



## МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«МИРЭА – Российский технологический университет»

**РТУ МИРЭА**

---

---

**Институт информационных технологий (ИИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

### **ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА УЧЕБНУЮ ПРАКТИКУ**

#### **Ознакомительная практика**

**Студенту 1 курса учебной группы ИКБО-33-21**

**Дмитриеву Павлу Вячеславовичу**

**Место и время практики:** РТУ МИРЭА кафедра ИиППО, с 09 февраля 2022 г.  
по 31 мая 2022 г.

**Должность на практике:** студент

#### **1. СОДЕРЖАНИЕ ПРАКТИКИ:**

- 1.1. Изучить: как инструменты командной разработки Git \ Trello и т.п. инструменты помогают в разработке и развитии проектов
- 1.2. Практически выполнить: найти особенности инструментов командной разработки Git \ Trello
- 1.3. Ознакомиться: с влиянием инструментов командной разработки Git \ Trello и т.п. инструментов на достижение целей проекта

**2. ДОПОЛНИТЕЛЬНОЕ ЗАДАНИЕ:** подготовить доклад на научно-техническую конференцию студентов и аспирантов РТУ МИРЭА, подготовить презентационный материал

**3. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ:** В процессе практики рекомендуется использовать периодические издания и отраслевую литературу годом издания не старше 5 лет от даты начала прохождения практики

Руководитель практики от кафедры  
«09» февраля 2022 г.

Задание получил  
«09» февраля 2022 г.

Подпись

(Степанов П. В.)


Подпись

(Дмитриев П. В.)

**СОГЛАСОВАНО:**

Заведующий кафедрой:

«09» февраля 2022 г.


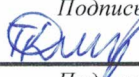
  
\_\_\_\_\_  
Подпись (Болбаков Р. Г.)

**Проведенные инструктажи:**

Охрана труда:

Инструктирующий

Инструктируемый

  
\_\_\_\_\_  
Подпись  
  
\_\_\_\_\_  
Подпись

«09» февраля 2022 г.

Болбаков Р. Г., зав. каф.


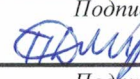
ИиППО

Дмитриев П. В.

Техника безопасности:

Инструктирующий

Инструктируемый

  
\_\_\_\_\_  
Подпись  
  
\_\_\_\_\_  
Подпись

«09» февраля 2022 г.

Болбаков Р. Г., зав. каф.


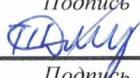
ИиППО

Дмитриев П. В.

Пожарная безопасность:

Инструктирующий

Инструктируемый

  
\_\_\_\_\_  
Подпись  
  
\_\_\_\_\_  
Подпись

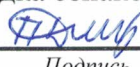
«09» февраля 2022 г.

Болбаков Р. Г., зав. каф.

ИиППО

Дмитриев П. В.

С правилами внутреннего распорядка ознакомлен:

  
\_\_\_\_\_  
Подпись

«09» февраля 2022 г.



## МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

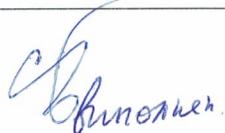
«МИРЭА – Российский технологический университет»

РТУ МИРЭА



### РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ ОЗНАКОМИТЕЛЬНОЙ ПРАКТИКИ

студента Дмитриева П. В. 1 курса группы ИКБО-33-21 очной формы обучения, обучающегося  
по направлению подготовки 09.03.04 Программная инженерия

Неделя	Сроки выполнения	Этап	Отметка о выполнении
1	09.02.2022	Подготовительный этап, включающий в себя организационное собрание (Вводная лекция о порядке организации и прохождения ознакомительной практики, инструктаж по технике безопасности, получение задания на практику)	Выполнен в срок
2	16.02.2022	Подготовительный этап (Участие в круглом столе на тему «Проблема достоверности информации в современном мире»)	Выполнен в срок
3	24.02.2022	Подготовительный этап (Участие в круглом столе на тему «Информационно-коммуникационные технологии для организации информационного процесса»)	Выполнен в срок
4	02.03.2022	Подготовительный этап (Участие в круглом столе на тему «Информационная и библиографическая культура»)	Выполнен в срок
5	09.03.2022	Исследовательский этап (Поиск, отбор и анализ материалов для выполнения задания по практике)	Выполнен в срок
8	30.03.2022	Представление руководителю (в составе бригады) структурированного материала: аналитический обзор предметной области	Выполнен в срок
10	29.03.2022	Технологический этап (анализ основных методологий разработки программных продуктов)	Выполнен в срок
14	11.05.2022	Представление руководителю (в составе бригады) результатов выполненной работы по основным подходам к разработке программных продуктов	Выполнен в срок
14	11.05.2022	Согласование с руководителем доклада на научно-техническую конференцию студентов и аспирантов РТУ МИРЭА	не выполнен

