

# ИСПОЛЬЗОВАНИЕ КОРПОРАТИВНЫХ ПРАКТИК РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПРИ ПОДГОТОВКЕ СПЕЦИАЛИСТОВ В ОБЛАСТИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

## THE USE OF CORPORATE PRACTICES IN SOFTWARE DEVELOPMENT IN EDUCATIONAL PROCESS OF COMPUTER SCIENCE STUDENTS

**A. Antonikov  
A. Nesterov  
T. Ermakova**

*Summary.* This paper discusses the problems of integration into the educational process the best practices of software development, such as: use of version control systems; use of continuous integration services; use of services for continuous analysis and measurement of code quality; use of bug tracking systems.

*Keywords:* quality of education, educational process, version control systems, continuous integration services, bug tracking systems.

**Антоников Александр Александрович**

Аспирант, ГАОУ ВО г. Москвы «Московский городской педагогический университет»  
antonikov.alexandr@gmail.com

**Нестеров Андрей Владимирович**

Д.ф.-м.н., профессор, ГАОУ ВО г. Москвы «Московский городской педагогический университет»  
andrenesterov@yandex.ru

**Ермакова Татьяна Николаевна**

К.т.н., доцент, ГАОУ ВО г. Москвы «Московский городской педагогический университет»  
ermaktat@bk.ru

*Аннотация.* В настоящей работе рассматриваются проблемы интеграции в образовательный процесс лучших практик разработки программного обеспечения, таких, как: использование систем контроля версий; использование сервисов непрерывной интеграции; использование сервисов для непрерывного анализа и измерения качества кода; использование систем отслеживания ошибок.

*Ключевые слова:* качество образования, образовательный процесс, система контроля версий, сервис непрерывной интеграции, система отслеживания ошибок.

## Введение

Одним из ключевых показателей качества образования считается востребованность выпускника на рынке труда. Суть современной модели образовательной системы заключается в обеспечении выпускника не только определенным количеством профессиональных знаний [1], но и набором необходимых ключевых компетенций, позволяющих оперативно адаптироваться в быстро изменяющихся социально-экономических условиях и использовать полученные знания в процессе создания новой, способной конкурировать на рынке продукции и услуг. Одним из таких постоянно изменяющихся рынков труда является рынок информационных технологий, поэтому задача подготовки специалиста, востребованного на данном рынке, является актуальной. Одним из возможных решений для образовательного учреждения в процессе подготовки таких специалистов может быть включение в образовательный процесс лучших практик разработки программного обеспечения, таких, как:

1. Использование систем контроля версий;
2. Использование сервисов непрерывной интеграции;
3. Использование сервисов для непрерывного анализа и измерения качества кода;
4. Использование систем отслеживания ошибок.

## 1. Использование системы контроля версий в образовательном процессе

Одними из важнейших инструментальных средств, служащих для разработки программного обеспечения (VCS), являются системы управления версиями [2]. Данные системы предназначены для решения следующих задач:

1. Хранение предыдущих версий файлов;
2. Возможность получить доступ к предыдущим версиям хранимых файлов;
3. Возможность просматривать изменения, внесенные пользователями;
4. Сохранение и просмотр комментариев со ссылкой на авторов внесенных изменений.

Образовательные организации, реализующие образовательные программы по подготовке специалистов в области IT-технологий, должны внести дисциплины, связанные с изучением возможностей таких систем, в свои учебные планы [3–4]. Это решение позволит подготовить обучающегося к тем ситуациям, с которыми он столкнется в своей профессиональной деятельности. Необходимость добавления изучения VCS в образовательные программы заключается в том, что в современных условиях для выполнения текущих рабочих задач необходимо использовать инструменты, обеспечивающие поддержку распределенной командной работы [4].

Применение преподавателем в образовательном процессе системы контроля версий имеет несколько потенциальных преимуществ, например:

1. мониторинг успеваемости обучающихся [5];
2. управление материалами курсов, заданиями;
3. мониторинг и визуализация вклада членов студенческих команд;
4. обнаружение плагиата.

Рассмотрим пример использования студентами системы контроля версий в случае выполнения заданий преподавателя. Возможная последовательность их действий может выглядеть следующим образом:

1. создание репозитория;
2. обновление локальных данных из репозитория;
3. работа с файлами (добавление, изменение);
4. фиксирование изменений в репозитории.

В зависимости от сложности выполняемого задания данные этапы могут повторяться в течение всего времени, требующегося для выполнения обучающимися поставленной задачи.

