

УДК 378.147

## ОПЫТ КОМАНДНОГО ПРОЕКТНО-РОЛЕВОГО ОБУЧЕНИЯ ПРОГРАММИРОВАНИЮ

**Чернышев Станислав Андреевич<sup>1,2</sup>,**

кандидат технических наук, доцент кафедры интеллектуальных систем и защиты информации; доцент кафедры информатики,  
chernyshev.s.a@bk.ru

- <sup>1</sup> Санкт-Петербургский государственный университет промышленных технологий и дизайна, Россия, 191186, г. Санкт-Петербург, ул. Большая Морская 18.
- <sup>2</sup> Санкт-Петербургский государственный экономический университет, Россия, 191023, г. Санкт-Петербург, ул. Садовая 21.

В статье описывается опыт применения командного проектно-ролевого обучения программированию студентов. Цель данного подхода – обучить студентов работать над проектами различной сложности в составе команды при выполнении каждым из ее членов одной из дополнительных ролей: менеджер, программный архитектор или тестировщик. Это формирует у студентов представление о минимальном составе команды, необходимом для разработки программных продуктов и систем, распределении обязанностей, самом процессе и важности написания технического задания, методики приемо-сдаточных испытаний, используемых инструментов для управления проектами и т. д. Для отслеживания результатов продвижения команды по проекту, автор предлагает использовать сервисы GitHub и Trello, а также еженедельные презентации по проекту каждой из команд с проверкой соответствия кодовой базы проекта и состояния среды командного взаимодействия представляемой презентации. Использование таких инструментов позволяет производить оценку как команды в целом, так и каждого из студентов в частности. Также рассматриваются существующие достоинства, недостатки и возможные проблемы, с которыми предстоит столкнуться при использовании данного подхода обучения студентов программированию. При этом, реализованные проекты могут использоваться как портфолио при поиске работы, либо в качестве основы для выпускной квалификационной работы. Среди существующих недостатков особенно выделяется требование к уровню компетенций самого преподавателя, так как это непосредственно влияет на сам результат использования командного проектно-ролевого обучения программированию.

**Ключевые слова:** командное проектно-ролевое обучение, проект, программирование, управление проектами, образование

### Введение

Постоянно возрастающие потребности рынка информационных технологий в разработчиках программных продуктов привели к тому, что на уровне федеральных средств массовой информации начали говорить о их нарастающем дефиците [1, 2]. При этом на специализированных форумах все чаще стали появляться публикации о проблемах высшего образования в России при подготовке it-специалистов [3–5]. В ряде случаев доходит до того, что студенты сами берут на себя инициативу и проводят мероприятия, где рассказывают другим студентам о реально применяющихся технологиях на рынке: «У нас был кое-какой опыт разработки приложений и мы умели в актуальные технологии в IT. Большинству из них не учат в универе, по крайней мере в нашем, и нас это не устраивало. Мы хотели, чтобы перваки, которые ещё не определились, нашли себя. Предмет «Введение в направление», им в этом не помогал, а на деле

оказался пересказом учебного плана с пачкой пассивной агрессии от преподавателя» [6].

Там же можно встретить от работающих в it-сфере преподавателей анализ ситуации, которая складывается в современном образовательном процессе [7, 8], ее возможные пути развития или решения.

В столичных учебных заведениях или университетах, которые входят в первые 50 позиций рейтинга Forbse [9] основную задачу по подготовке качественных специалистов взяли на себя кафедры, которые открываются под патронажем компаний, действующих на рынке информационных технологий или в кооперации с ними. К сожалению, количество специалистов, которые эти кафедры способны обучить, ограничено и их основная целевая функция – покрыть дефицит начинающих it-специалистов в самих компаниях. А что делать остальным компаниям, у которых нет средств для выстраивания аналогичного подхода? Или ВУЗам, на которые it-гиганты не об-

рашают свое внимание, но они также занимаются подготовкой it-специалистов?

Для начала необходимо признать тот факт, что система образования в её текущем виде и то, как организован процесс обучения таких специалистов в большинстве высших учебных заведениях страны, не справляются. Выпускаемые it-специалисты должны обладать рядом актуальных компетенций для рынка. К сожалению, в ФГОС ВО 3++ универсальные, профессиональные и общепрофессиональные компетенции, а также требования к формированию умений применять знания на практике, сформулированы в слишком общем виде. Отчасти это связано с тем, что за период обучения одного студента в компаниях, которые разрабатывают программные продукты, может несколько раз смениться стек используемых технологий. Так, например, в области front-end разработки, в течение полугодия может выйти новый фреймворк или библиотека, которые начнут активно использоваться. В соответствии с этим, образовательная программа ВУЗа должна проявлять сверх-мобильность и подстраиваться под постоянно меняющиеся тенденции it-индустрии.

Одним из инструментов, позволяющих студентам в процессе обучения нарабатывать актуальные для рынка компетенции является проектное обучение, которое уже длительное время применяется во многих странах мира [10–14] и активно внедряется в образовательный процесс в России [15–19]. Самые наилучшие результаты этот метод приносит при его использовании в процессе подготовки специалистов, чья профессиональная деятельность будет в последующем связана с проектной формой работы [18].

