ОГБПОУ «ТОМСКИЙ ТЕХНИКУМ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

УП 03. Моделирование и анализ предметной области

Тема: «Изучение и анализ предметной области»

Выполнил:

студент 691 группы

Маметьев С. И.

Проверил:  
преподаватель

Маюнова А. Ю.

г. Томск – 2021 г

Содержание

[Описание предметной области 1](#_Toc71804613)

[Описание функций системы 2](#_Toc71804614)

[Диаграмма вариантов использования 3](#_Toc71804615)

[Диаграммы классов 4](#_Toc71804616)

[Диаграммы деятельности 10](#_Toc71804617)

[Диаграммы последовательности 15](#_Toc71804618)

[Диаграммы состояния 20](#_Toc71804619)

[Структурная схема базы данных 25](#_Toc71804620)

[Блок-схемы 28](#_Toc71804621)

[Выводы по работе 34](#_Toc71804622)

[Ответы на контрольные вопросы 34](#_Toc71804623)

# Описание предметной области

Учёт банковских операций – это информационная система, которая служит для обеспечения возможности кассирам совершать операции с клиентом, следить за приходом/расходом, суммой на счете дебит/кредит, записывать и хранить номер операции, дата и время операции, стадию обработки операции, а также ФИО кассира и клиента и остаток в кассе.

Виды деятельности:

1. Совершение операции с клиентом;
2. Отслеживание прихода и расхода, суммы на счетах дебит/кредит, остаток в кассе;
3. Запись и хранение операций, их информацию;
4. Хранение ФИО кассира и ФИО клиента.

Для данной предметной области можно выделить следующих актеров:

|  |  |
| --- | --- |
| Актер | Общее описание |
| Клиент | Пользователь, который обращается к кассиру для совершения операции. |
| Кассир | Пользователь, который обращается к системе для совершения операций с клиентом. |
| БД | Хранилище данных, в которое заносятся операции, ФИО кассира и ФИО клиента, а также отслеживается приход и расход, суммы на счетах дебит/кредит и остаток в кассе. |
| Администратор | Сотрудник, который может управлять БД, восстанавливать её работу в случае неисправности и другое. |

ИС предоставляет следующие возможности:

1. Система предоставляет информацию о кассире и клиенте, операциях и счетах;
2. Кассир обращается к системе, когда клиент хочет совершить операцию. Система предлагает заполнить форму, которую потом проверит система на правильность данных. Если она заполнена верно, система её зафиксирует.
3. БД является внешним хранилищем: ничего не отправляет, только предоставляет запрашиваемую информацию.
4. Администратор следит за БД и управляет ей.

Для данной предметной области можно выделить следующие прецеденты:

|  |  |
| --- | --- |
| Регистрация | Запускается пользователем, позволяет пользователю зарегистрироваться как “Клиент” |
| Авторизация | Запускается пользователем, позволяет получить соответствующий доступ к данным. |
| Создание, изменение и удаление операции | Запускается Кассиром, позволяет ввести в форму информацию об операции и отправить её системе на проверку. Если операция оформлена верно и данные тоже верны, то операция фиксируется. Также Кассир и Администратор могут изменить или удалить операцию, если произошла ошибка данных. |
| Предоставление информации об операции | Запускается Кассиром, формируется запрос в БД о предоставлении информации об операции. |
| Запись и хранение данных | База данных записывает и хранит все данные. |

# Описание функций системы

1. Система дает возможность создавать операцию, записать ФИО кассира и клиента, выбрать приход или расход, посмотреть суммы на счетах дебет/кредит, остаток в кассе.
2. Все обращения к общей базе данных проходят через систему для того, чтобы избежать повторяющихся операций и неполной информации.

# Диаграмма вариантов использования

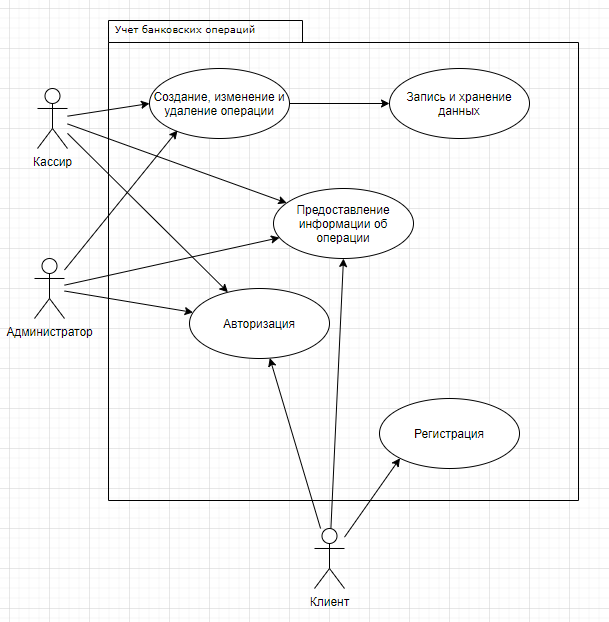


Рис. 1. Диаграмма вариантов использования

# Диаграммы классов

Рассмотрим вариант использования «Создание, изменение и удаление операции». Для данного варианта использования необходимо рассмотреть 2 класса – Кассир и Операция.