16	31.05.2022	Подготовка окончательной версии отчета (Оформление материалов отчета в полном соответствии с требованиями на оформление письменных учебных работ студентов)	
----	------------	--	---


Руководитель практики от  
кафедры  
Обучающийся

/Степанов П. В., ассистент  
каф. ИиППО/  
/Дмитриев П. В./

**Согласовано:**

Заведующий кафедрой



/Болбаков Р.Г., к.т.н., доцент/

## СОДЕРЖАНИЕ

1 ВВЕДЕНИЕ .....	1
2 ОСОБЕННОСТИ ПОПУЛЯРНЫХ ИНСТРУМЕНТОВ КОМАНДНОЙ РАЗРАБОТКИ .....	2
1) GitHub .....	2
2) BitBucket.....	3
3) GitLab.....	4
4) Gogs .....	4
5) GitFlic.....	5
6) Gitea .....	6
7) Gitee .....	6
8) ProjectLocker.....	6
9) RhodeCode .....	7
3 ВЛИЯНИЕ ИНСТРУМЕНТОВ КОМАНДНОЙ РАЗРАБОТКИ НА ДОСТИЖЕНИЕ ЦЕЛЕЙ ПРОЕКТА .....	8
4 ЗАКЛЮЧЕНИЕ .....	9
5 СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ .....	10

# 1 ВВЕДЕНИЕ

Исследуемая тема - инструменты командной разработки Git \ Trello и т.п. инструменты.

Существует множество как платных, так и бесплатных инструментов командной разработки, имеющие одно и то же применение, но разные функции и преимущества. Различные сервисы для постановки задач, хранилища исходного кода, системы управления проектами, облачные хранилища данных относятся к разным категориям инструментов, но все они используются в командной разработке проекта. Среди этих решений можно выделить основной инструмент, без которого работу над проектом нельзя будет назвать командной – хранилище исходного кода. Они предназначены для параллельной работы над одним проектом. Включая в себя системы ветвления проекта, инструмент позволяет разрабатывать решения одновременно нескольким людям в одном проекте. Большинство этих веб-репозиторий построено на системе контроля версий Git.

Git – это система контроля версий, отслеживающая изменения в коде, для облегчения работы программистов и оптимизации процесса разработки. Git был разработан в 2005 году Линусом Торвальдсом, создателем Linux, чтобы другие разработчики могли вносить свой вклад в ядро Linux. Git популярен благодаря скорости работы, оптимальному дизайну и поддержке нелинейной разработки. В Git есть полезная функция, называемая commit'ом. Эта функция загружает отредактированные файлы на сервер. Остальные разработчики могут посмотреть на сервере всю историю коммитов кода.



## 2 ОСОБЕННОСТИ ПОПУЛЯРНЫХ ИНСТРУМЕНТОВ КОМАНДНОЙ РАЗРАБОТКИ

### 1) GitHub

GitHub – это крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Он построен на Git'е и предлагает контроль версий, а также имеет собственный ряд удобных функций для командной разработки. Основными из них являются отслеживание ошибок в коде, клонирование репозитория, управление доступом, статистика, вики. Основные услуги GitHub предоставляет бесплатно, однако есть и расширенные тарифные планы с расширенными пакетами опций, пользующиеся спросом у крупных и масштабных проектов. В основном бесплатные аккаунты используются для размещения проектов с открытым исходным кодом. GitHub позволяет создавать неограниченное количество частных репозиториях с неограниченным количеством соавторов даже для бесплатных планов.

GitHub создан для облегчения отслеживания контроля версий и командной работы над проектом. С помощью pull запросов пользователь может предложить свои изменения в исходном коде проекта. Каждый разработчик имеет доступ ко всем обновлениям и изменениям в коде. Эти поправки называются коммитами (commit).

GitHub имеет множество функций для комфортной работы в команде таких, как:

- Визуализация текстовых файлов, использующихся для описания проекта, инструкции и документации
- График активности создания коммитов
- Встроенная программа просмотра pdf-файлов
- Рассылка по электронной почте
- Графики об участниках, частоте обновления кода
- Функционал GitHub'а позволяет пользоваться им не только как инструментом для создания проектов, но и как облачным

хранилищем для различных файлов, изображений, таблиц и кода.