Целью данной статьи является обобщение, наработанного за последние 3 года, опыта автора командному проектно-ролевому обучению разработке программных продуктов с использованием актуального стека технологий. Проблем, с которыми пришлось столкнуться и используемых инструментов для управления сформированными в процессе обучения командами.

### **Подготовительная работа**

В том случае, если студенты еще не изучали ни одного языка программирования или не имеют достаточной компетенции в написании кода, первоочередной задачей является их предварительная подготовка. Для этого

используется традиционный подход с лекциями и практическими работами, в дополнении к которому студенты решают задачи по программированию на таких сервисах, как: Codewars [20], LeetCode [21], Stepik [22] и Яндекс.Контест [23]. Основная цель этого этапа – автоматизация навыков кодирования: циклы, ветвления, работа со структурами данных и т. д., чтобы не было необходимости при постановке простейших задач искать решение в интернете. На изучение базовых конструкций языка, структур данных и способов работы с ними отводится 2–3 месяца.

После этого студентам дается два индивидуальных проекта: калькулятор и список дел (ToDo). На каждый из этих проектов максимум отводится по две недели. По результатам их выполнения преподаватель формирует списки, разделяя студентов по ряду критериев: мотивированность, успеваемость, опыт написания кода и т. д.

Дополнительно, к оцениванию студенческих работ преподавателем используется метод анонимной перекрестной проверки (cross-check) студентами друг друга. Проверяется только конечный результат – само приложение без исходного кода, то, насколько им удобно пользоваться, наличие базового и дополнительного функционала. Важным при этом условием является донести до студентов, что эта проверка не влияет на оценку, выносимую преподавателем, чтобы у них не возникло желание «завалить» друг друга. При этом выполнять проверку могут даже те студенты, которые еще не выполнили задание. Это позволяет им взять для своих реализаций заинтересовавший их функционал и мотивировать на скорейшее завершение собственного проекта.

### **Способ организации команд и распределение ролей**

Исходя из сложности назначаемого или выбираемого проекта команда состоит из двух или трех человек. Основное условие – вне зависимости от назначаемой дополнительной роли все участвуют в написании кода программного проекта, то есть базовая роль каждого студента в проекте – программист.

В команде из двух человек, одному из них дополнительно назначается роль менеджера, а второму – программного архитектора. При этом каждый из них должен участвовать в написании тестового окружения для проекта.

При наличии в команде трех человек, выделяются следующие роли: менеджер, программный архитектор, тестировщик.

Менеджер отвечает за постановку, декомпозицию и распределение задач, а также следит за тем, чтобы задачи были выполнены в обозначенные сроки и постоянно актуализирует их статус, переключается на помощь другим участникам команды, если у тех возникли трудности. Программный архитектор продумывает из каких классов и модулей будет состоять проект, как они будут взаимодействовать между собой и т. д. Тестировщик пишет тесты к коду проекта и поддерживает их в актуальном состоянии.

По результатам оценивания подготовительного этапа студенты делятся на три категории:

1. Целеустремленные (ЦУ) – студенты, показавшие результат выше среднего по итогам первого этапа.
2. Неуверенные (Н) – студенты, которые показали приемлемый результат, но в их ответах и структуре проекта видна неуверенность в своих силах.
3. Бездельники (Б) – студенты, целенаправленно не выполнявшие задания первого этапа.

В первых проектах, где преподаватель сам назначает студентов по командам, идеальным вариантом является состав из двух целенаправленных и одного неуверенного, а бездельников лучше вообще не привлекать к командно-проектной работе, оставив на традиционном варианте обучения (лабораторные/практические работы). В случае небольшого количества ЦУ-студентов команды можно распределить следующим образом: 1 ЦУ и 2 Н, либо 1 ЦУ и 1 Н.

«Бездельники», как показывает опыт, являются деструктивным элементом и стараются всеми силами переложить свои обязанности на других членов команды. Порой доходит даже до того, что в этом стремлении перестают поддерживать хоть какую-то связь с командой.

Начиная с 3-го или 4-го проекта у студентов появляется возможность самим сформировать команду для проекта.

### Выбор проектов и их типы

Проекты делятся на два типа:

- инженерные (для бакалавров);
- инженерно-исследовательские (для магистров).

Первые два или три проекта назначаются команде из студентов бакалавриата преподавателем, у них имеются четкие требования по срокам, функционалу разрабатываемого программного обеспечения и условиям оценивания. 70–75 % времени, отводимого для проекта, отдаются на написание кода, а 20–25 % на документацию и отчетность. С каждым последующим проектом время, отводимое на документацию, должно увеличиваться, но не более чем до 40 %.

Начиная с 3-го или 4-го проекта у студентов бакалавриата имеется возможность самим предложить проект или выбрать из списка, который предоставляет преподаватель. Если это их первый самостоятельный проект в сформированном составе, то он не должен быть слишком сложным, поскольку преподавателю придется часто взаимодействовать с командой. При этом, те решения, которые он будет предлагать, могут являться довольно высокоуровневыми и лежать за пределами текущей компетенции студентов. В этом случае им предлагается сначала реализовать минимально жизнеспособный продукт (minimal valuable product, mvp), а его полноценную реализацию отложить на следующий проект.

