

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Институт информатики, математики и электроники
Факультет информатики
Кафедра программных систем

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

«РАЗРАБОТКА АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ МОДЕЛИРОВАНИЯ ВОЛОКОННО-ОПТИЧЕСКИХ ДАТЧИКОВ СО СТРУКТУРНЫМ РЕЗЕРВИРОВАНИЕМ»

по направлению подготовки 02.03.02

Фундаментальная информатика и информационные технологии
(уровень бакалавриата)

направленность (профиль) «Информационные технологии»

Студент _____ А.А. Лукьянов

Руководитель ВКР

к.т.н., доцент _____ Е.В. Сопченко

Консультант _____ Г.И. Леонович

Самара 2018

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Институт информатики, математики и электроники
Факультет информатики
Кафедра программных систем

УТВЕРЖДАЮ
Заведующий кафедрой

_____ А.Н. Коварцев
«___» _____ 2017 г.

ЗАДАНИЕ НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
(бакалавр)

студенту _____ Лукьянов Андрей Александрович
группа _____ 6413-020302D

Тема работы: _____ Разработка автоматизированной системы моделирования
волоконно-оптических датчиков со структурным резервированием

Цель работы (и/или исходные данные) разработать автоматизированную систему
моделирования волоконно-оптических датчиков со структурным резервированием

Структурные части работы (перечень вопросов, подлежащих разработке):

- 1) Произвести анализ предметной области: изучить типовые структуры и основные
принципы проектирования волоконно-оптических датчиков и сенсорных групп с
резервированием (ДСГР), изучить методы и алгоритмы моделирования ДСГР
- 2) Выполнить обзор существующих систем-аналогов
- 3) Разработать информационно-логический проект по методологии UML
- 4) Разработать и реализовать программное и информационное обеспечение, провести
тестирование и отладку
- 5) Оформить документацию по ВКР
- 6) Подготовить презентацию по разработанной системе

Научный руководитель
доцент, кафедра программных систем

_____ (Е.В. Сопченко)
«___» _____ 2017 г.

Задание принял к исполнению

_____ (А.А. Лукьянов)
«___» _____ 2017 г.

ПРИЛОЖЕНИЕ

к заданию на выпускную квалификационную работу бакалавра
студенту А.А. Лукьянову группа № 6413-020302D

Тема: «Разработка автоматизированной системы моделирования волоконно-оптических датчиков со структурным резервированием»

Исходные данные к работе:

1) Характеристики объекта автоматизации:

- 1) объект автоматизации – моделирование волоконно-оптических датчиков и сенсорных групп с резервированием (ДСГР);
- 2) виды автоматизируемой деятельности:
 - процесс визуализации работы с моделью;
 - процесс формирования и обновления базы данных с топологическими схемами;
 - процесс моделирования обрывов оптоволокна;
 - процесс вычисления времени получения сигнала от Брэгговских решеток.
- 3) количество топологий неограниченно;
- 4) количество решеток в топологии от 1 до 1000;
- 5) количество оптоволоконных участков в топологии неограничено;
- 6) количество входных параметров – 2;
- 7) количество ролей пользователей – 1;

2) Требования к информационному обеспечению:

- 1) Обеспечение контроля целостности базы данных.

3) Требования к техническому обеспечению:

- тип ЭВМ – IBM PC совместимый;
- монитор с разрешающей способностью не ниже 800 x 600;
- тактовая частота процессора – не менее 1 ГГц;
- объем оперативной памяти – не менее 256 МБ;
- объём свободного дискового пространства – не менее 2 ГБ;
- клавиатура;

- манипулятор – мышь.

4) Требования к программному обеспечению:

- операционная система – Windows XP/7/8/10;
- СУБД – Oracle 5.1;
- среда программирования – IntelliJ Idea 2017;
- язык программирования – Java 8;
- среда проектирования – StarUML 5.0.

5) Общие требования к проектируемой системе.

5.1 Функции, реализуемые системой:

1) Функции пользователя:

- чтение справк о работе с системой;
- создание новых топологий и редактирование имеющихся с помощью конструктора;
- удаление топологий;
- ввод длины волны решетки и длины оптоволокна при создании и редактировании топологий;
- выбор топологии для проведения моделирования;
- проведение вычислительного эксперимента с математической моделью ДСГР;
- оценка длины волны и времени прихода сигнала от решетки, как параметров достоверности.

2) Общесистемные функции:

- визуализация результатов работы;
- выдача справочной информации о системе.

5.2 Технические требования к системе:

- 1) режим работы – диалоговый;
- 2) температура окружающего воздуха – 18-25°C;
- 3) влажность окружающего воздуха – 45-75%;
- 4) система должна удовлетворять санитарным правилам и нормам СанПин 2.2.2/2.4.2198-07;

- 5) условия работы средств вычислительной техники должны соответствовать ГОСТ 12.1.005, 12.1.007.

Научный руководитель,

к.т.н, доцент

Е.В. Сопченко

РЕФЕРАТ

Пояснительная записка: 40 с, 19 рисунков, 20 источников.

Графическая часть: 18 слайдов презентации PowerPoint.

ВОЛОКОННО-ОПТИЧЕСКИЙ ДАТЧИК, РЕЗЕРВИРОВАНИЕ, БРЭГГОВСКИЕ РЕШЕТКИ.

Объектом исследования является волоконно-оптические датчики на Брэгговских решетках.

Цель работы – разработка автоматизированной системы моделирования волоконно-оптических датчиков со структурным резервированием.

В процессе работы была спроектирована автоматизированная система моделирования волоконно-оптических датчиков со структурным резервированием, позволяющая вычислить степень резервирования каждого датчика в оптоволоконной сети.

СОДЕРЖАНИЕ

Введение	8
1 Описание и анализ предметной области	9
1.1 Основные понятия и определения	9
1.1.1 Волоконно-оптические датчики	9
1.1.2 Линейное резервирование	10
1.1.3 Кольцевые структуры	11
1.1.4 Системное резервирование	12
1.1.5 Надежность оптоволоконной сети	12
1.1.6 Достоверность информации	14
1.2 Описание систем-аналогов	14
1.2.1 Система «Автоматизированная система моделирования и анализа волоконно-оптических датчиков на брэгговских решетках»	14
1.2.2 Система «Trace Mode»	15
1.3 Постановка задачи	17
2 Проектирование системы	19
2.1 Структурная схема системы	19
2.2 Разработка прототипа интерфейса пользователя	20
2.3 Информационно-логический проект системы	24
2.3.1 Диаграмма вариантов использования	26
2.3.2 Диаграмма состояний	27
2.3.3 Диаграмма деятельности	28
2.3.4 Диаграмма последовательности	30
2.3.5 Логическая модель данных	31
2.4 Выбор и обоснование комплекса программных средств	31
2.4.1 Выбор и обоснование операционной системы	31
2.4.2 Выбор и обоснование языка программирования	34
2.4.3 Выбор и обоснование СУБД	36
Заключение	37
Список использованных источников	38

ВВЕДЕНИЕ

Создание перспективных образцов ракетно-космической и авиационной техники требует решения целого комплекса научно-технических проблем, в том числе разработки высокопроизводительных, энергоэффективных, малогабаритных и надежных сенсорных сетей при решении задач многопараметрического контроля, измерения и управления. С точки зрения устойчивости к дестабилизирующим факторам, габаритов, энергопотребления, простоты конструкции и эксплуатационной надежности наиболее перспективными для создания сенсорных сетей являются оптические, оптомеханические и волоконно-оптические датчики (ВОД) [1].

Быстрое развитие автоматизированных систем контроля и управления стало причиной роста потребности в датчиках различных физических величин. Большая надежность, стабильность, помехоустойчивость, долговечность и простота интегрирования в микроконтроллерные системы управления являются не менее важными критериями наравне с метрологическими характеристиками датчика. Всем этим требованиям удовлетворяют волоконно-оптические датчики [2].

Таким образом, помехи в работе волоконно-оптических датчиков могут серьезно нарушить производство, либо стать причиной серьезных отказов, которые могут повлечь за собой опасность для жизни людей. Одной из таких помех является обрыв оптоволокну, по которому идет контрольный сигнал к датчику и ответ от него. Чтобы минимизировать потери от обрывов оптоволокну, необходимо применять методы резервирования при проектировании датчиков. Так как элементов в сети может быть огромное количество, появилась потребность в автоматизированной системе.

Цель работы — разработать автоматизированную систему моделирования волоконно-оптических датчиков со структурным резервированием.

1 Описание и анализ предметной области

1.1 Основные понятия и определения

1.1.1 Волоконно-оптические датчики

Волоконно-оптический датчик или сенсор (ВОС) — небольшое по размерам устройство, в котором оптическое волокно используется как в качестве линии передачи данных, так и в качестве чувствительного элемента, способного детектировать изменения различных величин [3]. Пример волоконно-оптического датчика представлен на рисунке 1.

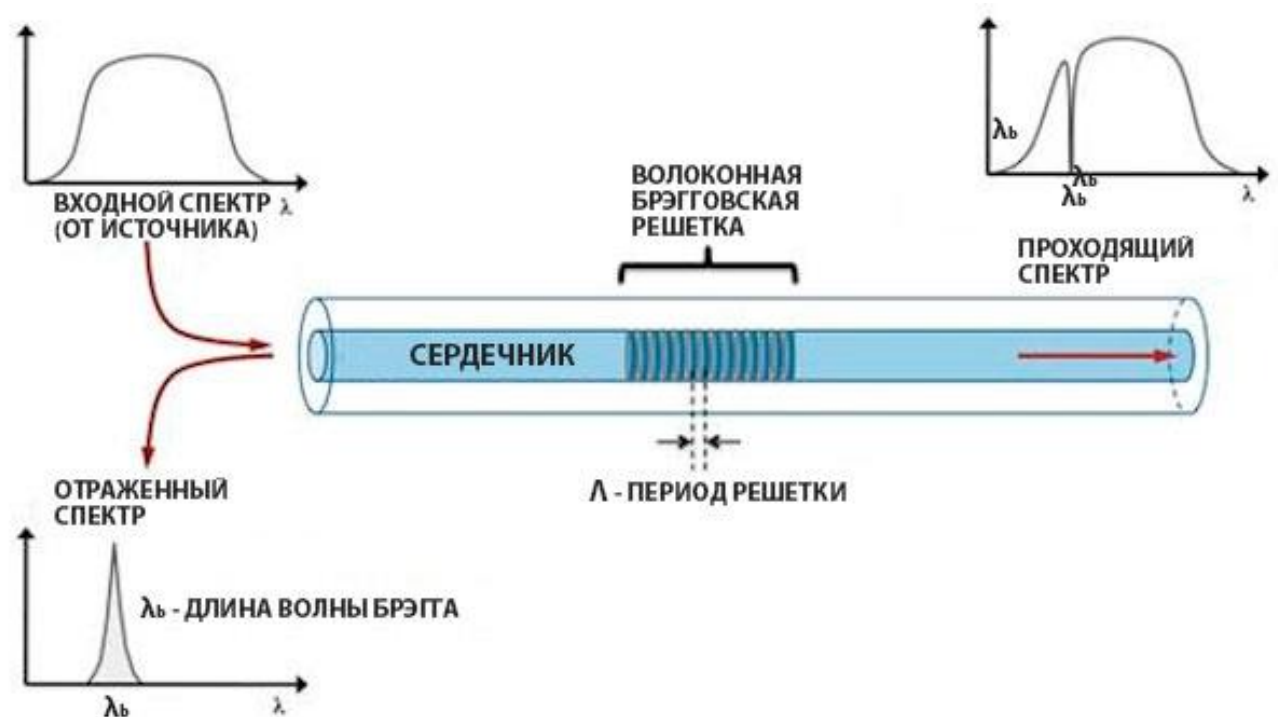


Рисунок 1 – Волоконно-оптический датчик

Решетка Брэгга – элемент, расположенный внутри оптоволокна и имеющий большое количество точек отражения, расположенных на определенном интервале друг от друга. При прохождении через волоконную брэгговскую решетку узкого спектра света, его отражение регистрируется измерительными приборами. Если на оптоволокно воздействуют извне, деформируя его, длина волны отраженного света изменяется, что позволяет отследить изменения измеряемого параметра.

