

УДК 004.78

М. Ю. БАБИЧ, А. Б. ЛЕЩЕНКО*Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Украина*

РАЗРАБОТКА ПОДСИСТЕМЫ УПРАВЛЕНИЯ ПОЧТОВЫМИ ТЕРМИНАЛАМИ

Рассмотрена задача автоматизации доставки почтовых отправлений. Для решения поставленной задачи был проведен анализ особенностей автоматизации доставки почтовых отправлений, рассмотрены существующие системы, особенности их функционирования и размещения. Полученная модель отражает структуру хранимых данных. На этапе выбора инструментальных средств были проанализированы основные платформы для реализации распределенных систем. Для разработки подсистемы обслуживания почтовых терминалов построены UML диаграммы, которые включают диаграммы прецедентов, состояний и классов. Определен комплекс технических и программных средств почтового терминала. Система почтовых терминалов реализована на платформе InterSystems Caché, которая поддерживает создание Web-приложений, стандарт XML, протокол SOAP и Web-сервисы.

Ключевые слова: *информационная система, автоматизация, автоматизированный почтовый терминал, InterSystems Caché.*

Введение

Важным направлением развития служб доставки, как и большинства сфер человеческой деятельности, является автоматизация, которая позволяет улучшить качество предоставляемых услуг и, тем самым, занять более выгодные позиции на рынке предоставления услуг. Примером такой автоматизации является создание сети автоматизированных почтовых терминалов.

Почтовый терминал – это станция приёма и выдачи малогабаритных отправлений: заказов Интернет-магазинов и компаний дистанционной торговли, пакетов документов и отправлений корпоративных клиентов, а также посылок. Почтовый терминал оборудован встроенными ячейками – ящичками разного размера для хранения заказов, и центральной консолью с сенсорным экраном для управления процессом получения и отправки заказов. Количество ячеек может варьироваться от 32 до 256. Терминалы размещаются в общественных местах: в торговых центрах, крупных розничных сетях, супермаркетах. Способ доставки через почтовый терминал является альтернативой курьерским компаниям и почтовым службам при заказе товаров дистанционным путем.

Почтовые терминалы позволяют сократить затраты на «последнюю милю» – самый дорогостоящий этап логистики, который заключается в передаче товара клиенту в руки. Логистические затраты интернет-магазинов сокращаются примерно до 20%, а для других видов дистанционной торговли (каталожной, сетевому маркетингу) – сокращают сроки доставки [1].

Сети почтовых терминалов хорошо распространены в Европе, первый терминал разработала

австрийская фирма «Кeba», в 2004 году он поступил в реализацию, после чего почтовые терминалы стала использовать в своей почтовой сети фирма DHL в Германии. Сегодня в Германии существует целая сеть почтовых терминалов, в которую входит более 4500 тысяч единиц. Благодаря такому широкому распространению технология доставки покрывает более 90% территории Германии и занимает 30% всего рынка доставки [2]. В 2012 преимущества использования уникальных систем оценили в 23 странах мира. Прибалтика, Польша и другие европейские страны приближаются к Германии по распространенности такого сервиса доставки, который успешно конкурирует с обычной почтой и курьерскими службами, доставляющими товары интернет-пользователям. Как показывает статистика, большинство покупателей забирают свои посылки в день доставки. Показатели среднего срока хранения отличаются в разных странах. В России посылки лежат в ячейках 1,7 дня, в странах Прибалтики – 1,6. Данные компании DHL показывают, что с увеличением числа автоматизированных почтовых станций пропорционально возрастает число их клиентов [3]. В настоящее время на рынке почтовых услуг в России [4] успешно работают компании Logibox [5], QIWI Post [6], PickPoint [7, 8]. Поэтому актуальна тема предлагаемой публикации, в которой предлагается подход для автоматизации доставки почтовых отправлений.

