

5. Фаулер М. Архитектура корпоративных программных приложений. М.: И.Д. Вильямс, 2008. 544 с.
6. Фаулер М. Рефакторинг: улучшение существующего кода. СПб.: Символ-Плюс, 2009. 432 с.
7. Физерс М. Эффективная работа с унаследованным кодом. М.: И.Д. Вильямс, 2009. 400 с.
8. Эккель Б. Философия Java. Библиотека программиста. 4-е изд. СПб.: Питер, 2010. 640 с.

ИСПОЛЬЗОВАНИЕ ГРУППОВЫХ ПРОЕКТОВ В УЧЕБНОМ ПРОЦЕССЕ

В.Ю. Пирогов, зав. кафедрой прикладной информатики,
электронная почта: pirogov@shadrinsk.net
ФГБОУ «ШГПИ», Шадринск, Россия

Аннотация

В статье рассматриваются различные аспекты создания проектов информационных систем в контексте процесса обучения. В статье содержится анализ моделей групповых проектов, характеристики этих моделей и некоторый опыт автора по использованию этой формы учебной работы.

Ключевые слова: *информационные системы, базы данных, программное обеспечение, жизненный цикл, дипломный проект, проектная группа, рабочая группа, промежуточный контроль.*

Abstract

This paper discusses different aspects of projection of information systems in the context of the learning process. The paper contains analysis of models of group projects, the characteristics of these models and some experience of the author in using of this form of teaching work.

Keywords: *information system, database, software application, life cycle, degree project, group project, working group, intermediate control.*

Общие замечания

Под термином «проект» в данной статье будем понимать целенаправленную деятельность по созданию программного обеспечения (далее ПО) или, в более узком смысле, информационной системы (далее ИС). Проект является важнейшим элементом учебного процесса по инженерным специальностям, так или иначе связанным с программированием и базами данных (далее БД). Именно в проекте студент на практике может понять особенности реальной разработки ПО, те сложности и проблемы, о которых говорится на лекциях, но которые могут быть усвоены только при создании реального (пусть в значительной степени и учебного, т.е. облегченного) продукта. Примером может служить в частности влияние структуры БД на создаваемое ПО. Особенности этого влияния, первичность структуры БД по отношению к ПО, хорошо усваи-

вается при создании достаточно больших проектов. К большим проектам мы относим в частности проекты, создаваемые в рамках написания выпускной квалификационной работы (далее ВКР), о чем мы также будем говорить в статье.

О моделях проектирования

Очень часто даже весьма подготовленный программист плохо знает различные этапы проектирования ПО, полагая, что главное в разработке ПО это программирование. Иногда не совсем просто переубедить его в этом вопросе. Использование проектов при обучении, позволяет показать студентам все этапы проектирования (этапы жизненного цикла, см. [1]): анализ и исследование, проектирования системы, программирование, тестирование¹, внедрение и эксплуатация. Перечисленные этапы могут состоять также из некоторого количества мелких разделов. Например, в первом этапе можно выделить разделы: сбор информации, систематизация, формулирование и анализ требований, составление технического задания и др. Особенности этих мелких разделов можно досконально понять только в сочетании теоретических знаний и практических навыков, которые как раз можно получить в работе над проектом.

Важным вопросом использования учебных проектов является выбор модели проектирования или модели жизненного цикла. Эмпирически сложилась практика использования некоторого гибрида – сочетание каскадной и циклической модели. Это чисто учебная практика, связанная, во-первых с тем, что в качестве заказчиков выступает преподаватель, а во вторых с тем, что студенты могут иметь разную подготовку, что приводит, часто, к необходимости корректировки требований в процессе работы над проектом².

Опыт использования проектов

Использование проектного подхода обучения на кафедре прикладной информатики и экономики началось несколько лет назад³ и используется при обучении дисциплинам, связанным с информационными системами. В частности к ним можно отнести такой предмет как Проектирование информационных систем, а также серию дисциплин, которая может быть названа, как «Информационные системы в конкретной области знаний»: Информационные системы в средних образовательных учебных заведениях, Информационные системы в

¹ Этап тестирования, часто не выделяют, как самостоятельный этап, относя его частично к этапу программирования, а частично к этапу внедрения. Тестирование можно также рассматривать и как процесс, параллельный основным процессам проектирования.

² При работе над проектами ВКР несколько иная ситуация и мы рассмотрим ее в разделе «Проекты в выпускных квалификационных работах».

³ Большую работу проводили в частности ст. преподаватели кафедры С.Ю. Парфенов, М.В., Балахнов, Е.В., Осокина. Е.В. Осокиной была подготовлена и защищена диссертация по теме коллективного проектирования [2].

дошкольных образовательных учебных заведениях, Информационные системы в налогообложении, Банковские информационные системы и т.д. Использование проектов по той или иной инженерной дисциплине показало, что студенты к концу работы над проектом начинают довольно профессионально разбираться в различных вопросах, связанных с разработкой ИС. В частности это касается понимания разных этапов разработки информационных систем. Опыт показал, что после успешного участия в подобных проектах у студентов пятых курсов уже не вызывает больших затруднений корректное составление технического задания проекта для ВКР.