Волоконно-оптические сенсоры могут быть трех типов:

- точечные;
- распределенные;
- квазираспределенные.

Точечные сенсоры позволяют проводить измерения и контролировать параметры в определенной точке. Обычно сенсоры данного типа обладают довольно высокой точностью и небольшими размерами. В зависимости от типа сенсорного элемента локализация сенсоров может достигать $0,1 \text{ см}^2$, как в случае чувствительного элемента на основе решетки Брэгга [4].

Распределенные сенсоры осуществляют постоянный контроль параметров по длине (объему) объекта в любой его точке, где установлен сенсорный световод. Принцип действия таких сенсорных систем основан на анализе изменения параметров по длине световода и на нелинейных эффектах. Недостатком распределения измеряемого параметра по длине является относительно невысокая точность измерения.

Квазираспределенный сенсор представляет собой массив точечных сенсоров, которые основаны на волоконных решетках Брэгга и объединены одним общим световодом. Каждый элемент имеет свои уникальные характеристики, что дает возможность провести анализ его параметров независимо от других элементов. Точность таких систем определяется точностью отдельных сенсоров, а массив может объединять до 100 элементов. Сенсорные массивы позволяют проводить мониторинг сложных объектов, инженерных сооружений, мостов, туннелей, корпусов кораблей и летательных аппаратов. Именно квазираспределенные системы по сравнению с электрическими аналогами обладают малой массой и размерами, что особенно важно для авиации и космонавтики [4].

1.1.2 Линейное резервирование

Высокий уровень надежности современных сетей оптической связи обеспечивается реализацией комплекса различных мер, среди которых

ключевыми являются средства полного или частичного восстановления связи в аварийных ситуациях. Традиционно для этого применялось резервирование — целенаправленное введение в систему определенной избыточности с целью увеличения степени связности отдельных ее узлов, то есть количества независимых путей передачи информации [5].

Увеличение количества передающих трактов, на которые осуществляется переключение при аварийных ситуациях, называется линейным резервированием.

В простейшем случае резервные волноводы находятся в том же кабеле, что и основные. Но это не гарантирует надежность системы. Для уменьшения риска одновременного обрыва основного и резервного кабеля, их прокладывают по разным маршрутам.

Линейное резервирование может быть осуществлено по схеме 1+1 и 1:1. В первом случае информация передается по двум трактам сразу и приемник выбирает сигнал, содержащий наименьшее количество шумов либо наиболее сильный сигнал. В схеме 1:1 передача переключается на резервный тракт сразу при возникновении неисправности в основном [5]. Данная схема линейного резервирования представлена на рисунке 2 [5].

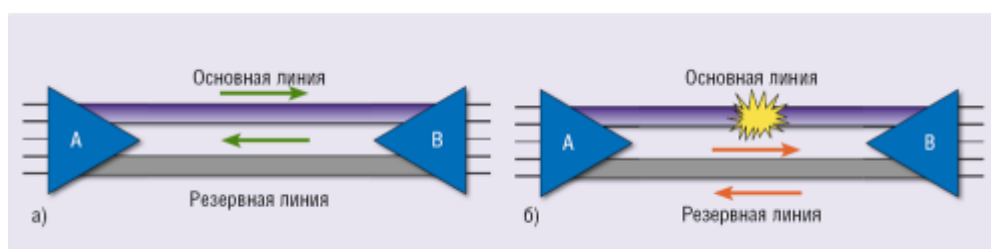


Рисунок 2 – Схема работы участка сети с линейным резервированием
в а) нормальном состоянии, б) при обрыве

1.1.3 Кольцевые структуры

Волоконно-оптические системы часто построены на основе кольцевых топологий. В большинстве случаев линейная часть кольцевой структуры в сетях связи общего пользования строится на основе пары волокон (так называемое сдвоенное кольцо). Таким образом, сигнал может передаваться в

направлении по часовой стрелке или против. Один из маршрутов выполняет роль основного тракта, а другой – резервного.

Резервирование в кольцевых структурах может выполняться по схемам 1+1 и 1:1, что говорит об отсутствии отличий от линейного резервирования. В схеме 1:1 при возникновении обрыва, на узлах, находящихся на границах вышедшего из строя участка, возникает закольцовывание трафика. Пример данной схемы резервирования представлен на рисунке 3 [5].

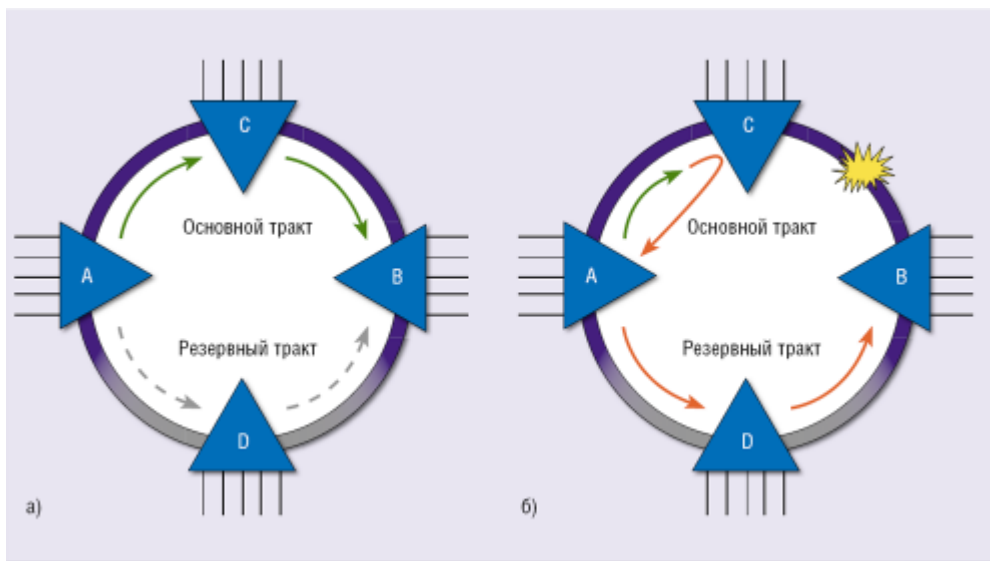


Рисунок 3 – Схема работы участка цепи с кольцевым резервированием
в а) нормальном состоянии и б) после обрыва

1.1.4 Системное резервирование

Системное резервирование представляет собой одновременное увеличение количества дополнительных волокон и приемопередающих устройств. При отказе в одном из приемопередающих устройств или при обрыве основного волокна происходит переключение на резервные волокна. Пример системного резервирования представлен на рисунке 4 [5].

1.1.5 Надежность оптоволоконной сети

Надежность – свойство сети связи выполнять заданные функции, то есть обеспечивать возможность передачи требуемой информации на заданных направлениях с установленной нормами достоверностью в течение требуемого промежутка времени.

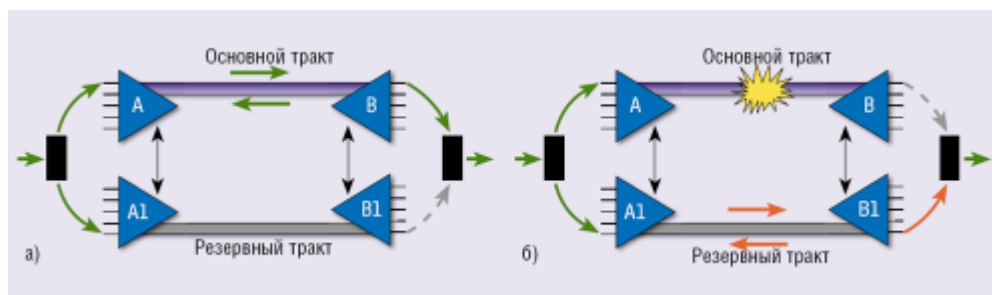


Рисунок 4 – Схема участка цепи с системным резервированием

Рассмотрим оценку надежности по такому комплексному показателю, как коэффициент готовности (K_r), который характеризует безотказность и ремонтпригодность. Если оценивается надежность единичной волоконно-оптической линии связи (ВОЛС), то в случае обрыва оптического кабеля (аварии) время восстановления связи (t_v) будет равняться времени восстановления данного кабеля. Если конкретный оптический кабель является фрагментом сети ВОЛС, то вполне возможно, t_v будет равняться времени, необходимому для организации обходных путей. Это время в большинстве случаев значительно меньше, чем время восстановления оптического кабеля. Основными составляющими t_v можно считать время: на определение места повреждения, на восстановление кабельной канализации, на прокладку аварийной вставки и на монтаж аварийной вставки с комплексом измерений. Время восстановления кабельной канализации может варьироваться от нуля до нескольких часов в зависимости от характера повреждений. Время прокладки аварийной вставки зависит от ее величины.

Коэффициент готовности (K_r) – вероятность того, что сеть связи в произвольно выбранный момент будет исправна.

$$K_r = \frac{T}{T + t_v},$$

где T – среднее время между отказами (наработка на отказ); t_v – среднее время восстановления связи [6].

Четыре фактора, влияющих на коэффициент готовности:

- отказоустойчивость оборудования;
- автоматическое защитное переключение;

- методика и технологическая дисциплина эксплуатации;
- характер трассы и защитные мероприятия [7].

1.1.6 Достоверность информации

Достоверность передачи данных характеризует вероятность получения искажения для передаваемого бита данных. Часто этот показатель называют интенсивностью битовых ошибок (Bit Error Rate, BER). Величина BER для каналов связи без дополнительных средств защиты от ошибок составляет, 10^{-4} — 10^{-6} , в оптоволокне — 10^{-9} . Значение BER в 10^{-4} говорит о том, что в среднем из 10000 бит искажается значение одного бита. Искажения бит происходят как из-за наличия помех на линии, так и по причине искажений формы сигнала ограниченной полосой пропускания линии. Для повышения достоверности передаваемых данных нужно повышать степень помехозащищённости линии, снижать уровень перекрёстных наводок в кабеле, а также использовать более широкополосные линии связи [8].

1.2 Описание систем-аналогов

Обзор систем аналогов показал их малое общее количество. В качестве примера рассмотрим две из них.

1.2.1 Система «Автоматизированная система моделирования и анализа волоконно-оптических датчиков на брэгговских решетках»

Интерфейс системы представлен на рисунке 5.

Можно выделить основные достоинства данной системы:

удобное отображение графиков;
 большое количество параметров, позволяющее гибко настраивать объект моделирования.