Постановка задачи

На основе анализа статистических данных существующих систем можно сделать вывод о необходимости внедрения почтовых терминалов в нашу повседневную жизнь. Разработка сети почтовых терминалов подразумевает проектирование модели

предметной области, на основании которой будет реализована система, которая в свою очередь, позволит автоматизировать процесс выдачи и приема посылок клиентами служб экспресс доставки. При этом разрабатываемая система автоматизации почтовых терминалов должна предоставлять различные услуги для следующих групп пользователей:

1. Клиент (отправка/получение посылки, оплата доставки);
2. Сервисный инженер (локальный доступ к терминалу, управление VPN-соединением);
3. Экспедитор (выгрузка/загрузка посылок);
4. Инкассатор (инкассация терминала).

Результаты

Почтовый терминал реализован на базе промышленного компьютера, к которому через интерфейс RS-232 подсоединены: контроллер ячеек, сканер и принтер штрих-кодов, принтер чеков и купюроприёмник (рис. 1).



Рис. 1. Устройства почтового терминала

Программное обеспечение терминала реализовано на базе СУБД Caché [9 - 11], так как эта система поддерживает создание Web-приложений, стандарт XML, протокол SOAP [12] и Web-сервисы, что позволяет разрабатывать на ней приложения, обладающие высокой производительностью и надежностью (рис. 2).

Использование в качестве сервера базы данных СУБД Caché позволило применить объектно-ориентированный принцип проектирования информационной системы и использовать различные способы доступа к данным.

Все почтовые терминалы работают в общей сети. Для контроля работы почтовых терминалов и для управления ими используется сервер. Взаимодействие почтовых терминалов с сервером осуществляется по протоколу SOAP (Simple Object Access Protocol – простой протокол доступа к объектам на базе языка XML), который определяет правила передачи сообщений по Internet между различными

прикладными системами. С помощью SOAP можно создавать Web-службы настолько же оперативно, насколько быстро будут написаны сообщения SOAP для вызовов программ для существующих приложений, а затем добавить эти приложения в простые Web-страницы. Но, кроме того, есть возможности использовать вызовы SOAP в выделенных приложениях и создавать приложения, которые можно переносить на Web-страницы других разработчиков, и тем самым избежать трудоемкого и дорогостоящего процесса разработки.

Архитектура подсистемы представляет собой трехзвенную модель «клиент - сервер». Почтовый терминал взаимодействует с сервером по средствам протокола SOAP. Сервер обменивается информацией с внешним сервером по протоколу HTTP. Внешний сервер выполняет две функции: формирование штрих-кода почтового отправления и расчёт стоимости доставки. Сервер сети терминалов выполняет множество функций, и является связующим звеном между внешним сервером и терминалом. Архитектура всей системы представлена на рисунке 2.