## 2) BitBucket

BitBucket – это веб-сервис для хостинга проектов и их совместной разработки, основанный на системе контроля версий Git. По назначению и основным предлагаемым функциям аналогичен GitHub, от которого отличается определёнными преимуществами в плане размещения не публичных репозиториев — возможностью их бесплатного хостинга с ограничением на размер команды не более пяти человек и меньшей арендной платой при большем размере команды, а также управление правами доступа на уровне отдельных ветвей проекта.

К плюсам можно отнести:

- BitBucket абсолютно бесплатен для пяти пользователей. Это включает в себя неограниченное количество частных репозиториев
- Цены за подписку намного выгоднее чем на GitHub

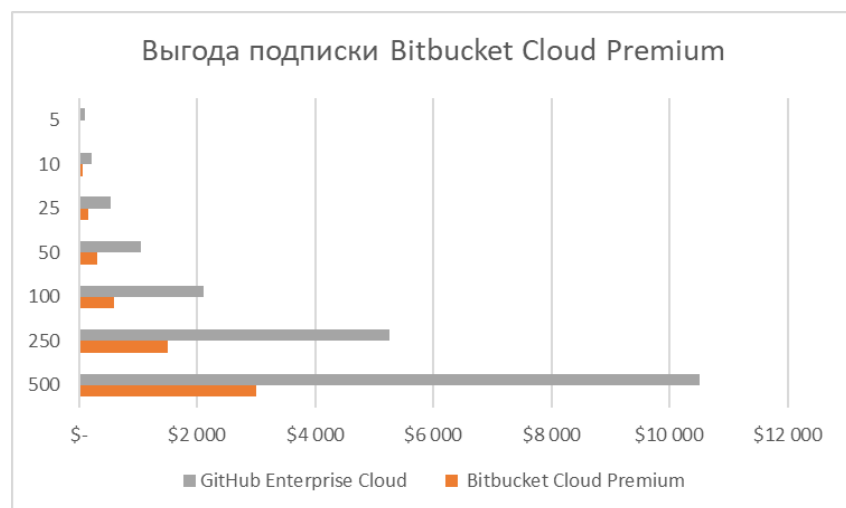


Рисунок 1 – Диаграмма сравнения подписок GitHub и BitBucket

- Поиск BitBucket сканирует ваш синтаксис, чтобы найти определения, которые соответствуют вашему запросу, а не просто имена переменных
- Одним из чрезвычайно полезных аспектов BitBucket является то, что каждый репозиторий может иметь свою собственную вики, доступную по щелчку переключателя. Эта функциональность означает,



что вы можете включить всю необходимую информацию и заметки в платформу, держа их под рукой в течение всего цикла разработки

- BitBucket имеет мощную интеграцию Trello

### 3) GitLab

GitLab - еще одна популярная платформа для хостинга исходного кода, которая предлагает множество функций, аналогичных Github, поскольку оба они контролируют версии размещенных проектов с использованием Git. GitLab позволяет управлять правами доступа и пользователями, автоматизировать процессы. Изначально разработчики создали GitLab для управления исходным кодом для совместной работы в команде. Позже его доработали, и GitLab стал полноценным решением, охватывающим весь жизненный цикл DevOps. GitLab предоставляет инструменты интеграции и непрерывной доставки или CI / CD, а также метрики для отслеживания качества кода, производительности и юзабилити-тестирования. У него также есть платное самостоятельное приложение.

GitLab имеет широкий функционал, в который входят:

- Планирование (процесс разработки становится более наглядным, повышается эффективность коллективной работы)
- Тестирование (настраиваемые инструменты проверки кода на ошибки и оценки качества работают в автоматическом режиме и помогают быстрее обнаружить проблему)
- Мониторинг (сбор статистики о проекте и анализ производительности после каждого изменения в коде)
- Контроль времени (отчеты по затраченному времени на выполнение задач)

### 4) Gogs

Gogs - это кроссплатформенный, автономный сервис с открытым исходным кодом. Он служит инструментом контроля версий, который позволяет разработчикам создавать отдельные двоичные дистрибутивы для разных платформ. Веб-инфраструктура Gogs основана на Maseron, который

чрезвычайно быстр. Кроме того, он включает в себя возможность аутентификации из внешних источников, так как он может быть интегрирован с серверами LDAP, PAM, SMTP и даже GitHub.