Класс «Paymaster»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс пользователя, который может создавать новую операцию. |
| Атрибуты | ID: Int – Идентификатор кассира  LastName: Nvarchar(50) – фамилия кассира  FirstName: Nvarchar(50) – имя кассира  MiddleName: Nvarchar(50) – отчество кассира Login: Nvarchar(50) – логин кассира Password: Nvarchar(30) – пароль кассира |
| Операции | create() – создать операцию edit() – редактировать данные об операции |

Класс «Operation»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, в объектах которого содержится информация. |
| Атрибуты | ID: Int – Идентификатор операции  DateOperation: Datetime – Дата и время операции  Client\_ID: Int – Идентификатор клиента  Paymaster\_ID: Int – Идентификатор кассира  Processing\_Stage: Nvarchar(50) – стадия обработки операции  Type: Nvarchar(50) – тип операции (приход или расход)  Amount: Money – сумма денег, которую клиент хочет получить или вложить |
| Операции | create() – создать операцию edit() – редактировать данные об операции show() – показать операцию |

Конечная диаграмма класса

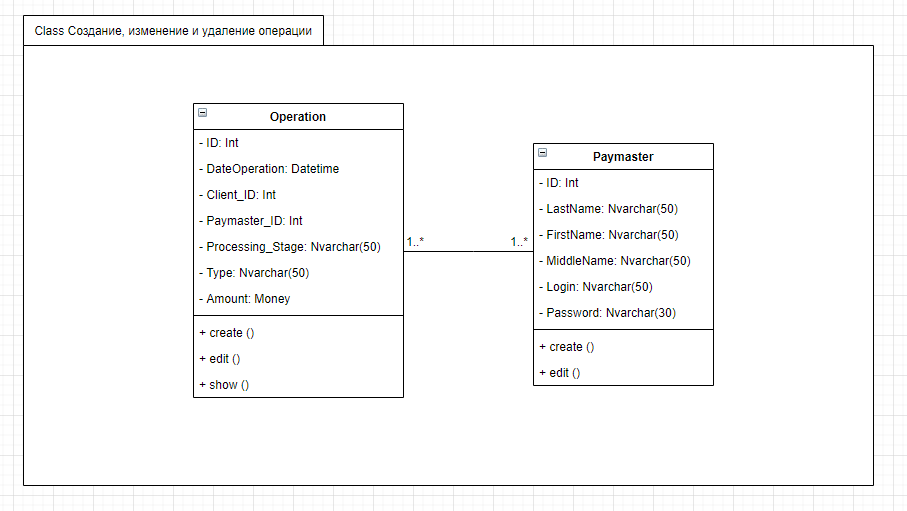


Рис. 2. Диаграмма класса «Создание, изменение и удаление операции»

Рассмотрим вариант использования «Регистрация». Для данного варианта использования необходимо рассмотреть 1 класс – Клиент.

Класс «Client»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс пользователя, который может зарегистрировать свой аккаунт. |
| Атрибуты | ID: Int – идентификатор клиента  LastName: Nvarchar(50) – фамилия клиента  FirstName: Nvarchar(50) – имя клиента  MiddleName: Nvarchar(50) – отчество клиента Login: Nvarchar(50) – логин клиента Password: Nvarchar(30) – пароль клиента |
| Операции | register() – зарегистрировать аккаунт |

Конечная диаграмма класса «Регистрация»

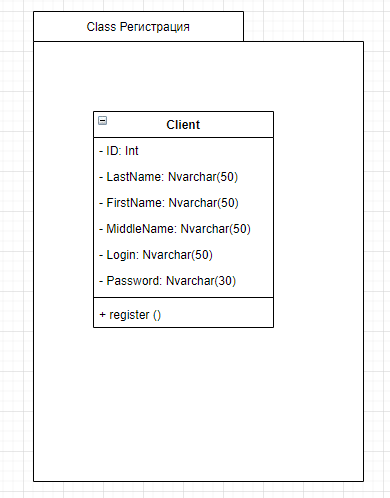


Рис. 3. Диаграмма класса «Регистрация»

Рассмотрим вариант использования «Авторизация». Для данного варианта использования необходимо рассмотреть 3 класс – Клиент, Кассир и Администратор.

Класс «Client»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс пользователя, который может авторизоваться в свой аккаунт. |
| Атрибуты | ID: Int – идентификатор клиента  LastName: Nvarchar(50) – фамилия клиента  FirstName: Nvarchar(50) – имя клиента  MiddleName: Nvarchar(50) – отчество клиента Login: Nvarchar(50) – логин клиента Password: Nvarchar(30) – пароль клиента |
| Операции | Authorize() – авторизоваться в свой аккаунт |

Класс «Paymaster»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс пользователя, который может авторизоваться в свой аккаунт. |
| Атрибуты | ID: Int – Идентификатор кассира  LastName: Nvarchar(50) – фамилия кассира  FirstName: Nvarchar(50) – имя кассира  MiddleName: Nvarchar(50) – отчество кассира Login: Nvarchar(50) – логин кассира Password: Nvarchar(30) – пароль кассира |
| Операции | Authorize() – авторизоваться в свой аккаунт |

Класс «Administrator»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс пользователя, который может авторизоваться в свой аккаунт. |
| Атрибуты | ID: Int – Идентификатор администратора Login: Nvarchar(50) – логин администратора  Password: Nvarchar(30) – пароль администратора |
| Операции | Authorize() – авторизоваться в свой аккаунт |

Конечная диаграмма классов «Авторизация»

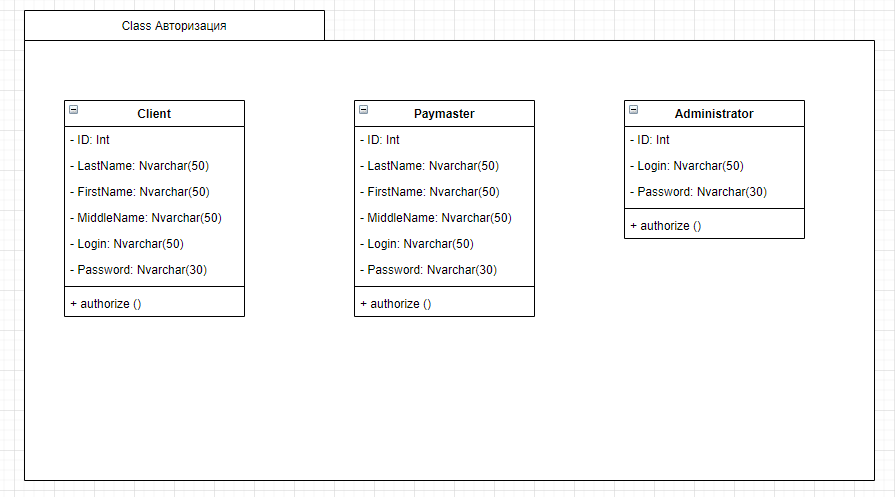


Рис. 4. Диаграмма класса «Авторизация»

Рассмотрим вариант использования «Предоставление информации об операции». Для данного варианта использования необходимо рассмотреть 1 класса – Операция.