У каждого инженерного проекта должна быть целевая аудитория: преподаватели, кафедра, сами студенты и т. д. Тот факт, что им будут пользоваться, дает дополнительную мотивацию студентам. В случае автора, примером таких проектов являются:

- CRM для управления студенческими проектами;
- telegram-бот для проведения голосований и веб-сервис для анализа их результатов;
- telegram-бот расписания кафедры и веб-сервис для внесения изменений и формирования расписания;
- telegram-бот для отслеживания посещения занятий студентами;
- программное обеспечение для аннотации изображений обучающей выборки;
- различные игры и т. д.

Проекты для магистров более комплексные и организуются таким образом, чтобы каждый участник мог взять его часть в свою магистерскую диссертацию. Так, например, проект по разработке мультиагентной системы планирования поставок продуктов состоит из двух частей: сама мультиагентная система и фреймворк для разработки распределенных мультиагентных систем.

## Программные продукты для отслеживания выполнения проектов

Каждая команда создает github-репозиторий [24] под проект и подключает к нему преподавателя. Это позволяет постоянно мониторить в каком состоянии находится кодовая база проекта, документация, чистоту кода и принимаемые в нем архитектурные решения (рис. 1).

Для отслеживания текущего состояния проекта, времени выполнения, постановки и декомпозиции задач используется сервис Trello [25] (рис. 2).

Одновременно с этим формируется Google-таблица, доступная в режиме «только чтение» всем студентам. В нее заносятся данные по командам, ссылки на github и trello, а также оценки (цветом) или замечания преподавателя по результатам текущего состояния проекта (рис. 3).

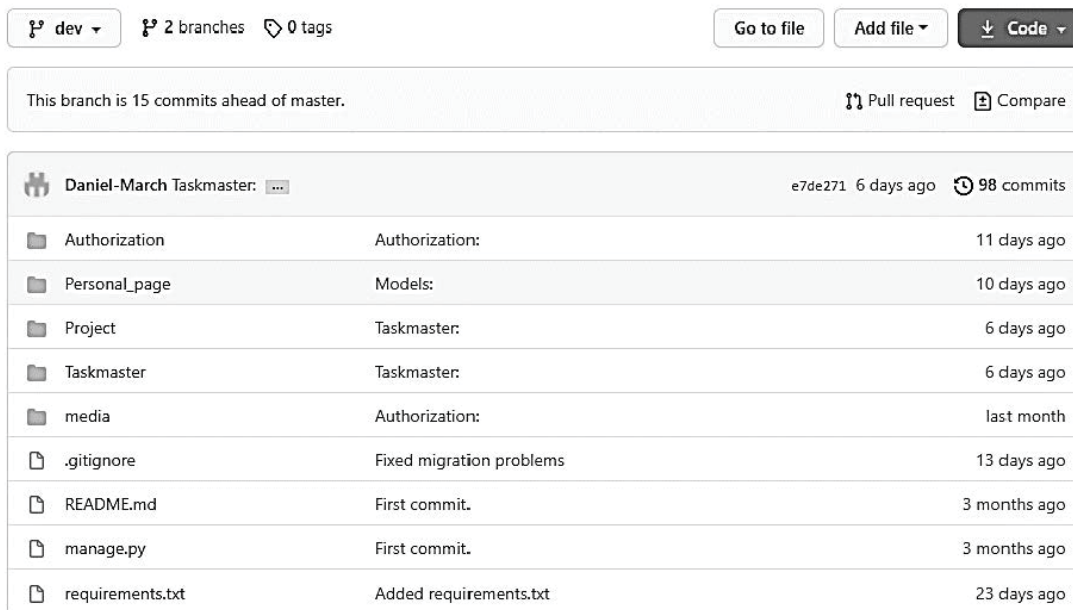


Рис. 1. Репозиторий CRM для управления студенческими проектами  
Fig. 1. CRM Repository for Student Project Management