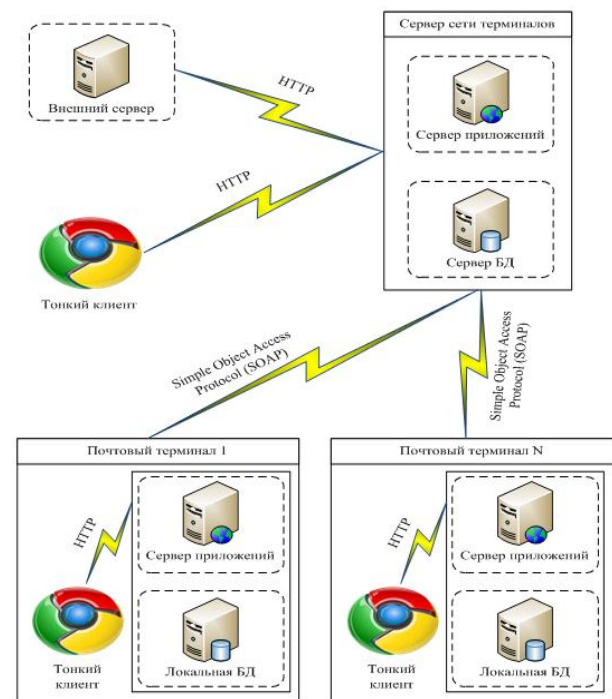


Рис. 2. Архитектура системы

Диаграмма прецедентов для подсистемы почтового терминала представлена на рисунке 3 [13]. Для работы с почтовым терминалом клиент должен авторизоваться. При отправке посылки клиент может оплатить услуги доставки. Если он этого не сделал, то оплачивать будет получатель. Создать товарно-транспортную накладную (ТТН) можно заранее, используя личный кабинет или во время отправки на почтовом терминале.