В числе важнейших характеристик Gogs также следует назвать:

- поддержку протоколов HTTP(s) и SSH;
- поддержку SMTP и LDAP;
- возможность создания как приватных, так и публичных репозиторий;
- интеграцию с социальными сетями (пока что поддерживаются GitHub, Google+, а также китайские сервисы QQ и Weibo);
- возможность работы в связке с сервисами непрерывной интеграции и DevOps-сервисами

## 5) GitFlic

GitFlic – это первый российский сервис для хранения исходного кода и работы с ним. Основан на системе контроля версий Git. Разработчики планировали создать инструмент командной разработки наподобие GitHub. Сервис пока что нельзя назвать аналогом иных инструментов, поскольку он появился меньше года назад и имеет сильно урезанный функционал.

Из предложенных опций можно отобразить:

- Запросы на слияние (использование ветвей проекта и управление правами участников)
- Трекер задач (определение задач для каждого коммита)
- Обсуждение кода (комментирование участков кода)
- Уведомления в Telegram (получение отчетов о процессе разработки проекта)

GitFlic имеет самые базовые функции инструмента командной разработки, а большинство дополнительных опций находятся в разработке, поэтому ими сейчас нельзя воспользоваться. У GitFlic'а есть все шансы в будущем догнать такие сервисы, как GitHub и GitLab, но успех зависит только от его команды разработчиков.

## **6) Gitea**

Gitea— это система для организации работы с Git-репозиториями. Этот сервис создала группа разработчиков, ушедших из Gogs. Основным преимуществом над остальными инструментами командной разработки состоит в малом потреблении ресурсов. Существует интеграция с сервисами Slack и Discord. Gitea хвалят за простоту установки и удобный интерфейс.

Основные возможности Gitea:

- Отображение активности по шкале времени
- Средства для миграции и клонирования репозитория и wiki из других инструментов командной разработки
- Доступ к репозиторию по протоколам SSH и HTTP/HTTPS
- Поддержка MySQL, SQLite3, PostgreSQL

## **7) Gitee**

Gitee – это китайская платформа для хостинга с открытым исходным кодом, построенная на системе контроля версий Git. Данным сервисом пользуются более 5 млн разработчиков и более 100 тыс. компаний. Gitee поддерживают множество частных компаний Китая, которые планируют полностью отказаться от GitHub, ввиду возможности внешней блокировки.

Gitee имеет большой и полезный функционал, из которого можно выделить:

- Онлайн-IDE
- Поддержка IP-ограничения для доступа к репозиторию
- Поддержка многоуровневых задач и связанных задач
- Создание еженедельных отчетов о работе
- Автоматический анализ качества кода
- Канбан-доска задач

## **8) ProjectLocker**

ProjectLocker - это служба хостинга кода корпоративного уровня с закрытым исходным кодом, которая поддерживает системы контроля версий Git и SVN. ProjectLockers имеет веб-систему управления пользователями и

РАМ. Он также предлагает автоматическое резервное копирование и непрерывную интеграцию buildlocker.

Обладает такими опциями как:

- Автоматическое развертывание
- Лента активности проекта
- Отслеживание ошибок
- Вики с вложениями
- Автоматическое резервное копирование данных

## **9) RhodeCode**

RhodeCode – это платформа для управления репозиториями и совместной работы на основе системы контроля версий Git, Mercurial и Subversion. Сервис организует совместную разработку с контролем прав доступа к различным репозиториям и рецензированием кода. Имеет те же коммиты и пулл-реквесты, что и остальные инструменты командной разработки. RhodeCode предоставляет возможность вносить изменения в код через веб-интерфейс. Сервис включает в себя online-редактор с подсветкой синтаксиса, изменений в коде.

### **3 ВЛИЯНИЕ ИНСТРУМЕНТОВ КОМАНДНОЙ РАЗРАБОТКИ НА ДОСТИЖЕНИЕ ЦЕЛЕЙ ПРОЕКТА**

Инструменты командной разработки оказывают позитивное влияние на развитие проекта. С их помощью возможно разделить один проект на множество проектов с разными подзадачами. Например, инструмент командной разработки GitHub предусматривает возможность деления команды разработчиков на разные группы с разными доступами к репозиториям. В каждом репозитории могут разрабатываться определенные области проекта. Управляющий проектом имеет доступ ко всем репозиториям и может контролировать разработку. С помощью такого инструмента намного удобнее ориентироваться в разработке, так как все находится на своих местах, и никто друг другу не мешает.

