

ОБУЧЕНИЕ СТУДЕНТОВ РЕШЕНИЮ ЗАДАЧ УПРАВЛЕНИЯ УЧЕБНЫМ ПРОЦЕССОМ СРЕДСТВАМИ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Кафедра информатики.

Научный руководитель - Е. В. Баранова

Применение средств информационных технологий (ИТ) в управлении процессом обучения в школе - одно из основных направлений подготовки современных учителей информатики¹. Для этого учитель информатики должен уметь разрабатывать новые и (или) модифицировать ранее созданные информационные системы². Существующая подготовка студентов педвузов по направлению 540200 «Физико-математическое образование» и специальности 030100 «Информатика» обеспечивает формирование знаний и умений в области программирования, необходимых для создания программных средств,³ Однако этих знаний и умений недостаточно. Представляется целесообразным обучать будущих учителей информатики разработке информационных систем (ИС). Данная деятель-

ность включает в себя не только стадию программирования, но и другие этапы «жизненного цикла» программного обеспечения: анализ проблемы, проектирование, внедрение и т. д.

Охарактеризуем основные составляющие методики обучения *разработке* информационных систем управления учебным процессом. Под системой *управления учебным процессом* будем понимать совокупность действий администрации школы и учителей, направленных на организацию учебного процесса, а также учет и хранение результатов процесса обучения. *Управление учебным процессом* предполагает решение ряда *задач*:

- ведение персональной информации (данные об учениках и учителях) и информации учебного назначения;

- ведение учебного плана;
- тематическое планирование уроков;
- распределение учебной нагрузки учителей;
- составление школьного расписания;
- ведение классных журналов.

Решение задач управления учебным процессом средствами ИТ в первую очередь предполагает создание информационных систем, позволяющих автоматизировать или повысить качество выполнения вышеуказанных задач работниками школы. Общеизвестной методологией разработки ИС на сегодняшний момент является *объектно-ориентированная методология*³. Согласно данной методологии процесс разработки включает три основных этапа: а) анализ, б) проектирование, в) программная реализация. Соответственно, процесс обучения созданию информационных систем также будет состоять из трех основных этапов. Обозначим их следующим образом.

1. Обсуждение результатов анализа задачи управления учебным процессом.
2. Проектирование архитектуры ИС, позволяющей решить данную задачу.
3. Программная реализация ИС.

Охарактеризуем каждый этап обучения: наиболее важные элементы деятельности по разработке ИС и действия, направленные на приобретение студентами новых навыков, знаний и умений.

Этап **анализа** достаточно сложен. Умение анализировать предметную область и определять требования к будущей системе приобретается с опытом. У студентов, естественно, такой опыт минимален. Поэтому преподаватель должен сам предварительно провести анализ разрабатываемой информационной системы. Подготовленные материалы позволяют управлять ходом рассуждений студентов при обсуждении задачи и анализе информационных средств ее решения.

Основная цель на этапе **анализа** – получение модели требуемого поведения системы⁴. Модель должна быть представлена графически (на диаграммах) и в текстовом виде (документация системы). В ходе обучения не-

обходимо сосредоточиться на построении диаграмм. Это вызвано тем, что, как правило, процесс обучения прерывается во времени (1-2 занятия через несколько дней, неделю перерыва). Студент неизбежно забывает часть промежуточных результатов своей работы. *Построение модели в графическом виде* в ущерб разработке документации призвано, помимо достижения целей разработки системы на этапе анализа, решить еще одну задачу: сделать работу студента более наглядной, сформировать у него четкое представление о результате будущей работы – образе готовой ИС.

Первый этап обучения начинается с описания задачи. Студенты при помощи преподавателя определяют **субъекты** (людей и внешние информационные системы), задействованные в рассматриваемой задаче управления учебным процессом, основные **функции**, которые данные субъекты выполняют в контексте задачи; устанавливают примерный перечень **действий** и их последовательность при выполнении **субъектом** определенной функции.

На основе данных описания задачи преподаватель обсуждает со студентами основную **цель** создания системы. Как правило, использование средств ИТ в любой деятельности человека преследует следующие цели: облегчение труда человека и повышение качества работы. Поставленная цель определяет характер и сложность будущей системы. В дальнейшем цель должна быть конкретизирована – студенты определяют основные **требования** к системе, ее будущие характеристики. Для того чтобы определить эти требования, необходимо ответить на вопрос: «Что система должна выполнять, чтобы достичь поставленной цели?» Цели и основные требования к системе студенты описывают в текстовом файле (документации).

После описания задачи, формулирования **цели и требований к системе студенты** осуществляют анализ *сценариев* ее работы. Под *сценарием* понимается один из возможных **путей** в графе, отвечающем потоку событий для определенного варианта использования системы⁵. Опишем методику проведения анализа сценариев⁶.

1. Разработчики на основе выделенных *вариантов использования* системы перечисляют сценарии, существенные для работы системы (не углубляясь в детали). Например, для редактора школьного расписания основными сценариями работы будут: составление расписания, проверка расписания, формирование индивидуального расписания учителя и т. д.

