

UDC 004.8

MODERN MACHINE LEARNING TECHNOLOGIES

BARSHCHEVSKY Evgeny Georgievich

Candidate of Sciences in Technology, Professor
State University of the Sea and River Fleet named after Admiral S.O. Makarov
St. Petersburg, Russia

The relevance of the work is due to the widespread use of machine learning in various fields of science, technology, and human activity. The article discusses the composition of machine learning, categories of machine learning, the work of machine learning, machine learning software tools.

Keywords: machine learning, machine learning category, clustering, classification.

СИСТЕМЫ КОНТРОЛЯ ВЕРСИЙ ДЛЯ ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ

БУНЬКИН Виктор Иванович

кандидат технических наук, доцент кафедры цифровой экономики
НОЧУ ВО «Московский финансово-промышленный университет «Синергия»
г. Москва, Россия

В статье приводится описание двух популярных средств контроля версий – СКВ git и облачного онлайн-сервиса GitHub. Дается краткая справка по данным системам и рекомендации по их использованию при изучении различных дисциплин, связанных с программированием.

Ключевые слова: система контроля версий, технологии программирования, СКВ git, GitHub.

При изучении программирования, а также других вопросов, связанных с бурно развивающимися информационными технологиями, стоит задача не только получить

соответствующие знания, но и научиться навыкам работы с такими инструментальными средствами, которые в своей работе активно используют профессионалы.

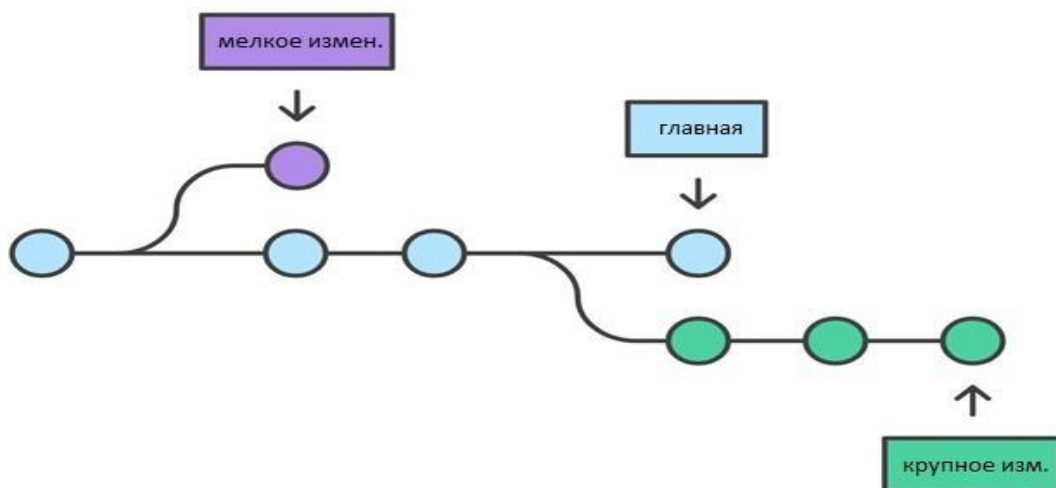


Рисунок 1. Основные приемы работы с СКВ git

Поскольку реальная промышленная разработка ПО давно перестала быть уделом одиночек, а требует умения работы в команде, то будет разумно, начиная с самых первых шагов в программировании, прививать студентам навыки совместной работы над проектами (например, в рамках изучения курса «Разработка кода информационных систем» или «Технология программирования»). Часть заданий соответственно должна носить форму групповых проектов, предполагающих общий результат и единый совместный отчет по работе.

Даже при выполнении индивидуальных заданий (например, лабораторных работ в рамках изучения курса «Основы алгоритмизации и программирования») студентам стоит почувствовать, что разработка программного обеспечения давно стала чем-то большим, чем просто набором исходного кода программы. В частности, в реальной работе используется различный дополнительный инструментарий (багтрекеры, системы для управления проектами, системы контроля версий (СКВ) и т. д.). Использование такого ресурса, как GitHub помогает «не только отслеживать изменения в версиях ..., но и практикоориентированными методами показать студентам способы организации и ведения совместной работы над проектами» [1].

Также нужно принять во внимание эффективность работы преподавателя: при наличии в группе даже десятка учащихся преподавателю не хватает отведенного аудиторного времени для качественной проверки всех сданных работ. Поэтому хотелось бы так организовать учебный процесс, чтобы максимально использовать онлайн-сервисы. Например, желательно иметь следующие возможности для преподавателя и учащихся:

- публиковать в сети задания и оценки;
- просматривать студенческие листинги, контролируя время их размещения и авторство;
- устанавливать dead-line т. е. запрещать с определенного момента запись работ;
- создавать комментарии не только в целом к проекту, но и к определенным строкам программы;
- указывать на недочеты в проделанной

работе и размещать указания к выполнению заданий;

- в любой момент студенту пользоваться своим и общим репозиториями;
- автоматически получать студентом уведомления о проверке работы и комментариях преподавателя.