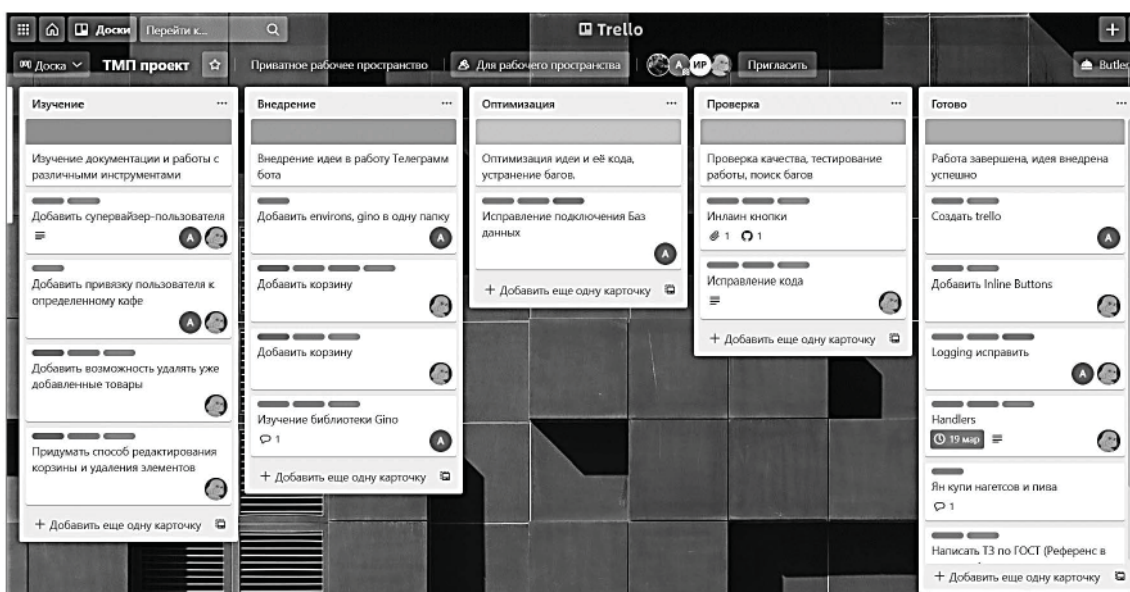


Рис. 2. Trello-доска проекта telegram-бота заказа продуктов в кафе  
Fig. 2. Trello-board of the telegram-bot project for ordering products in a cafe

участники	Группа	ПРОЕКТ	github	Trello	инструктаж	15.02.2021	20.02.21(ТЗ)
1-МДА-9	1-МДА-7	Телеграмм-бот расписания кафедры + софт для редактирования БД с расписанием	<a href="https://github.com/saplat">https://github.com/saplat</a>	<a href="https://trello.com/b/QahA1d">https://trello.com/b/QahA1d</a>			
1-МДА-9	1-МДА-7	RPG Rogue-like с генерируемыми комнатами на локациях.	<a href="https://github.com/Sgoolf">https://github.com/Sgoolf</a>	<a href="https://trello.com/roguelike">https://trello.com/roguelike</a>			
1-МДА-7	1-МДА-9	Телеграмм-бот для стоматологической клиники	<a href="https://github.com/NikaRo">https://github.com/NikaRo</a>	<a href="https://trello.com/user7707">https://trello.com/user7707</a>			
1-МДА-7	1-МДА-9	Приложение под Android "Кулинарная книга"	<a href="https://github.com/Mgai-T">https://github.com/Mgai-T</a>	<a href="https://trello.com/b/CADdd">https://trello.com/b/CADdd</a>			
1-МДА-7	1-МДА-9	Программа для распознавания жестов схожих	<a href="https://github.com/sion-dar">https://github.com/sion-dar</a>	<a href="https://trello.com/b/peTRV">https://trello.com/b/peTRV</a>			
1-МДА-7	1-МДА-9	Программа для управления приложением на компьютере посредством жестов кистей рук	<a href="https://github.com/itepland">https://github.com/itepland</a>	<a href="https://trello.com/mabnas">https://trello.com/mabnas</a>			
1-МДА-9	1-МДА-7	Название: Dungeon 6; Жанр: rogue-like dungeon crawler Игрок исследует случайно сгенерированные этажи подземелья, сражается с монстрами и боссами, получает улучшения экипировки, навыков и способности, разовые и постоянные. Цель игры - добраться до 36 этажа и победить финального босса.	<a href="https://github.com/Axc317">https://github.com/Axc317</a>	<a href="https://trello.com/b/3r6826">https://trello.com/b/3r6826</a>			
1-МДА-7	1-МДА-9	Телеграмм-бот для отслеживания потраченных за месяц денег	<a href="https://github.com/Timur1">https://github.com/Timur1</a>	<a href="https://trello.com/b/aNJK89">https://trello.com/b/aNJK89</a>			
1-МДА-7	1-МДА-9	Игра с gps ориентированием. PvP, PvE - бои. Система турниров, данжей, личных территорий, классов и навыков, групп, кланов, гильдий. Возможно не всё из этого	<a href="https://github.com/AniMA">https://github.com/AniMA</a>	<a href="https://trello.com/">https://trello.com/</a>			

Рис. 3. Google-таблица преподавателя

Fig. 3. Teacher Google Sheets

### Стадии проекта

Вне зависимости от тематики выбранного студентами проекта, он состоит из следующих стадий:

**1. Формирование технического задания и методики приема-сдаточных испытаний.** Если это первый проект для студентов, то проводится занятие с объяснением, что такое техническое задание (ТЗ), для чего оно нужно и разбирается пример его написания. Аналогичным образом разбирается методика приема-сдаточных испытаний. Эти документы студенты должны писать параллельно, что позволяет заранее продумывать, какой функционал будет в разрабатываемом программном продукте и каким образом они будут закрываться пункты ТЗ. Написание документов проходит в несколько итераций, преподаватель обсуждает с командой требования к проекту, стек технологий, который будет использоваться для его реализации и дает ряд замечаний для их последующего исправления.

