

Министерство образования и науки Российской Федерации  
ФГБОУ ВО РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ  
ГИДРОМЕТЕОРОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ  
(РГГМУ)

Институт Информационных систем и геотехнологий  
**КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ**

БАКАЛАВРСКАЯ РАБОТА

На тему «Проектирование информационной системы, отслеживающей суда  
в локальной акватории»

**Исполнитель** Быков Данила Денисович

**Руководитель** кандидат технических наук, доцент – Колбина Ольга Николаевна

**«К защите допускаю»**

**Заведующий кафедрой** \_\_\_\_\_

(подпись)

---

(ученая степень, ученое звание)

---

(фамилия, имя, отчество)

«\_\_\_» \_\_\_\_\_ 2023г.

Санкт–Петербург

2023 г.

## Оглавление

ВВЕДЕНИЕ .....	3
ГЛАВА 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ .....	5
1.1 Анализ предметной области .....	5
1.2 Для кого и какой цели создаётся данная ИС? .....	5
1.3 Обоснование необходимости разработки ИС .....	6
1.4 Анализ разрабатываемой системы .....	6
1.5 Анализ аналогов и доступных решений на рынке. ....	7
1.6 Выделение необходимого функционала.....	9
1.7 Сроки реализации проекта .....	11
1.8 Техничко-экономические показатели.....	13
1.9 Выбор CASE-средств и их обоснование.....	16
1.10 Подходы SWOT и VCM .....	17
ГЛАВА 2. ПРОЕКТИРОВАНИЕ ИС .....	20
2.1 Моделирование предметной области.....	20
2.2 Сценарии на основе диаграммы вариантов использования .....	23
2.3 Модель данных.....	31
2.4 Диаграмма последовательности .....	33
2.5 Диаграмма компонентов .....	35
2.6 Диаграмма развертывания .....	37
ГЛАВА 3. РЕАЛИЗАЦИЯ ИС .....	39
3.1 Обоснование выбора средств разработки.....	39
3.2 Метод реализации геолокационного функционала.....	43
3.3 Описание реализации базы данных ИС .....	44
3.4 Описание технологии работы с ИС.....	47
ЗАКЛЮЧЕНИЕ .....	51
СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ .....	53
ПРИЛОЖЕНИЕ А .....	56
ПРИЛОЖЕНИЕ Б.....	60

## ВВЕДЕНИЕ

Современные технологии позволяют людям автоматизировать различные аспекты своей профессиональной деятельности — от автоматической инвентаризации склада до управления сложными механизмами на наукоемких производствах. Однако, не смотря на бесконечную автоматизацию всего и вся, человеческий фактор остаётся неотъемлемой частью процессов, проходящих в различных сферах жизни. Для таких случаев разрабатываются технологии, нацеленные на облегчение профессиональной деятельности.

Одной из таких сфер деятельности является мореплавание. Ежедневно, десятки тысяч судов посещают десятки тысяч портов. Обслуживание судов и акваторий являются сложными процессами, регламентированные как и со стороны закона соответствующего государства, так и со стороны бизнеса, управляющего судами.

Среди всевозможных процессов, протекающих в сфере мореплавания, в качестве примера деятельности, всё ещё зависящей от человека, можно привести отслеживание судов. Далеко не всегда удаётся или является целесообразным оставить слежение за судами автоматизированным системам. Особенно важным участие человека является в отслеживании судов на территории портов и берегов. Какой бы совершенной не была бортовая система торгового судна или техническое обеспечение у берегового поста охраны, они не способны разрешить внеплановые ситуации. В данном контексте, технические средства нацелены на упрощение работы персонала постов наблюдения, береговой охраны, охраны портов и т.п.

*Объектом исследования* является информационная система береговой радиолокационной системы.

*Предметом исследования* является слежение за судами в локальной акватории и документирование параметров их перемещения.

*Целью работы* является разработка информационной системы для береговой радиолокационной системы, способной отслеживать движение судов в локальной акватории, записывать данные движения и просматривать их.

Для достижения данной цели необходимо решить следующие задачи:

- Проанализировать предметную область;
- Создать систему для хранения данных;
- Исследовать способы реализации информационной системы;
- Разработать функционирующий прототип приложения на выбранной платформе.