Недостатками системы являются:

ограниченный функционал;
 отсутствие возможности моделирования датчиков с резервированием;
 отсутствие ограничения массы датчиков;

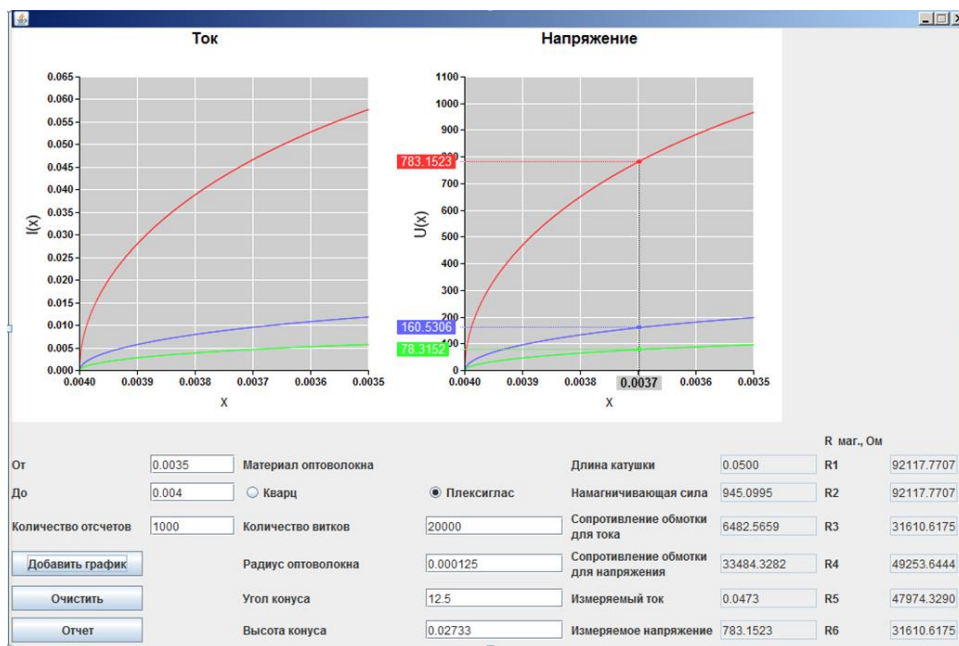


Рисунок 5 – Интерфейс автоматизированной системы

отсутствие показателя достоверности информации.

1.2.2 Система «Trace Mode»

Trace Mode – это высокотехнологическая программная система для автоматизации технологических процессов (АСУ ТП), телемеханики, диспетчеризации, учета ресурсов (АСКУЭ, АСКУГ) и автоматизации зданий [9]. Пример интерфейса программы представлен на рисунках 6 и 7. Данная система обладает следующими достоинствами:

- большой выбор видов объекта моделирования;
- наличие функции резервирования датчиков;
- наличие развитых средств обеспечения надежности и отказоустойчивости.

Недостатками данной системы являются:

- система предоставляется платно;
- высокая сложность обучения из-за наличия большого количества функций.

Рассмотренные системы не реализуют весь необходимый функционал и не пригодны для выполнения поставленной задачи. Поэтому разработка

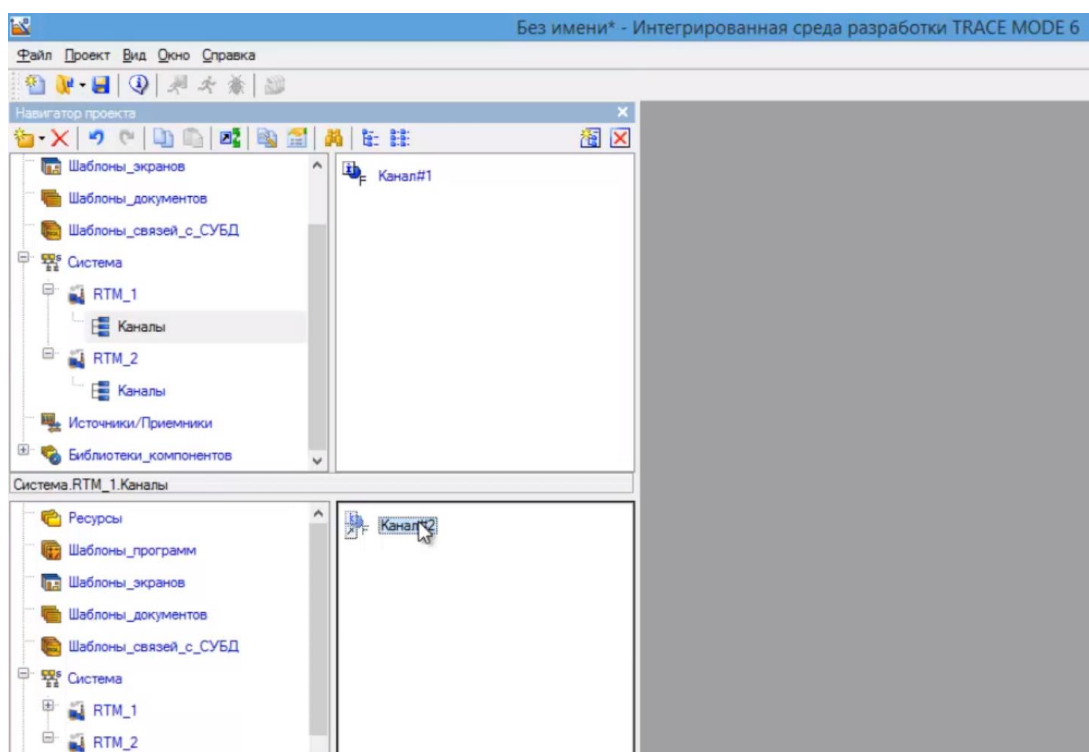


Рисунок 6 – Пример интерфейса системы Trace Mode

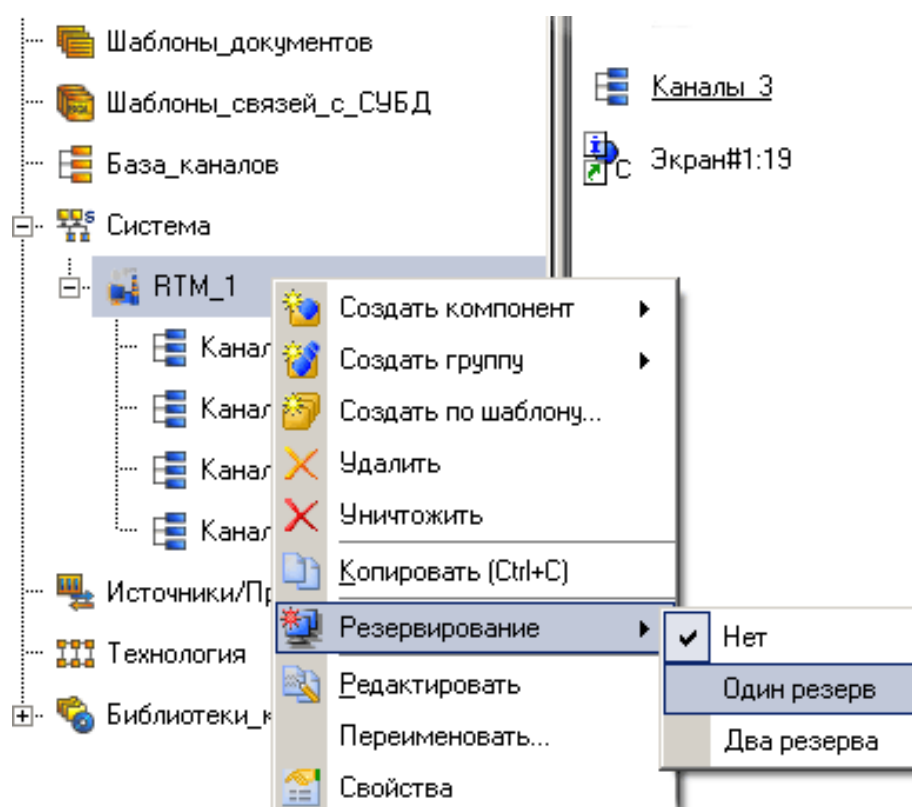


Рисунок 7 – Пример работы системы Trace Mode

автоматизированной системы моделирования волоконно-оптических датчиков со структурным резервированием является своевременной и необходимой.

1.3 Постановка задачи

Перед автором поставлена задача – разработать автоматизированную систему моделирования волоконно-оптических датчиков со структурным резервированием, в которой должны быть реализованы функции администрирования системы, управления входными и выходными параметрами, автоматического формирования параметров, моделирования волоконно-оптических датчиков и сенсорных групп с резервированием (ДСГР).

Система должна представлять собой приложение для персонального компьютера.

В системе должна быть реализована только роль пользователя.

Доступ к системе свободный, без регистрации. Он должен иметь возможность прочитать справку по работе с системой. Также должна быть возможность перейти в раздел работы с топологиями, где пользователь, предварительно выбрав необходимую топологию, должен удалить или изменить топологию. Также должна присутствовать возможность создания новой топологии. При создании и редактировании топологий, система должна предоставлять возможность выбирать топологии и решетки и размещать их на панели редактирования (количество решеток в топологии от 1 до 1000). Также пользователь при размещении элементов должен ввести необходимые данные. В разделе моделирования должна присутствовать возможность выбора необходимой топологии и редактирования ее, и создания новой топологии. После выбора топологии пользователь должен иметь возможность начать моделирование, после чего система должна отобразить результаты проведенного моделирования.

1) Функции пользователя:

чтение справок о работе с системой;

создание новых топологий и редактирование имеющихся с помощью конструктора;

удаление топологий;

ввод длины волны решетки и длины оптоволокна при создании и редактировании топологий;

выбор топологии для проведения моделирования;

проведение вычислительного эксперимента с математической моделью ДСГР;

оценка длины волны и времени прихода сигнала от решетки, как параметров достоверности.

2) Общесистемные функции:

визуализация результатов работы;

выдача справочной информации о системе.

При разработке программного обеспечения будет использоваться технология Rapid Application Development (RAD) – быстрая разработка приложений. RAD предполагает небольшую команду и короткий, но тщательно проработанный производственный график. Разработка осуществляется по спирали, когда по мере создания программы уточняются и реализуются все более и более мелкие детали.

Во время проектирования будут использоваться две основные методологии: методология ООАП (Object-Oriented Analysis/Design) технология разработки программных систем, в основу которой положена объектно-ориентированная методология представления предметной области в виде объектов, являющихся экземплярами соответствующих классов [10] и методология UML графического описания для объектного моделирования в области разработки программного обеспечения, моделирования бизнес-процессов, системного проектирования и отображения организационных структур [11].

2 Проектирование системы

2.1 Структурная схема системы

Структурная схема представляет собой совокупность элементарных звеньев объекта и связей между ними. Схема предназначена для отражения общей структуры программы, то есть её основных блоков, частей и главных связей между ними. Из структурной схемы должно быть понятно, зачем нужно данное приложение и что оно делает в основных режимах работы, как взаимодействуют его части [12].

Приведем структурную схему разрабатываемой системы и описание некоторых подсистем (рисунок 8).

