

Раздел 3. ИННОВАЦИОННЫЕ ПОДХОДЫ К ПРОФЕССИОНАЛЬНОЙ ПОДГОТОВКЕ И ПЕРЕПОДГОТОВКЕ ПЕРСОНАЛА В ОБРАЗОВАТЕЛЬНЫХ ОРГАНИЗАЦИЯХ РАЗЛИЧНОГО УРОВНЯ

УДК 658.15:331.346

С. А. Власов

S. A. Vlasov

*ФГАОУ ВО «Российский государственный
профессионально-педагогический университет», Екатеринбург*

Russian State Vocational Pedagogical University, Yekaterinburg

sergey.vlasov@rsvpu.ru

ВЫБОР МЕТОДОЛОГИИ ПРОЕКТИРОВАНИЯ ПОДСИСТЕМЫ РЕГИСТРАЦИИ КОМАНДИРОВОЧНЫХ УДОСТОВЕРЕНИЙ CHOOSING A METHODOLOGY FOR DESIGNING A SUBSYSTEM FOR REGISTERING TRAVEL CERTIFICATES

***Аннотация.** Рассматривается методология SADT, которая представляет собой совокупность методов, правил и процедур, предназначенных для построения функциональной модели объекта какой-либо предметной области, может использоваться для моделирования широкого круга систем и определения требований и функций, а затем для разработки системы, которая удовлетворяет этим требованиям и реализует эти функции.*

***Abstract.** The SADT methodology is considered, which is a set of methods, rules and procedures designed to build a functional model of an object in a subject area, can be used to model a wide range of systems and determine requirements and functions, and then to develop a system that meets these requirements and implements these functions.*

***Ключевые слова:** модель; графика блоков и дуг; процесс; хранилище; база данных.*

***Keywords:** model; block and arc graphics; process; storage; database.*

В практической деятельности часто встречаются неоднозначные ситуации с командировочными удостоверениями. Для систематизации таких случаев предлагается выбор методологии проектирования подсистемы регистрации командировочных удостоверений.

1. Методология SADT

Принцип этой методологии основывается на графическом представлении блочного моделирования. Связь между блоками происходит посредством интерфейсных дуг, которые определяют, когда и каким образом функции выполняются и управляются. Применяются электронные таблицы Excel для заполнения бланка командировочного удостоверения и последующего вывода его на печать. В качестве приложения используют журнал регистрации.

Рассмотрим бизнес-процесс формирования документа Командировочное удостоверение. Схема бизнес-процесса представлена на рис. 1.



Рис. 1. Процесс формирования командировочного удостоверения

Таблица 1

Левая и правая половина журнала регистрации удостоверений

№ п/п	ОАО «Уралмаш»	Журнал регистрации командировочных удостоверений				
	ФИО	Должность	Место работы	Место командировки	Предприятие	№ приказа
1	Петров Иван Петрович	Главный инженер	Технический департамент	Москва	ВНИИЖТ	17.04.21 г. № 025-ОК
2	Сидоров Николай Иванович	Начальник техотдела	Технический департамент	Тюмень	ОАО «Газпром»	18.04.21 г. № 036-ОК

№ п/п	ОАО «Вектор»	Журнал регистрации командировочных удостоверений				
	ФИО	Дата и номер командировочного удостоверения	Дата выезда в командировку	Дата прибытия из командировки	Срок командировки, дней	Цель командировки
1	Петров Иван Петрович	17.04.2021 № 25	17.05.2021	18.05.2021	30	Повышение квалификации
2	Сидоров Николай Иванович	18.04.2021 № 036	18.04.2021	30.04.2021	12	Обмен опытом

Для удобства в работе используется закрепление областей рабочей таблицы. Закрепление областей позволяет видеть на экране строку заголовков и два левых столбца, с порядковым номером и фамилией. Для этого выполняется команда «Вид/Закрепить области» табличным курсором.

Далее создают формулы автоматизация заполнения столбца «Дата и номер командировочного удостоверения».

2. Методология объектного проектирования и анализа на языке UML

Построение диаграммы вариантов использования (прецедентов). Use case – это описание сценария поведения, которому следуют действующие лица (Actors) (рис. 2).

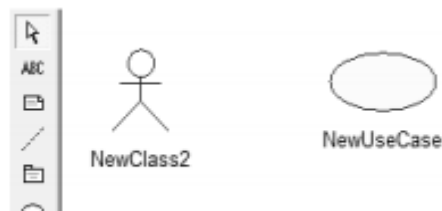


Рис. 2. Создание прецедента

Для создания действующего лица используется кнопка Actor, изображённая на рис. 3.