Не менее полезным является инструмент командной разработки Trello. Его интерфейс похож на магнитную доску с приклеенными на ней небольшими стикерами, поэтому Trello очень популярен среди разработчиков. С его помощью разработчики могут наблюдать за выполнением поставленных задач и ходом развития проекта. На доске можно выстроить подробный план разработки и параллельно с ним подзадачи со своими дедлайнами к каждому пункту плана, с которыми должен справиться определенный человек или команда. Кроме того, разработчики могут оставлять на ней заметки для себя, чтобы не забыть что-нибудь доработать или изменить. Также можно создать отдельную карточку для отчетов о проделанной работе. Используя такой инструмент командной разработки как Trello, команде станет намного проще довести проект до конца, так как, имея под рукой полностью расписанный план, разработчику станет легче выполнять свою работу постепенно.

Инструменты командной разработки играют важную роль в разработке и развитии проектов. Они централизуют почти все процессы работы команды над проектом, сохраняют время на разработку, которое могло быть потрачено на определение стадии готовности проекта и проделанной работы и обеспечивают комфорт при командной работе.

## 4 ЗАКЛЮЧЕНИЕ

Изучив инструменты командной разработки, можно сделать вывод, что все они по-своему хороши. Каждый инструмент имеет основную функцию – систему контроля версий, поэтому при выборе сервиса лучше обращать внимание на его дополнительные опции и удобство пользования. Для новичков и программистов-любителей больше всего подходит GitHub, так как он имеет приятный и минималистичный дизайн и все главные опции, которые понадобятся разработчику, присутствуют в бесплатном плане. Он является самым популярным веб-хостингом проектов, а значит любой, кому понадобится поделиться ссылкой, сможет открыть репозиторий. Для крупных проектов и компаний лучше подойдет GitLab или Gitee, так как их функционал в основном направлен на серьезную разработку, чем на частные небольшие работы.

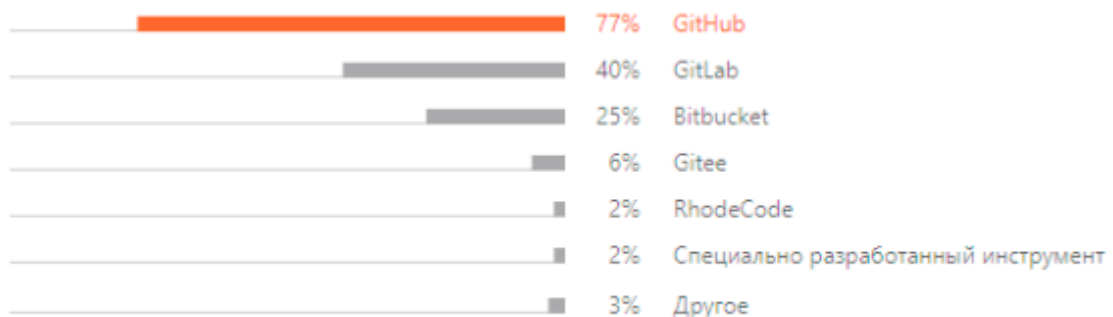


Рисунок 2 – Шкала популярности инструментов командной разработки

## 5 СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ

1. Материал с сайта Atlassian.com. Что такое Git? <https://www.atlassian.com/ru/git/tutorials/what-is-git> (Дата обращения: 06.05.2022)
2. Материал с сайта Netology.ru. Что такое Trello и как им пользоваться. <https://netology.ru/blog/trello> (Дата обращения: 06.05.2022)
3. Материал с сайта Itlogia.ru. GitLab и GitHub: в чем различия? [https://itlogia.ru/article/gitlab\\_i\\_github\\_v\\_chem\\_razlichiya](https://itlogia.ru/article/gitlab_i_github_v_chem_razlichiya) (Дата обращения: 01.05.2022)
4. Материал с сайта Atlassian.com. Краткий обзор Bitbucket. <https://www.atlassian.com/ru/software/bitbucket/guides/getting-started/overview> (Дата обращения: 02.05.2022)
5. Материал с сайта Habr.com. Китай поставил задачу превратить Gitee в местный аналог GitHub. <https://habr.com/ru/news/t/516158/> (Дата обращения: 29.04.2022)
6. Материал с сайта Gitflic.ru. GitFlic - первый российский сервис для хранения кода и работы с ним. <https://gitflic.ru/> (Дата обращения: 02.05.2022)
7. Материал с сайта Vc.ru. Шесть альтернатив GitHub для совместной разработки. <https://vc.ru/services/39321-shest-alternativ-github-dlya-sovmestnoy-razrabotki> (Дата обращения: 25.04.2022)