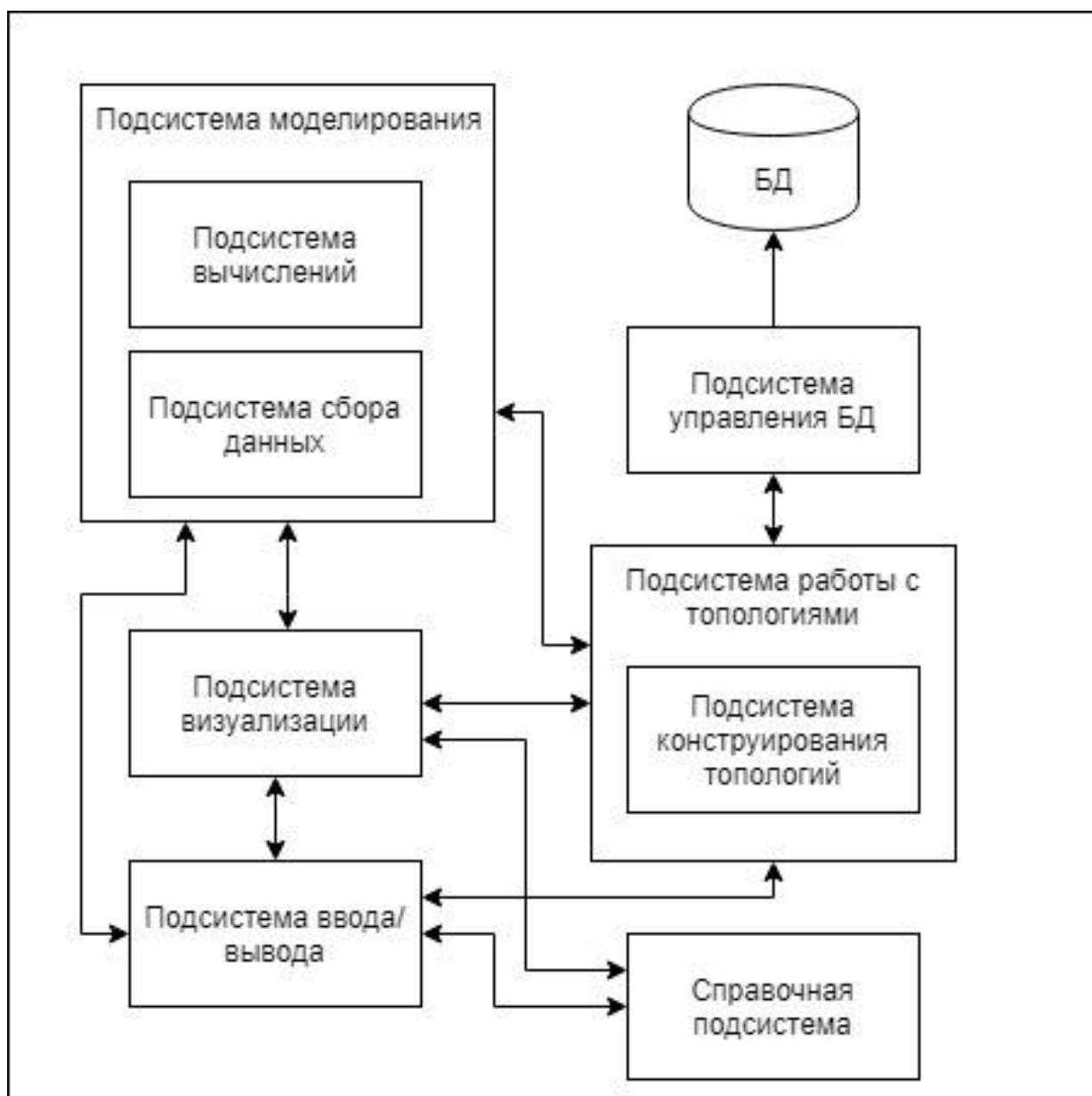


Рисунок 8 – Структурная схема

Система состоит из следующих подсистем:

- 1) подсистема моделирования, отвечающая за непосредственное проведения моделирования, в которую входят подсистема вычислений, выполняющая расчет степени резервирования, и подсистема сбора данных, которая собирает данные после вычислений для отчета;
- 2) подсистема работы с топологиями, отвечающая добавление, изменение и удаление топологий, в которую входит подсистема конструирования топологий, которая отвечает за ввод и размещение Брэгговских решеток и оптоволокна;
- 3) подсистема управления БД, отвечающая за связь с базой данных, в которой хранится информация о топологиях;
- 4) подсистема визуализации, отвечающая за отображение интерфейса пользователя и предоставление доступа к элементам управления системой;
- 5) подсистема ввода/вывода, отвечающая непосредственно за предоставление пользователю возможности вводить данные и видеть ответные данные системы;
- 6) справочная подсистема, отвечающая за предоставление пользователю справки о работе с системой.

2.2 Разработка прототипа интерфейса пользователя

Интерфейс пользователя является одним из важнейших элементов программы, это та часть программы, которая находится у всех на виду. Недочеты в пользовательском интерфейсе могут серьезно испортить впечатление даже о самых многофункциональных программах. Именно поэтому разработке и проектированию пользовательского интерфейса нужно уделять особое внимание.

Для выполнения начальной фазы разработки необходимо погрузиться целиком в задачи пользователей и создать прототип навигационной модели. Навигационная модель показывает, как необходимо распределять функции

или задачи между окнами приложения, она определяет, как пользователи смогут перемещаться как между различными задачами, так и внутри отдельной задачи [13]. Навигационная модель показывает, как необходимо распределять функции или задачи между окнами программы, она определяет, как пользователи смогут перемещаться между различными задачами и внутри отдельной задачи. На рисунке 9 приведена навигационная модель для разрабатываемой системы.

На рисунке 10 представлен прототип главной страницы, которая должна содержать соответствующее программе изображение, название и кнопки «Моделирование» и «Топологии». Пользователь может перейти в соответствующий раздел, нажав одну из этих кнопок. На рисунке 11 представлен прототип страницы работы с топологиями. Она должна содержать выпадающий список топологий, каждую из которых можно выбрать, панель с отображением выбранной топологии, кнопки «Изменить», «Удалить» и кнопка с отображением знака «+», что означает создание новой топологии. Пользователь может выбрать топологию, после чего она отобразится на панели, и удалить, либо изменить ее. При изменении топологии или создании новой, выполнится переход на страницу редактирования, которая представлена на рисунке 12. Страница должна содержать переключатели для выбора брэгговской решетки и оптоволокна, панель редактирования с возможностью перемещать элементы и указывать им необходимые данные и кнопку «Сохранить». После нажатия кнопки «Сохранить» выполнится переход обратно на страницу работы с топологиями. На рисунке 13 представлен прототип страницы моделирования. Она должна содержать список топологий, с возможностью выбора каждой из них, кнопку «+», предназначенную для создания новой топологии, кнопку «Изменить» для изменения выбранной топологии и кнопку «ОК». После нажатия кнопки «ОК» пользователю откроется страница с результатами моделирования, представленная на рисунке 14.

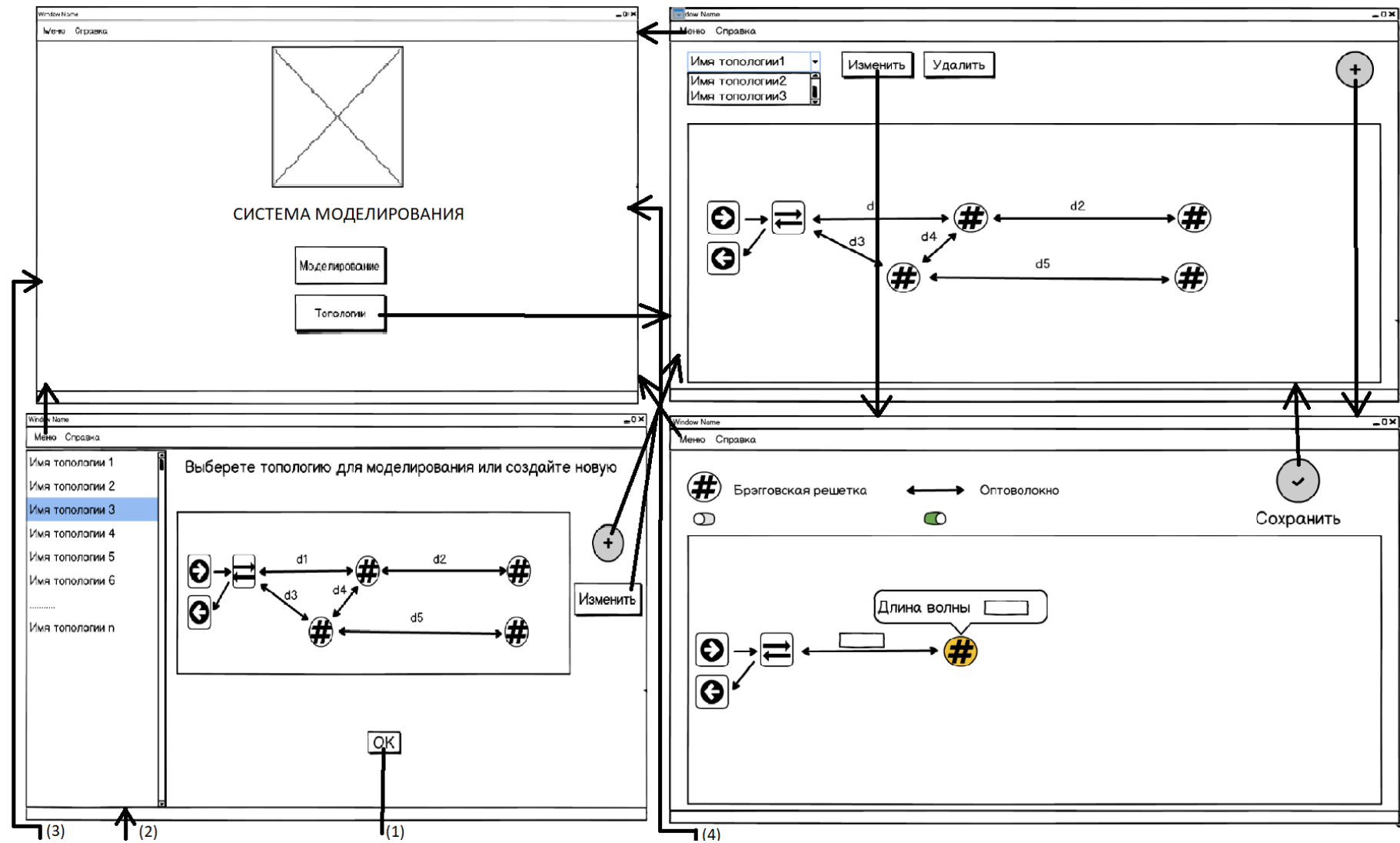


Рисунок 9 – Навигационная модель (начало)

