

## Опыт использования систем контроля версий в учебном процессе

*Шагбазян Давид Варданович*

*Нижегородский государственный университет им. Н.И.Лобачевского*

*Аспирант*

*Штанюк Антон Александрович*

*Нижегородский государственный университет им. Н.И.Лобачевского*

*доцент*

### Аннотация

В статье рассматриваются вопросы, связанные с использованием систем контроля версиями (VCS) в учебном процессе. Предлагается использовать Git, имеющий множество пользователей и наиболее динамичное развитие. Перечислены варианты применения VCS как студентами, так и преподавателями.

**Ключевые слова:** системы контроля версиями, git, github, изучение программирования, учебный процесс, репозиторий, коммит

### Version Control System in education process

*Shagbazyan David*

*Nizhny Novgorod State University*

*graduate student*

*Shtanyuk Anton*

*Nizhny Novgorod State University*

*Associate professor*

### Abstract

In article the questions connected with use of control systems of versions (VCS) in educational process when studying programmer disciplines are considered. As such system it is offered to use Git having a great number of users and the most dynamic development. The variants of using VCS by students and teachers are listed.

**Keywords:** version control systems, git, github, programming studying, education process, repository, commit.

Системы контроля версий (Control Version Systems, VCS) - представляют собой класс программного обеспечения, широко используемого в индустрии промышленной разработки ПО, а также при создании web-сайтов и систем документооборота [1]. Благодаря своим возможностям, эти системы обеспечивают непрерывность работы над

электронными документами, использование версий и распараллеливание процесса редактирования. В данной статье обсуждаются возможности применения этого класса профессиональных программных продуктов для обеспечения высокого качества учебного процесса, повышения производительности работы преподавателей, в первую очередь связанных с дисциплинами IT-направления (программирование, администрирование, тестирование).

В последние годы, процесс профессиональной подготовки студентов высших учебных заведений, претерпел ряд значительных изменений в плане использования современных информационных технологий. Информирование студентов производится с помощью специализированных программных комплексов [2]. Широкое распространение получили такие программные системы, как Moodle и E-learning, позволяющие организовать учебный процесс в рамках графика, задаваемого преподавателем. Студенты получают возможность работы online с материалами курса, используют форму обратной связи в виде заданий, в том числе тестовых. Автоматизация процесса тестирования позволяет контролировать успеваемость по основным параметрам курса, выставлять оценки. Появилась возможность взаимодействия между преподавателем и студентами в режиме реального времени, при проведении online-консультаций. При этом, однако, остается актуальной задача верификация присылаемых на проверку исходных кодов программ, определение личного вклада студента в работу над этими кодами. До сих пор широко применяется практика устных ответов и написание программ по памяти в результате непосредственного общения преподавателя со студентами.

Анализ текстов программ преподавателем до сих пор может вызывать ряд проблем. Далеко не всегда студенты придерживаются правил оформления своих результатов, приносят (или присылают) на проверку некачественные тексты. На преподавателя ложится дополнительная нагрузка в плане расшифровки этих текстов, а о верификации и потенциальной компилируемости данного кода речь может и не идти совсем.

Необходимо отметить, что VCS начинают применяться в преподавательской практике [3, 4]. В последние годы появилось несколько сообщений об успешном применении контроля версий на занятиях по программированию. Как правило, эти системы применяют молодые преподаватели, нередко совмещающие работу в ВУЗе с работой в компаниях-разработчиках ПО, где подобные инструменты не редкость. Необходимо всячески поддерживать эти инициативы, сближающие процесс обучения с реалиями современных производственных процессов.

Тем не менее, процесс использования VCS в преподавательской практике идет довольно медленно. Этому есть ряд объяснений:

1. Наличие психологического барьера у преподавателей старшего поколения перед современными инструментами, недоверие.
2. Недостаточная информированность в плане возможностей данного класса программного обеспечения.

3. Отсутствие (или недостаточная степень) обратной связи между процессами обучения в ВУЗе и требованиями производственного процесса в рамках инженерной деятельности.

К основным преимуществам от использования VCS для студентов можно отнести:

1. Использование сетевых хранилищ документов (исходных файлов программ) с возможностью доступа к ним из любой точки мира.
2. Распределенный характер хранилища, позволяющий иметь несколько копий документов на различных компьютерах и повышающий защищенность системы в целом.
3. Предпосылки к формированию у студентов способностей к коллективной разработке ПО, свободному и плодотворному обмену информацией, чему в немалой степени способствует открытый характер хостинга проектов.
4. После окончания обучения, личный аккаунт студентов сохраняется за ними и в дальнейшем играет роль резюме, помогающий устроиться на работу в престижные компании.

