Введение

Почтовая марка – это специальный знак почтовой оплаты, выпускаемый и продаваемый национальными (и иными) почтовыми ведомствами и обладающий определённой номинальной стоимостью (номиналом).

Впервые необходимость в оплате почтовых услуг появилась с появлением самой почты, при этом почта, как и в настоящее время, представляла собой передачу информации в письменном виде от отправителя к получателю. До середины 17 века монополия на почту принадлежала государствам, имеющим необходимость получать как можно раньше важную информацию. В 1653 году в Париже появилась регулярная почтовая служба. Поскольку оплачивать отправку письма мог как отправитель, так и получатель, то письмо обертывалось билетами почтового сбора, и для оплаты на билете проставлялся специальный штемпель. К началу 19 века механизм оплаты отправления писем эволюционировал до оттиска на специальном штемпельном листе. Этот лист мог иметь различную форму (круглую, овальную, восьмиугольную), а разнообразные водяные знаки препятствовали подделкам. В 1840 году идея штемпельного листа трансформировалась в традиционную марку, на которой с одной стороны изображен сложный для подделки рисунок, а с другой имелось покрытие из клеевого слоя. Первая почтовая марка вошла в историю под названием «Черный пенни». С этого момента для оплаты почтовых услуг следовало приобрести марку, наклеить ее на конверт, проставить печать и отправить письмо адресату [ссылка на википедию – почтовая марка].

В настоящее время роль почты при отправлении писем заметно уменьшилась с развитием Интернета. Пользователь может отправить электронное письмо, содержащее информацию, всего за несколько секунд, при этом не требуется оплачивать отправку письма. Таким образом, роль почтовых марок большей частью сводится к объекту коллекционирования. Стоимость отдельных марок, существующих в мире в единственном экземпляре, может превышать миллион долларов США.

Во время курсового проектирования необходимо разработать автоматизированную систему учета почтовых марок, которая позволит пользователю вести собственный каталог марок и получать информацию в виде отчета.

При проектировании системы будет использоваться методология UML (Unified Modeling Language – унифицированный язык моделирования), которая применяется для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это – открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования, в основном, программных систем. UML не является языком программирования, но на основании UML-моделей возможна генерация кода[2].

1. Описание и анализ предметной области

1.1 Основные характеристики почтовой марки

Почтовая марка – это специальный знак почтовой оплаты, выпускаемый и продаваемый национальными (и иными) почтовыми ведомствами и обладающий определённой номинальной стоимостью (номиналом).

Каждая марка имеет следующие обязательные параметры:

* Страна-изготовитель. Параметр содержит название страны, в которой была изготовлена марка (см. рисунки 1, 2, 4);
* Год выпуска. Параметр содержит год, в который марка была выпущена (см. рисунок 1);
* Гашение. Параметр содержит информацию о том, было ли произведено гашение марки, или нет (см. рисунок 1);



Рисунок 1 – Немецкая почтовая марка, выпущенная в 1965 году

* Тема. Параметр содержит информацию о тематике изображения марки (см. рисунок 1);
* Серия. Параметр содержит описание общей темы, события или повода выпуска марок, имеющих похожий стиль оформления (см. рисунок 2);
* Номинал. Параметр содержит цену марки, с помощью которой была произведена оплата почтовых услуг (см. рисунки 1, 2);
* Тираж. Параметр содержит данные о количестве марок, выпущенных одной партией (как правило, на марке нет сведений о тираже);
* Зубцовка. Данный параметр, в свою очередь, разделяется на три независимых признака:

1. технология перфорирования (см. рисунок 3);
2. количество зубцов на каждой стороне марки;
3. направление линий зубцовки (см. рисунок 4);



Рисунок 2 – «Романовская» серия марок, 1913 год

* Бумага. Параметр содержит информацию об основных характеристиках бумаги, на которой печатался тираж марки;
* Размер. Разные марки могут быть разного размера;
* Цвет. Дешевые марки могут быть напечатаны в черно-белом варианте, тогда как более дорогие используют большее число различных цветов.

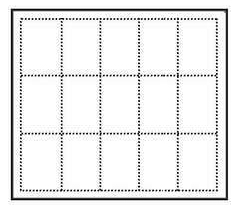


Рисунок 3 – Схема марочного листа с рамочной зубцовкой



Рисунок 4 – Частичная зубцовка на марке США 1954 года

В качестве примера можно рассмотреть образец почтовой марки, выпущенной в 1914 году (см. рисунок 5). На данной марке изображен Георгий Победоносец, убивающий дракона, надпись под изображением гласит: «Въ пользу воиновъ и ихъ семействъ». Поверх изображения сделано гашение с пометкой «Образец». Серия отсутствует. Номинал марки составляет 10 копеек, хотя ее цена на момент выпуска составляла 11 копеек. Изображение является трехцветным, не считая черного цвета гашения. Из-за отсутствия дополнительной информации (размер, бумага, тираж) некоторые параметры не могут быть корректно идентифицированы.



Рисунок 5 – Образец почтовой марки Российской империи 1914 года

1.2 Обзор существующих систем-аналогов

1.2.1 Программный комплекс «Моя филателия»

На сегодняшний день наиболее известной системой подобного рода является программный комплекс «Моя филателия» (см. рисунок 6). С ее помощью пользователь может создавать и редактировать коллекции марок, вести справочники по различным характеристикам марок, таким как базовые данные (страна, год выпуска), внешние характеристики (форма, цвет, размер), а также содержание (оформление, тема). Благодаря данным справочникам пользователь может добавлять в коллекцию исчерпывающие записи о марках, которые содержат не только подробные характеристики марки, но и информацию о возможных продавцах или покупателях данной марки. Кроме того, для удобства пользователя реализована функция поиска нужной марки по запросу, содержащему некоторые данные о марке[3].

