

Специальность: 09.02.07 Информационные системы и программирование

Дисциплина/МДК: МДК 03.02 Обеспечение качества функционирования
компьютерных систем

ОТЧЕТ

По лабораторной работе № 3 Тема: Построение диаграммы последовательностей с помощью специализированных языков

Выполнил студент гр.И-9-23

Барган С.Н

Оценка

(оценка прописью)

Проверил преподаватель

Лихторенко О.С.

г.Москва
2024 г.

Краткая теория: Диаграмма последовательностей - это один из видов диаграмм в UML (Unified Modeling Language), который используется для моделирования взаимодействий между объектами в системе с акцентом на порядок выполнения сообщений. Она показывает, как объекты взаимодействуют друг с другом во времени, представляя последовательность сообщений и временные отношения их передачи.

Тема: Диаграмма последовательностей

Цель: Научиться построению диаграммы последовательностей с помощью специализированных языков

Порядок выполнения работы:

1. Определение цели диаграммы.
2. Аудитория.
3. Сбор требований.
4. Установление последовательности сообщений
5. Создание диаграммы

Листинг:

000	@startuml
001	actor Пользователь #lightgreen
002	participant "Интерфейс создания счета" as InvoiceInterface #lightgreen
003	participant "Система управления счетами" as InvoiceSystem #lightgreen

004	participant "База данных счетов" as InvoiceDatabase #lightgreen
005	Пользователь -> InvoiceInterface:: Иницирует создание счета
006	activate InvoiceInterface #pink
007	InvoiceInterface -> Пользователь: Запрашивает информацию для счета
008	Пользователь -> InvoiceInterface: Вводит информацию и отправляет
009	InvoiceInterface -> InvoiceSystem: Передает данные для создания счета
010	activate InvoiceSystem #pink
011	InvoiceSystem -> InvoiceDatabase: Сохраняет информацию о счете
012	activate InvoiceDatabase #pink
013	InvoiceDatabase --> InvoiceSystem: Подтверждение сохранения
014	InvoiceSystem --> InvoiceInterface: Подтверждение о создании счета
015	InvoiceInterface -> Пользователь: Отображает сообщение о создании счета
016	@enduml

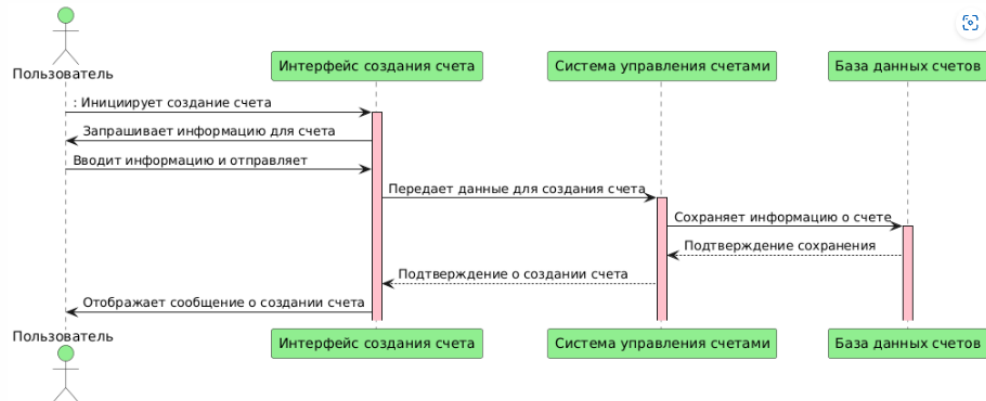
Изображение диаграммы:

```

actor Пользователь #lightgreen
participant "Интерфейс создания счета" as InvoiceInterface #lightgreen
participant "Система управления счетами" as InvoiceSystem #lightgreen
participant "База данных счетов" as InvoiceDatabase #lightgreen
Пользователь -> InvoiceInterface: Иницирует создание счета
activate InvoiceInterface #pink
InvoiceInterface -> Пользователь: Запрашивает информацию для счета
Пользователь -> InvoiceInterface: Вводит информацию и отправляет
InvoiceInterface -> InvoiceSystem: Передает данные для создания счета
activate InvoiceSystem #pink
InvoiceSystem -> InvoiceDatabase: Сохраняет информацию о счете
activate InvoiceDatabase #pink
InvoiceDatabase --> InvoiceSystem: Подтверждение сохранения
InvoiceSystem --> InvoiceInterface: Подтверждение о создании счета
InvoiceInterface -> Пользователь: Отображает сообщение о создании счета
@enduml

```

https://www.plantuml.com/plantuml/png/BLDDJ1985DvpYwckkG@Buy8DQgyerX4ReP0aHw357-YaUX6Ln9k8ecl6cozmf1R-Kr3c7P1Ympwy-Rb-_dU6-x8Z1ze1Wc3r2CjyPvJ3e4OKwd-srtARPxBF2-6nIFVxqdNS81Sagz8RnRF03TGrch_KaAaX9K1OK40nH9g2Whgb45r1bV4UQtZz2k8XYU5VyeT1Grz1e1XAKER8gQkw4S



Вывод: В ходе работы я приобрел навык умения написания диаграммы последовательности