**2. Разработка и отчетность по проделанной работе.** Раз в неделю производится общий сбор (очно или дистанционно) команд и преподавателя. Один человек из команды (каждый раз новый) выступает перед аудиторией с презентацией проекта, кратко рассказывая его суть, задачи, которые были поставлены на текущую неделю, что было выполнено, что нет, почему, с какими проблемами столкнулись, как решали, планы на следующую неделю. Также показывает текущее состояние

проекта, после чего отвечает на вопросы преподавателя и других студентов. Такой способ помогает студентам делиться накопленным опытом между собой и подстегивает соревновательный процесс между командами. Если команда не видит, как решить возникшую проблему в ходе разработки, преподаватель дает пару рекомендаций в каком направлении следует двигаться или, при необходимости, сразу объясняет способ ее решения. В течение этой стадии преподаватель читает лекции об алгоритмах и структурах данных, принципах, паттернах и методологиях разработки программного обеспечения, а также библиотекам и фреймворкам используемого языка программирования. Все лекции записываются и выкладываются для студентов в открытый доступ.

**3. Приемка проекта.** Она осуществляется в дату, которая прописана в ТЗ и в соответствии с подготовленной в самом начале методикой приема-сдаточных испытаний.

**4. Участие на конференции или статья в журнал с результатами выполненного проекта.** На данную стадию проходят работы, у которых имеется потенциал. Статья или тезисы доклада пишутся итерационно, с их постоянной вычиткой со стороны преподавателя. При этом редакторские правки самим преподавателем вносятся только на финальном этапе. Вплоть до этого момента студенты сами вносят исправления в соответствии с полученными замечаниями.

## Оценивание

Использование таких инструментов как github, trello и еженедельный отчет по проекту позволяют проводить оценку как команды в целом, так и каждого студента в частности. Например, средствами trello можно проследить какому студенту какая задача была назначена, в какие сроки она была выполнена, на сколько детально и правильно ведется среда командного взаимодействия, какой функционал по проекту на данный момент реализован и т. д.

GitHub используется для отслеживания изменений кодовой базы проекта, кто их вносил, какие архитектурные решения применялись и на сколько читаемый код пишут студенты.

Таким образом, каждую неделю во время отчета по проекту выполняется проверка – соответствует ли состояние дел тому, о чем говорят студенты. Если всё нормально и нет претензий по презентации или текущему состоянию проекта, то ему присваивается зеленая карточка. При ненадлежащей презентации проекта, отсутствии признаков выполненного функционала, о котором заявлено в ходе презентации в github или trello, то есть, когда команда реально забыла их обновить, проекту присваивается желтая карточка. В случае систематического повторения таких ошибок – красная. Также красная карточка присваивается проекту при невыполнении всех условий, либо когда студенты начинают «хитрить»:

- За разрабатываемый функционал в trello отвечает один студент, а код по нему в github заливает другой;
- В код проекта на github вносит изменения только один студент;
- Студентами хаотично переназначаются задачи в trello;
- Вместо документации (ТЗ и методика приемо-сдаточных испытаний) студенты прикрепляют в trello документы, не относящиеся к их проекту;

По результату семестра, когда у команды нет красных карточек и более 70 % из них – зеленые, каждому члену команды ставится оценка «отлично». При наличии двух красных карточек – «хорошо». Команда, у которой проекту было назначено от трех до шести красных карточек, но он был выполнен в срок, и студенты правильно отвечали на дополнительные вопросы – «удовлетворительно».

Во всех остальных случаях студенты получают свою оценку в ходе проведения экзамена или зачета. В этом процессе также могут участвовать и те, кто желает улучшить свою оценку, которая была поставлена по результатам их работы.

## Достоинства, недостатки и проблемы проектно-ролевого обучения

Что касается большинства достоинств и недостатков, то они аналогичны стандартному методу проектного обучения [19]. К более выделяющимся достоинствам командного проектно-ролевого обучения программированию можно отнести то, что у студентов появляется опыт взаимодействия друг с другом в соответствии с назначаемыми ролями в проекте, понимание процесса разработки программных продуктов и систем, опыт использования актуального для рынка стека технологий. Проекты, расположенные на github-репозитории, представляют собой портфолио для потенциального работодателя. Таким образом, они могут использоваться как при поиске работы, так и для участия в различных студенческих конкурсах, конференциях или в качестве выпускной квалификационной работы.

Среди проблем, что возникли при использовании проектно-ролевого обучения программированию выделяются:

- Попытка студентов с ролью менеджера не писать код. Решается довольно просто, но, если запустить, это может привести к конфликтам внутри команды.
- До студентов не сразу доходит мысль, что программирование – это не только написание кода, но еще и постоянное общение, обсуждение вариантов решения, умение слушать и т. д. Эта проблема решается сама собой в процессе продвижения по проекту.
- Некоторые студенты могут резко передумать обучаться, в связи с чем возрастает нагрузка по объему работ на его бывшую команду. В этом случае снижается объем требований к проекту, так как перераспределение людей в командах из-за ушедшего студента только усугубит ситуацию. Также может быть рассмотрен вариант, что на замену ушедшему студенту, один из членов команды приглашает знакомого.
- Студенты не могут сработаться в рамках команды. Здесь нет простого решения, самым оптимальным вариантом является ор-

ганизация простого проекта для каждого из расформированной команды.