Рассмотрим также предполагаемые преимущества применения VCS в образовательном процессе:

1. процесс отслеживания прогресса выполнения заданий студентами станет более простым и прозрачным для преподавателя, например, быстрый просмотр журнала учета изменений позволит увидеть информацию о том, как студент выполнял полученное задание: делал ли он его постепенно и размеренно или задание было выполнено в последний момент;
2. обучающиеся в процессе работы с данными системами приобретут навыки выполнения основных операций в системе;
3. несмотря на то, что применение VCS необходимо и полезно в образовательном процессе, но имеющееся ограничение в ресурсах, предоставляемых для выполнения заданий, и зачастую небольшой объем данных ресурсов не позволят студентам проводить эксперименты и использо-

вать возможности систем контроля версий в полном объеме;

4. грамотно подобранные задания позволят студентам оценить преимущества применения VCS в дальнейшей деятельности;
5. применение VCS в образовательном процессе окажет поддержку в формировании портфолио студента, так как в системе контроля версий могут храниться все выполняемые им задания и история их выполнения.

В процессе внедрения систем контроля версий в образовательный процесс можно столкнуться со следующими проблемами:

1. учитывая то, что изначально не предполагалось использование VCS в образовательных организациях, у обучающихся могут возникнуть некоторые трудности в работе. Одним из вариантов решения данной проблемы является написание руководства пользователя, в котором будут описаны этапы работы с системой;
2. применение VCS на начальном этапе повысит затраты труда преподавателя, которые связаны с подготовкой курса и настройкой системы, кроме этого, в процессе планирования курса необходимо будет учитывать отсутствие у студентов опыта работы с системами контроля версий;
3. так как большая часть студентов предпочитает откладывать выполнение заданий на длительный срок, то для них использование контроля версий не повлечет за собой кардинальных изменений, а поэтапная оценка прогресса выполнения задания также принесет определенные трудности. Чтобы решить данную проблему, можно добавить в курс несколько контрольных точек, прохождение которых будет оцениваться и влиять на итоговую оценку по дисциплине. Для максимального использования студентами контроля версий им необходимо структурировать назначенную работу с учетом конкретных этапов и провести оценку возможных последствий прохождения конкретных контрольных точек;
4. естественно, что применение в образовательном процессе системы контроля версий не позволит автоматически повысить качество обучения или увеличить знания студентов по дисциплине, однако позволит более прозрачно управлять процессом освоения конкретного курса [6–7].

## 2. Использование сервисов непрерывной интеграции в образовательном процессе

Непрерывная интеграция (Continuous Integration) — это одна из практик разработки программного

обеспечения, когда разработчики регулярно объединяют изменения программного кода в центральной репозитории, а после этого автоматически выполняется сборка, тестирование и запуск. Термин «непрерывная интеграция» часто используется на стадии сборки или интеграции процесса выпуска программного обеспечения, в него входит компонент автоматизации (например, сервис непрерывной интеграции или сборки) и компонент культуры разработки (например, обучение частой интеграции). Используя сервисы автоматизированной интеграции в образовательном процессе, преподаватели получают дополнительные возможности по:

1. мониторингу успеваемости обучающихся;
2. получению проектов, выполняемых студентами в рамках полученных от преподавателя заданий, для дальнейшей проверки [8–10].

В процессе выполнения заданий студентам необходимо будет создавать конфигурации сборок проекта и вносить в них изменения по мере работы над выполнением задания. Результатом выполнения задания будет сборка проекта и/или история сборок проекта.

Для успешного внедрения данной практики разработки программного обеспечения преподавателю необходимо подготовить методические указания по работе с сервисами непрерывной интеграции, где будет описан базовый функционал системы и приведен пример настройки сборки проекта.

### 3. Использование сервисов для непрерывного анализа и измерения качества кода

Статический анализ кода — это процесс выявления ошибок и недочетов в исходном коде программ, осуществляемый без реального выполнения исследуемых программ. Статический анализ можно рассматривать как автоматизированный процесс обзора кода.

Программы статического анализа кода позволяют решить следующие категории задач:

1. нахождение ошибок в программах;
2. рекомендации по оформлению кода. Существующие статические анализаторы предоставляют возможности по проверке соответствия исходного кода принятому в компании стандарту оформления кода. В данном случае идет речь о контроле количества отступов в различных конструкциях, использовании пробелов/символов табуляции;
3. подсчет метрик. Метрика программного обеспечения — это мера, которая позволяет получить численное значение некоторого свойства программного обеспечения или его спецификаций,

например, количество строк кода, цикломатическая сложность [11].

При использовании статического анализа необходимо учитывать, что возможны ложно-положительные срабатывания и сложности с диагностикой утечек памяти и ошибок, связанных с многопоточностью.

Применение сервисов статического анализа кода в образовательном процессе приведет к автоматизации процесса проверки выполненных заданий преподавателем.

В качестве одного из результатов выполнения задания студенту необходимо будет предоставить отчет об успешном прохождении статического анализа, для чего в процессе работы потребуется настроить сервис анализа кода для применения в проекте и итерационно исправлять замечания, выявленные сервисом.

### 4. Использование систем отслеживания ошибок в образовательном процессе

Система отслеживания ошибок представляет собой информационную систему, в которой аккумулированы найденные ошибки, пожелания пользователей, с ее помощью можно следить и управлять процессом устранения неполадок и принятия/не принятия пожеланий. В некоторых случаях багтрекинг-систему можно использовать для управления проектами.