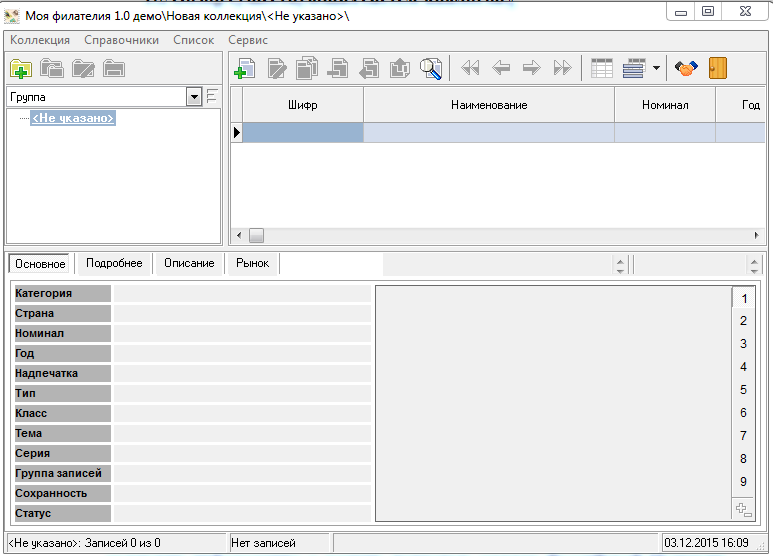


Рисунок 6 – Экранная форма программного комплекса «Моя филателия»

1.2.2 Филателия. Почтовые марки России и СССР

В качестве систем-аналогов также можно рассмотреть сайты, на которых размещена информация о марках, доступных для покупки. Это связано с тем, что люди, коллекционирующие марки, интересуются как характеристиками своих марок, так и возможностями приобрести новые, продать лишние или совершить обмен. Сайты, посвященные торговым маркам, содержат необходимую информацию и обеспечивают надежный доступ к ней. Кроме того, популярность сайта зависит от своевременного обновления новостной ленты, содержащей оригинальный контент.

Сайт, расположенный по адресу [4] (рисунок 7), содержит информацию о редких почтовых марках Российской Империи, РСФСР и СССР. Эти марки сгруппированы как по годам выпуска, так и по общей теме изображения.

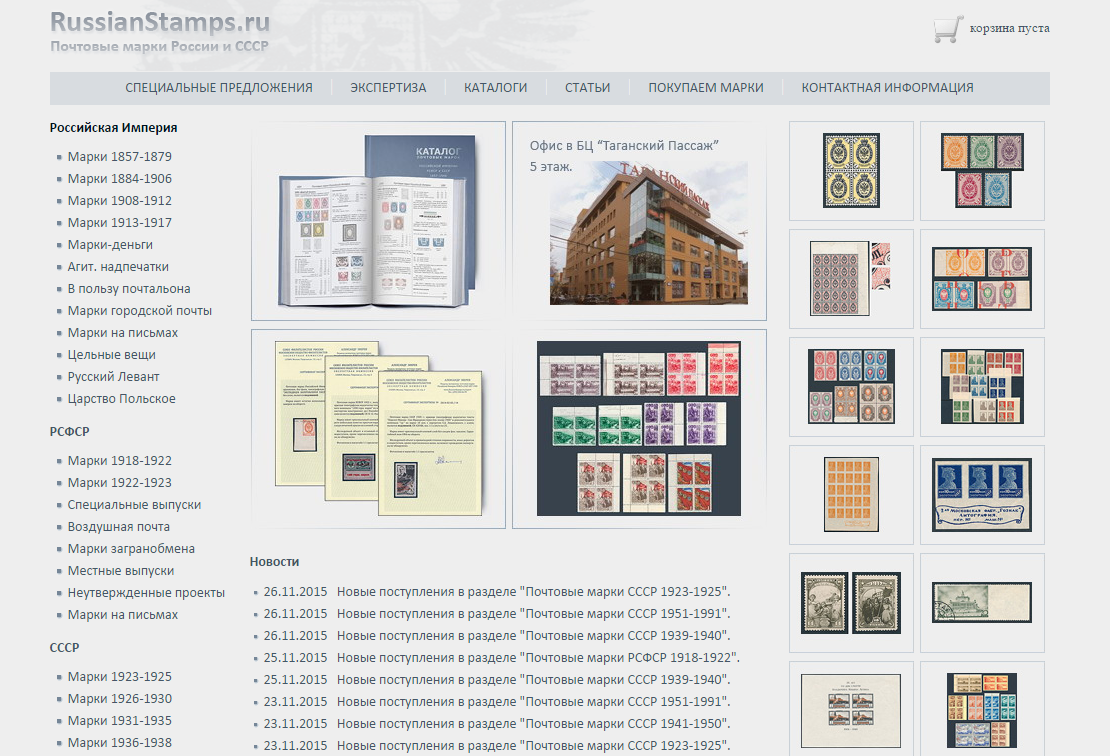


Рисунок 7 – Главная форма сайта «Филателия. Почтовые марки России и СССР»

В разделе «Специальные предложения» можно приобрести марки комплектами по более низкой цене. Раздел «Экспертиза» определяет понятие экспертизы марки, ее условия, сроки и стоимость проведения. Раздел «Каталоги» предлагает посетителю сайта приобрести различные каталоги марок. Раздел «Покупаем марки» позволяет посетителю продать марки магазину или отдать на реализацию на комиссионной основе. Посетитель также имеет возможность найти нужную марку по ее году выпуска и купить у лиц, указанных в разделе «Контактная информация». Благодаря своевременным обновлениям сайта посетители имеют доступ к информации о новых марках, появившихся в продаже.

1.2 Диаграмма объектов предметной области

Диаграмма объектов представляет статическую составляющую взаимодействующих между собой объектов, она должна включать в себя только те объекты предметной области, которые потом преобразуются в диаграмму классов.

