

Pullet

(так себе название, конечно)

Руководство пользователя

Pullet

Это просто! Вам больше не придется принимать письма от студентов по почте! Пользуйтесь всеми возможностями системы контроля версий *GitHub* и сервисом *Pullet* для приема и учета лабораторных работ, а также построения статистики успеваемости студентов.

Эта короткая инструкция поможет Вам в использовании *Pullet*.

Оглавление:

1.	Краткое описание	3
а.	Основные возможности.....	3
2.	Условия использования и системные требования.....	4
а.	Общие указания.....	4
б.	Механизм взаимодействия	4
с.	Создание собственного репозитория	4
3.	Типовые действия	5
а.	Главная страница.....	5
б.	Панель управления.....	5

1. Краткое описание

Pullet – веб-сервис, призванный облегчить работу преподавателей за счет частичной автоматизации процесса приема и учета лабораторных работ, сдаваемых очно или удаленно. Используя крупнейший веб-сервис хостинга проектов и исходного кода *GitHub* и механизм *Pull Requests*, ***Pullet*** упрощает процесс резюмирования студенческой отчетности и программного кода, контроля за сроком сдачи работ и ведения статистики успеваемости.

а. Основные возможности

- Построение временной линии студенческих работ с указанием дат подачи и принятия;
- Вывод подробной информации о содержании работы;
- Подтверждение приема работы от преподавателя;
- Группировка работ по авторам;
- Добавление сроков сдачи работы, определяемых кодовым словом;
- Вывод сведений об успеваемости на основе информации о сданных работах;
- Экспорт полученных данных.

2. Условия использования и системные требования

а. Общие указания

Для корректной работы с *Pullet* необходимо иметь аккаунт на сервисе *GitHub*. Как только пользователь сервиса проходит авторизацию через сайт github.com, он получает доступ ко всем возможностям сервиса *Pullet*.

б. Механизм взаимодействия

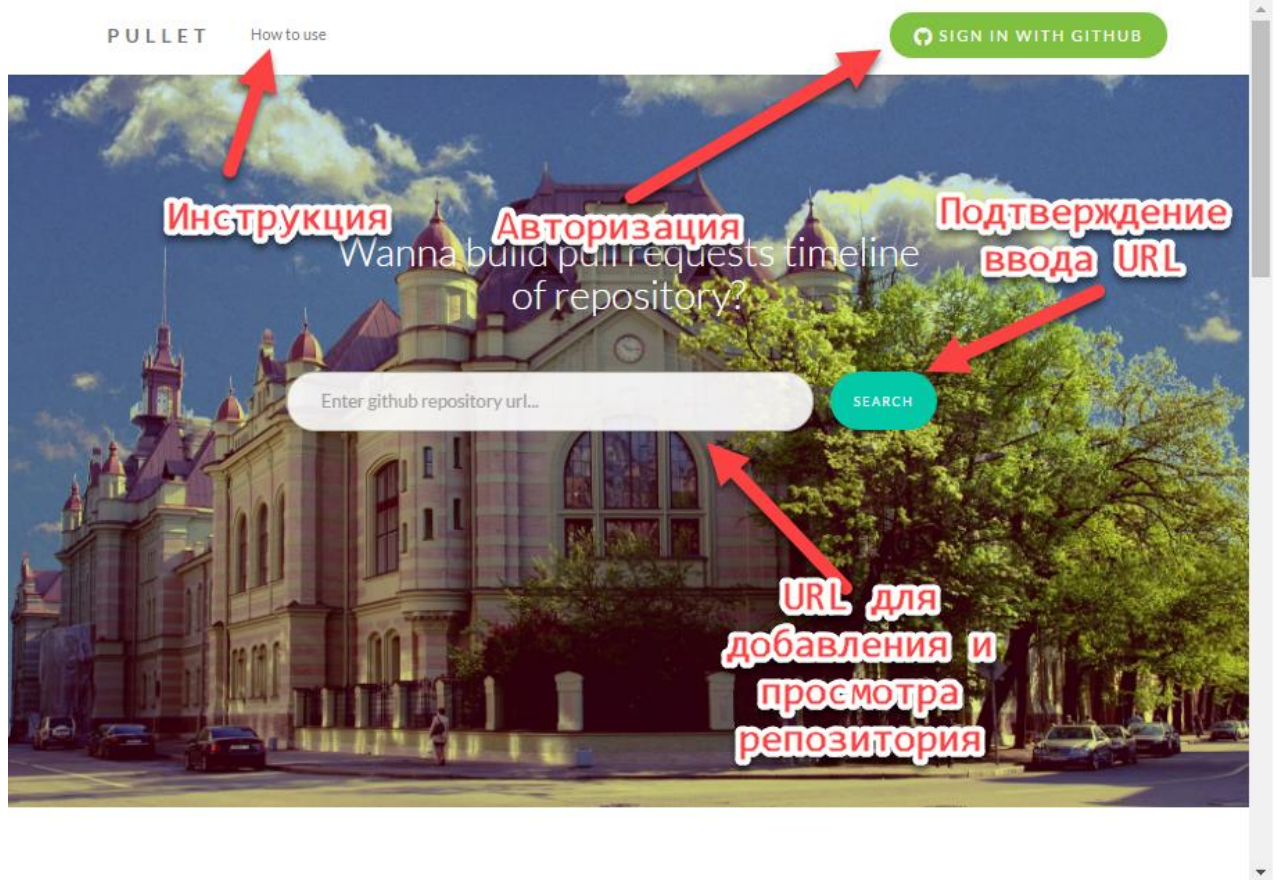
Взаимодействие между преподавателем и студентом происходит в [GitHub-репозитории](#) с использованием механизма [Pull Requests](#), позволяющего использовать все преимущества системы контроля версий и командного режима разработки.