Применение систем в образовательном процессе достаточно тривиально, все задания, которые студентам необходимо выполнить, назначаются студентам в багтрекинг-системе. В процессе работы над заданием обучающийся меняет статус назначенной ему задачи, преподаватель, в свою очередь, имеет достаточно удобный интерфейс, позволяющий отслеживать выполнение заданий, при возникновении замечаний в процессе проверки можно вернуть студенту задание для доработки. Помимо этого, можно представить прохождение курса проектом, с точкой старта, точкой окончания, списком заданий для выполнения с оцененной и указанной трудоемкостью, сроком выполнения. В таком случае можно сказать, что студент дополнительно ознакомится с проектным подходом, а преподаватель, получив обратную связь в виде фактических затрат времени на выполнение заданий, получит возможность скорректировать программу освоения дисциплины.

### Заключение

Создание современной IT-инфраструктуры и активное ее использование в образовательном процессе позволит не только повысить востребованность выпускника

на рынке труда, а, следовательно, и качество образования, но и может открыть дополнительные возможности для взаимодействия образовательного учреждения с потенциальными работодателями, например, предоставить компаниям, сотрудничающим с образовательными учреждениями, доступ к системе контроля версий

для оценки способностей потенциальных сотрудников. В свою очередь, интеграция различных информационных сервисов и систем в единую систему позволит образовательному учреждению создать единое информационное образовательное пространство, которым можно будет пользоваться в любое время и в любом месте.

#### ЛИТЕРАТУРА

1. Каптерев А. И. Профессиональное знание как объект управления: мультидисциплинарный подход: Монография. — Verlag. — Saarbrücken. — Deutschland. — 2012. — 442 с.
2. Gudkova, I.A., Romashkova, O.N., Samoylov, V. E. Determination of the range of the guaranteed radio communication in wireless telecommunication networks of IEEE802.11 standard with the use of ping program // В сборнике: CEUR Workshop Proceedings 8. Сер. "ITMM 2018 — Proceedings of the Selected Papers of the 8th International Conference "Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems"" 2018. С. 54–59.
3. M. Cochez, V. Isomöttönen, V. Tirronen, and J. Itkonen. How Do Computer Science Students Use Distributed Version Control Systems? pages 210–228. Springer, 2013.
4. M. Cochez, V. Isomöttönen, V. Tirronen, and J. Itkonen. The use of distributed version control systems in advanced programming courses. In Proceedings of the 9th International Conference on ICT in Education, Research and Industrial Applications: Integration, Harm.
5. Барсукова К.Н., Чискидов С. В., Павличева Е. Н. Актуальные проблемы автоматизации учета результатов обучения студентов в вузе (на примере ИМИИЕН ГБОУ ВПО МГПУ) // Информационные ресурсы России. — 2015. — № 3 (145). — С. 37–39.
6. Ромашкова О.Н., Ермакова Т. Н. Этапы реализации методики выбора информационной модели для оценки показателей качества обучения // В сборнике: Междисциплинарные исследования в области математического моделирования и информатики Материалы 7-й научно-практической интернет-конференции. отв. ред. Ю. С. Нагорнов. 2016. С. 318–321.
7. Ромашкова О.Н., Ермакова Т. Н. Применение инфокоммуникационных технологий для анализа показателей качества обучения образовательного комплекса // В сборнике: Технологии информационного общества X Международная отраслевая научно-техническая конференция: сборник трудов. 2016. С. 388–389.
8. Федин Ф.О., Фролов П. А., Чискидов С. В., Павличева Е. Н. Автоматизация разработки электронных учебно-методических комплексов для кафедр организаций высшего образования // Информационные ресурсы России. 2017. № 6 (160). С. 38–42.
9. Антониюков А.А., Чискидов С. В., Павличева Е. Н. Разработка модуля системы дистанционного обучения для проверки знаний в области программирования // Информационные ресурсы России. 2012. № 3 (127). С. 32–34.
10. Ермакова Т.Н., Ромашкова О. Н., Пономарева Л. А. Модернизированная структура управления образовательной системой // Вестник Брянского государственного технического университета. 2019. № 6 (79). С. 84–91.
11. Gaidamaka, Y.V., Romashkova, O.N., Ponomareva, L.A., Vasilyuk, I. P. Application of information technology for the analysis of the rating of university // В сборнике: CEUR Workshop Proceedings 8. Сер. "ITMM 2018 — Proceedings of the Selected Papers of the 8th International Conference "Information and Telecommunication Technologies and Mathematical Modeling of High-Tech Systems"" 2018. С. 46–53.

© Антониюков Александр Александрович ( antonikov.alexandr@gmail.com ),

Нестеров Андрей Владимирович ( andrenerstov@yandex.ru ), Ермакова Татьяна Николаевна ( ermaktat@bk.ru ).

Журнал «Современная наука: актуальные проблемы теории и практики»