Связи между объектами показывают отношения между ними, при необходимости в диаграмме можно привести и атрибуты (свойства) объектов. На рисунке 8 приведена диаграмма объектов для рассматриваемой системы.

Пользователь составляет коллекцию, состоящую из некоторого числа всех возможных марок. Эта коллекция хранится в базе данных. Имея коллекцию, пользователь может создать отчет по ней, который будет храниться в файле заданного формата.

1.4 Постановка задачи

В рамках курсового проектирования перед авторами поставлена задача – разработать автоматизированную систему учета почтовых марок, которая позволит пользователю вести свою коллекцию почтовых марок, редактировать ее и производить поиск записей коллекции по заданному критерию.

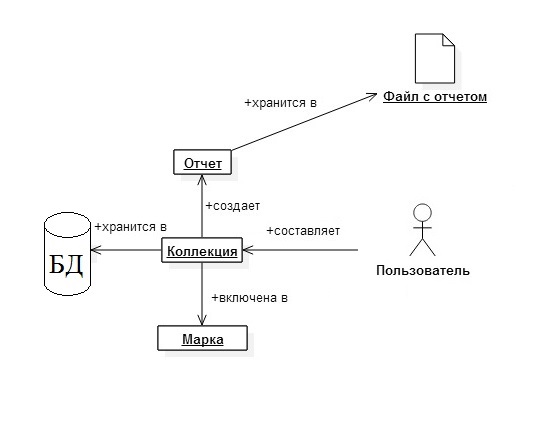


Рисунок 8 – Диаграмма объектов предметной области

В системе должны быть предусмотрены возможности создания справочников, содержащих достаточное число марок, различающихся по стране изготовления, номиналу, году выпуска, типу бумаги и иным параметрам, указанным в п.2 приложения к заданию на курсовой проект. На основе справочной информации пользователь может создать и в дальнейшем редактировать собственную коллекцию марок. Как справочник, так и коллекция должны удовлетворять принципам ведения баз данных. При необходимости, пользователь сможет сохранить коллекцию в виде отчета в файл, настроив его параметры.

В системе также должна быть обеспечена возможность получения справочной информации как о самой системе, так и о предоставляемых ею возможностях.

Таким образом, система должна выполнять следующие функции:

1. работа со справочниками БД:

* добавление записи;
* удаление записи;
* редактирование записи;

1. ввод информации о почтовой марке;

* добавление описания;
* задание параметров марки;
* прикрепление фотографии;

1. редактирование информации о марках;
2. удаление информации о марках;
3. формирование отчета:
   1. настройка параметров отчета;
   2. импортирование отчета в файл заданного формата;
4. отбор информации по заданному критерию (параметру марки);
5. выбор критериев отбора;
6. выдача справочной информации о системе.
7. ПРОЕКТИРОВАНИЕ СИСТЕМЫ
8. Структурная схема системы

Система (греч. – «составленное из частей», «соединение» от «соединяю») – множество элементов, находящихся в отношениях и связях друг с другом, которое образует определённую целостность, единство.

Как следует из определения, отличительным (главным свойством) системы является ее целостность: комплекс объектов, рассматриваемых в качестве системы, должен обладать общими свойствами и поведением. Очевидно, необходимо рассматривать и связи системы с внешней средой. В самом общем случае понятие «система» характеризуется:

* наличием множества элементов;
* наличием связей между ними;
* целостным характером данного устройства или процесса.

Система должна представлять собой совокупность элементов (объектов, субъектов), находящихся между собой в определенной зависимости и составляющих некоторое единство (целостность), направленное на достижение определенной цели. Система может являться элементом другой системы более высокого порядка (надсистема) и включать в себя системы более низкого порядка (подсистемы). То есть систему можно рассматривать как набор подсистем, организованных для достижения определенной цели и описанных с помощью набора моделей (возможно, с различных точек зрения), а подсистему – как группу элементов, часть которых составляет спецификацию поведения, представленного другими ее составляющими.

На рисунке 9 представлена структурная схема системы. В состав системы входят следующие подсистемы:

1. подсистема работы со справочниками, которая позволяет добавлять, удалять и редактировать записи;
2. подсистема работы с марками, в состав которой входят:

Подсистема работы с алгоритмами

Подсистема моделирования  
работы МТ




Рисунок 9 – Структурная схема системы

1. подсистема работы с коллекцией, которая позволяет добавлять марки из каталога в коллекцию, а также задавать дополнительные параметры.
2. подсистема поиска, позволяющая пользователю найти нужную информацию по введенным критериям (параметрам марки);
3. подсистема формирования отчета, которая позволяет создать отчет, содержащий информацию о марках, соответствующих введенным критериям поиска;
4. файловая подсистема, которая отвечает за сохранение отчета и изображений марок в файлах заранее определенного типа;
5. справочная подсистема, которая содержит сведения о системе (руководство пользователя) и о её разработчиках.
6. Разработка прототипа интерфейса системы

Прототипирование интерфейса программного обеспечения – этап разработки программного обеспечения, процесс создания прототипа программы – макета черновой версии программы, обычно – с целью проверки пригодности предлагаемых для применения концепций, архитектурных и технологических решений, а также для представления программы заказчику на ранних стадиях процесса разработки [6]. В ходе разработки был получен прототип интерфейса главного окна, который представлен на рисунке 10.