Конечно, использование учебных проектов целесообразно только на старших курсах (не ранее третьего). Это требование вытекает просто из необходимости овладения студентами знаниями и умениями, требуемыми при разработке ИС (программирование, проектирование БД, знание языков БД и т.д.).

Основные принципы проектирования ИС

Учебные проекты отличаются от реальных проектов, прежде всего тем, что итогом работы является не только создаваемая информационная система, которая, в частности, не всегда оказывается готова к установленному сроку, а некоторый обобщенный результат, связанный, прежде всего с оценкой работы группы. Остановимся на некоторых ключевых моментах, связанных с организацией учебного проектирования.

Выбор задачи

Выбор темы разработки осуществляется при совместном обсуждении преподавателя и уже сформированной группы разработчиков. Обычно предварительно группе дается время на обдумывание возможной темы. В процессе такого обдумывания часто формируется и сама группа. Если группа не сформулировала тему, то преподаватель предлагает несколько тем на выбор. Обычно тема разработки касается автоматизации той или иной профессиональной деятельности: поликлиника, больница, деканат, отдел кадров и т.д. В процессе обсуждения оговаривается также предварительный объем работы и используемые для разработки инструменты (языки, системы программирования, сетевые технологии, СУБД, сервера и т.д.). Не вводится никакого ограничения на средства и методы разработки. Тут главным критерием является хорошее знание инструментария разработчиками¹. Важно провести оценку способности группы справиться с поставленной задачей. Последующий промежуточный контроль позволяет, впрочем, скорректировать объем работы и требования к создаваемой системе.

¹ Хорошее владение разработчиками используемыми инструментами и технологиями является важным условием и в реальных разработках.

Еще одним критерием выбора инструментария разработки является его соответствие поставленным задачам. Например, для создания динамически-генерируемых html-страниц целесообразнее выбрать какой либо из известных скриптовых языков: php, Perl, Python и т.п., а не С++ или Delphi. Выбор же СУБД обычно определяется теми СУБД, которые участники проекта изучили. Впрочем, бывают и исключения, когда участники проекта предлагали использование незнакомых им инструментов, чтобы в процессе разработки изучить эти инструменты. Такой подход можно только приветствовать, если, конечно, участники проекта обладают достаточной квалификацией и упорством, чтобы справиться с поставленной задачей.

Количество членов рабочих групп и распределение ролей

Опыт использования учебного коллективного проектирования показывает, что оптимальное количество членов группы составляет три человека. В некоторых случаях возможны отклонения в ту или другую сторону, но не более чем на одного человека. Большие группы трудно управляемы, в результате в них часть студентов может оказаться не удел и отследить это не всегда удается. Стандартное распределение ролей для группы из трех человек при создании ИС с клиент-серверной архитектурой такое: один человек – разработка клиентской части приложения, один человек – разработка серверной части (на стороне СУБД), один человек – документация и тестирование. Кроме того, последний член группы может также выполнять функции координатора, связующего звена между группой и преподавателем, который играет роль заказчика. Кроме этого, часть работы группа должна выполнять коллективно: анализ предметной области, составление технического задания, проектирование базы данных, проектирование пользовательского интерфейса. Это очень важное условие, отражающее основной принцип такого проектирования: каждый отвечает за свой фронт работы, но хорошо понимает, чем занимаются другие, и имеет хорошее представление о системе в целом. Конечно, при разработке члены группы разработки общаются друг с другом, кто-то обращается за помощью к другому, иногда что-то делают совместно. С учебной точки зрения это совсем не плохо, и даже должно приветствоваться, хотя с точки зрения общих принципов разработки ПО это не совсем корректно¹.

Промежуточный контроль

¹ Одним из принципов разработки является принцип ортогональности, требующий максимальной изолированности разработчиков (групп разработчиков) друг от друга. Каждый должен отвечать за свой фронт работы. Программисты, например, не должны заниматься тестированием системы.

Важнейшим элементом коллективной разработки проектов является промежуточный контроль. Промежуточный контроль, обычно осуществляется на лабораторных занятиях. При обычной разработке промежуточный контроль позволяет оценить сроки, ресурсы и подкорректировать требования. При выполнении учебных проектов промежуточный контроль играет важную учебную роль. Иногда при таком контроле выясняется, что поставленная задача слишком сложна для данного состава группы. И здесь можно принять разные решения. Например, можно переформировать группы. Но чаще всего переформулируются требования в сторону упрощения задачи. Иногда выявляются общие проблемы: проблемы, с которыми встретились сразу несколько групп разработчиков. Тогда эти проблемы выносятся на семинарское обсуждение или же рассматриваются на лекции. Во время промежуточного контроля преподаватель проводит также индивидуальное консультирование.