Многие современные LMS (обучающие информационные системы) поддерживают часть из перечисленных выше возможностей. Например, к таким ИС можно отнести систему, внедренную в учебный процесс в Университете «Синергия» (г. Москва).

Однако для приобщения студентов к инструментарию профессиональных разработчиков ПО целесообразно задействовать при обучении СКВ git (см. рисунок 1) в связке с облачным сервисом GitHub (web-репозиторий для проектов с открытым исходным кодом, использующих git для контроля версий). Важной особенностью git является то, что можно работать на локальном компьютере с периодическим обновлением репозитория (синхронизация) на GitHub.

Базовым элементом репозитория git является коммит (commit) – зафиксированное пользователем состояние репозитория. К любому из коммитов можно вернуться и посмотреть изменения и дополнения по сравнению с другими коммитами. Именно это позволяет преподавателю прямо в тексте сданных программ указывать на недочеты и ошибки. Используя GitHub, легко находить изменения в коде, внесенные для исправления предыдущих ошибок. Это позволяет не просматривать заново листинг у каждого студента, а работать только с теми участками кода, в которые были внесены изменения.

GitHub предоставляет также широкий набор дополнительных инструментов, в частности – возможность просматривать «историю» любого файла, выяснять авторство любой строки кода проекта, визуализировать топологию ветвей и коммитов, ставить студентам задачи в багтрекере, создавать электронную документацию, отображать оценки за работы через простую вики-разметку (используется для оформления текста на веб-сайтах), собирать и визуализировать многочисленные статистики. Также на этом сервисе имеется

возможность администрирования командной работы, что дает преподавателю широкие возможности по управлению учебным процессом.

Еще одним опробованным применением является сопровождение руководителем написания курсовой работы, если использовать указанные технологии совместно с технологией TeX, так как все форматирование TeX хранит в текстовом виде.

При использовании GitHub могут возникнуть и некоторые проблемы (см. также [1]):

Во-первых, не все студенты осознают всю широту возможностей СКВ, и пытаются использовать git+GitHub, как простой облачный файлообменник (аналогично iCloud, Dropbox, Google Диск и т. п.).

Во-вторых, поскольку многие используют ОС Windows, то часто у них возникает проблема с конвертацией кодировок файлов из Win1251 в UTF.

В-третьих, любая СКВ рассчитана на работу в основном с текстовыми файлами. Поэтому преподавателю удобно работать с репозиториями студентов только, когда там хранятся лишь файлы, необходимые для создания и дальнейшей компиляции проекта. Однако студенты могут работать в различ-

ных IDE. И потому нам приходится создавать конфигурационные и прочие файлы самостоятельно (отдельно, например, для Qt, Code::Blocks или Visual Studio). Эту проблему можно решить, используя еще одну промышленную технологию: автоматизированные IDE-независимые сборщики (например, CMake). Такой подход пока используется не очень широко. В то же время, если нет жестких требований к языку программирования (например, при изучении «Компьютерной графики»), то надеяться, что на компьютере преподавателя будет развернута именно та система, которой пользовался студент – нереально. Поэтому студенты должны хранить в репозитории помимо исходных кодов также исполняемые файлы с готовым приложением. Но это не дает возможности проверить соответствие предоставленной программы с размещенным текстом кода.

В целом внедрение указанной связки технологий в учебный процесс существенно улучшает качество учебного процесса, облегчает работу преподавателя, и дает студентам чувство сопричастности современным технологиям промышленной разработки программного обеспечения.

СПИСОК ЛИТЕРАТУРЫ

1. Шалтунович А.В. Организация совместной разработки веб-приложений в рамках социальной сети github // Вестник Нижневартковского гос. университета. – 2021. – № 3. – URL:<https://vestnik.nvsu.ru/2311-1402/article/view/49183>.

VERSION CONTROL SYSTEMS FOR TEACHING PROGRAMMING

BUNKIN Victor Ivanovich

Candidate of Sciences in Technology, Associate Professor at the Department of Digital Economics
Moscow Financial and Industrial University «Synergy»
Moscow Russia

The article provides a description of two popular version control tools – VCS git and the online cloud service GitHub. A brief summary of these systems and recommendations for their use in the study of various disciplines related to programming are given.

Keywords: version control system, programming technologies, VCS git, GitHub.