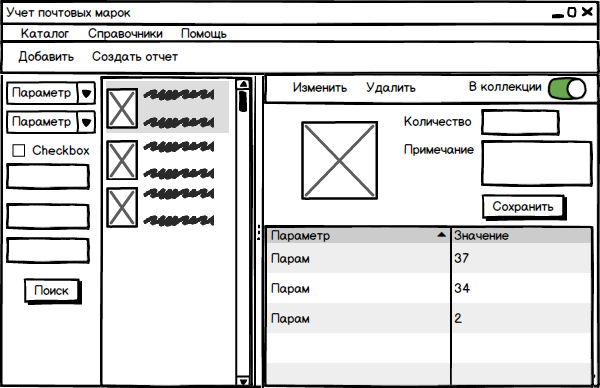


Рисунок 10 – Прототип главного окна

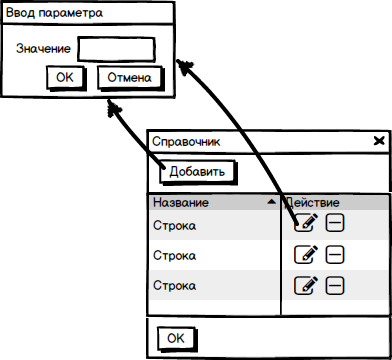


Рисунок 11 – Прототип окна управления справочником

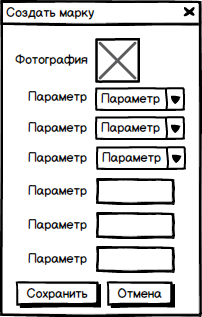


Рисунок 12 – Прототип окна создания и редактирования марки

В пункте меню "Каталог" пользователь может выбрать команду создания отчета, создания новой записи или выхода из программы. При сохранении файла открывается соответствующее диалоговое окно.

В пункте меню "Справочники" пользователь может выбрать команду открытия окна управления справочником одного из параметров марки. Прототип диалогового окна изображен на рисунке 11. Нажимая на соответствующие кнопки, пользователь может создавать, редактировать и удалять записи в выбранном справочнике. В случаях добавления и изменения записей, открывается дополнительное диалоговое окно, где пользователь может ввести значение записи.

В пункте меню "Помощь" пользователь может выбрать команду открытия справочной информации в виде интернет-страницы.

Пользователь может задать необходимые критерии (параметры марки) – год выпуска, страна, тема, зубцовка и т.д. – и найти требуемую информацию, нажав кнопку «Поиск». Тогда в центре главного окна появится список марок, удовлетворяющих заданным критериям. Нажав левой кнопкой мыши на интересующую марку, пользователь сможет получить более подробную информацию о марке.

При нажатии на кнопки «Добавить» и «Изменить» открывается окно, прототип которого представлен на рисунке 12. В данном окне пользователь может вводить значения параметров марки. При нажатии на кнопку «Сохранить» в случае корректных данных создается запись о марке. При нажатии на кнопку «Отмена» диалоговое окно закрывается, а введенные данные не сохраняются.

После активации переключателя «В коллекции» пользователь может изменить параметры выбранной марки, доступные только для коллекции. После нажатии на кнопку «Сохранить» введенные данные сохраняются.

1. UML–проект

UML (Unified Modeling Language – унифицированный язык моделирования) – язык графического описания для объектного моделирования в области разработки программного обеспечения. UML является языком широкого профиля, это открытый стандарт, использующий графические обозначения для создания абстрактной модели системы, называемой UML-моделью. UML был создан для определения, визуализации, проектирования и документирования в основном программных систем. UML не является языком программирования, но в средствах выполнения UML-моделей как интерпретируемого кода возможна кодогенерация [7].

Использование UML не ограничивается моделированием программного обеспечения. Его также используют для моделирования бизнес-процессов, системного проектирования и отображения организационных структур.

UML позволяет также разработчикам программного обеспечения достигнуть соглашения в графических обозначениях для представления общих понятий (таких как класс, компонент, обобщение (generalization), объединение (aggregation) и поведение), и больше сконцентрироваться на проектировании и архитектуре.

UML содержит стандартный набор диаграмм и нотаций для моделирования самых разнообразных видов систем: систем программного обеспечения, технических средств и организационно-экономических систем [8]:

1. диаграммы вариантов использования (use case diagrams);
2. диаграммы классов (class diagrams);
3. диаграммы поведения системы (behavior diagrams):
   * диаграммы состояний (statechart diagrams);
   * диаграммы деятельностей (activity diagrams;
   * диаграммы взаимодействия (interaction diagrams):
   * диаграммы последовательности (sequence diagrams);
   * кооперативные диаграммы (collaboration diagrams).
4. диаграммы реализации (implementation diagrams):
   * диаграммы компонентов (component diagrams);
   * диаграммы размещения (deployment diagrams).

**Диаграмма вариантов использования**

Диаграмма вариантов использования описывает функциональное назначение системы, т.е. то, что система будет делать в процессе своего функционирования. Диаграмма вариантов использования является исходным концептуальным представлением или концептуальной моделью системы в процессе ее проектирования и разработки.

Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества сущностей или актантов, взаимодействующих с системой с помощью так называемых вариантов использования. При этом актантом или действующим лицом называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. В свою очередь вариант использования служит для описания сервисов, которые система предоставляет актанту.

Общая диаграмма вариантов использования разработанной системы представлена на рисунке 13. На данной диаграмме актант – пользователь, который может управлять каталогом марок и собственной коллекцией марок, добавляя и убирая марки из нее, храня дополнительную информацию. Также пользователь может управлять справочниками характеристик марок, вести поиск и создавать отчеты с учетом фильтров. Диаграмма вариантов использования при работе с марками изображена на рисунке 14. Пользователь может создавать, редактировать и просматривать записи о марках, а также удалять их из каталога.