Защита проектов

Выставление оценок за проекты проводится по определенной процедуре. Собирается комиссия из трех-четырех преподавателей, а каждая группа разработчиков готовит не большой доклад-презентацию. После представления, проект рассматривается более детально с выяснением вклада каждого из участников. Проверяется:

А) выполненная работа;

Б) компетентность в той части, которую выполнял конкретный член команды;

В) компетентность в других частях работы выполненной работы.

Оценка выставляется после совещания членов комиссии. Тут важна значимость всего мероприятия. Если это достигнуто, то оно (мероприятие) влияет не только тех, кто в данный момент защищал свои работы, но и на тех, кто в будущем будет это делать.

Важным моментом защиты проектов является четко сформулированные требования к проекту и критерии оценки. Требования к проекту формулируются при выборе темы проекта и определении состава проектной группы. Впоследствии при промежуточном контроле требования могут измениться. Критерии оценки должны быть известны участникам проектов заранее и не меняется в течение всего срока разработки. Это может быть набор критериев к интерфейсу пользователя, к технологии взаимодействия клиентской и серверной части, например, использовании хранимых процедур. Важным критерием, который остается неизменным уже в течение многих лет, является доскональное знание

разработанного автором¹ материала (программного кода, структуры базы данных, системы тестов и т.п.).

Проекты в выпускных квалификационных работах

Использование групповых проектов в выпускных работах студентов обладает рядом специфических особенностей. Традиционно выпускная квалификационная работа по инженерным специальностям должна решать реальную практическую задачу, т.е. являться реальным проектом, у которого есть реальный заказчик. Организация группового проектирования в таком случае может оказаться сложной задачей. С другой стороны сама задача может быть достаточно объемной и требовать именно групповой разработки.

На практике были апробированы две схемы групповой разработки выпускных проектов. Условно их можно назвать со слабым и сильным внешним контролем. Под внешним контролем будем понимать контроль со стороны руководителя или со стороны заказчика.

Проект с сильным внешним контролем

Данный способ организации работы предполагает, что руководитель (ВКР) или заказчик непосредственно занимается координацией работы группы. В этом случае руководитель ВКР или представитель от заказчика, по сути, становятся руководителями проектной группы. Такой руководитель должен

1. Принимать участие в разработке проекта и в его календарном планировании.
2. Быть постоянно в курсе сделанной каждым из участников работы.
3. Осуществлять при необходимости корректировку работы каждого из участников.

Использование такого типа контроля целесообразно применять при наличии жестких требований со стороны заказчика, которые к тому же, могут меняться в течение срока разработки.

Проект со слабым внешним контролем

Такой способ организации предполагает, что в группе проектировщиков должен быть выбран свой координатор. В задачу его будет входить

1. Участие в анализе и проектировании ИС.
2. Общее планирование работы, в том числе календарное планирование.
3. Отслеживания объема сделанных работ.
4. Связь с заказчиком и руководителем ВКР. Такая связь необходима, так как в процессе работы над проектом могут возникнуть различные обстоятельства, требующие вмешательства и принятия решений на более высоком уровне.

¹ По сути, проверка на авторство.

Наличие такого координатора может являться также рычагом, с помощью которого заказчик или руководитель ВКР может воздействовать на ход выполнения проекта.

Использование данного типа контроля может быть эффективно в первую очередь в том случае, если в группе разработчиков имеется лидер, способный выполнять функции координатора.

Распределение ролей в проекте

При распределении ролей и формулировке задач в группе следует избегать возникновения простоя отдельных участников проекта. Например, проектирование базы, при неумелой организации работы проектной группы, может затормозить работу члена группы, занимающегося разработкой клиентской части ИС. В действительности здесь нет никаких противоречий, поскольку тот же интерфейс пользователя может разрабатываться в отсутствие базы данных. Еще один элемент работы группы – тестирование. Член группы, отвечающий за тестирование, должен начать работу над тестами еще на самом раннем этапе разработки. Процесс тестирования, таким образом, можно организовать как параллельный другим разработкам процесс.

ЛИТЕРАТУРА

1. Пирогов, В.Ю. Информационные системы и базы данных: организация и проектирование [Текст] : учебное пособие / В.Ю. Пирогов. – СПб. : ВНУ-Санкт-Петербург, 2009. – 528 с.

2. Осокина, К.В. Использование метода коллективного проектирования при обучении будущих специалистов в области информационных технологий разработке информационных систем [Текст]: диссертация на соискание учёной степени кандидата педагогических наук : 13.00.02 / К.В. Осокина. : Рос-сийский государственный профессионально-педагогический университет. – Екатеринбург, 2011. – 171 с.

СОЗДАНИЕ ИГРОВЫХ ПРОЕКТОВ КАК СРЕДСТВО ОБУЧЕНИЯ СТУДЕНТОВ ПРОГРАММИРОВАНИЮ

П.В. Родченков

Научный руководитель: А.И. Газейкина, к.п.н., доцент
ФГБОУ ВПО «УрГПУ», Екатеринбург, Россия

Аннотация

Данная статья посвящена исследованию углублённого обучения программированию студентов младших курсов с помощью создания игровых программных продуктов: обсуждаются проблемы выбора языка программирования и методи-