Класс «Operation»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, в объектах которого содержится информация. |
| Атрибуты | ID: Int – Идентификатор операции  DateOperation: Datetime – Дата и время операции  Client\_ID: Int – Идентификатор клиента  Paymaster\_ID: Int – Идентификатор кассира  Processing\_Stage: Nvarchar(50) – стадия обработки операции  Type: Nvarchar(50) – тип операции (приход или расход)  Amount: Money – сумма денег, которую клиент хочет получить или вложить |
| Операции | show() – показать информацию об операции |

Рассмотрим вариант использования «Запись и хранение данных». Для данного варианта использования необходимо рассмотреть 1 класса – Операция.

Класс «Operation»

|  |  |
| --- | --- |
| **Параметр** | **Значение** |
| Комментарий | Класс, в объектах которого записывается и хранится информация. |
| Атрибуты | ID: Int – Идентификатор операции  DateOperation: Datetime – Дата и время операции  Client\_ID: Int – Идентификатор клиента  Paymaster\_ID: Int – Идентификатор кассира  Processing\_Stage: Nvarchar(50) – стадия обработки операции  Type: Nvarchar(50) – тип операции (приход или расход)  Amount: Money – сумма денег, которую клиент хочет получить или вложить |
| Операции | Record() – записать данные об операции с последующим её хранением. |

# Диаграммы деятельности

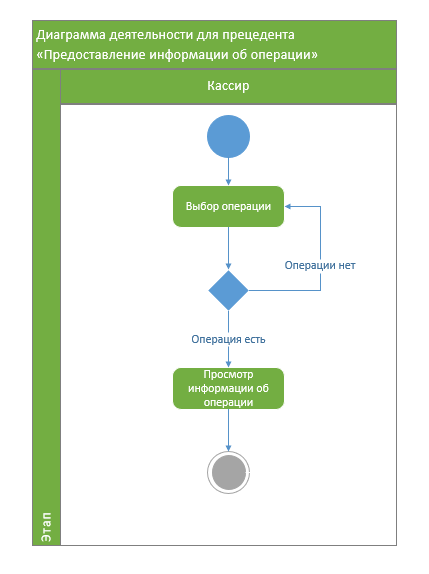


Рис. 5. Диаграмма деятельности «Предоставление информации об операции»

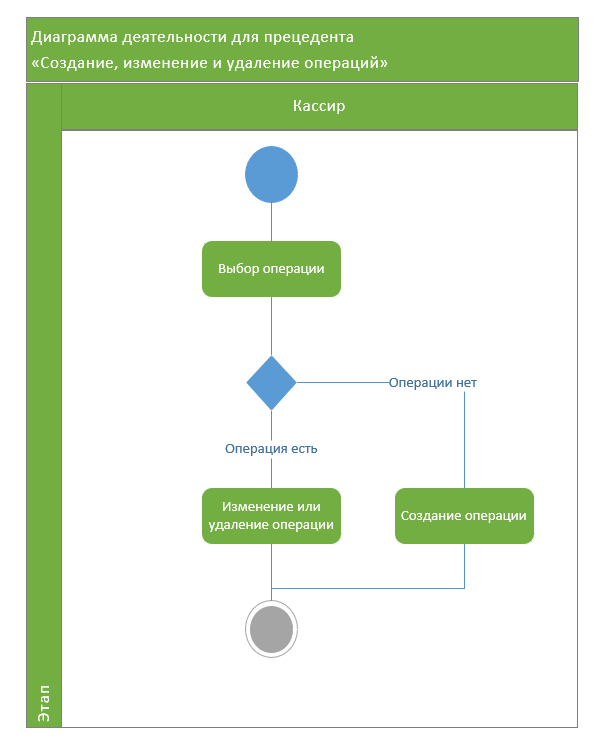


Рис. 6. Диаграмма деятельности «Создание, изменение и удаление операций»

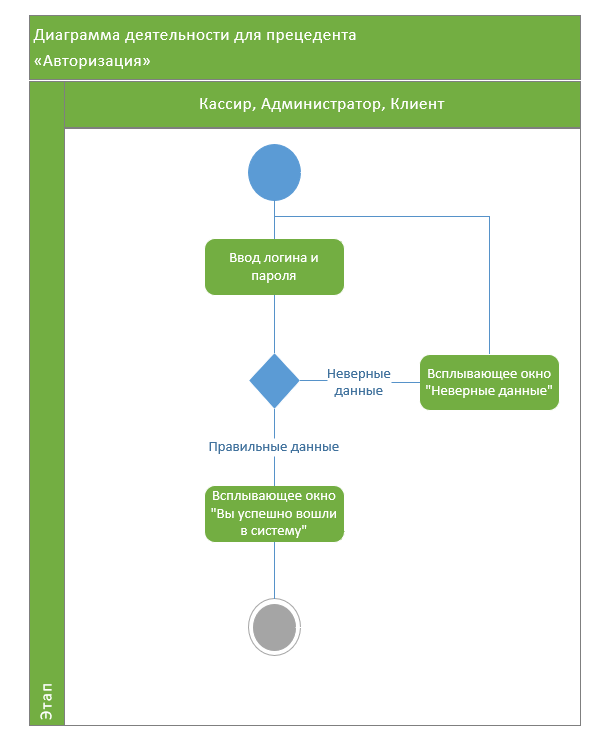


Рис. 7. Диаграмма деятельности «Авторизация»

