Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Проектирование программного обеспечения

Лабораторная работа №7

Тема: «Интерфейс» системы видеоконтроля за «объектом»

Студент: Лемешевский В.О.

ФИТ 3 курс 2 группа

Преподаватель: Курилец А.В.

Минск 2024

**1. Постановка задачи (описание функциональных требований):**

**Оператор**

Настройка объектов наблюдения: оператор должен иметь возможность добавлять и редактировать объекты наблюдения, указывая их название, местоположение и описание.

Управление камерами: оператор должен иметь возможность добавлять, редактировать и удалять камеры, а также настраивать их параметры.

Просмотр видеозаписей: оператор должен иметь доступ к видеозаписям по дате, времени и объекту наблюдения.

**Руководитель системы**

Анализ данных: руководитель должен иметь возможность генерировать отчёты о событиях и активности объектов, используя доступные данные.

Настройка уведомлений: руководитель должен иметь возможность настраивать уведомления о событиях (например, движение или тревога) для определённых объектов.

Мониторинг и аналитика: администратор должен иметь инструменты для мониторинга производительности системы и анализа данных о событиях и активности.

**Администратор системы**

Управление пользователями: администратор должен иметь возможность создавать и управлять учетными записями пользователей, назначать роли и права доступа.

Безопасность и аутентификация: обеспечивать безопасность системы, включая аутентификацию пользователей и управление доступом к данным и функциональности.

Управление базой данных: добавление, удаление обьектов.

В данной лабораторной работе необходимо составить диаграмму развёртывания и компонентов, которая должна содержать: список компонент, из которых будет состоять подсистема, существующие связи между компонентами подсистемы, список физических устройств, на которых будет работать подсистема и определить соединения.

**2. Описание программных средств:**

Draw.io – это бесплатное онлайн-приложение для создания диаграмм и схем. Оно позволяет пользователям создавать диаграммы благодаря широкому набору инструментов и функций.

Название: Draw.io

Версия: Веб-приложение

Разработчик: JGraph Ltd.

Адрес : https://www.draw.io/

Режим использования: Онлайн

Доступность на платформах: любой веб-браузер.

Draw.io позволяет создавать различные типы диаграмм, такие как блок-схемы, организационные диаграммы, UML-диаграммы, сетевые диаграммы и многое другое. Он также поддерживает импорт и экспорт файлов в различных форматах, включая PNG, JPEG, PDF и SVG.

**3. Описание практического задания**

Первой диаграммой была выбрана диаграмма деятельности – рисунок 1. Она посвящена процессу авторизации пользователя и извлечения видеозаписей из архива сервера.

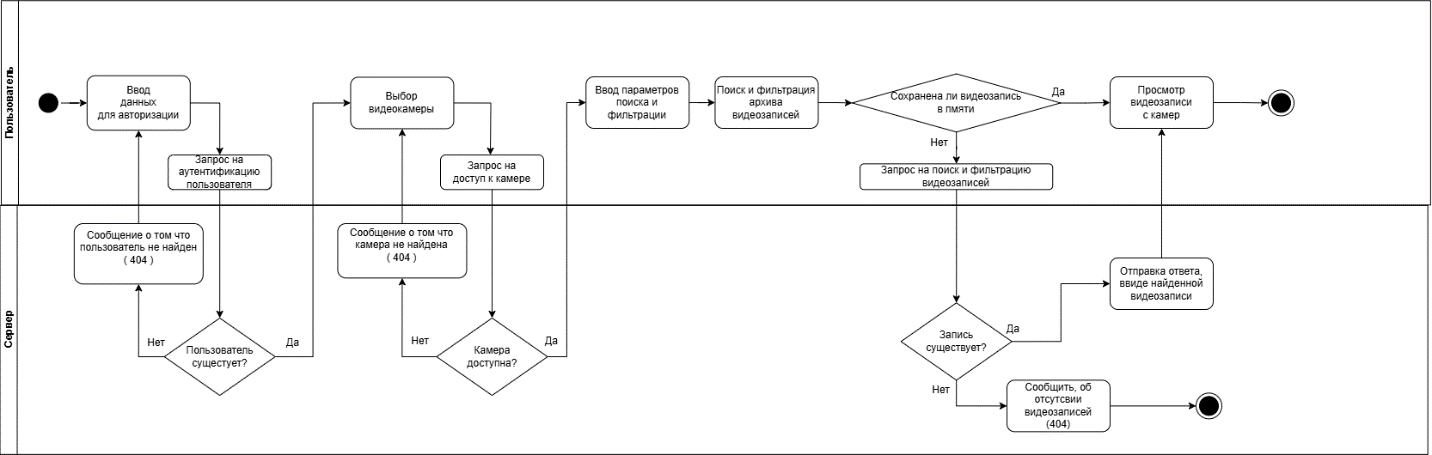


Рисунок 1 – Диаграмма деятельности

Основными элементами диаграммы выступают операции: «Ввод данных для авторизации», «Поиск и фильтрация архива видеозаписей», «Выбор видеокамеры», «Ввод параметров поиска и фильтрации», «Просмотр видеозаписи с камер», а так же использованы блоки для предоставления HTTP запросов и ответов пользователя и сервера по типу «Запрос на аутентификацию пользователя», «Сообщение о том, что камера не найдена (404) » и т.д. Использовано 4 условных блока для проверки авторизации, доступа к камерам, видеозаписям и т.д.

Вторая диаграмма – диаграмма состояния. Отобразим на ней процесс отправки уведомлений о произошедшем событии.