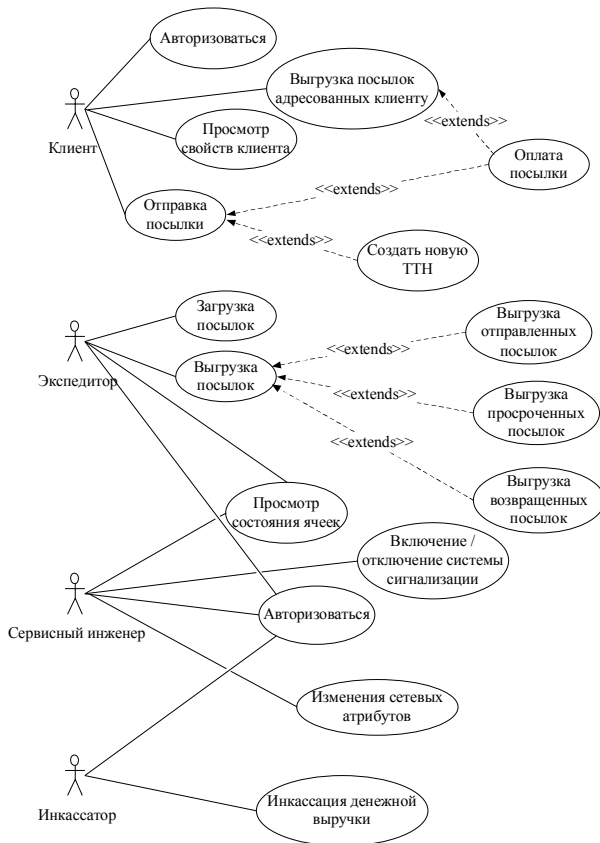


Рис. 3. Диаграмма Use Case

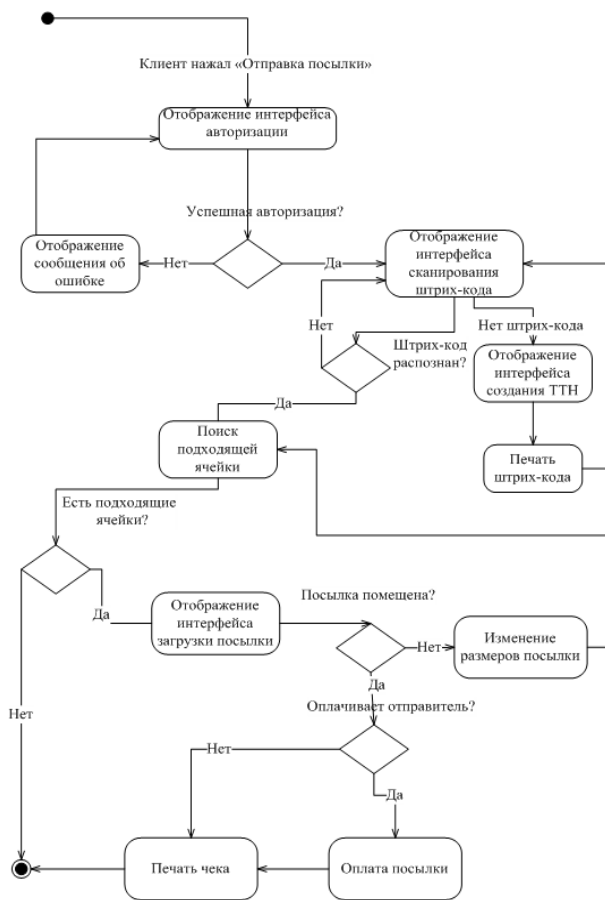


Рис. 4. Отправка посылки клиентом

В обязанности экспедитора входит загрузка и выгрузка посылок с почтового терминала. Посылки для выгрузки бывают трех типов: посылки, загруженные клиентами для отправки; просроченные посылки – те посылки, которые получатель не забрал в течение определенного срока; посылки, качеством которых получатели не удовлетворены и решили вернуть их (для электронной коммерции).

Процесс отправки посылки клиентом показан в виде диаграммы состояний (рис. 4).

Чтобы отправить посылку клиент должен сформировать ТТН (рис. 5). Клиент делает соответствующий запрос к контроллеру. Контроллер создает и отправляет клиенту форму. Форма состоит из нескольких полей. Основные поля – номер получателя, адрес доставки, необходимый размер ячейки и т.д. Клиент заполняет форму и отправляет контроллеру. Контроллер проверяет правильность данных и начинает процесс создания штрих кода. Для обеспечения уникальности штрих кода в рамках системы необходимо выполнять генерацию штрих кода на сервере терминалов. Для этой задачи и предназначен BarCodeWebService. После того, как сервер сгенерировал уникальный штрих код и занес данные о нем в БД, контроллер может распечатать штрих код, чтобы клиент наклеил его на посылку.

Для печати штрих-кодов и чеков используется технология Zen Reports и FOP (Formatting Objects Processor). Zen Reports является высокоуровневым механизмом извлечения данных из базы Caché и преобразования их в xml [14].

Данный способ (рис. 6) предполагает создание класса Zen-отчёта с использованием Caché-студии, наследуемый от %ZEN.Report.reportPage, в котором необходимо заполнить блоки XData ReportDefinition и XData ReportDisplay. FOP – XSL: FO процессор для преобразования документов в формате XSL:FO в PDF, RTF и другие форматы [15].

Как уже было отмечено, сервер сети почтовых терминалов взаимодействует с сервером почтовой службы по протоколу HTTP. В качестве примера приведем фрагмент программного кода, который синхронизирует данные об отделениях для отправки посылок:

```

s wr=##class(%XML.Writer).%New()
s wr.Indent=1
s oStream=##class(%GlobalCharacterStream).%New()
d wr.OutputStream(oStream)
d wr.RootElement("file")
d wr.Element("auth")
d wr.Write(..%GetParameter("Auth"))
d wr.EndElement()
d wr.Element("citywarehouses")
d wr.EndElement()
d wr.EndRootElement()
s req = ##class(%Net.HttpRequest).%New()
s req.ContentType="text/xml"
s req.Server="novaposhta.ua"
    
```

```

d req.EntityBody.CopyFrom(oStream)
d req.Post("xml.php",0)
d ##class(NPPS.Data.Ref.Department.City).%KillExtent()
d ##class(NPPS.Data.Ref.Department.Area).%KillExtent()
s reader = ##class(%XML.Reader).%New()
s status = reader.OpenStream(req.HttpResponse.Data)
if $$$ISOK(status){
Do reader.Correlate("city", "NPPS.API.XML.City")
  While (reader.Next(.object, .status)) {
    &sql(select ID into :regId from
NPPS_Data_Ref_Department.Area
where AreaNameUkr = :object.AreaNameUkr)
    if regId=""{

```

```

s reg = ##class(NPPS.Data.Ref.Department.Area).%New()
s reg.AreaNameUkr = object.AreaNameUkr
d reg.%Save()
s regId = reg.%Id()
}
s region=
##class(NPPS.Data.Ref.Department.Area).%OpenId(regId)
s city = ##class(NPPS.Data.Ref.Department.City).%New()
s city.NameUkr = object.NameUkr
s city.NplD = object.Npld
s city.Region = region
d city.%Save()
q status

