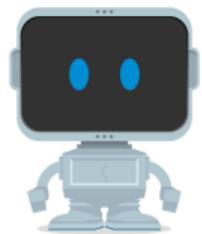


Про то, как
эффективно работать
в распределенной команде,
но может и нет

Стас Рудаков
stas@garage22.net

Minsk Python Meetup
December 2017



DataRobot

Беспочвенное утверждение 1

Беспочвенное утверждение 1

Технологии и рынок труда меняются так быстро, что любой переход на позицию, не требующую ежедневной работы с кодом, ухудшает карьерные перспективы для разработчика в Беларуси.

Беспочвенное утверждение 1

Технологии и рынок труда меняются так быстро, что любой переход на позицию, не требующую ежедневной работы с кодом, ухудшает карьерные перспективы для разработчика в Беларуси.

Но это не значит, что вы не должны быть способны перейти на такую позицию.

Основные вызовы в распределенных командах

Для индивида

Основные вызовы в распределенных командах

Для индивида

- ▶ Самомотивация
- ▶ Нехватка общения
- ▶ Стирается граница личное-рабочее
- ▶ Мультипликация проблем

Основные вызовы в распределенных командах

Для индивида

- ▶ Самомотивация
- ▶ Нехватка общения
- ▶ Стирается граница личное-рабочее
- ▶ Мультипликация проблем

Для команды

Основные вызовы в распределенных командах

Для индивида

- ▶ Самомотивация
- ▶ Нехватка общения
- ▶ Стирается граница личное-рабочее
- ▶ Мультипликация проблем

Для команды

- ▶ Эффективная коммуникация
- ▶ Культурные различия
- ▶ Часовые пояса
- ▶ Эффективное стимулирование членов команды
- ▶ Установление доверительных отношений между членами команды

Технологический сдвиг

- ▶ Collaborative Documents: G Suite, Office365
- ▶ Github / Gitlab / Bitbucket
- ▶ Чаты: Slack, HipChat
- ▶ Видео-конференции: Hangouts, Zoom, Slack
- ▶ Больше облаков: Jira, Confluence, Trello, Яндекс.Коннект
- ▶ CI: Travis, Codeship, Circle
- ▶ Инфраструктура: AWS, Google Cloud, Digital Ocean
- ▶ Real time collaboration: Teletype for Atom, ...

Беспочвенное утверждение 2

Беспочвенное утверждение 2

Разработчику нет смысла не становиться более эффективным.

Самоуправление

- ▶ Планируйте свое время
- ▶ Ставьте себе цели
- ▶ Делайте отчеты о своей работе
- ▶ Будьте прозрачны
- ▶ Отвечайте за свою карьеру

Беспочвенное утверждение 3

Беспочвенное утверждение 3

Нет смысла работать без цели добиться успеха.

Коммуникация I

- ▶ Коммуникация асинхронна по умолчанию
- ▶ Нужны четкие точки входа
 - ▶ Куда писать email, чтобы его прочитала вся команда?
 - ▶ Куда пойти, чтобы пообщаться с командой в чате?
 - ▶ Как понять, какая команда отвечает за ту или иную подсистему?
 - ▶ Как просить PR review у команды, которая отвечает за эту область кода?
- ▶ Нужно прояснить, как эскалировать вопрос, если методы выше не работают
- ▶ Нужно прояснить временные зоны, в которые есть смысл ожидать ответа

Коммуникация II

- ▶ Все члены команды следят за эффективной коммуникацией
 - ▶ Если есть возможность/знания, дают ответы на запросы
 - ▶ Если нет возможности/знаний, пытаются привлечь нужных людей
- ▶ Худшее при коммуникации — молчание на другой стороне
- ▶ Если есть проблемы с поддержанием коммуникации, нужно определять ключевые индикаторы и цели

Коммуникация III: Документ vs Дискуссия

- ▶ Дискуссия хорошо подходит, если идея не сформирована
- ▶ Документы (спецификации, предложения) хорошо подходят, если идея уже сформирована, но вам нужно услышать чужое мнение
 - ▶ Написать документ дешевле, чем написать код
 - ▶ Написанный документ можно переиспользовать
 - ▶ Асинхронная коммуникация
- ▶ Даже если вы не хотите слышать ничье мнение, все равно опишите свою идею

Коммуникация IV: Code Review

Коммуникация IV: Code Review

Цели

- ▶ Общее владение кодом
- ▶ Коммуникация об изменениях
- ▶ Поддержание ожидаемого уровня качества кода
- ▶ Обучение коллег, выработка единого стиля
- ▶ Быстрые итерации

Не-цели

- ▶ Повышение самооценки
- ▶ Уберегание драгоценного репозитория от плохих изменений



Code Review: Opinion vs Facts

```
1 def do_something(strange , really):
```

Листинг 1: D

```
1 def do_something(  
2     strange , really  
3 ):
```

Листинг 2: D

```
1 def do_something(strange ,  
2                 really):
```

Листинг 3: D

```
1 def do_something(  
2     strange , really):
```

Листинг 4: D

Code Review: Ad-Hoc Requests

“Кстати, раз уж ты правишь код, не мог бы ты исправить еще ... и ...”

Code Review: Ad-Hoc Requests

“Кстати, раз уж ты правишь код, не мог бы ты исправить еще ... и ...”

“Нет!”

Code Review: Ad-Hoc Requests

“Кстати, раз уж ты правишь код, не мог бы ты исправить еще ... и ...”

“Нет!”

Просто создавайте тикеты.

Code Review: вместо тысячи слов...

“Мне кажется, эту функцию можно переписать так, чтобы внутренний цикл . . . , а потом использовать паттерн . . .”

Code Review: вместо тысячи слов...

“Мне кажется, эту функцию можно переписать так, чтобы внутренний цикл ..., а потом использовать паттерн ...”

```
1     for i in range(0, len(items), batch_size):  
2         ...  
3         items[i:i+batch_size]
```

Brian Anderson: советы

`http://brson.github.io/2017/04/05/minimally-nice-maintainer`

- ▶ Отвечайте быстро
- ▶ Говорите “спасибо”
- ▶ Хвалите
- ▶ Говорите “Да”
- ▶ Устанавливайте четкие ожидания
- ▶ Признавайте ошибки
- ▶ Будьте эмоциональны



nott approved these changes 2 days ago • edited



[View changes](#)

This is def a good step forward ❤️

My only concern is `singular/plural` - feel free to address it in a follow-up PR if you keep XXX-1234 open.

Стас Рудаков

<mailto:stas@garage22.net>

<http://staaas.net/talks.html>