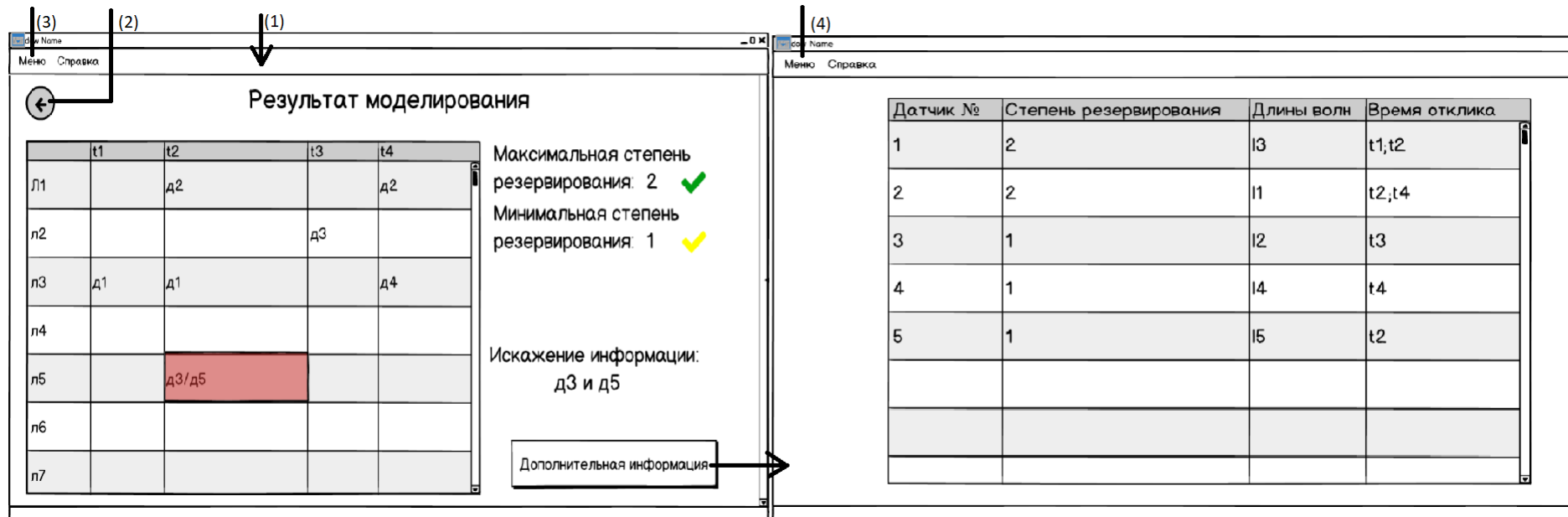


Рисунок 9 – Навигационная модель (окончание)

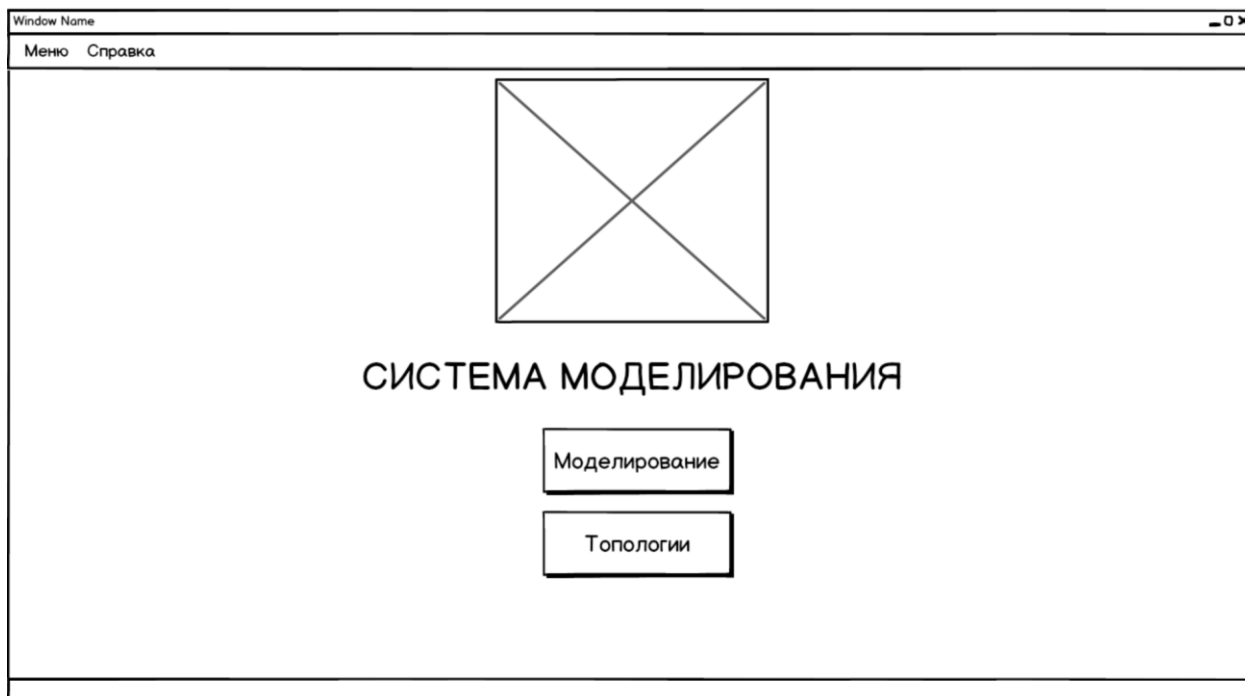


Рисунок 10 – Прототип главной страницы

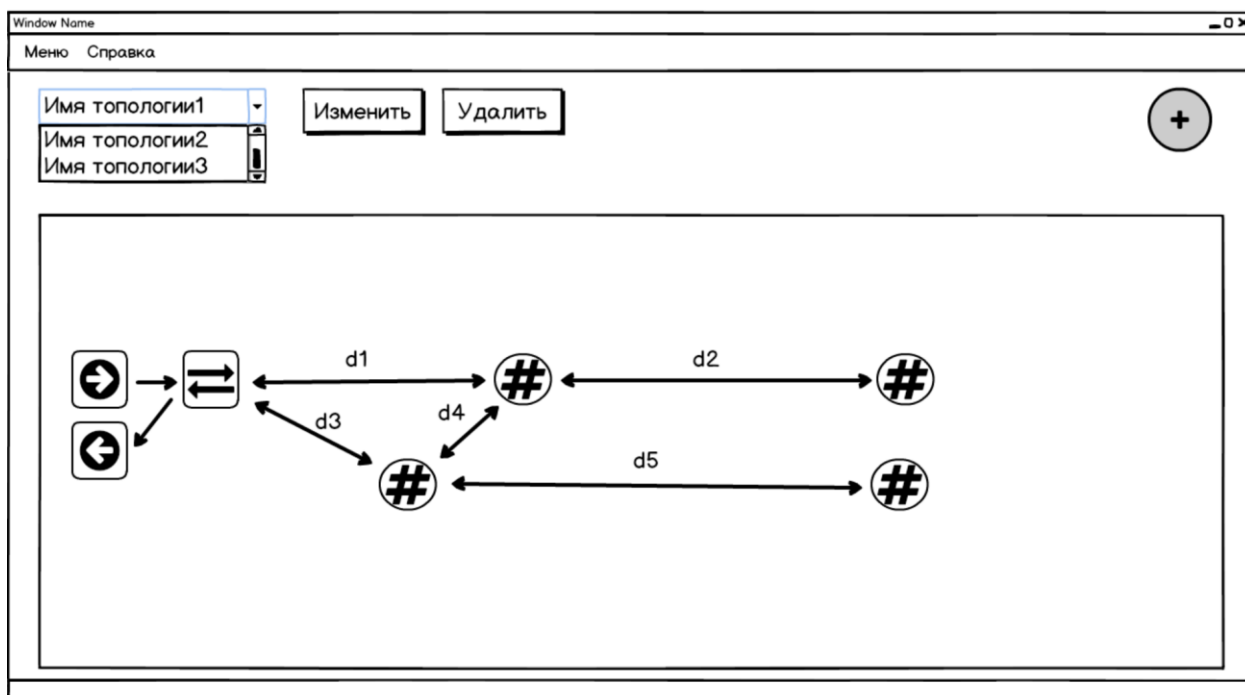


Рисунок 11 – Прототип страницы работы с топологиями

1.1 Информационно-логический проект системы

Унифицированный язык моделирования (UML) является стандартным инструментом для создания "чертежей" программного обеспечения. С помощью UML можно визуализировать, специфицировать, конструировать и

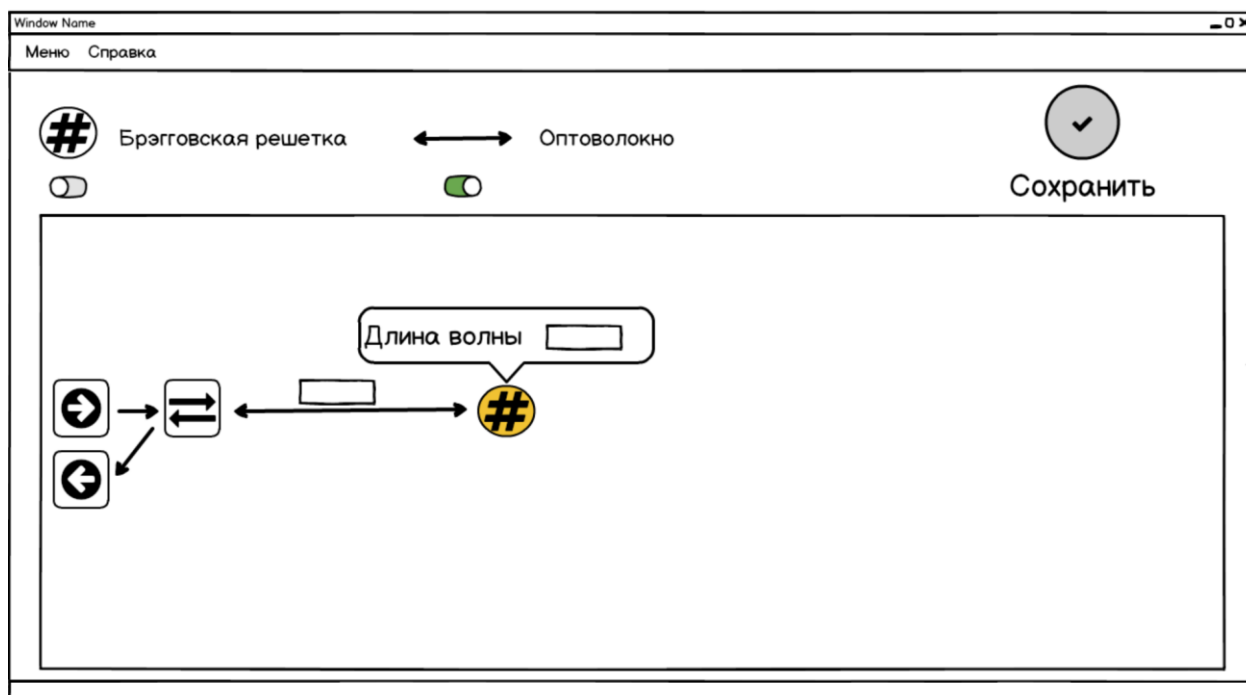


Рисунок 12 – Прототип страницы редактирования топологий

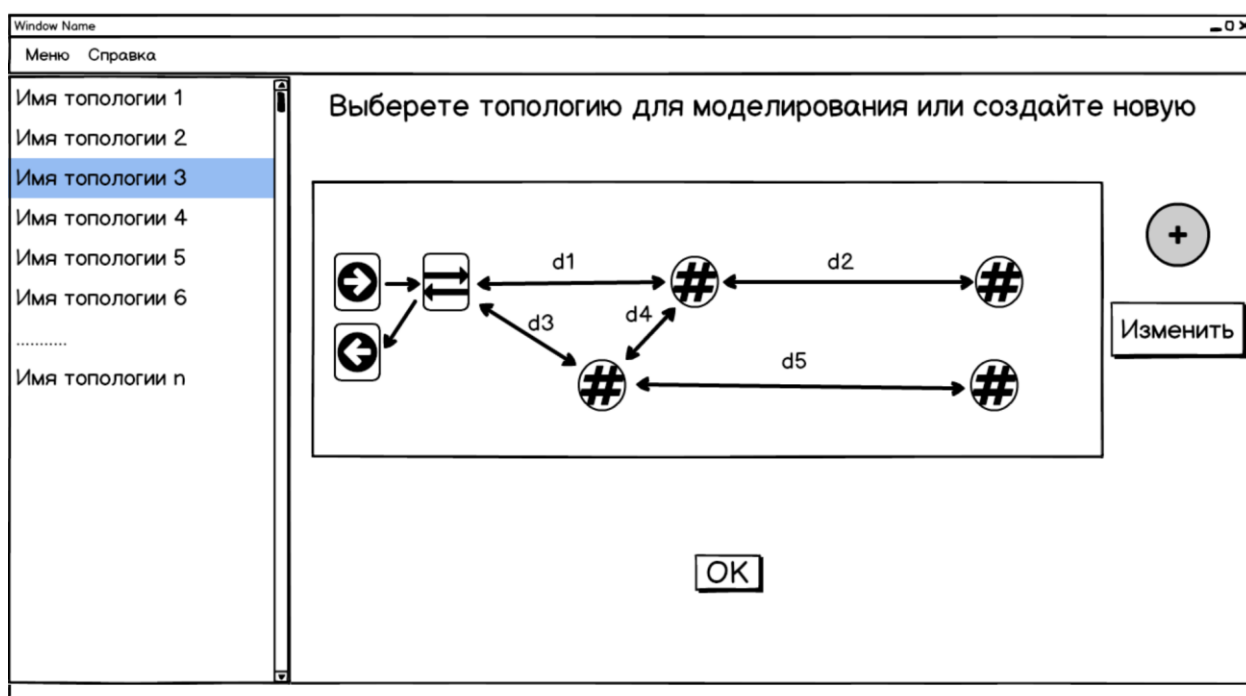


Рисунок 13 – Прототип страницы моделирования

и документировать артефакты программных систем.