```

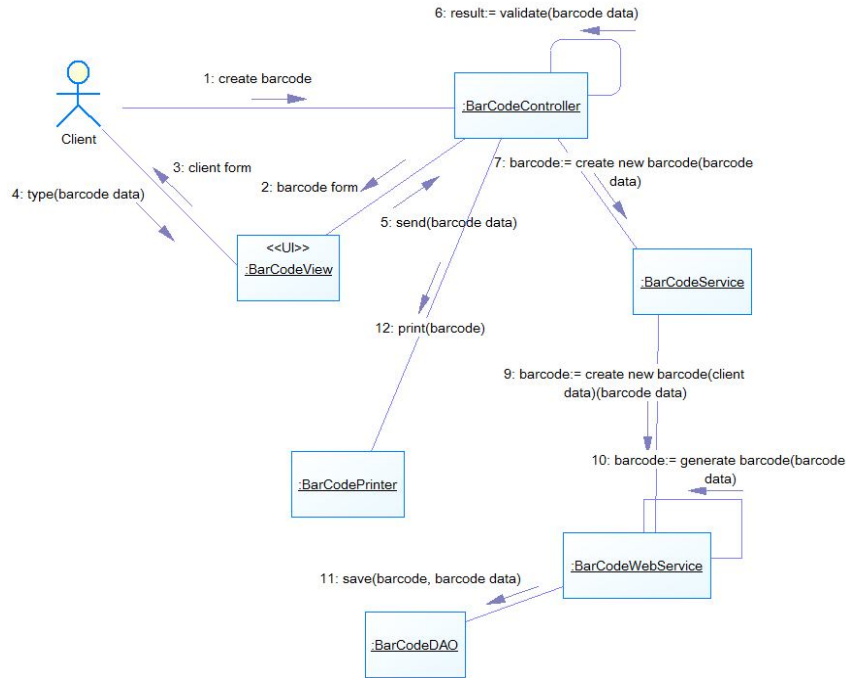


Рис. 5. Создание штрих кода ТТН

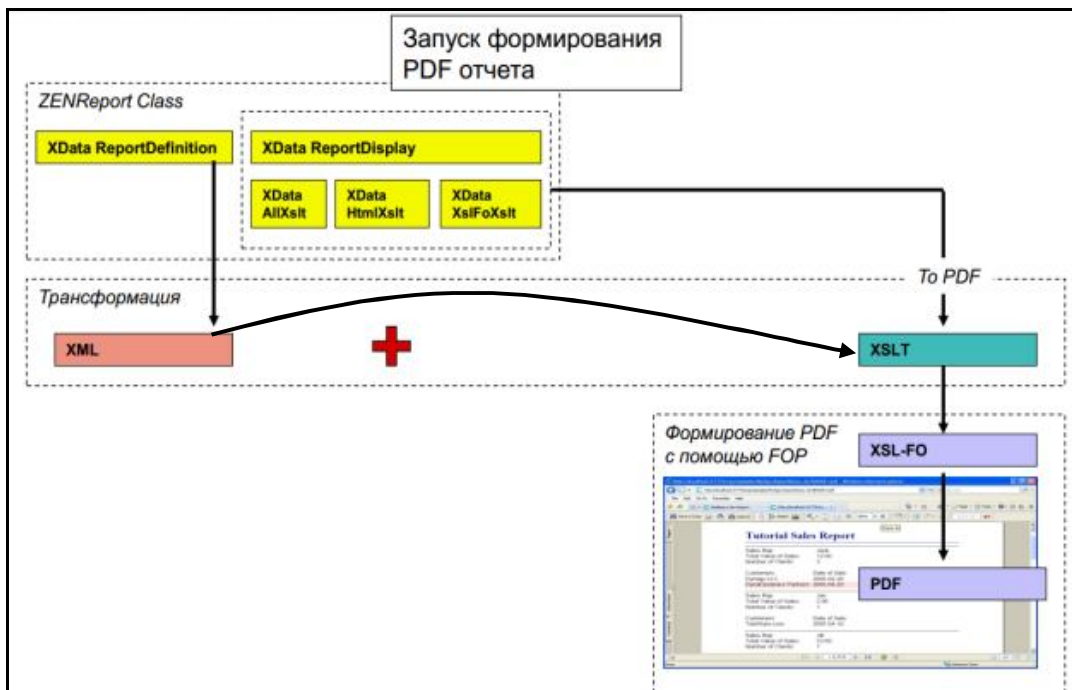


Рис. 6. Формирование отчета классом Zen Reports

Введите последние 5 цифр
товарно-транспортной накладной (ТТН)

Ваше графическое изображение ТТН
указано в поле ввода
(Пароль: 12345678901234)
ПН для отправки:
1969. Дата: 05.11.2014 22:22
www.logibox.ru

1 2 3
4 5 6 ←
7 8 9 0

НАЗАД ПРОДОЛЖИТЬ ГЛАВНОЕ МЕНЮ

Рис. 7. Форма ввода номера ТТН

Выберите ячейку

Страница 1 из 11

#	Номер ячейки	Состояние
1	1	EMPTY
2	2	EMPTY
3	3	EMPTY
4	4	FULL
5	5	EMPTY
6	6	EMPTY
7	7	EMPTY

НАЗАД ПРОДОЛЖИТЬ ГЛАВНОЕ МЕНЮ

Рис. 8. Форма выбора ячейки для загрузки посылкой

Заключение

Разработанная информационная система обслуживания почтовых терминалов, позволяет автоматизировать процесс получения и отправки почтовых отправлений.

При разработке системы обслуживания почтовых терминалов сформированы основные диаграммы UML, которые включают диаграммы прецедентов, состояний и классов. Были выделены виды пользователей подсистемы (клиенты, экспедиторы, инкассаторы и сервисные инженеры) и их возможные действия на почтовом терминале. Проведено моделирование основных процессов, таких как отправка посылки клиентом, загрузка/выгрузка посылок экспедитором и получение посылки клиентом. Кроме того, была описана структура хранимых классов подсистемы.