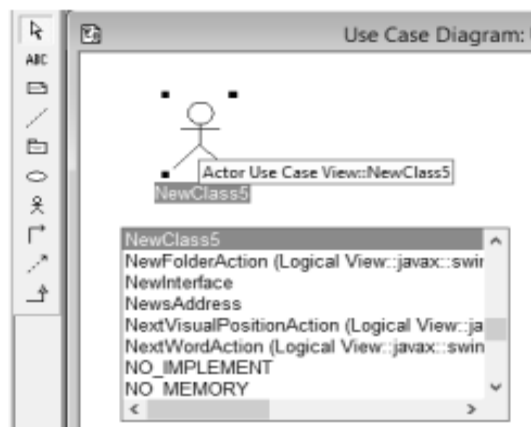


Рис. 3. Создание действующего лица (роли)

Левой кнопкой мыши по изображению действующего лица (актера/роли) в меню открывается «Спецификация» и в поле «Name» добавляется название актера как показано на рис. 4.



Рис. 4. Присвоение имени действующему лицу

При открытии меню «Спецификация» заполняем информацию во вкладках:

- General – имя (Name), стереотип (Stereotype), приоритет (Rank), является ли прецедент абстрактным (Abstract) и текстовое описание прецедента (Documentation);
- Diagrams – диаграммы, содержащие данный прецедент;
- Relations – связи, в которых данный прецедент участвует;
- Files – добавление файлов, содержащих дополнительную информацию о классе.

С помощью кнопки «Unidirectional Association» (однаправленная ассоциация) панели инструментов создается ассоциация (связь) между актером и прецедентом.

Для работы подсистемы регистрации командировочных удостоверений на диаграмме вариантов использования выделены следующие актёры:

- командированный – человек, имеющий доступ к редактированию данных своей карточки командировочного удостоверения;
- ответственный – человек, имеющий неограниченный доступ к карточке командировочного удостоверения;
- бухгалтер – человек, который имеет неограниченный доступ к чтению карточек командировочного удостоверения;
- пользователь вне роли – человек неавторизованный в системе и/или пользователь системы, у которого нет доступа к выбранной карточке командировочного удостоверения.

Система предоставляет следующие возможности участникам:

- актёр «Командированный» использует систему для просмотра информации о карточке командировочного удостоверения, к которой привязан, а также имеет доступ к печати бланков и загрузке сканов оригиналов;
- актёр «Ответственный» использует систему для создания карточки командировочного удостоверения актёра «Командированный», отслеживания его передвижения по карточке командировочного удостоверения, имеет доступ к редактированию данных карточки, а также доступ к печати бланков и загрузке сканов оригиналов;
- актёр «Бухгалтер» использует систему для чтения данных карточки командировочного удостоверения актёра «Командированный» и печати готовой карточки командировочного удостоверения;
- актёр «Пользователь вне роли» использует систему для просмотра части информации о карточке командировочного удостоверения актёра «Командированный».

На основании вышеизложенного можно выделить актуальность системы создания выбранной методологии:

1) авторизация – запускается неавторизованным пользователем. Позволяет определённому лицу при правильном наборе логина и пароля авторизоваться в системе и получить доступ к данным карточки в зависимости от своей должности и места в иерархии отношений к карточке;

2) создание карточки – запускается командированным сотрудником или ответственным за него сотрудником. Позволяет создать карточку и занести первичные данные в систему;

3) редактирование карточки – запускается ответственным сотрудником. Позволяет изменить введённые в карточку данные в случае ошибки при её создании;

4) печать бланков – запускается командированным или ответственным за него сотрудником. Позволяет распечатать шаблоны командировочных удостоверений, заполненные первичными данными и распечатывать бланки прибытия и убытия;

5) загрузка сканов – запускается командированным или ответственным за него сотрудником. Позволяет загружать отсканированные копии заполненных бланков командировочного удостоверения;

6) печать заполненной карточки командировочного удостоверения – запускается бухгалтером или ответственным за командировочного сотрудника. Позволяет распечатать заполненную карточку командировочного удостоверения.

3. Хранилище данных

Хранилище данных информационного ресурса состоит из следующих таблиц:

- таблица «Users» – в ней хранится информация о пользователях системы;
- таблица «Trips» – в ней хранится основная информация, например, о командировочном удостоверении. Кого командировали, кто ответствен за командированного, зачем его командировали, куда командировали, когда и до какого числа, номер документа оригинала, индексы, отсканированная копия оригинала;
- таблица «Mission» – в ней хранится информация о перемещениях пользователя. Куда и когда прибыл, куда и когда убыл, кто поставил метку о прибытии, сканы оригинала бланка прибытия-убытия.

3.1. Построение диаграммы состояний

Построение диаграммы состояний в программном продукте IBM Rational Rose происходит следующим образом:

- создание диаграммы состояний с помощью команд «New – Statechart Diagram»;
- добавление таких компонентов, как State (состояния), Start State (начало), End State (завершение), State Transition (состояние перехода).

На диаграмме имеются два специальных состояния начальное и конечное. Начальное состояние выделяется черной точкой, а конечное черной точкой в белом кружке, оно соответствует состоянию объекта перед его уничтожением. На диаграмме состояний может быть одно начальное состояние. Типовая диаграмма состояний для вариантов использования показана на рис. 5.



Рис. 5. Типовая диаграмма состояний

3.2. Построение диаграммы компонентов

На диаграмме компонентов были выделены следующие объекты:

- а) База данных, характеризующая актуальное состояние предметной области и используется для удовлетворения информационных потребностей пользователей.