UML пригоден для моделирования любых систем: от информационных систем масштаба предприятия до распределенных Web-приложений и даже встроенных систем реального времени [13].

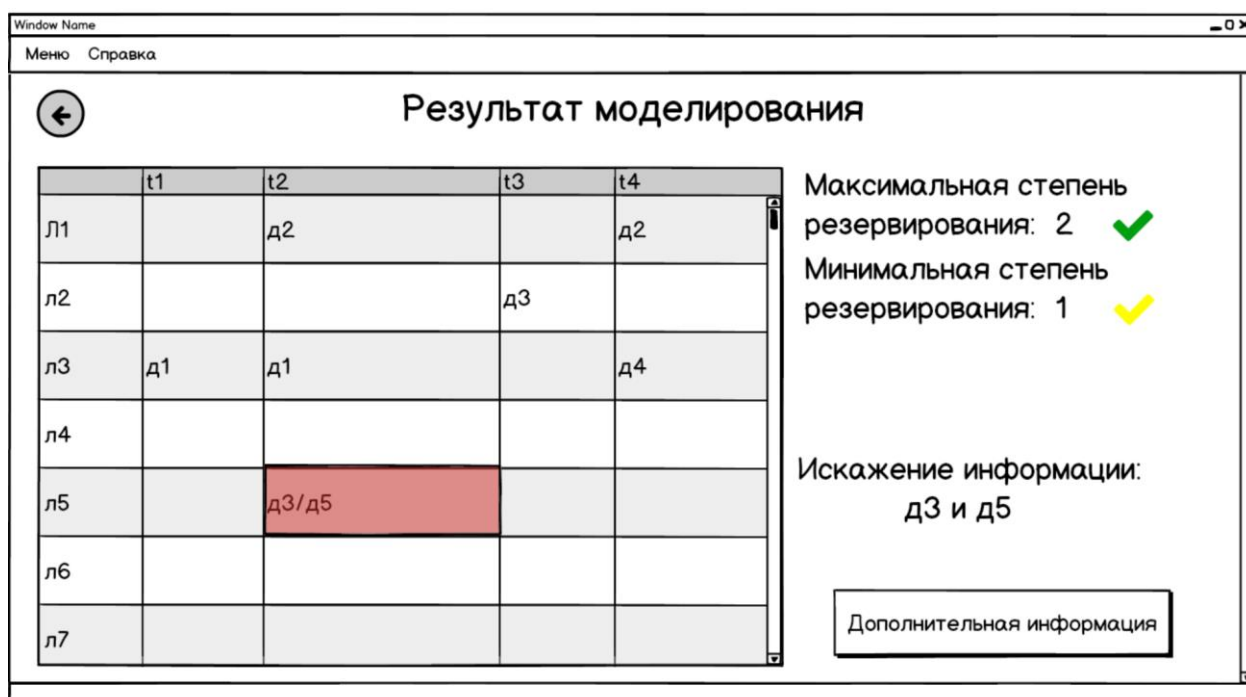


Рисунок 14 – Прототип страницы результатов моделирования

В нотации языка UML определены следующие виды канонических диаграмм:

вариантов использования (use case diagram);

классов (class diagram);

кооперации (collaboration diagram);

последовательности (sequence diagram);

состояний (statechart diagram);

деятельности (activity diagram);

компонентов (component diagram);

развертывания (deployment diagram).

2.2.1 Диаграмма вариантов использования

Диаграмма вариантов использования (use case diagram) – это наиболее общее представление функционального назначения системы.

Создание диаграммы вариантов использования имеет следующие цели [13]:

определить общие границы и контекст моделируемой предметной области на начальных этапах проектирования системы;

сформулировать общие требования к функциональному поведению проектируемой системы;
разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей;
подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Актер (actor) - согласованное множество ролей, которые играют внешние сущности по отношению к вариантам использования при взаимодействии с ними (это может быть любой объект, субъект или система, взаимодействующая с моделируемой бизнес-системой извне, т.е. человек, техническое устройство, программа и т.п.).

Вариант использования - внешняя спецификация последовательности действий, которые система или другая сущность могут выполнять в процессе взаимодействия с актерами (он определяет набор действий, совершаемый системой при диалоге с актером) [13].

На рисунке 15 показаны варианты использования актером «Пользователь». Пользователь может работать удалять топологии, а так же изменять и добавлять их, указывая все необходимые параметры. Пользователь имеет возможность провести моделирование, выбрав перед этим необходимую топологию. Так же имеется возможность ознакомиться со справкой и переходить в главное меню.

2.2.2 Диаграмма состояний

Диаграммы состояний применяются для того, чтобы объяснить, каким образом работают сложные объекты.

Состояние (state) - ситуация в жизненном цикле объекта, во время которой он удовлетворяет некоторому условию, выполняет определенную деятельность или ожидает какого-то события. Состояние объекта определяется значениями некоторых его атрибутов и присутствием или отсутствием связей с другими объектами.

Диаграмма состояний показывает, как объект переходит из одного состояния в другое, и служат для моделирования динамических аспектов системы [14].

На рисунке 16 изображена диаграмма состояний системы. Запустив программу, система находится в состоянии ожидания действий пользователя. Пользователь может перейти в разделы «Моделирование» и «Топологии», после чего система соединится с базой данных, запросит список топологий и выведет его на экран. Также в этих разделах пользователь может создавать и редактировать топологии, переводя систему в состояние редактирования. В этом состоянии пользователь имеет возможность выбирать оптоволоконно и оптоволоконную решетку и помещать их на панель редактирования. После нажатия кнопки «Сохранить», система соединится с базой данных и сохранит новую или обновленную топологию. В разделе «Моделирование» после выбора топологии и нажатия на кнопку «ОК», система перейдет в состояние моделирование, где проводятся математические вычисления. Пользователь может завершить работу с программой путем закрытия окна, нажав на кнопку «X».

1.1.1 Диаграмма деятельности

Диаграмма деятельности (activity diagram) – UML-диаграмма, на которой показано разложение некоторой деятельности на её составные части. Под деятельностью (activity) понимается спецификация исполняемого поведения в виде координированного последовательного и параллельного выполнения подчинённых элементов — вложенных видов деятельности и отдельных действий, соединённых между собой потоками, которые идут от выходов одного узла ко входам другого [15].

На рисунке 17 изображена диаграмма деятельности для пользователя. Пользователь имеет возможность в любое время прочитать справку по работе с программой.