- Студенты не сразу понимают, что программирование – это сложно и большинство времени разработчик проводит в поисках способов решения возникающих проблем, а возникают они постоянно. Здесь важную роль играет общение и примеры из реальной работы.
- У студентов нет опыта написания статей. Единственный совет, как решить эту проблему – проявить терпение, и она решится сама собой. Замечания по тексту стоит давать только с обоснованием почему такой вариант не подходит и как лучше это исправить.
- У студентов нет опыта публичных выступлений. Спустя пару месяцев еженедельных презентаций по проектам про эту проблему можно будет забыть.

Из недостатков наиболее выделяется то, что при недостаточной мотивации и контроля со стороны преподавателя проект не будет доведен до логического завершения. При этом такой подход к образовательному процессу можно организовать только при наличии у преподавателя реального опыта работы на различных позициях в сфере разработки программного обеспечения.

### Заключение

Так как разработка программных продуктов и систем ведется коллективами разработчиков, то умение студентов работать в команде, способность правильно донести с какой проблемой столкнулся или поддержать обычный разговор, рассказать над каким функционалом сейчас ведется работа и умение признавать свои ошибки являются весо-

мыми факторами, на которые работодатель обращает свое внимание. Для компании куда безопасней, в плане рисков, на позицию junior-разработчика взять человека, у которого развиты эти навыки, но он не дотягивает в плане технических компетенций, чем наоборот.

Командное проектно-ролевое обучение программированию не является панацеей и при его использовании вместо традиционного обучения необходимо учитывать достаточно большой объем факторов: начальный уровень подготовки студентов, их готовность к командной работе, наличие большого варианта проектов для реализации и преподавателя с необходимым набором компетенций, мотивированность и т. д. Такой формат обучения хорошо зарекомендовал себя в рамках студенческих кружков и для мотивированных студентов, настроенных на получении актуальных на рынке труда компетенций. К тому же, после нескольких студенческих проектов, таких студентов можно привлекать к реальным заказам.

К сожалению, если такой подход использовать при обучении программированию повсеместно, то это может привести к некоторым негативным последствиям, так как не все студенты поступают на it-направления с целью последующего трудоустройства в этом секторе экономики. Такая категория студентов, как показывает опыт, не проявляет даже минимальной активности и старается переложить весь объем работы на остальных членов команды. Чтобы такие студенты не выступали в качестве деструктивного элемента, их лучше всего оставить на традиционном варианте обучения: лекции + практические/лабораторные работы.

### СПИСОК ЛИТЕРАТУРЫ

1. Алексеев В. Катастрофический дефицит. Цифровому прорыву предрекли острую нехватку IT-специалистов. URL: [https://www.dp.ru/a/2020/01/24/Katastroficheskiy\\_deficit](https://www.dp.ru/a/2020/01/24/Katastroficheskiy_deficit) (дата обращения: 3.05.2020)
2. Морозова А. «IT-специалистов действительно не хватает». Эксперты рассказали о спросе на IT-специалистов. URL: <https://vc.ru/hr/107883-it-specialistov-deystvitelno-ne-hvataet-eksper-ty-rasskazali-o-sprose-na-it-specialistov> (дата обращения: 3.05.2020)
3. Возможно ли обучить хорошего программиста в условиях вуза в России. URL: <https://habr.com/ru/post/530294/> (дата обращения: 3.05.2020)
4. Честный рассказ об образовании в IT. Стоит ли оно того? URL: <https://habr.com/ru/post/511584/> (дата обращения: 4.05.2020)
5. Коробцов А.С. Качество инженерного образования: лозунги и реальность // Инженерное образование. – 2020. – № 27. – С. 27–36.
6. Мы поступили в универ и сами показали преподам, как учить студентов. Теперь собираем самые большие аудитории. URL: <https://habr.com/ru/company/ruvds/blog/521218/> (дата обращения: 6.05.2020)

