

2.1 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 деятельности (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.

Диаграмма вариантов использования разработанной системы представлена на рисунке 10. На данной диаграмме актант – пользователь, который имеет возможности загружать алгоритм из файла, сохранять алгоритм и трассу в файл. Пользователю предоставляются возможности создать алгоритм вручную или выбрать один из базовых алгоритмов, запустить алгоритм в одном из режимов демонстрации. Также пользователь может задать значения операндов в выбранном формате и посмотреть справочную информацию.

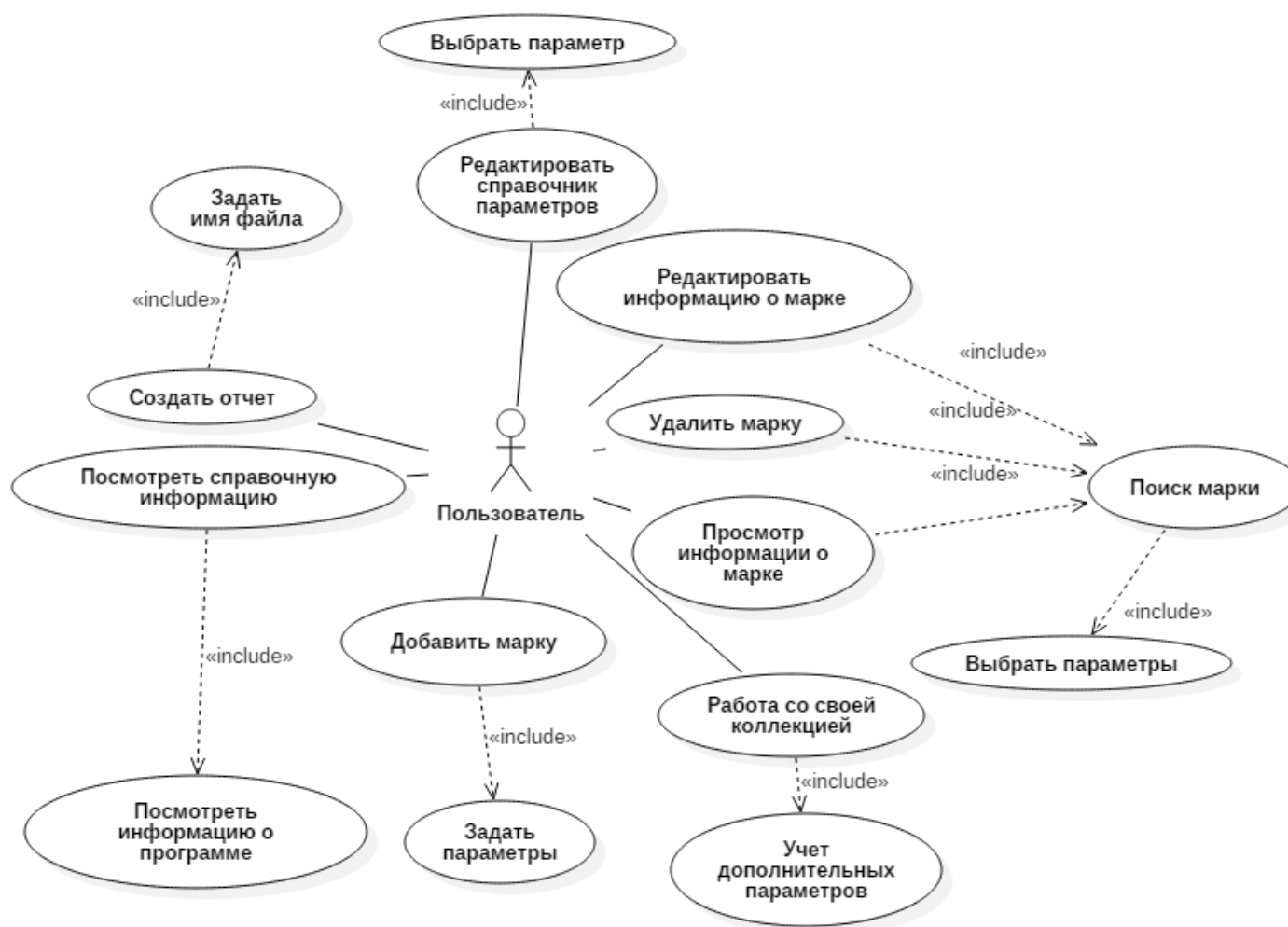


Рисунок 10 – Графическое представление автоматизированной системы учета почтовых марок

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: Сохранить отчет в файл.

Краткое описание. Дает возможность пользователю сохранить информацию о марках в каталоге. Пользователь открывает окно сохранения выбрав команду «Создать алгоритм» в меню «Каталог» или нажав на аналогичную кнопку в главном окне приложения.

Актант. Пользователь.

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

Основной поток событий.

1. Пользователь выбрал в меню «Каталог» пункт «Создать отчет».
2. На экране появляется стандартное окно для сохранения файла. В нем пользователь выбирает папку для сохранения файла, вводит имя файла вручную и нажимает кнопку «Сохранить». Файл сохраняется. В случае, если файл с таким именем уже существует, выдается сообщение о перезаписи, файл либо сохраняется, либо пользователь возвращается в окно сохранения файла. Вариант использования завершается успешно.

A1: Нажата кнопка «Отмена»

Альтернативы.

A1: Нажата кнопка «Отмена»

A1.1: Система закрывает окно сохранения и возвращается в главную форму приложения.

Постусловия. Отчет сохранен в файл.

Вариант использования: Создать новую марку в каталоге.

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»

A1: Заполнены не все обязательные поля.

A2: Нажата кнопка «Отмена»

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

Альтернативы.

A1: Заполнены не все обязательные поля.

A1.1: 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.

A2: Введены значения, не подходящие по ограничениям.

A2.1: 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.