с. Создание собственного репозитория

Преподавателю необходимо создать новый репозиторий, в корневой каталог которого студенты будут добавлять свои проекты/лабораторные работы с помощью *Pull Requests* (далее PR). Каждая работа при этом должна являться отдельным PR. После создания репозитория, преподаватель получит возможность добавлять с помощью *Pullet* «дэдлайны» - сроки сдачи каждой конкретной работы. Соответствие срока сдачи работе проверяется при помощи «matching phrase» - кодового слова, которое должно быть указано в названии PR, поданного студентом.

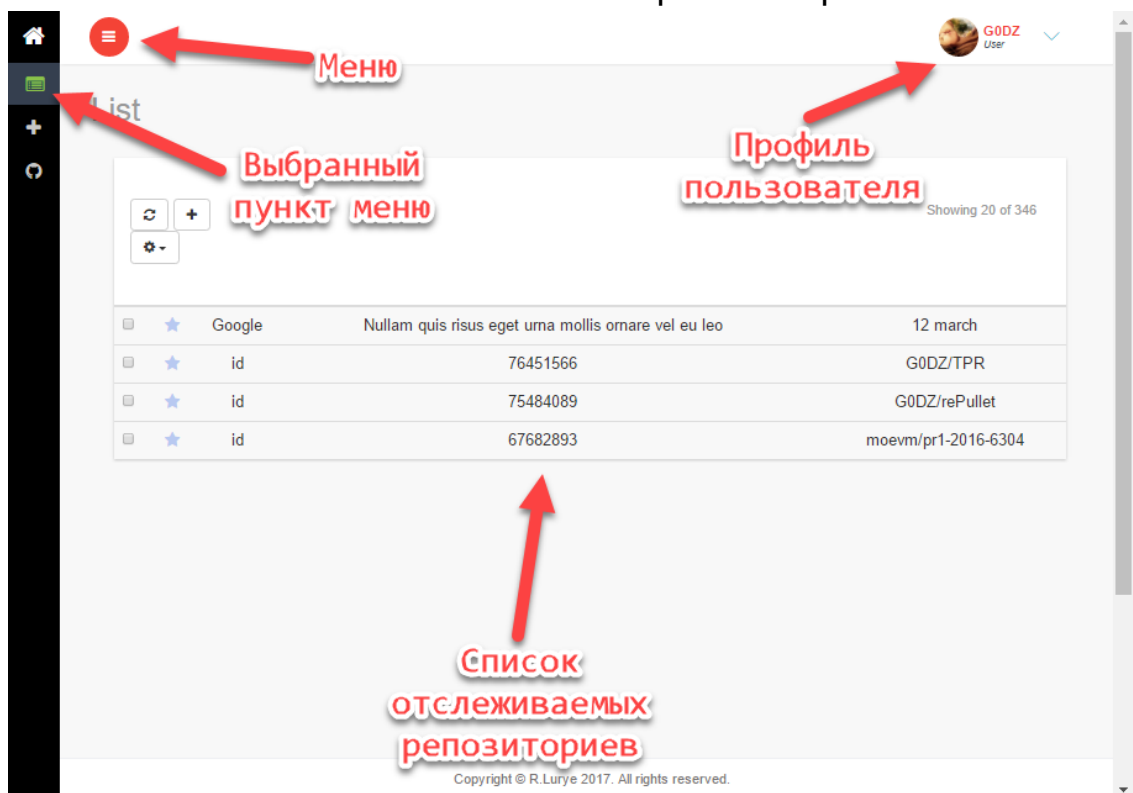
3. Типовые действия

а. Главная страница



б. Панель управления

- Список отслеживаемых репозиториях



• Добавление нового репозитория

URL GitHub репозитория

Выбранный пункт меню

Подтверждение ввода URL

Repo url:

Submit

GitHub repository url

Copyright © R.Lurye 2017. All rights reserved.

• Просмотр репозитория

GitHub аккаунт студента

Кодовое слово

Период сдачи работы (отмечен серым)

Подробная информация о Pull Request

Временная шкала

Pull Request лабораторной работы

Рейтинг успеваемости студентов

Сроки сдачи работ (видны только создателю репозитория)

Repository

PR Name

0apm Lab2

Lab4

2016 Dec Jan 2017 Feb Mar

Pull Request info

Property	Value
Title	Lab4
Author	0apm
Created at	2016-12-17 13:50
Status	open
Rework	3
Report	1

Pull Requests Rating

#	GitHub login	GitHub name	GitHub link	Opened PR	Closed PR			Rework
					In time	Delay	All	
1	0apm	0apm	https://github.com/0apm	3	1	1	2	4

Deadline Periods

Add New Datarange

Lab2 | 24-11-20 to 22-03-20 X

Save & Update Page

Copyright © R.Lurye 2017. All rights reserved.