsdom",
9.     "test:coverage": "npm test -- --coverage",
10.    ...
11.  },
12.  ...
13.}
```

Полезные ссылки

Читать:

[YDKJS: Getting Started](#)

Смотреть:

[The Wierd History Of JavaScript](#)
[JavaScript: How It's Made](#)

**Спасибо
за внимание!**

Алексей Опалев

a.opalev@corp.mail.ru