Можно выделить ряд преимуществ, которые получают преподаватели от использования VCS:

1. Доступ к работам студентам организуется через единое хранилище (репозиторий).
2. Имеется возможность организации параллельной работы студентов над своими задачами.
3. Имеются средства контроля над интенсивностью работы, индивидуального вклада студента.
4. Возможно использование средств автоматического контроля результатов работы (использование непрерывной интеграции CI).

Отдельно хочется подчеркнуть возможности VCS в плане оценивания личного вклада студента при получении результата. Несмотря на то, что современные тенденции поощряют свободный обмен информацией между участниками производственных и учебных процессов, для преподавателя остается актуальной задача оценивания личного вклада студента в представляемый результат. Студент может получить этот результат как самостоятельно, так и путем заимствования из разных источников. Особенно опасно полное заимствование, свидетельствующее об отсутствии личного вклада в работу, нежелание выполнять задание и пренебрежение графиком учебного процесса. Личная беседа с таким студентом может быть не всегда эффективна, поскольку студент может быть подготовлен к возможным вопросам заранее. Система контроля версий дает возможность оценить реальный вклад студента в результат, путем фиксации истории изменений. В процессе работы над задачей, VCS фиксирует изменения и формирует их в виде, так называемых "коммитов". Каждый коммит сопровождается уникальным комментарием и набор этих коммитов дает наглядное

представление о работе студента над заданием. Преподавателю остается только проанализировать эту историю и убедиться в объеме проведенной работы.

Системы контроля версий поощряют ветвление, которое не только препятствует случайным потерям нужной информации, но и делают процесс работы над задачей более четким (Рисунок 1).

Для группы создается хранилище, в котором главная ветка (master) редактируется самим преподавателем, в нее помещается документация по выполнению работы.

Студенты выполняют копирование центрального хранилища в свои аккаунты, а затем работают в индивидуальных ветках, согласно распределению. После окончания работы, все данные студентов направляются в центральное хранилище, где они поступают на проверку преподавателю.



Рисунок 1 – Вариант ветвления в VCS

В качестве рекомендуемой системы контроля версий следует использовать Git [5], обладающая рядом конкурентных преимуществ среди других систем:

1. Свободное распространение.
2. Наличие большого числа программ, имеющих встроенную поддержку данной VCS.
3. Распределенный характер, повышающий надежность хранения данных.
4. Динамичное развитие и широкая популярность в мире.

Помимо использования вместе с Git популярной системы хостинга проектов GitHub [6], с помощью которой сейчас происходит распространение не только свободного программного обеспечения, но и свободных книг, документов, есть возможность организовать в рамках учебного процесса автоматическую проверку присылаемых текстов программ с помощью серверов непрерывной интеграции. Такой сервер непрерывно осуществляет мониторинг хранилища исходных текстов программ и если студенты вносят в него изменения, то извлекает исходные тексты и выполняет построение

программы по заданному сценарию, а потом применение тестового набора. Благодаря этому удастся выявить большинство ошибок в присылаемых текстах программ и подготовить для преподавателя материал для собеседования.

### **Библиографический список**

1. Штанюк А.А. Системы контроля версий в электронном документообороте. Международное научное издание Современные фундаментальные и прикладные исследования. 2013. № 4 (11). С. 60-62.
2. Якимов А.С., Малиевский Я.Г., Баженов Р.И. Разработка серверного модуля информационной системы для информирования студентов университета. Постулат. 2016. № 8 (10). С. 5.
3. Штанюк А.А. Системы управления версиями при изучении программирования // Международное научное издание Современные фундаментальные и прикладные исследования. 2015. № 3 (18). С. 19-21.
4. Гусева Т.Ф., Комиссарова О.Р. Применение системы контроля версий в образовательном процессе // Современная техника и технологии. 2016. № 11. Ч. 2. URL: <http://technology.snauka.ru/2016/11/11369> (дата обращения: 02.10.2017).
5. Git. URL: <https://git-scm.com> (дата обращения 10.12.2017)
6. GitHub. URL: <http://www.github.com> (дата обращения 10.12.2017).