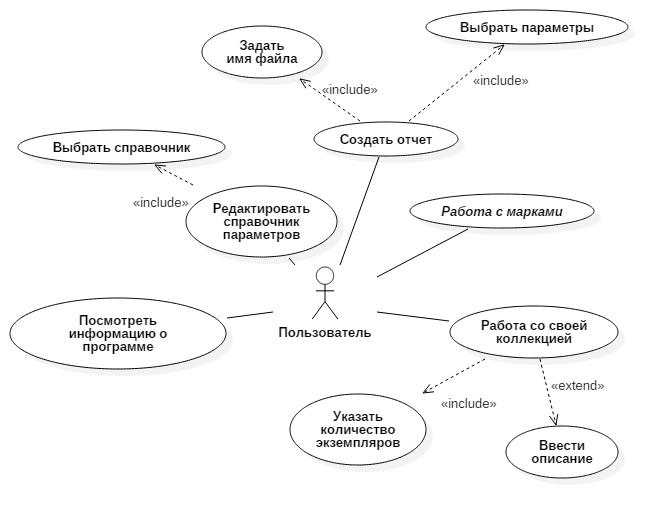


Рисунок 13 – Общая диаграмма вариантов использования системы

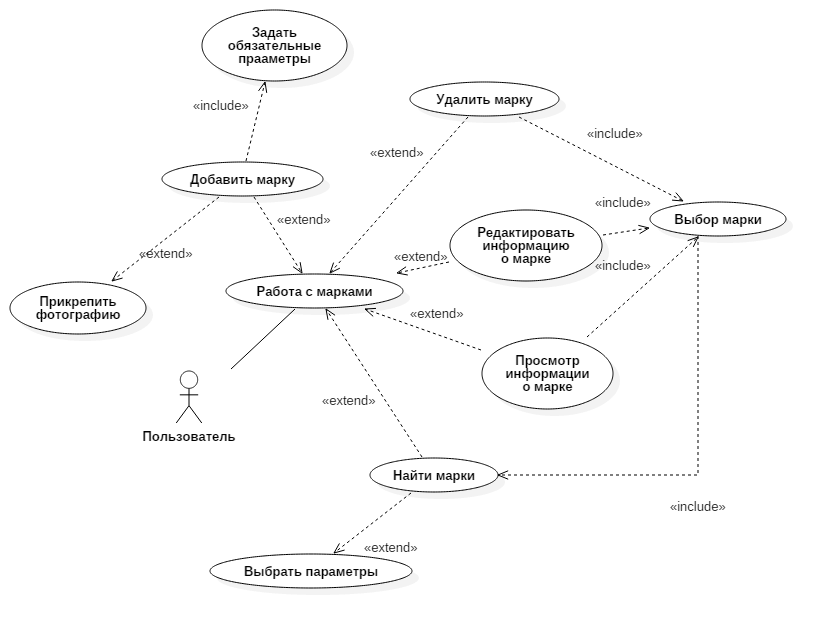


Рисунок 14 –Диаграмма вариантов использования системы   
(работа с марками)

Каждый вариант использования должен быть задокументирован как некий сценарий работы с системой. Сценарий – определенная последовательность действий, которая описывает действия актеров и поведение моделируемой системы в форме обычного текста. Ниже приведены примеры оформления сценариев.

***Вариант использования:* Просмотр информации о марке.**

*Краткое описание.* Дает возможность пользователю просмотреть хранящуюся в системе информацию о конкретной интересующей марке. Информация, содержащая названия параметров и их значения, представлена в форме таблицы в главном окне приложения.

*Актант.* Пользователь.

*Предусловия.* Компьютер пользователя включён, на экране – главное окно программы « Автоматизированная система учета почтовых марок».

*Основной поток событий.*

1. На экране появляется основная форма программы. Пользователь вводит критерии поиска марок в каталоге и нажимает кнопку «Поиск».
2. В главном окне приложения отображается список найденных марок.
3. Пользователь кликом мыши выбирает нужную марку в списке.
4. В главном окне приложения отображается информация о выбранной марке.

*Постусловия.* Отображена информация о марке.

***Вариант использования:* Добавить марку в каталог.**

*Краткое описание.* Дает возможность пользователю создать новую запись в о марке в системе. Пользователь вводит параметры марки в диалоговом окне. После создания пользователь может работать с новой записью.

*Актант.* Пользователь.

*Предусловия.* Компьютер пользователя включён, на экране – главное окно программы «Автоматизированная система учета почтовых марок.

*Основной поток событий.*

1. Пользователь нажимает на кнопку «Добавить марку».
2. На экране появляется диалоговое окно с полями ввода для параметров добавляемой марки.
3. Пользователь вводит значения, соответствующие характеристикам марки, в поля ввода. После ввода информации пользователь нажимает на кнопку «Сохранить»

А1: Заполнены не все обязательные поля.

А2: Нажата кнопка «Отмена»

1. Система закрывает окно создания и возвращается в главную форму приложения.

*Альтернативы.*

А1: Заполнены не все обязательные поля.

А1.1: Система выводит сообщение «Некорректные данные», содержащее информацию, какие конкретно обязательные поля не были заполнены. При закрытии сообщения при пользователь возвращается в диалоговое окно. Вариант использования завершается успешно.

А2: Введены значения, не подходящие по ограничениям.

А2.1: Система выводит сообщение «Некорректные данные», содержащее информацию, какие поля были заполнены некорректно, а так же ограничения для них. При закрытии сообщения при пользователь возвращается в диалоговое окно. Вариант использования завершается успешно.

*Постусловия.* Запись о марке добавлена в систему.

### Диаграмма классов

Диаграммы классов – это наиболее часто используемый тип диаграмм, которые создаются при моделировании объектно-ориентированных систем, они показывают набор классов, интерфейсов и коопераций, а также их связи. На практике диаграммы классов применяют для моделирования статического представления системы, они служат основой для целой группы взаимосвязанных диаграмм – диаграмм компонентов и диаграмм размещения. На рисунке 15 представлена диаграмма сущностных классов приложения. На рисунке 16 представлена диаграмма интерфейсных классов приложения.