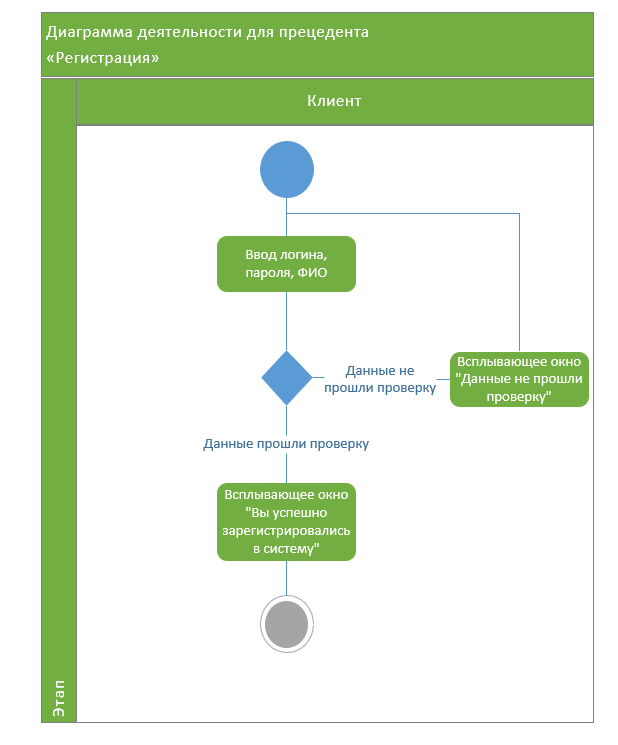


Рис. 8. Диаграмма деятельности «Регистрация»

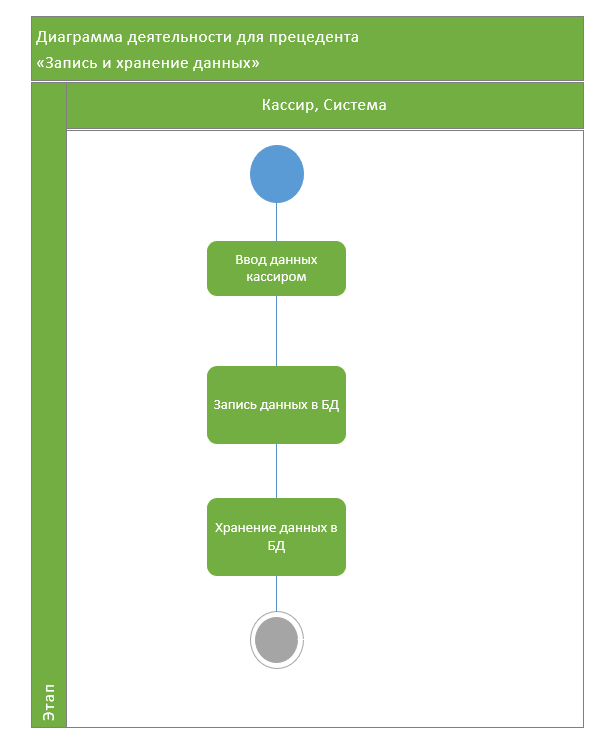


Рис. 9. Диаграмма деятельности «Запись и хранение данных»

# Диаграммы последовательности

Диаграмма последовательности подпотока «Создание операции» для прецедента «Создание и изменение операции»

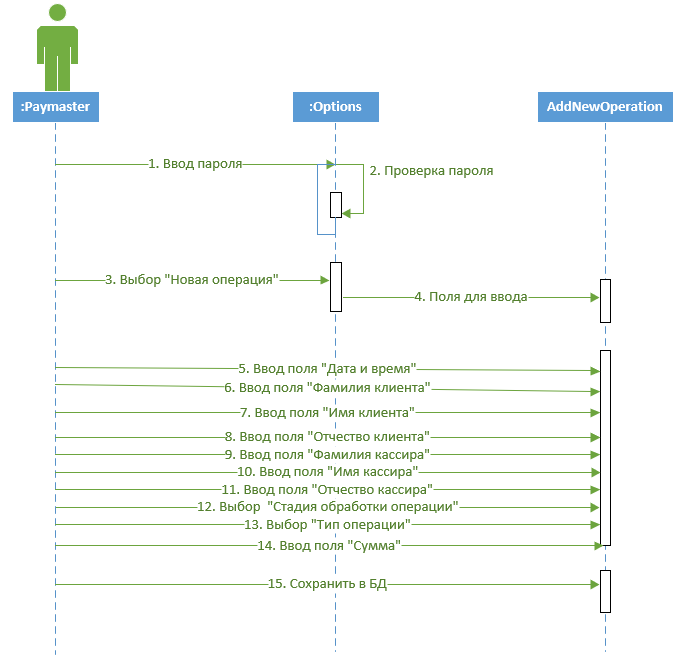


Рис. 10. Диаграмма последовательности подпотока «Создание операции»

Диаграмма последовательности для прецедента «Регистрация»