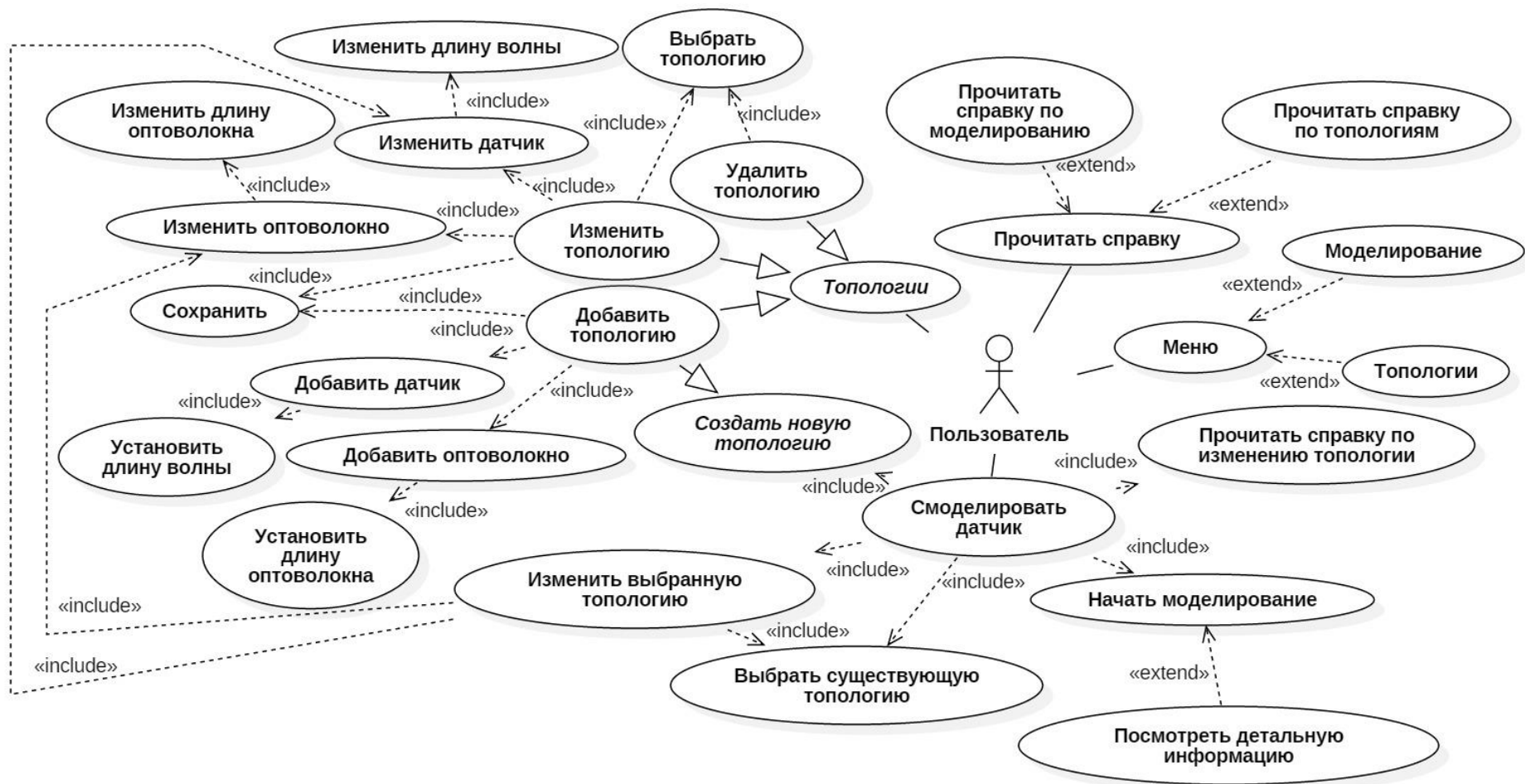


Рисунок 15 – Диаграмма вариантов использования

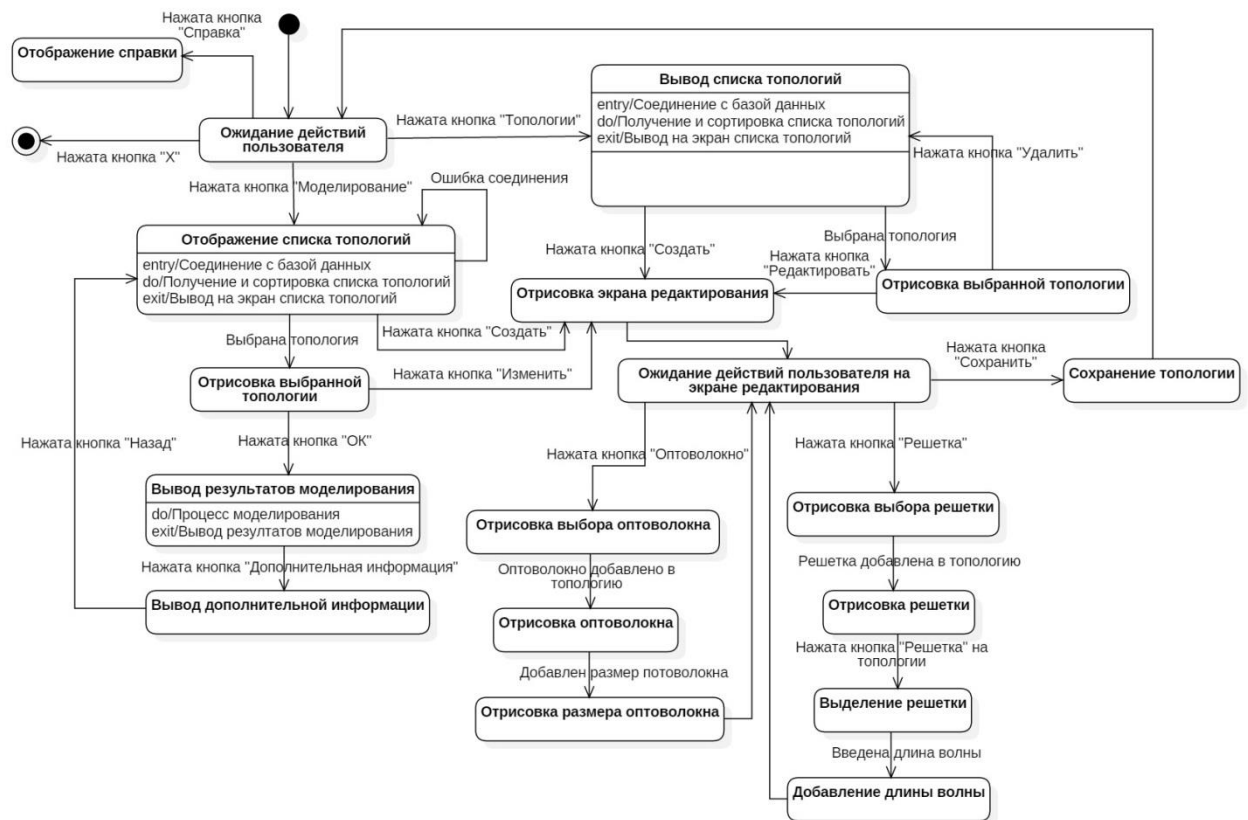


Рисунок 16 – Диаграмма состояний

Если ему необходимо отредактировать он должен зайти в соответствующий раздел и либо изменить или удалить топологию, перд этим выбрав необходимую, либо создать новую. В окне редактирования пользователю следует выбирать оптоволоконно или оптоволоконную решетку и помещать их на панель редактирования, указав для них необходимые данные. Если же пользователю нужно провести моделирование, он должен перейти в соответствующий раздел, выбрать топологию и начать моделирование. В результатах моделирования имеется возможность посмотреть дополнительную информацию. Для закрытия программы пользователь должен нажать кнопку «X».

2.2.3 Диаграмма последовательности

Диаграмма последовательности (*sequence diagram*) – диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл (создание-деятельность-уничтожение) и взаимодействие

(отправка запросов и получение ответов). Таким образом, данный вид диаграмм отображает взаимодействие объектов в динамике [16].

На рисунке 18 представлена диаграмма последовательности для процесса моделирования датчика от запуска программы до конечного результата.

2.2.4 Логическая модель данных

Логическая модель описывает понятия предметной области, их взаимосвязь, а также ограничения на данные, налагаемые предметной областью [17].

На рисунке 19 представлена логическая модель данных разрабатываемой системы. Так как база данных в данной системе играет вспомогательную роль, модель данных состоит всего из четырех сущностей: Брэгговская решетка, в которой находятся брэгговские решетки всех топологий;

смежные вершины, которая содержит решетки, которые соединены оптоволоком, и непосредственно оптоволокну;

оптоволокну, содержащее длину;

топологии, содержащая по одной вершине из каждой топологии.

2.3 Выбор и обоснование комплекса программных средств

2.3.1 Выбор и обоснование операционной системы

В настоящее время Windows добросовестно трудятся на персональных компьютерах во многих странах мира, и является наиболее востребованной. Последние версии оказались стабильными и практичными в работе. Пользователь, не имеющий глубоких знаний, уже при установке системы получает знания о процессах, действиях и комплектации. Во всем многообразии семейства систем, выпущенных в различное время, переход к новой версии интуитивно понятен и не вызывает затруднений [18].

Под операционную систему написано большое количество ПО, как платных программ, так и много бесплатных аналогов. Разработчики приложений в обязательном порядке закладывают совместимость с версиями.

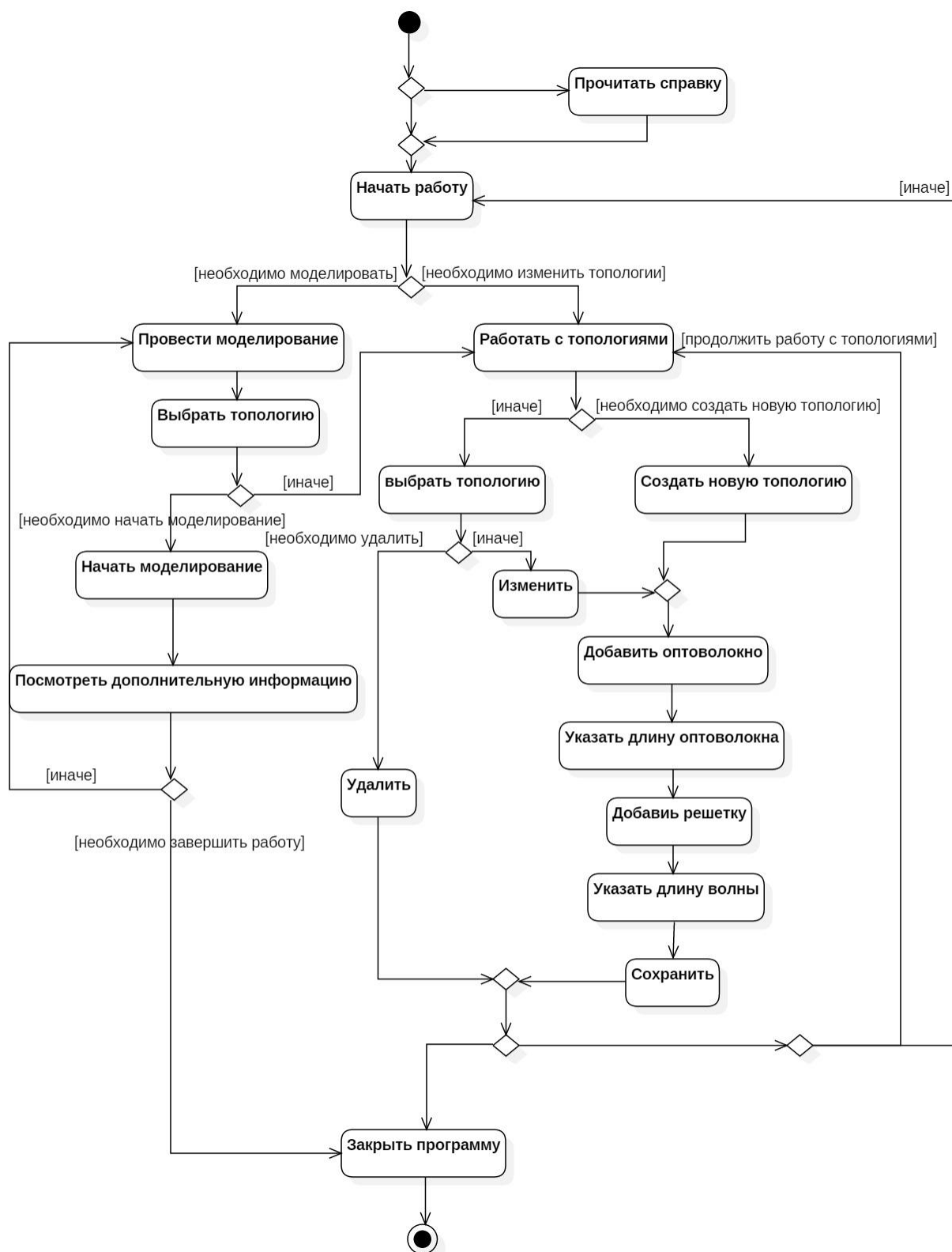


Рисунок 17 – Диаграмма детельности

Опыт, накопленный за длительный промежуток при работе с системой, гигантский. На любые неполадки с ОС или приложением всегда найдется ответ, как исправить ситуацию. Популярность Windows с течением времени