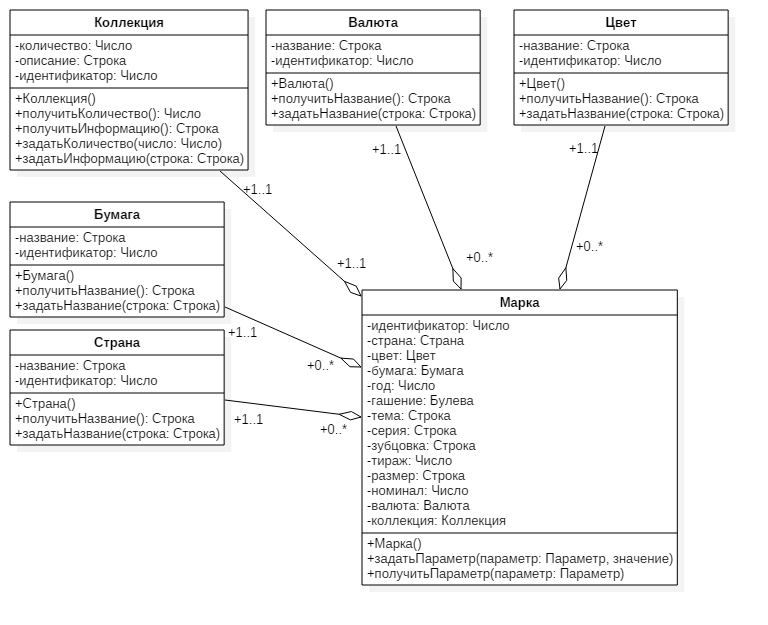


Рисунок 15 – Диаграмма сущностных классов приложения

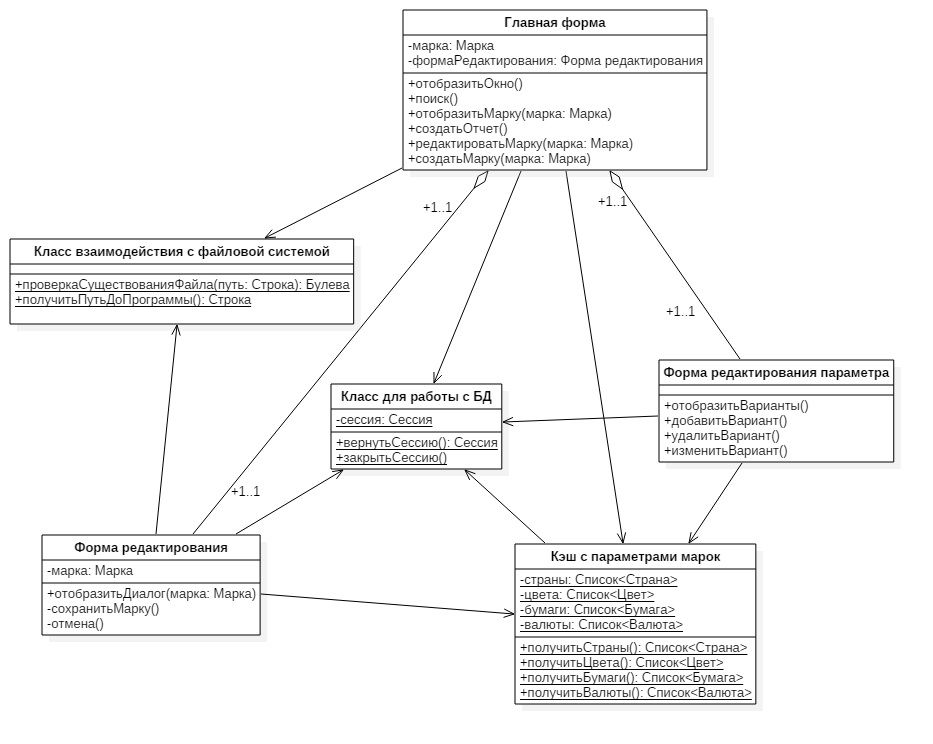


Рисунок 16 – Диаграмма интерфейсных классов приложения

### Диаграмма состояний

Главное назначение диаграммы состояний – описать возможные последовательности состояний и переходов, которые в совокупности характеризуют поведение моделируемой системы в течение всего ее жизненного цикла. Диаграмма состояний представляет динамическое поведение сущностей, на основе спецификации их реакции на восприятие некоторых конкретных событий. Системы, которые реагируют на внешние действия от других систем или от пользователей, иногда называют реактивными. Если такие действия инициируются в произвольные случайные моменты времени, то говорят об асинхронном поведении модели.

Общая диаграмма состояний системы представлена на рисунке 17. После запуска приложения отображается главное окно программы. При нажатии на кнопку «Добавить» открывается окно создания марки, при нажатии на кнопку «Справка» отображается справочная информация о системе, при нажатии на кнопку «Поиск» осуществляется поиск марок по введенным критериям поиска, при нажатии на марку из списка отображается список её параметров. При нажатии на кнопку «Изменить» открывается окно редактирования параметров марки, при нажатии на кнопку «Удалить» происходит удаление марки. По нажатию на «Х» окно закрывается.