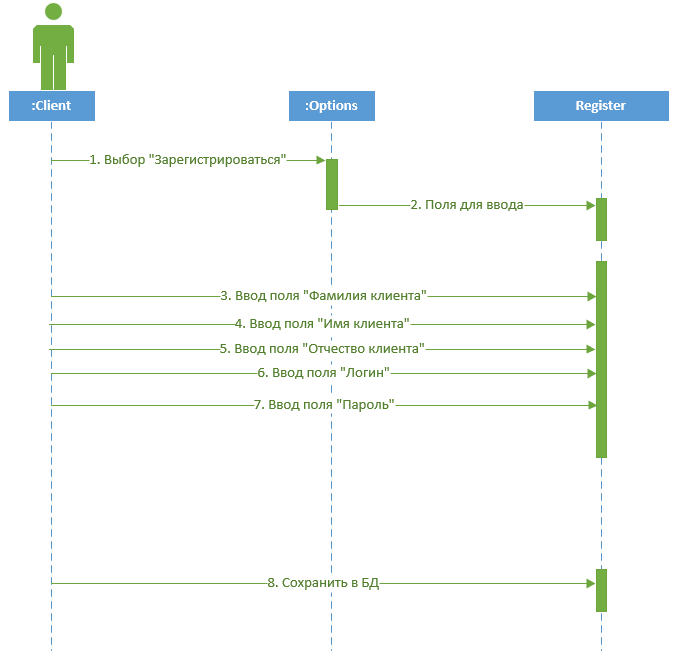


Рис. 11. Диаграмма последовательности прецедента «Регистрация»

Диаграмма последовательности для прецедента «Авторизация»

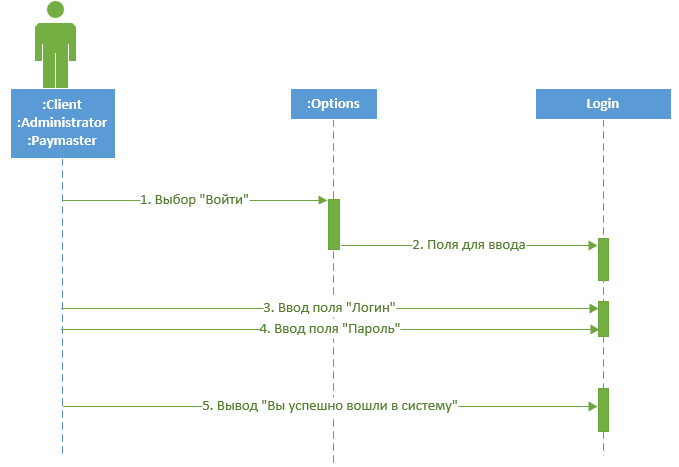


Рис. 12. Диаграмма последовательности прецедента «Авторизация»

Диаграмма последовательности подпотока «Изменение операции» для прецедента «Создание и изменение операции»

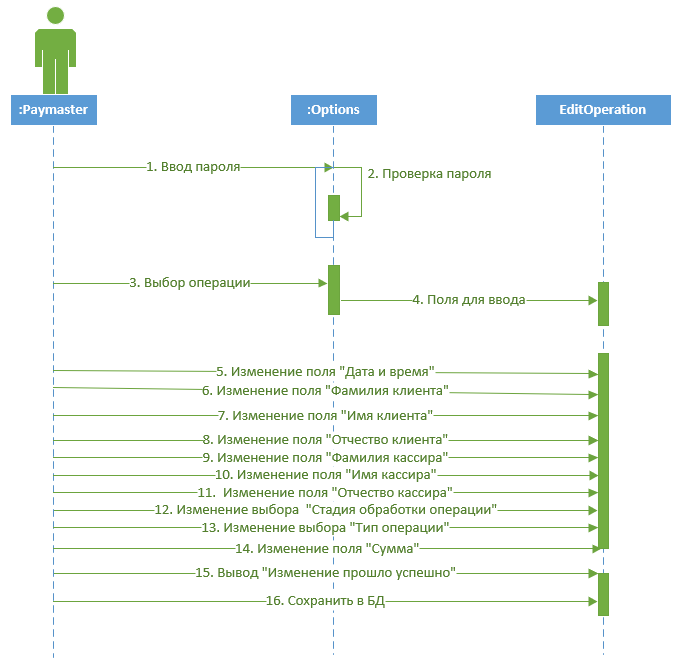


Рис. 13. Диаграмма последовательности подпотока «Изменение операции»

Диаграмма последовательности подпотока «Удаление операции» для прецедента «Создание и изменение операции»

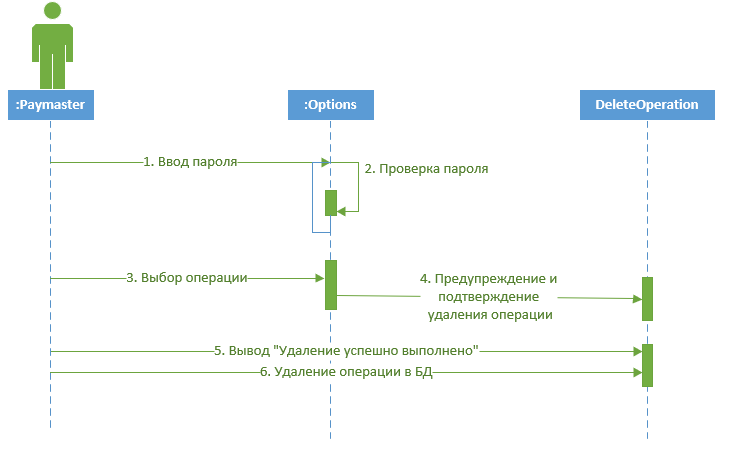


Рис. 14. Диаграмма последовательности подпотока «Удаление операции»

# Диаграммы состояния

Диаграмма состояния для прецедента «Предоставление информации об операции»