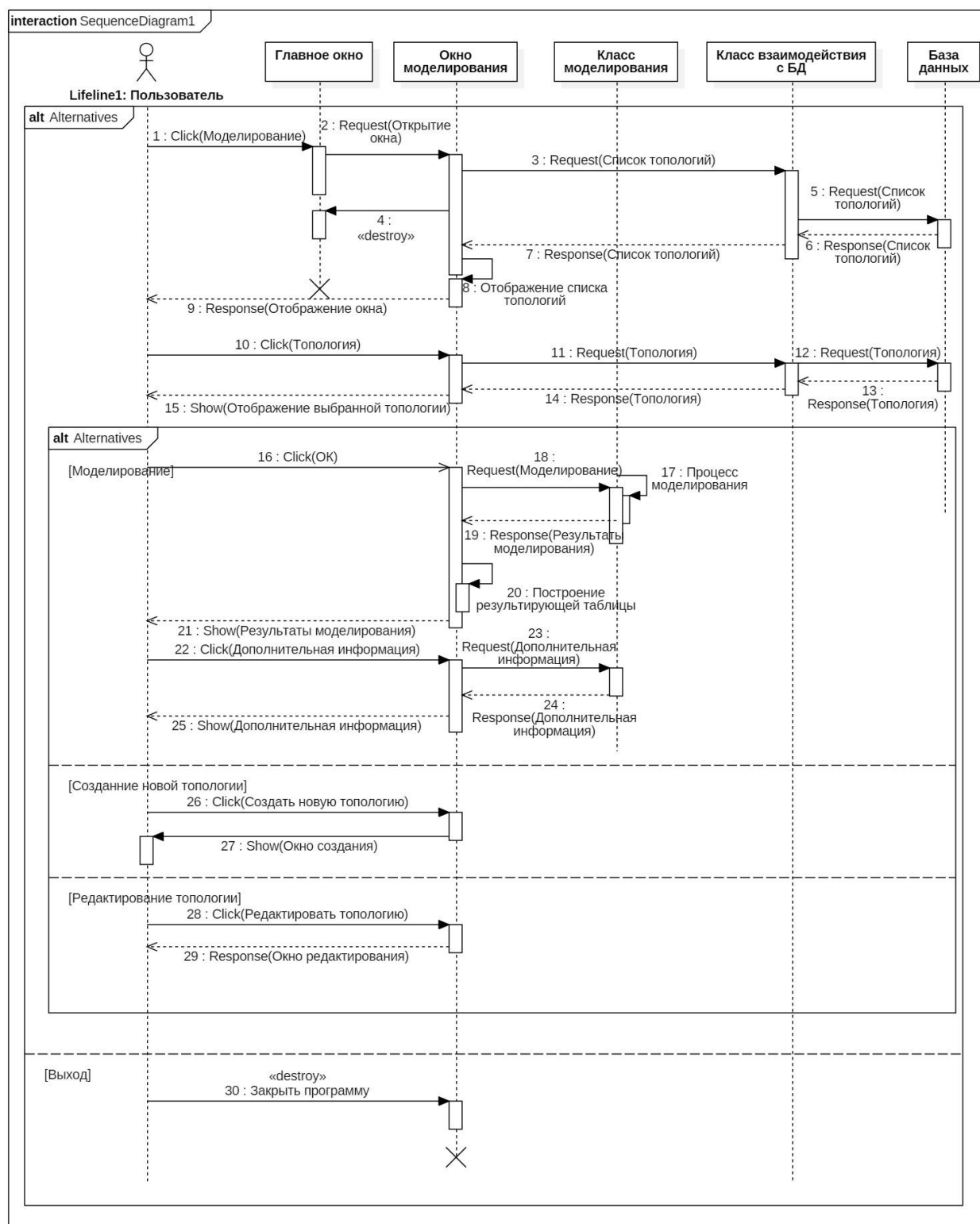


Рисунок 18 – Диаграмма последовательности

не изменяется. Только последняя, 10-я версия, по утверждению официального блога Microsoft, установлена на 67 млн планшетов и компьютеров. Мы живем в информационном мире и роль компании в изменении этого мира переоценить невозможно.



Рисунок 19 – Логическая модель данных

2.3.2 Выбор и обоснование языка программирования

Язык программирования Java – это высокоуровневый объектно-ориентированный язык, разработанный в компании Sun Microsystems. В настоящее время Java-технологии находят широкое применение в различных сферах. Это вызвано особенностями создания и запуска Java-приложений [19].

Достоинства

Java-технологии имеют много особенностей, отличающие их от других технологий разработки программного обеспечения:

1) Переносимость

Программы, написанные на языке Java, после однократной трансляции в байт-код могут быть исполнены на любой платформе, для которой реализована виртуальная Java-машина. Наиболее эффективно возможности реального компьютера может использовать только программа, написанная с использованием «родного» машинного кода;

2) Безопасность

Функционирование программы полностью определяется (и ограничивается) виртуальной Java-машиной. Отсутствуют указатели и другие механизмы для непосредственной работы с физической памятью и прочим аппаратным обеспечением компьютера. Дополнительные ограничения снижают возможность написания эффективно работающих Java-программ;

3) Надежность

В языке Java отсутствуют механизмы, потенциально приводящие к ошибкам: арифметика указателей, неявное преобразование типов с потерей точности и т.п. Присутствует строгий контроль типов, обязательный контроль исключительных ситуаций. Многие логические ошибки обнаруживаются на этапе компиляции. Наличие дополнительных проверок снижает эффективность выполнения Java-программ;

4) Сборщик мусора

Освобождение памяти при работе программы осуществляется автоматически с помощью «сборщика мусора», поэтому программировать с использованием динамически распределяемой памяти проще и надежнее. При интенсивной работе с динамически распределяемой памятью возможны ошибки из-за того, что «сборщик мусора» не успел освободить неиспользуемые области памяти;

5) Стандартные библиотеки

Многие задачи, встречающиеся при разработке программного обеспечения, уже решены в рамках стандартных библиотек. Использование объектно-ориентированного подхода позволяет легко использовать готовые объекты в своих программах.

Для запуска приложения необходима установка JRE, содержащего полный набор библиотек, даже если все они не используются в приложении. Отсутствие библиотеки необходимой версии может воспрепятствовать запуску приложения;

6) Самодокументируемый код

Имеется механизм автоматического генерирования документации на основе комментариев, размещенных в тексте программ;

7) Многообразие типов приложений

На языке Java возможно реализовать абсолютно разные по способу функционированию и сфере использования программы.

2.3.3 Выбор и обоснование СУБД

Oracle Database или Oracle RDBMS – объектно-реляционная система управления базами данных компании Oracle [20].

Важнейшим преимуществом Oracle перед конкурентами (и, прежде всего, перед SQL Server) является идентичность кода (в оценке Gartner Group - консолидация кода) различных версий сервера баз данных Oracle для всех платформ, гарантирующая идентичность и предсказуемость работы Oracle на всех типах компьютеров, какие бы не входили в ее состав. Все варианты сервера Oracle имеют в своей основе один и тот же исходный программный код и функционально идентичны, за исключением некоторых опций, которые, например, могут быть добавлены к Oracle Database Enterprise Edition и не могут - к Oracle Database Standard Edition. Таким образом, для всех платформ существует единая СУБД в различных версиях, которая ведет себя одинаково и предоставляет одинаковую функциональность вне зависимости от платформы, на которой она установлена.

ЗАКЛЮЧЕНИЕ

В процессе выполнения квалификационной работы был проведен обзор предметной области и спроектирована автоматизированная система моделирования волоконно-оптических датчиков со структурным моделированием.

В первой части описаны волоконно-оптические датчики и некоторые виды резервирования, рассмотрены системы-аналоги, отмечены их основные достоинства и недостатки с точки зрения пользователя. Так же была сформулирована постановка задачи.

Во второй части была выбрана и обоснована архитектура системы, описаны структурная схема системы и прототип интерфейса системы. Также разработан информационно-логический проект, в который вошли все канонические диаграммы, был произведен выбор программного обеспечения.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) Электронные и волоконно-оптические элементы и устройства систем управления, измерения и контроля [Электронный ресурс] // Единый портал инновационной деятельности Самарской области. URL: <http://www.startupsamara.ru/Bases/Competences/View?researchId=38> (дата обращения 18.12.2017).
- 2) Жижин В. Волоконно-оптические датчики: перспективы промышленного применения [Электронный ресурс] // Время электроники: [сайт]. URL: <http://www.russianelectronics.ru/leader-r/review/2193/doc/54040/> (дата обращения: 04.04.2018).
- 3) Волоконно-оптический датчик [Электронный ресурс] // Электронная энциклопедия «Википедия». URL: – https://ru.wikipedia.org/wiki/Волоконно-оптический_датчик (дата обращения 10.12.2017).
- 4) А. А. Ключников, А. С. Левицкий, Г. М. Федоренко «Волоконно-оптические информационно-измерительные системы – путь к повышению надежности эксплуатации генераторов АЭС, ТЭС и ГЭС» Киев: Институт электродинамики НАН Украины, 2011. – 9с.
- 5) Увеличение надежности систем оптической связи [Электронный ресурс] // Электронный журнал сетевых решений «LAN». URL: <https://www.osp.ru/telecom/2011/12/13012037> (дата обращения 10.12.2017).
- 6) Эксплуатация ВОЛС [Электронный ресурс] // Сайт фирмы «ЮНИТЕСТ». 2001 – 2016. URL: <http://www.unitest.com/pdf/reliability.pdf> (дата обращения 16.12.2017).
- 7) Надежность волоконно-оптических систем связи [Электронный ресурс] // Студенческая электронная библиотека ЭБС «Консультант студента». URL: <http://www.studentlibrary.ru/doc/ISBN9785991201094-SCN0007.html> (дата обращения 16.12.2017).
- 8) Помехоустойчивость линии [Электронный ресурс] // Электронная энциклопедия «Википедия». URL: https://ru.wikipedia.org/wiki/Помехоустойчивость_линии (дата обращения 16.12.2017).

9) Разработка оптоволоконного датчика электрических параметров на основе решеток Брегга и программного комплекса для автоматического моделирования его параметров [Электронный ресурс] // Самарский национальный исследовательский университет имени академика С.П. Королева. URL: http://repo.ssau.ru/bitstream/Informacionnye-tehnologii-i-nanotehnologii/Razrabotka-optovolokonno-go-datchika-elektricheskikh-parametrov-na-osnove-reshetok-Bregga-i-programmnogo-kompleksa-dlya-avtomaticheskogo-modelirovaniya-ego-parametrov-64059/1/paper%20270_1507-1511.pdf (дата обращения 18.12.2017).

10) Леоненков, А. В. Нотация и семантика языка UML [Электронный ресурс] // А.В. Леоненков. – Интернет-университет информационных технологий. <http://www.intuit.ru/departament/pl/umlbasics> (дата обращения 19.09.2017).

11) UML [Электронный ресурс] // Электронная энциклопедия «Википедия». URL: <https://ru.wikipedia.org/wiki/UML> (дата обращения 19.09.2017).

12) Структурная схема [Электронный ресурс] // Академик: [сайт]. [2000–2017]. URL: <http://dic.academic.ru/dic.nsf/ruwiki/1642106> (дата обращения: 1.10.2017).

13) Зеленко Л.С. Методические указания к лабораторному практикуму по дисциплине «Технологии программирования». Самара: СГАУ, 2014. 65 с.

14) Лекция 3. Виды диаграмм UML [Электронный ресурс] // НОУ Интуит: [сайт]. URL: <http://www.intuit.ru/studies/courses/1007/229/lecture/5954?page=4> (дата обращения: 1.12.2017).

15) Диаграмма деятельности [Электронный ресурс] // Википедия: электронная энциклопедия. 2001-2017. URL: https://ru.wikipedia.org/wiki/Диаграмма_деятельности (дата обращения: 09.12.2017).

16) Диаграмма последовательности [Электронный ресурс] // Википедия: электронная энциклопедия. 2001-2017. URL: https://ru.wikipedia.org/wiki/Диаграмма_последовательности (дата обращения: 18.12.2017).

17) Нормальные формы отношений [Электронный ресурс] // Cit Forum: [сайт]. URL: <http://citforum.ru/database/dblearn/dblearn06.shtml> (дата обращения: 05.04.2018).

18) Windows [Электронный ресурс] // Википедия: электронная энциклопедия. 2001-2017. URL: https://ru.wikipedia.org/wiki/Oracle_Database (дата обращения: 24.12.2017).

19) Java [Электронный ресурс] // Википедия: электронная энциклопедия. 2001-2017. URL: <https://ru.wikipedia.org/wiki/Java> (дата обращения: 24.12.2017).

20) Oracle Database [Электронный ресурс] // Википедия: электронная энциклопедия. 2001-2017. URL: https://ru.wikipedia.org/wiki/Oracle_Database (дата обращения: 24.12.2017).