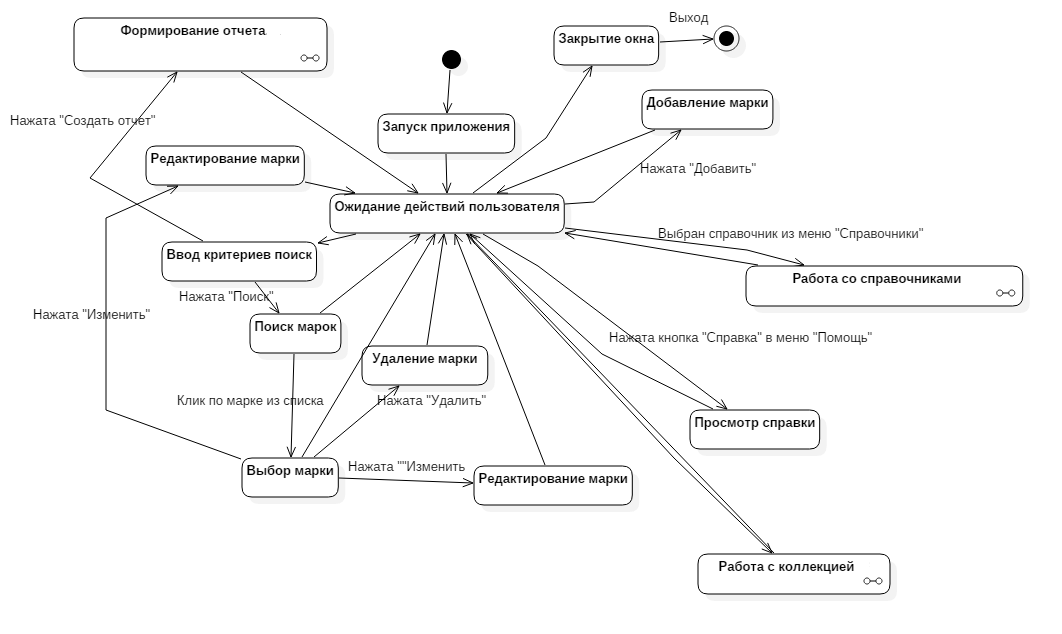


Рисунок 17 – Общая диаграмма состояний системы

На рисунке 18 изображена диаграмма состояний «Работа со справочниками». При нажатии на кнопку «Добавить» открывается окно ввода нового значения. При нажатии на кнопку «Изменить» открывается окно ввода нового значения. При попытке сохранения значения происходит его проверка. Если данное значение уже присутствует в справочнике, второе такое же добавить нельзя. При нажатии на кнопку «Удалить» значение, если оно не используется, удаляется. При нажатии на «Х» форма управления справочником закрывается.

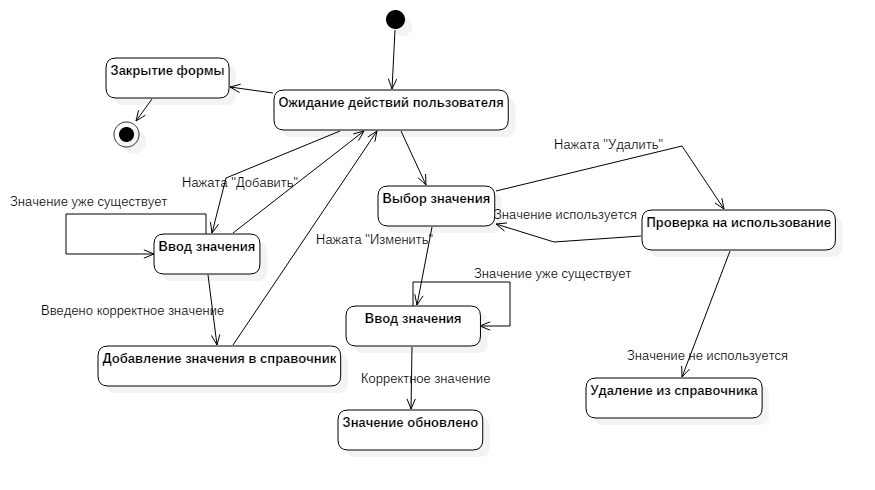


Рисунок 18 – Диаграмма состояний «Работа со справочниками»

### Диаграмма последовательностей

Для моделирования взаимодействия объектов в языке UML используются соответствующие диаграммы взаимодействия. Так, взаимодействия объектов можно рассматривать во времени, и тогда для представления временных особенностей передачи и приема сообщений между объектами используется диаграмма последовательности.

На диаграмме последовательности изображаются объекты, которые непосредственно участвуют во взаимодействии. Для диаграммы последовательности ключевым моментом является динамика взаимодействия объектов во времени. Взаимодействия объектов реализуются посредством сообщений, которые посылаются одним объектом другим и также образуют порядок по времени своего возникновения.

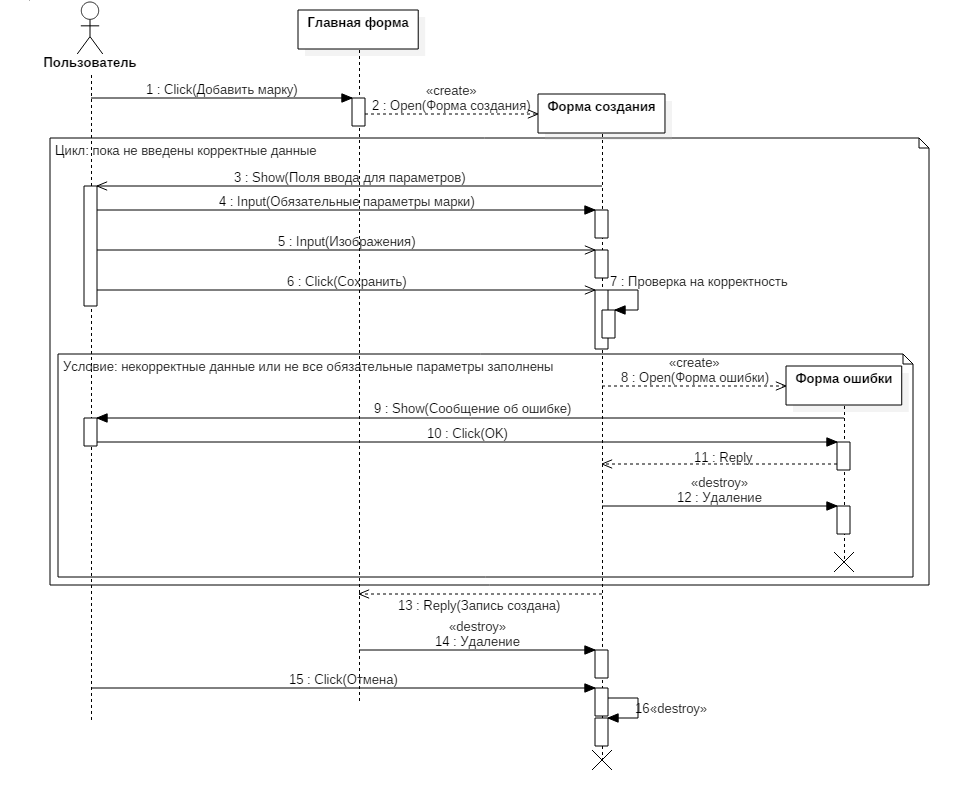


Рисунок 19 – Диаграмма последовательности варианта использования   
«Добавить марку в каталог»