2. Основной поток событий в сценарии тщательно прорабатывают, устанавливая, какие объекты участвуют в сценарии, каковы обязанности каждого объекта и как они взаимодействуют в терминах операций. Тем самым четко распределяются области влияния абстракций.

3. Далее сценарии расширяются, чтобы учесть исключительные ситуации и вторичное поведение (альтернативные варианты развития сценария).

Этап **проектирования** предназначен для построения логической и физической структуры будущей системы. Результат проектирования - полная спецификация системы: подробное описание всех ее элементов (отраженное на диаграммах и в документации)⁷. На занятиях выполнить полную спецификацию разрабатываемой программы невозможно из-за недостатка времени, поэтому студенты выполняют проектирование на меньшем уровне детализации, чем в реальном процессе разработки. Проектирование логической структуры сводится к построению иерархии классов, реализующих поведение системы: а) выделение классов; б) выявление их строения (набора свойств) и поведения (набора методов); в) определение связей между классами.

Кандидатами на роль классов являются абстракции, выявленные в ходе *анализа сценариев*. Процесс классификации достаточно сложен. Менее классифицировать объекты и выбирать наиболее существенные критерии для объединения объектов в один класс, как правило, приобретается с опытом разработки информационных систем, поэтому трудно дать какие-либо строгие инструкции для отыскания классов.

Множество классов системы на протяжении ее жизненного цикла изменяется и пополняется. Исходный набор абстракций - кандидатов на роль классов редко совпадает с той совокупностью классов, которые, в конечном счете, реализуются на практике. Студенты должны установить классы как минимум трех типов⁸.

1. Классы *сущностей* - абстракции, представляющие понятия реального мира. Эти классы необходимы для обработки данных, которые будут использованы в работе информационной системы (например, учебный предмет, кабинет, ученик, учебный класс, учитель, время занятий и т. д.).

2. Классы *границ*, моделирующие пользовательский интерфейс системы и ее взаимодействие с внешними системами.

3. Классы *управления*, контролирующие потоки событий, присущих определенному варианту использования (например, гарантируют существование, непротиворечивость и актуальность данных и т. д.).

Следующая задача - построение физической архитектуры системы. Физическая архитектура - это набор *модулей*, которые хранят описания классов, реализующих ее логическую структуру, и *связей* между модулями. Физическую архитектуру необходимо представить на двух уровнях: уровне *реализации* и уровне *выполнения*.

Уровень *реализации* позволяет определить, из каких файлов (модулей) будет состоять система на этапе написания исходного кода программы: заголовочные файлы, исходные файлы библиотек и т. д. На уровне *выполнения* отражается структура уже созданного приложения: исполняемые файлы, подключаемые библиотеки, база данных, файлы настроек и т. д.

Цель этапа **программной реализации** - создание работоспособной версии программы на ранних стадиях и ее последующее развитие. Ожидаемыми результатами обучения студентов на данном этапе являются:

1) формирование навыков групповой работы (выполнение каждым студентом отдельного задания по программированию).

Необходимость приобретения данных навыков обусловлена тем, что профессиональные информационные системы разрабатываются, как правило, командой программистов - одному человеку такая работа не под силу;

2) развитие готовности студента к самостоятельной реализации поставленной перед ним задачи средствами ИТ (в данном случае - при помощи систем программирования).

Предложенный подход ориентирован на подготовку в магистратуре, но может использоваться и при обучении студентов бакалавриата. В первом случае акцент должен быть сделан на этапы анализа и проектирования - развитие навыков программирования у магистрантов играет второстепенную роль. При обучении в бакалавриа-

те использование методов анализа и проектирования способствует овладению студентами технологии объектно-ориентированного программирования.

Одним из важнейших средств обучения разработке ИС являются специализированные среды объектно-ориентированного моделирования⁹. Данные компьютерные оболочки позволяют графически представить модель проектируемой ИС в различных ракурсах (на диаграммах нескольких типов). Включение в учебные планы подготовки учителей информатики содержания обучения системам объектно-ориентированного моделирования - обязательное условие для реализации предлагаемой в статье методики обучения разработке информационных систем.

ПРИМЕЧАНИЯ

¹ Кузнецов Э. И. Общеобразовательные и прикладные аспекты изучения информатики и вычислительной техники в пединститутах. Автореф-т дисс.... д.п.н. М., 1990.

² Государственный образовательный стандарт высшего профессионального образования. Направление 540200 Физико-математическое образование. М., 2000.

³ Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. М., 1999.

⁴ Буч Г., Якобсон А., Рамбоу Дж. UML. Специальный справочник. СПб., 2002.

⁵ Кватрани Т. Визуальное моделирование с помощью Rational Rose 2002 и UML.: Пер. с англ. М., 2003.

⁶ Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++ М., 1999. С. 159.

⁷ Буч Г., Якобсон А., Рамбоу Дж. UML. Специальный справочник. СПб., 2002.

⁸ Кватрани Т. Визуальное моделирование с помощью Rational Rose 2002 и UML.: Пер. с англ. М., 2003.

⁹ Кватрани Т. Визуальное моделирование с помощью Rational Rose 2002 и UML.: Пер. с англ. М., 2003.