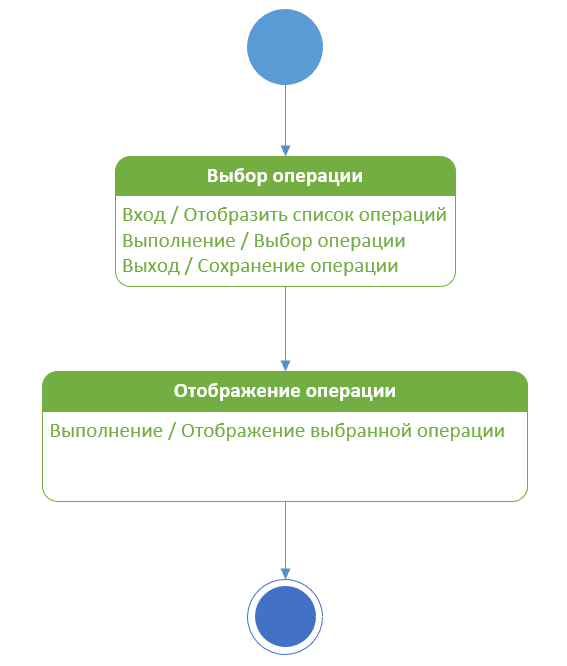


Рис. 15. Диаграмма состояния прецедента «Предоставление информации об операции»

Диаграмма состояния для подпотока «Создание операции» прецедента «Создание, изменение и удаление операции»

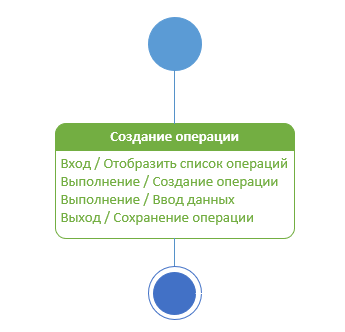


Рис. 16. Диаграмма состояния подпотока «Создание операции»

Диаграмма состояния для подпотока «Изменение операции» прецедента «Создание, изменение и удаление операции»

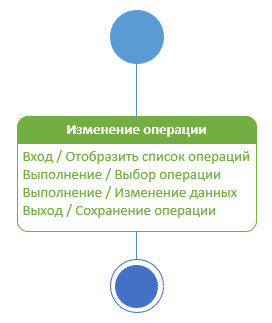


Рис. 17. Диаграмма состояния подпотока «Изменение операции»

Диаграмма состояния для подпотока «Удаление операции» прецедента «Создание, изменение и удаление операции»

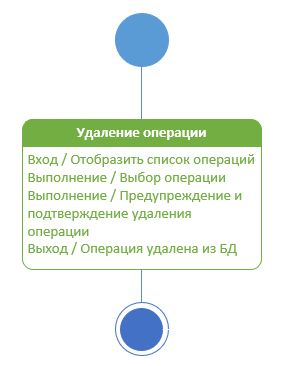


Рис. 18. Диаграмма состояния подпотока «Удаление операции»

Диаграмма состояния для прецедента «Регистрация»

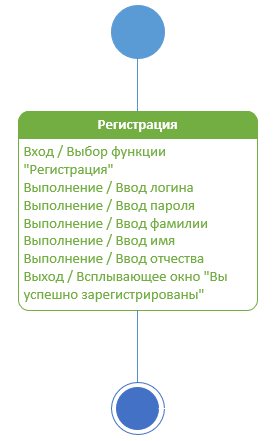


Рис. 19. Диаграмма состояния прецедента «Регистрация»

Диаграмма состояния для прецедента «Авторизация»

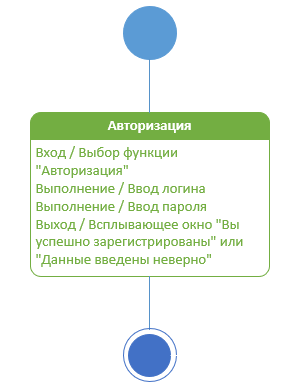


Рис. 20. Диаграмма состояния прецедента «Авторизация»

Диаграмма состояния для прецедента «Запись и хранение данных»

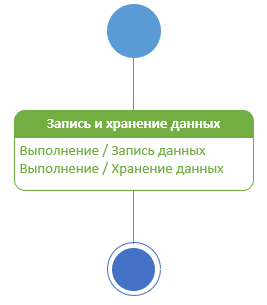


Рис. 21. Диаграмма состояния прецедента «Запись и хранение данных»

# Структурная схема базы данных

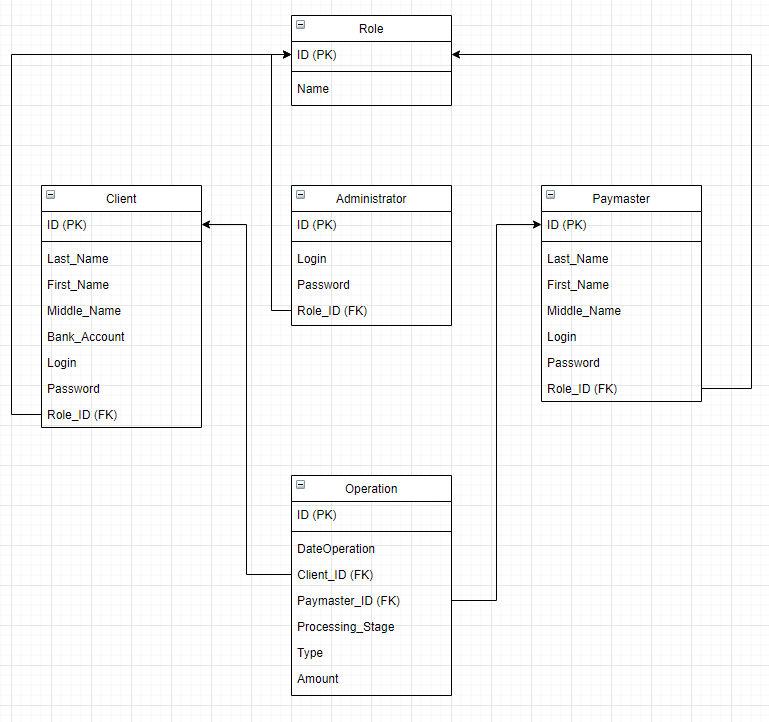


Рис. 22. Структурная схема базы данных

Таблица 1 - «Role»