На рисунке 19 представлена диаграмма последовательности варианта использования «Добавить марку в каталог». На главной форме пользователь наживает кнопку «Добавить». Система отображает новое диалоговое окно, содержащее поля ввода. Пользователь задает обязательные значения параметров и, возможно, прикрепляет фотографию, после чего нажимает кнопку «Сохранить». Происходит проверка корректности введенных значений. Если данные некорректны, то формируется форма ошибки и выводится сообщение об ошибке. Если значения параметров заданы корректно, то происходит создание записи о марке.

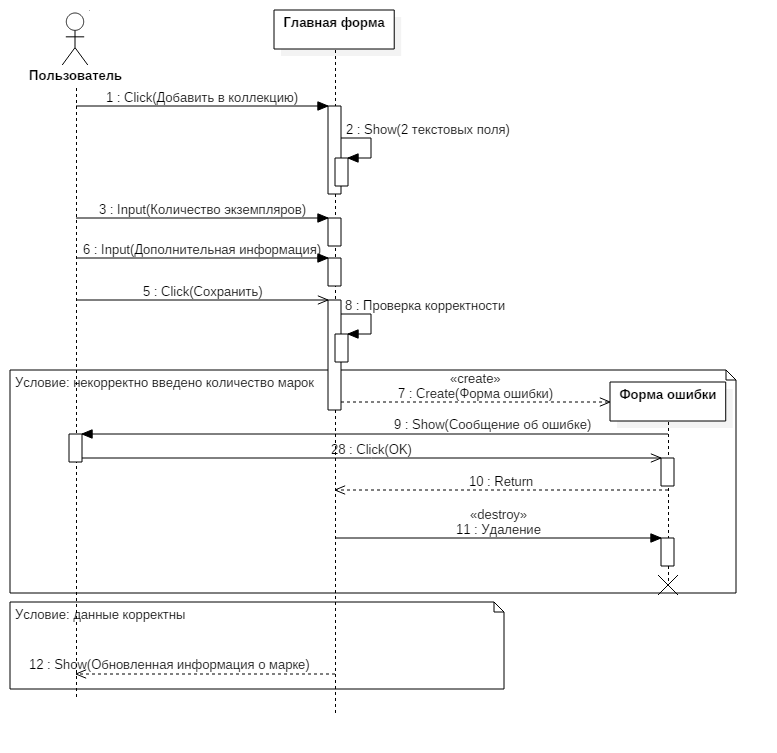


Рисунок 20 – Диаграмма последовательности варианта использования   
«Добавить марку в коллекцию»

На рисунке 20 представлена диаграмма последовательности варианта использования «Добавить марку в коллекцию». На главной форме пользователь нажимает кнопку «В коллекции». Система отображает поля ввода количества экземпляров и примечания. Пользователь задает значения параметров и нажимает кнопку «Сохранить». Происходит проверка корректности введенных значений. Если данные некорректны, то формируется форма ошибки и выводится сообщение об ошибке. Если значения параметров заданы корректно, то происходит добавление марки в коллекцию.

Диаграмма деятельности

Диаграммы деятельности используются для моделирования процесса выполнения операций. Каждое состояние на диаграмме деятельности соответствует выполнению некоторой элементарной операции, а переход в следующее состояние срабатывает только при завершении этой операции в предыдущем состоянии. На диаграмме деятельности отображается логика или последовательность перехода от одной деятельности к другой, при этом внимание фокусируется на результате деятельности. Сам же результат может привести к изменению состояния системы или возвращению некоторого значения. На рисунке 21 представлена диаграмма деятельности «Работа с марками».

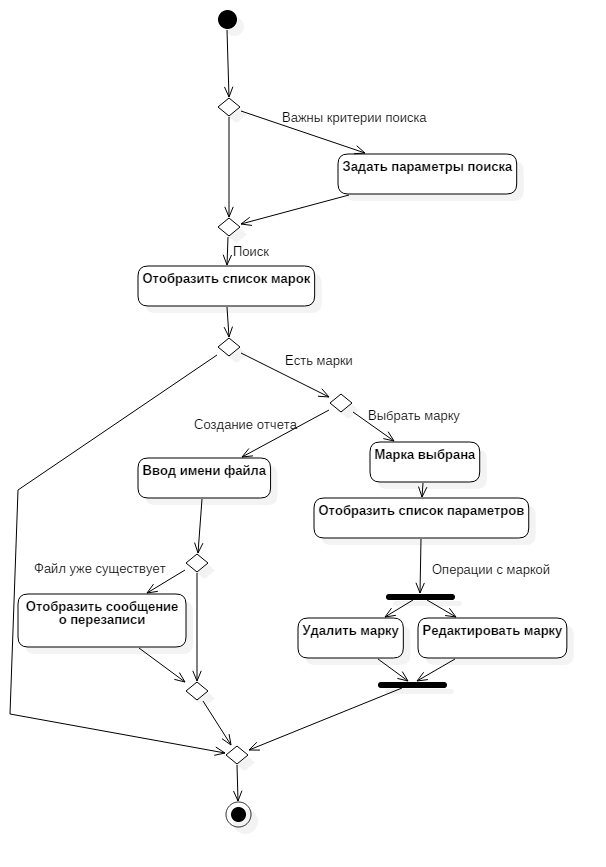


Рисунок 21 – Диаграмма деятельности «Работа с марками»