Литература

1. Напалкова, А. Последняя миля – на «автомате» [Текст] / А. Напалкова // Эксперт. – 2012. – № 7. – С. 24–25.
2. Packstation [Электронный ресурс]. – Режим доступа: <http://de.wikipedia.org/wiki/Packstation>. – 12.01.2014.
3. Близость к дому – главный критерий выбора почтомата [Электронный ресурс]. – Режим доступа: <http://www.logibox.ru/article/blizost-k-domu-glavnyj-kriterij-vybora-pochtomata-dostavki>. – 12.11.2014.
4. Гольдина, А. Новая сеть почтовых терминалов в России [Электронный ресурс] / А. Гольдина. – Режим доступа: <http://63.ru/text/itnews/487181.html>. – 12.11.2014.
5. Logibox – федеральная сеть автоматических станций по приему и вручению отправлений

[Электронный ресурс]. – Режим доступа: <http://www.logibox.ru/about/>. – 12.11.2014.

6. Терминал удобной доставки QIWI Post [Электронный ресурс]. – Режим доступа: <http://qiwiipost.ru/ru/inpost/news/news>. – 12.11.2014.

7. Почтоматы PickPoint - уникальная альтернатива почте и курьерской доставке [Электронный ресурс]. – Режим доступа: <http://pickpoint.ru/about/>. – 12.11.2014.

8. Почтомат PickPoint глазами покупателя: удобно или нет [Электронный ресурс]. – Режим доступа: <http://www.siteprojects.ru/blog/pochtomat-pickpoint-glazami-pokupatelya-udobno/>. – 12.11.2014.