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Описание |
| ID (PK) | INT | Идентификатор роли |
| Name | NVARCHAR(20) | Название роли |

Таблица 2 - «Administrator»

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Описание |
| ID (PK) | INT | Идентификатор админа |
| Login | NVARCHAR(50) | Логин администратора |
| Password | NVARCHAR(30) | Пароль администратора |
| Role\_ID (FK) | INT | Идентификатор роли |

Таблица 3 - «Paymaster»

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Описание |
| ID (PK) | INT | Идентификатор кассира |
| Last\_Name | NVARCHAR(50) | Фамилия кассира |
| First\_Name | NVARCHAR(50) | Имя кассира |
| Middle\_Name | NVARCHAR(50) | Отчество кассира |
| Login | NVARCHAR(50) | Логин кассира |
| Password | NVARCHAR(30) | Пароль кассира |
| Role\_ID (FK) | INT | Идентификатор роли |

Таблица 4 - «Client»

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Описание |
| ID (PK) | INT | Идентификатор клиента |
| Last\_Name | NVARCHAR(50) | Фамилия клиента |
| First\_Name | NVARCHAR(50) | Имя клиента |
| Middle\_Name | NVARCHAR(50) | Отчество клиента |
| Bank\_Account | MONEY | Банковский счет, в котором указано количество денежных средств |
| Login | NVARCHAR(50) | Логин клиента |
| Password | NVARCHAR(30) | Пароль клиента |
| Role\_ID (FK) | INT | Идентификатор роли |

Таблица 5 - «Operation»

|  |  |  |
| --- | --- | --- |
| Название поля | Тип данных | Описание |
| ID (PK) | INT | Идентификатор операции |
| DateOperation | DATETIME | Дата и время операции |
| Client\_ID (FK) | INT | Идентификатор клиента |
| Paymaster\_ID (FK) | INT | Идентификатор кассира |
| Processing\_Stage | NVARCHAR(50) | Стадия обработки операции |
| Type | NVARCHAR(50) | Тип операции (приход/расход, снятие или вложение) |
| Amount | MONEY | Сумма денег |

# Блок-схемы

Блок-схема «Предоставление информации об операции»

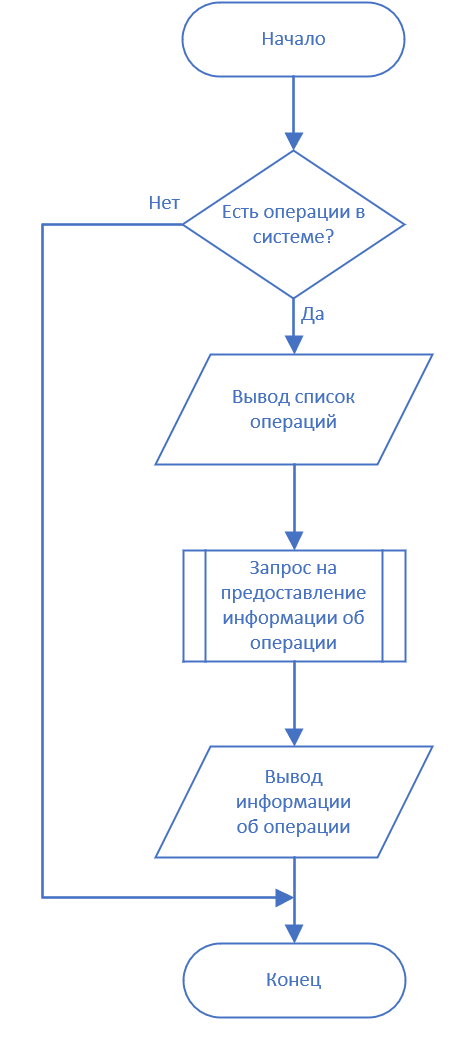


Рис. 23. Блок-схема «Предоставление информации об операции»

Блок-схема «Создание, изменение и удаление операции»

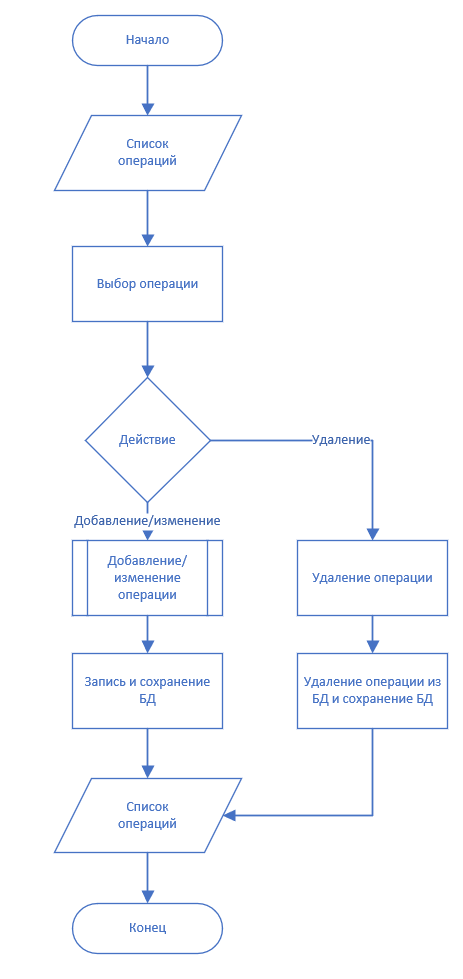


Рис. 24. Блок-схема «Создание, изменение и удаление операции»

Блок-схема подпроцесса «Добавление/изменение операции»