7. Эксперименты с тиграми и другие способы преподавать программирование студентам, которым скучно. URL: <https://habr.com/ru/company/croc/blog/526200/> (дата обращения: 6.05.2020)
8. Необразованная молодёжь. Ответ преподавателя-совместителя. URL: <https://habr.com/ru/post/339208/> (дата обращения: 7.05.2020)
9. Университеты для будущей элиты: 100 лучших российских вузов по версии Forbes–2020. URL: <https://www.forbes.ru/obshchestvo/403369-universitety-dlya-budushchey-elity-100-luchshih-rossiyskih-vuzov-po-versii-forbes> (дата обращения: 7.05.2020)
10. Казун А.П., Пастухова Л.С. Практики применения проектного метода обучения: опыт разных стран // Образование и наука. – 2018. – № 20 (2). – С. 32–59. DOI: 10.17853/1994-5639-2018-2-32-59
11. Burnik U., Košir A. Industrial product design project: building up engineering students' career prospects // Journal of Engineering Design. – 2017. – Vol. 28. – Issue 7–9. – P. 549–567. DOI: 10.1080/09544828.2017.1361512
12. Mo J.P.T., Tang Y.M. Project-based learning of systems engineering V model with the support of 3D printing // Australasian Journal of Engineering Education. – 2017. – Vol. 22. – Issue 1. – P. 3–13.
13. Ye C., Van Os J., Chapman D., Jacobson D. An Online Project-Based Competency Education Approach to Marketing Education // Journal of Marketing Education. – 2017. – Vol. 39. – Issue 3. – P. 162–175. DOI: 10.1177/0273475317724843
14. Yamin Y., Permanasari A., Redjeki S., Sopandi W. Application of Model Project Based Learning on Integrated Science in Water Pollution // Journal of Physics: Conference Series. – 2017. – Vol. 895. – Issue 1. – P. 1–8. DOI: 10.1088/1742-6596/895/1/012153
15. Сычёва С.М., Арзуманова Р.А. Проектное обучение – ключ к подготовке успешного специалиста // Вестник университета. – 2019. – № 6. – С. 32–37. DOI: 10.26425/1816-4277-2019-6-32-37
16. Балтина А.М. Проектное обучение в магистратуре как метод создания инноваций // Вестник Саратовского государственного социально-экономического университета. – 2017. – № 2 (66). – С. 8–11.
17. Гансуар К. Др., Неретина Е.А., Корокошко Ю.В. Опыт проектно-ориентированного обучения и организации командной работы студентов вуза // Интеграция образования. – 2015. – Т. 19. – № 2. – С. 22–30. DOI: 10.15507/Inted.079.019.201502.022
18. Трищенко Д. А. Проектное обучение в вузе: направления поиска внешнего заказчика // Вестник Московского государственного областного университета. Серия: Педагогика. – 2020. – № 2. – С. 105–115. DOI: 10.18384/2310-7219-2020-2-105-115
19. Трищенко Д.А. Опыт проектного обучения: попытка объективного анализа достижений и проблем // Образование и наука. – 2018. – № 20 (4). – С. 132–152. DOI: 10.17853/1994-5639-2018-4-132-152
20. Codewars. URL: <https://www.codewars.com> (дата обращения: 10.05.2020)
21. LeetCode. URL: <https://leetcode.com> (дата обращения: 10.05.2020)
22. Stepik. URL: <https://stepik.org> (дата обращения: 10.05.2020)
23. Яндекс.Контест: URL: <https://contest.yandex.ru> (дата обращения: 10.05.2020)
24. GitHub. URL: <https://github.com> (дата обращения: 11.05.2020)
25. Trello. URL: <https://trello.com> (дата обращения: 11.05.2020)

Дата поступления: 04.07.2020



UDC 378.147

## TEAM PROJECT-ROLE PROGRAMMING LEARNING EXPERIENCE

**Stanislav A. Chernyshev<sup>1,2</sup>,**

Ph.D. (Engineering), Associate Professor of the Department of Intelligent Systems and Information Security; Associate Professor of the Department of Informatics, chernyshev.s.a@bk.ru

<sup>1</sup> Saint Petersburg State University of Industrial Technologies and Design,  
18, Bolshaya Morskaya str., Saint-Petersburg, 191186, Russia

<sup>2</sup> Saint Petersburg state university of economics,  
21, Sadovaya str., Saint-Petersburg, 191023, Russia.

The paper describes team project-role learning programming experience to students. The purpose this approach is teach students to work on varying complexity projects as part of a team, while each of its members performs one of additional roles: manager, software architect, or tester. These forms understanding of students of the minimum team composition required for the development of software products and systems. How make distribution of responsibilities, the process and the importance of writing a technical assignment and acceptance test methodology. What the tools used for project management and etc. For track team progress on the project author used GitHub and Trello, as well as weekly presentations on the project of each of the teams. Checking the compliance of the project code base and the state of the team interaction environment is carried out during the presentation. Use such tools makes it possible to evaluate both the team as a whole and each student. Also discusses current advantages, disadvantages and potential problems that will have to be faced when using this approach to teaching students to program. Implemented projects can be used as a portfolio when looking for a job, or as a basis for final qualifying work. Among existing disadvantages stands out the requirement for the level of a teacher competence. It directly affects result of using the team project-role learning programming.