б) СУБД метаданные включают в себя системные таблицы (отношения), имена отношений и типы данных этих атрибутов.

в) Компонент – абстрактная единица, которая может выполнять определенные действия, связанные с решением поставленных задач.

Добавление компонента на диаграмму осуществляется с помощью операции главного меню «Tools – Create – Component» или с помощью операции контекстного меню «New – Component».

С помощью панели инструментов добавляем компоненты на диаграмму как показано на рис. 6 и 7.

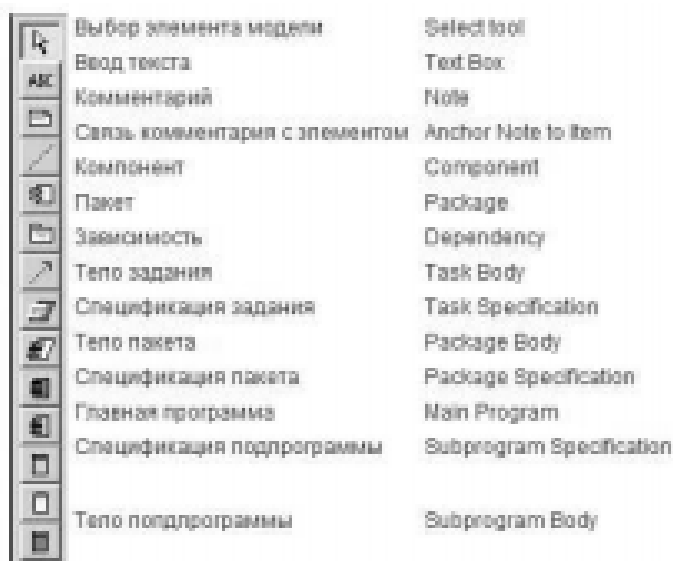


Рис. 6. Описание кнопок панели инструментов диаграммы компонентов в IBM Rational Rose

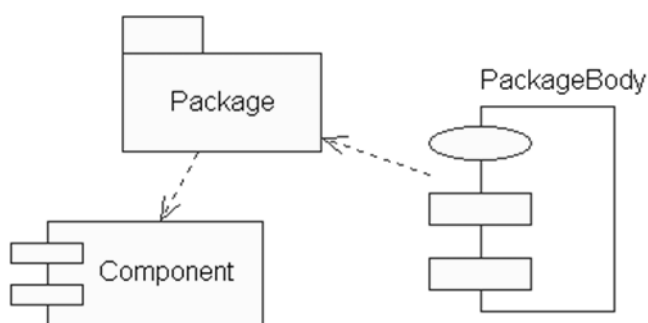


Рис. 7. Диаграмма компонентов после добавления компонента «Главная программа»

Редактирование свойств компонента осуществляется с помощью диалогового окна спецификации свойств.

3.3. Построение диаграммы размещения (развёртывания)

Каждый узел на диаграмме размещения представляет собой образ вычислительного устройства. Основные элементы отображены на рис. 8.

Диаграмма размещения (развёртывания) веб-версии продукта изображена на рис. 9. На ней показано взаимодействие файлов. Файлы *.css и *.js взаимодействуют с клиентом напрямую и хранятся на серверной стороне. Файлы *.php и *.sql также взаимодействуют на серверной стороне, но используют веб-сервер Apache и СУБД PHP My Admin соответственно.








Кнопка	Описание	Название
	Выбор элемента модели	Select tool
	Ввод текста	Text Box
	Комментарий	Note
	Связь комментария с элементом	Anchor Note to Item
	Процессор	Processor
	Соединение	Connection
	Устройство	Device

Рис. 8. Основные элементы для построения диаграммы размещения в IBM Rational Rose

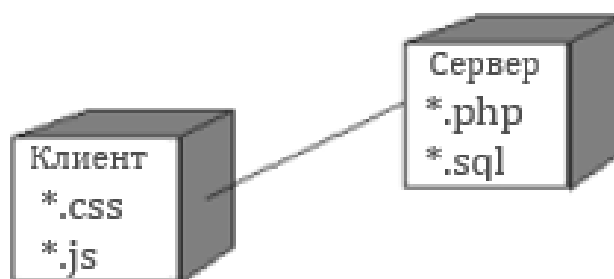


Рис. 9. Диаграмма размещения (развёртывания)

В рассмотренном варианте были выбраны наиболее подходящие методологии для создания диаграмм, отображающих основные компоненты и процессы программного продукта для работы подсистемы регистрации командировочных удостоверений.

Список литературы

1. *Командировочное удостоверение*. Бланк и образец заполнения формы Т-10. – Текст : электронный // Ассистентус : [сайт]. – URL: <https://assistentus.ru/forma/t-10-komandirovochnoe-udostoverenie/>.
2. *Фаулер, М. UML в кратком изложении. Применение стандартного языка объектного моделирования / М. Фаулер, К. Скотт ; пер. с англ. А. М. Вендрова; под ред. Л. А. Калиниченко*. – Москва : Мир, 1999. – 191 с. – ISBN 5-03-003331-9. – Текст : непосредственный.