Автономная некоммерческая организация «Профессиональная образовательная организация «Московский Международный Колледж»»

|  |  |
| --- | --- |
| Специальность: | 09.02.07 Информационные системы и программирование |
| Дисциплина/МДК: | МДК Разработка кода |

ОТЧЕТ

к практическому занятию № 3

Тема: Построение диаграммы последовательностей

|  |  |  |
| --- | --- | --- |
| Выполнил студент гр. И-9-23 | \_\_\_\_\_\_\_\_\_\_\_\_ | Исаев М. |
| Оценка | \_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_ |
|  |  | (оценка прописью) |
| Проверил преподаватель | \_\_\_\_\_\_\_\_\_\_\_\_ | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

# Кратка теория

# Кратка теория Диаграммы последовательностей - это один из типов диаграмм, используемых в унифицированном языке моделирования (UML), который показывает взаимодействия между объектами в системе. Они визуализируют порядок сообщений, передаваемых между объектами, тем самым иллюстрируя динамическое поведение системы. Это важно для проектирования и понимания сложных систем

# Легенда

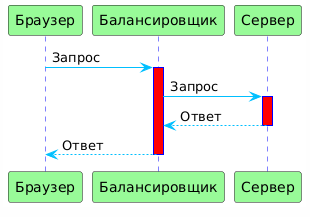
# При построении диаграммы последовательности используются следующие элементы: 1. Объекты - сущности, которые взаимодействуют друг с другом. 2. Сообщения - стрелки, показывающие обмен данными между объектами. 3. Линии времени - вертикальные линии, которые представляют время жизни объекта и порядок вызовов. 4. Параллельные процессы - возможность одновременного выполнения действий

### Цели работы:

 1. Изучение теоретических основ диаграмм последовательностей и их применения.   
2. Овладение инструментами и специализированными языками для построения диаграмм. 3. Создание наглядной диаграммы последовательности для выбранного сценария.  
4. Анализ и интерпретация созданной диаграммы  
  
**Порядок выполнения работы:**

 1. Изучение литературы - Ознакомление с базовыми понятиями UML и диаграмм последовательностей.  
2. Выбор специализированного инструмента - Определение софта для создания диаграмм (например, PlantUML, Lucidchart, Visual Paradigm).   
3. Определение сценария - Выбор конкретного сценария, который будет визуализироваться. 4. Построение диаграммы - Использование выбранного языка или инструмента для создания диаграммы, соблюдая правила нотации.  
5. Анализ диаграммы - Обсуждение результатов, удобство понимания и возможные улучшения

### Листинг 2



# Код

# @startuml participant "Браузер" as A #98FB98 participant "Балансировщик" as B #98FB98 participant "Сервер" as C #98FB98 skinparam sequence { ArrowColor DeepSkyBlue LifeLineBorderColor blue LifeLineBackgroundColor #A9DCDF} A -> B : Запрос activate B #ff0000 B -> C: Запрос activate C #ff0000 C --> B: Ответ deactivate C B --> A: Ответ deactivate B @enduml

### Выводы:

Построение диаграмм последовательностей с использованием специализированных языков позволяет визуализировать взаимодействия в системе, что упрощает процесс проектирования и анализа. Знание правил и инструментов создания диаграмм значительно повышает качество разработки программного обеспечения и понимание его архитектуры. В результате работы был достигнут запланированный результат – создана диаграмма последовательностей, которая помогает лучше понять динамическое поведение выбранного сценария.