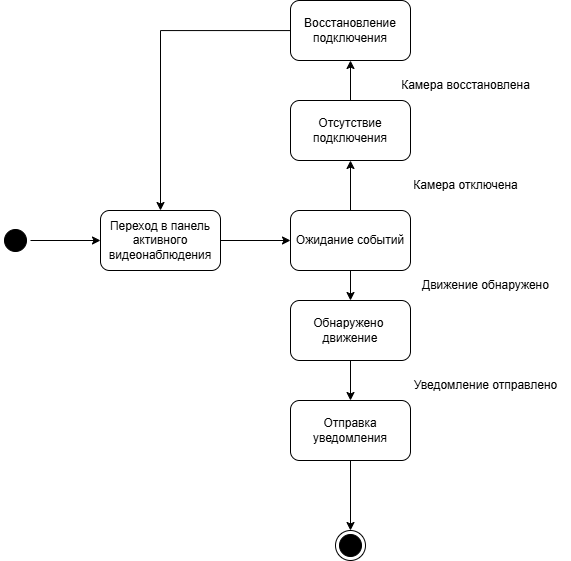


Рисунок 2 – Диаграмма состояния

Основными элементами диаграммы выступают следующие объекты: начальное псевдосостояние, состояния «Переход в панель активного видеонаблюдения», «Ожидание события», «Отправка уведолмения», «Восстановление подключения», а также конечное состояние.

Переходы определяют статус отправки уведомления, обнаружения события, восстановления связи и отключения камеры.

**Вывод:**

Я изучил методологию объектно-ориентированного моделирования средствами UML. Получил дополнительные навыки проектирования моделей информационной системы с применением возможностей UML диаграмм поведения в контексте программного средства «Интерфейс системы видеоконтроля за объектом».

**4. Ответы на теоретические вопросы**

1. **Укажите виды диаграмм поведения.**

Всего существует 3 вида диаграмм поведения:

* диаграмма деятельности;
* диаграмма состояний;
* диаграмма вариантов использования.

1. **Опишите назначение диаграммы деятельности.**

**Диаграмма активностей (видов деятельности)** - один из доступных [видов диаграмм](https://flexberry.github.io/ru/fd_editing-diagram.html), поддерживаемых [Flexberry Designer](https://flexberry.github.io/ru/fd_flexberry-designer.html). Она, как и [диаграмма состояний](https://flexberry.github.io/ru/fd_statechart-diagram.html), **отражает динамические аспекты поведения системы**. По существу, эта диаграмма представляет собой блок-схему, которая наглядно показывает, как поток управления переходит от одной деятельности к другой.

Активности на диаграмме “разбросаны” по беговым дорожкам, каждая из которых соответствует поведению одного из объектов (например, клиента, менеджера, веб-сервера, сервера БД и т.п.). Благодаря этому легко определить, каким из объектов выполняется каждая из активностей. Дорожка - часть области диаграммы деятельности, на которой отображаются только **те активности, за которые отвечает конкретный объект**. Предназначены дорожки для разбиения диаграммы в соответствии **с распределением ответственности за действия**. Имя дорожки может означать роль или объект, которому она соответствует.

**3. Опишите основные нотации, которые используются на диаграмме состояний.**

| **Элемент/Нотация** | **Предназначение** |
| --- | --- |
| Пример | Класс (Class) |
| Пример | Состояние (State) |
| Пример | Состояние (StateEx) |
| Пример | Составное состояние (Composite state) |
| Пример | Разделитель (Concurrent state) |
| Пример | История (History) |
| Пример | Глубокая история (Deep history) |
| Пример | Начальное состояние (Start state) |
| Пример | Конечное состояние (Final state) |
| ПримерПример | Синхронизатор/разветвитель (Complex transition) |
| Пример | Переход (Transition) |
| Пример | Сообщение (Event message) |
| Пример | Точка изгиба связей (Point) |
| Пример | Комментарий (Note) |
| Пример | Коннектор комментария (Note connector) |

**4. Укажите виды связей между объектами на диаграмме последовательностей.**

Синхронное сообщение — отправитель передаёт ход управления актёру-получателю, которому необходимо провести в прецеденте некоторое действие. Пока проводимое получателем действие не будет завершено (не будет получено ответное сообщение), отправитель теряет возможность производить какие-либо действия. Графически изображается как сплошная линия со стрелкой в виде закрашенного треугольника, после которой идёт прямоугольник, отражающий деятельность объекта, в конце которого находится ответное сообщение.

Ответное сообщение — данное сообщение является ответом на синхронное сообщение. Обычно, содержит какое-либо возвращаемое изначальному отправителю значение, также возвращающее ему управление (возможность действовать). Графически изображается пунктирной линией с открытой стрелкой.

Асинхронное сообщение — отправитель передаёт ход управления получателю, которому необходимо провести в прецеденте некоторое действие. Основное отличие от синхронного сообщения состоит в том, что отправитель не теряет возможности совершать другие действия. Графически изображается сплошной линией с открытой стрелкой.

Потерянное сообщение — сообщение без адресата.

Найденное сообщение — сообщение без отправителя.

Последние два вида стрелок (взаимодействий) используются крайне редко. В основном они используются для демонстрации взаимодействия имеющихся объектов в данном прецеденте с внешними системами.

**5. Какая диаграмма позволяет моделировать параллельные вычисления?**

Диаграммы деятельности используются для моделирования бизнес-процессов, технологических процессов, последовательных и параллельных вычислений. или диаграмма прецедентов, описывает отношения между актёрами (действующими лицами) и вариантами использования моделируемой системы (ее возможностями).