9. Построение RESTful web API в Cache [Электронный ресурс]. – Режим доступа: <http://habrahabr.ru/company/intersystems/blog/204576/>. – 12.11.2014.

10. Труб, И. И. СУБД Cache 5: работа с объек-

тами [Текст] / И. И. Труб. – М. : «Диалог-МИФИ», 2006. – 385 с.

11. Якунин, Ю. Технология ZEN/PROTOTYPE 6 [Текст] : учеб. пособие / Ю. Якунин, Р. Квитунов. – Красноярск, 2008. – 148 с.

12. Андреев, И. Использование протокола SOAP в распределенных приложениях [Электронный ресурс] / И. Андреев. – Режим доступа: <http://www.rsdn.ru/article/xml/soap.xml> - 13.03.2005.

13. Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования [Текст] : пер. с англ. / М. Фаулер. – СПб. : Символ-Плюс, 2011. – 192 с.

14. Using InterSystems Zen Reports [Текст] : учеб. пособие. – Cambridge MA, 2008. – 83 с.

15. Тидуэлл, Д. XSLT [Текст] : пер. с англ. / Д. Тидуэлл. – СПб. : Символ-Плюс, 2011. – 960 с.

Поступила в редакцию 12.11.2014, рассмотрена на редколлегии 18.11.2014

Рецензент: д-р техн. наук, проф., профессор кафедры инженерии программного обеспечения И. В. Шостак, Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Харьков.

РОЗРОБКА ПІДСИСТЕМИ УПРАВЛІННЯ ПОШТОВИМИ ТЕРМІНАЛАМИ

М. Ю. Бабич, О. Б. Лещенко

Розглянуто задачу автоматизації доставки поштових відправлень. Для вирішення поставленого завдання було проведено аналіз особливостей автоматизації доставки поштових відправлень, розглянуто основні існуючі системи, особливості їх функціонування та розміщення. Отримана модель відображає структуру збережених даних підсистеми. На етапі вибору інструментальних засобів було проаналізовано основні платформи для реалізації розподілених систем. Для розробки інформаційної підсистеми обслуговування поштових терміналів було побудовано основні UML діаграми, які включають діаграми прецедентів і класів. Визначено комплекс технічних і програмних засобів поштового терміналу. Система поштових терміналів реалізована на платформі InterSystems Caché, яка підтримує створення Web-додатків, стандарт XML, протокол SOAP і Web-сервіси.

Ключові слова: інформаційна система, автоматизація, автоматизований поштовий термінал, InterSystems Caché.

DEVELOPMENT OF POSTAL TERMINALS MANAGEMENT SUBSYSTEM

M. Y. Babych, A. B. Leshchenko

The problem of the postal items automating delivery. In order to solve this problem was analyzed features of postal items automate delivery, considered major existing systems, especially their functioning and placement. The resulting model reflects the structure of the stored data subsystem. At the stage of selection tools were analyzed the main platform for the implementation of distributed systems. In order to develop the postal terminals information subsystem were built basic UML diagrams, which include use case diagrams, classes and states diagrams. Defined set of technical resources and programs of postal terminal. The postal terminals system implemented on the InterSystems Caché platform, which supports the creation of Web-based applications, the standard XML, SOAP protocol and Web-services.

Keywords: information system, automation, automated postal terminal, InterSystems Caché.

Бабич Максим Юрьевич – магистрант каф. информационных управляющих систем, Национальный аэрокосмический университет им. Н. Е. Жуковского «Харьковский авиационный институт», Харьков, Украина, e-mail: maks.babich@gmail.com.

Лещенко Александр Борисович – канд. техн. наук, доцент, доцент каф. информационных управляющих систем, Национальный аэрокосмический университет им. Н. Е. Жуковского «Харьковский авиационный институт», Харьков, Украина, e-mail: lesch@xai.edu.ua.