**Keywords:** team project-role learning, project, programming, project management, education

### REFERENCES

1. Alekseyev V. *Katastroficheskiy defitsit. Tsifrovomu proryvu predrekli ostruyu nekhvatku IT-spetsialistov* [Catastrophic deficit. The digital breakthrough was predicted by an acute shortage of IT specialists]. Available at: [https://www.dp.ru/a/2020/01/24/Katastroficheskiy\\_deficit](https://www.dp.ru/a/2020/01/24/Katastroficheskiy_deficit) (accessed 3.05.2020)
2. Morozova A. «*IT-spetsialistov deystvitelno ne khvatayet*». *Eksperty rasskazali o sprose na IT-spetsialistov* [«IT specialists are really not enough». The experts spoke about the demand for IT specialists]. Available at: <https://vc.ru/hr/107883-it-specialistov-deystvitelno-ne-hvataet-eksperty-rasskazali-o-sprose-na-it-specialistov> (accessed 3.05.2020)
3. *Vozmozhno li obuchit khoroshego programmista v usloviyakh VUZa v Rossii* [Is it possible to train a good programmer in a university in Russia]. Available at: <https://habr.com/ru/post/530294/> (accessed 3.05.2020)
4. *Chestnyy rasskaz ob obrazovanii v IT. Stoit li ono togo?* [An honest story about education in IT. Is it worth it?] Available at: <https://habr.com/ru/post/511584/> (accessed 4.05.2020)
5. Korobtsov A.S. The quality of engineering education: slogans and reality. *Engineering Education*. 2020, no. 27, pp. 27–36. In Russ.
6. *My postupili v univer i sami pokazali prepodam, kak uchit' studentov. Teper sobirayem samyye bolshiye auditorii*. [We entered the university and showed teachers how to teach students. Now we are collecting the largest audiences] Available at: <https://habr.com/ru/company/ruvds/blog/521218/> (accessed 6.05.2020)
7. *Eksperty s tigrami i drugiye sposoby prepodavat programmirovaniye studentam, kotorym skuchno*. [Experimenting with tigers and other ways to teach programming to bored students.] Available at: <https://habr.com/ru/company/croc/blog/526200/> (accessed 6.05.2020)
8. *Neobrazovannaya molodozh. Otvet prepodavatelya-sovmestitelya*. [Uneducated youth. The answer of the part-time teacher.]. Available at: <https://habr.com/ru/post/339208/> (accessed 7.05.2020)
9. *Universitety dlya budushchey elity: 100 luchshikh rossiyskikh vuzov po versii Forbes–2020*. [Universities for the Future Elite: Forbes Top 100 Russian Universities – 2020.] Available at: <https://www.forbes.ru/obshchestvo/403369-universitety-dlya-budushchey-elity-100-luchshih-rossiyskih-vuzov-po-versii-forbes> (accessed 7.05.2020)
10. Kazun A.P., Pastukhova L.S. Praktiki primeneniya proyektного metoda obucheniya: opyt raznykh stran [Practices of using the project-based teaching method: the experience of different countries]. *Obrazovaniye i nauka*. 2018, no. 20 (2), pp. 32–59. DOI: 10.17853/1994-5639-2018-2-32-59

11. Burnik U., Košir A. Industrial product design project: building up engineering students' career prospects. *Journal of Engineering Design*. 2017, vol. 28, iss. 7–9, pp. 549–567. DOI: 10.1080/09544828.2017.1361512
12. Mo J.P.T., Tang Y.M. Project-based learning of systems engineering V model with the support of 3D printing. *Australasian Journal of Engineering Education*. 2017, vol. 22, iss. 1, pp. 3–13.
13. Ye C., Van Os J., Chapman D., Jacobson D. An Online Project-Based Competency Education Approach to Marketing Education. *Journal of Marketing Education*, 2017, vol. 39, iss. 3, pp. 162–175. DOI: 10.1177/0273475317724843
14. Yamin Y., Permanasari A., Redjeki S., Sopandi W. Application of Model Project Based Learning on Integrated Science in Water Pollution. *Journal of Physics: Conference Series*. 2017, vol. 895, iss. 1, pp. 1–8. DOI: 10.1088/1742-6596/895/1/012153
15. Sycheva S.M., Arzumanova R.A. Proyektnoye obucheniye – klyuch k podgotovke uspehnogo spetsialista [Project education is the key to the training of a successful specialist]. *Vestnik universiteta*. 2019, no. 6, pp. 32–37. DOI: 10.26425/1816-4277-2019-6-32-37
16. Baltina A.M. Proyektnoye obucheniye v magistrature kak metod sozdaniya innovatsiy [Project-based learning in master's studies as a method for creating innovations]. *Vestnik Saratovskogo gosudarstvennogo sotsial'no-ekonomicheskogo universiteta*. 2017, no. 2 (66), pp. 8–11.
17. Ganseuer C. Dr., Neretina E. A., Korokoshko Yu. V. Opyt proektno-orientirovannogo obucheniya i organizacii komandnoj raboty studentov vuza [Experience of project-oriented learning and organisation of teamwork among university students]. *Integracija obrazovanija*. 2015, vol. 19, no. 2, pp. 22–30. DOI: 10.15507/Inted.079.019.201502.022
18. Trishchenko D.A. Proyektnoye obucheniye v vuze: napravleniya poiska vneshnego zakazchika [Project education in the university: directions of searching for an external customer]. *Vestnik Moskovskogo gosudarstvennogo oblastnogo universiteta. Seriya: Pedagogika*. 2020, no. 2, pp. 105–115. DOI: 10.18384/2310-7219-2020-2-105-115
19. Trishchenko D.A. Opyt proyektnogo obucheniya: popytka obyektivnogo analiza dostizheniy i problem [Project-based learning experience: an attempt at an objective analysis of achievements and problems]. *Obrazovaniye i nauka*. 2018, no. 20 (4), pp. 132–152. 10.17853/1994-5639-2018-4-132-152
20. Codewars. Available at: <https://www.codewars.com> (accessed 3.05.2020)
21. LeetCode. Available at: <https://leetcode.com> (accessed 3.05.2020)
22. Stepik. Available at: <https://stepik.org> (accessed 3.05.2020)
23. Yandeks.Kontest [Yandex.Contest]. Available at: <https://contest.yandex.ru> (accessed 3.05.2020)
24. GitHub. Available at: <https://github.com> (accessed 3.05.2020)
25. Trello. Available at: <https://trello.com> (accessed 3.05.2020)

Received: 04.07.2020