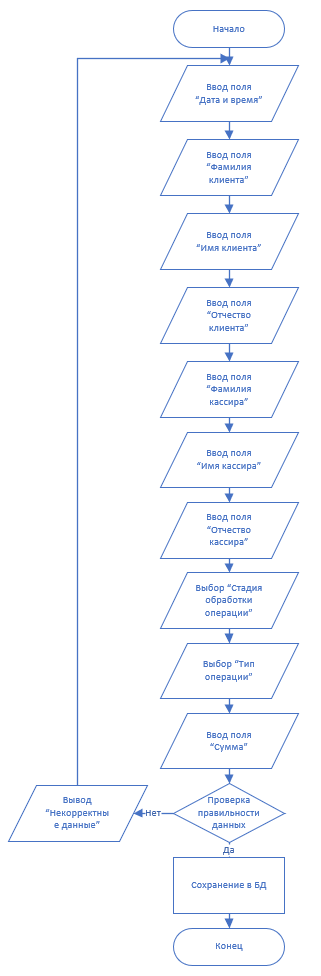


Рис. 25. Блок-схема подпроцесса «Добавление/изменение операции»

Блок-схема «Авторизация»

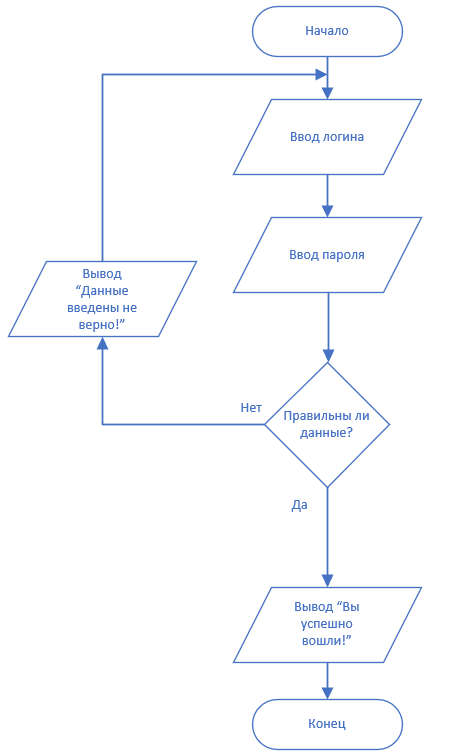


Рис. 26. Блок-схема «Авторизация»

Блок-схема «Регистрация»

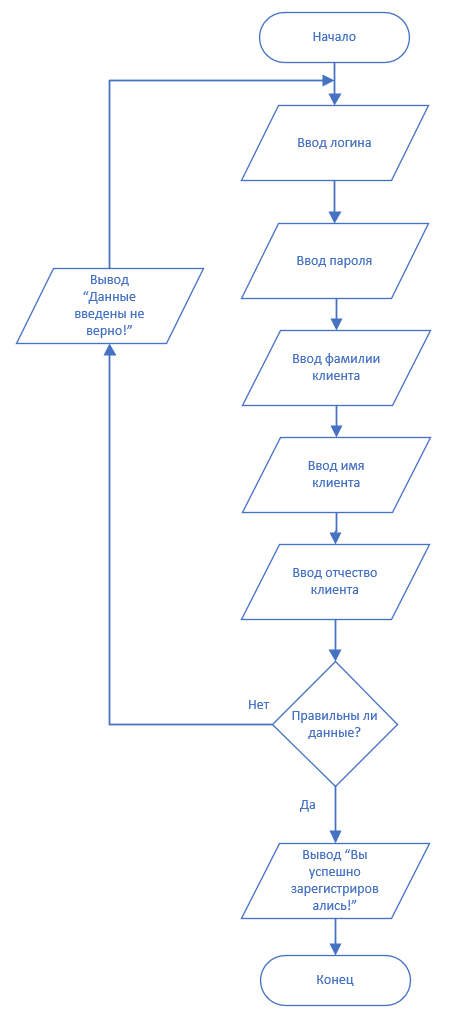


Рис. 27. Блок-схема «Регистрация»

Блок-схема «Запись и хранение данных»

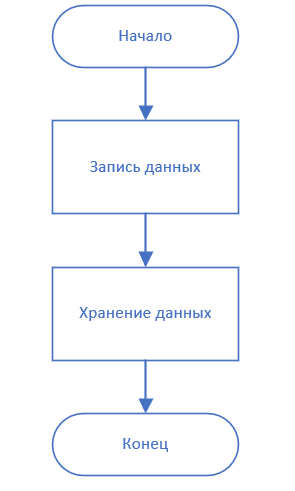


Рис. 28. Блок-схема «Запись и хранение данных»

# Выводы по работе

В процессе выполнения данной расчётно-графической работы была смоделирована система «Учёт банковских операций». Были разработаны прецеденты, диаграмма вариантов использования, диаграммы деятельности, диаграмма классов, диаграммы состояний, диаграмма последовательности системы, структурную схему базы данных и блок-схемы.

# Ответы на контрольные вопросы

1. Что такое Диаграмма деятельности?

Диаграмма деятельности – блок-схема, на которой показываются шаги в выполнении процесса, изображаемые в виде прямоугольников с сопряженными дугами горизонтальными сторонами и переходы между ними, показываемые стрелками.

1. Зачем нужна Диаграмма состояния?

Диаграммы состояний предназначены для представления жизненного цикла объекта в виде конечного автомата, где каждое состояние – это период жизни объекта, когда он удовлетворяет определенным условиям, а некоторое событие может привести переходу объекта в другое состояние, где может выполняться действие, предписанное данному переходу.

1. Что такое стереотипы классов?

Стереотип класса – это элемент расширения словаря UML, который обозначает отличительные особенности в использовании класса. Стереотип используется для обозначения классов-сущностей (классов данных), стереотип описывает пограничные классы, которые являются посредниками между ПС и внешними по отношению к ней сущностями – актерами, обозначаемыми стереотипом <>, а также стереотип описывает классы и объекты, которые управляют взаимодействиями.