Методы, стандарты и технологии, примененные в ВКР:

- Сравнительный анализ имеющихся на рынке аналогичных систем;
- Моделирование системы с помощью UML диаграмм;
- Подходы планирования разработки систем SWOT, VCM;
- Технологии баз данных и управления базами данных.

Используемый инструментарий:

В ходе выполнения практической части ВКР был использован следующий стек технологий: diagrams.net, Ramus, DbSchema, Microsoft Visual Studio, Python, PyQt6, MySQL, TCP/IP.

## ГЛАВА 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

### 1.1 Анализ предметной области

Морские сообщения играют огромную роль в мировой торговле. В столь важном экономическом звене надежность и эффективность являются самыми важными факторами. Множество ученых, инженеров и политиков создали необходимую техническую, информационную и правовую инфраструктуру для защиты судов и их операций.

Одной из немногих точек пересечения всех инфраструктур являются электронные навигационные технологии, также известные как «е-Навигация».

е-Навигация – Это комплекс процессов, нацеленных на сбор, обмен и анализ судоходной информации, реализуемые посредством электронных технологий и нацеленных на повышение уровня безопасности, качества и эффективности морской деятельности.

В сравнении как и с историей мореплавания, так и с историей самой е-Навигации, внедрение цифровых и информационных технологий является новшеством. Не смотря на уже выстроенный порядок и правила обмена морской информацией, в данной сфере не прекращаются инновации.

### 1.2 Для кого и какой цели создаётся данная ИС?

Данная информационная система создаётся как дополнение к радиолокационным решениям и может быть использована морскими специалистами как и в порту, так и на судах.

Сама ИС создаётся поверх уже существующего радиолокационного решения, и дополняет его способностью записывать получаемую информацию, воспроизводить её в удобном для пользователя формате и находить среди записей нужные детали. Внедрение поверх радиолокационного решения будет осуществлено системой плагинов, позволяющей работать с любым радаром. Также, в ключевые цели можно выделить уменьшение цены создания

радиолокационного решения за счёт внедрения в итоговую систему компьютера, вместо объединения всех компонентов в проприетарный аппарат.

### **1.3 Обоснование необходимости разработки ИС**

Навигационные технологии являются одним из множества основополагающих столбов современного мира. Большинство об этом не задумывается, но их применение многократно сократило затраты на перемещение людей и грузов. Помимо прямой пользы, сам рынок е-Навигации представляет значимые экономические выгоды. В рамках данного рынка продаются десятки и сотни дорогостоящих решений, решающие ежедневные проблемы в мореплавании.

Разработка навигационных решений всегда будет актуальна, так как перед людьми всегда будут стоять задачи поиска, анализа и документирования различных путей. Также, обилие решений в подобном рынке не является отрицательным качеством, так как многочисленность программных и аппаратных решений дает обильный выбор финальному пользователю.

Разрабатываемая ИС является экспериментальным аналогом существующих решений и предполагает нестандартную итерацию привычного функционала. Дополнение функционала стандартных радиолокационных систем, гибкость в внедрении и удешевление реализации создаст новое решение, которое продолжит конвейер инноваций в мире е-Навигации.

### **1.4 Анализ разрабатываемой системы**

В рамках разрабатываемой информационной системы планируется реализовать следующие функции:

- Отслеживание судов в пределах акватории поста береговой охраны;
- Хранение считываемых данных в течении месяца;
- Возможность просмотреть сохраненные данные в удобном для пользователя формате.

Данный функционал является совмещением агломерата функций судовых локационных систем и систем записи и учета большого количества информации. Для реализации данных функций необходимо учитывать и/или обрабатывать следующие факторы:

- Данные, уникально определяющие судно;
- Данные, описывающие положение, движение и деятельность судна;
- Хранение большого объема навигационных данных;
- Возможность связи с суднами;
- Отображение текущего состояния акватории;
- Отображение сохраненных данных;
- Метод связи с радаром;
- Метод управления функциями радара (при их наличии).

### **1.5 Анализ аналогов и доступных решений на рынке.**

Зачастую, подобные системы являются проприетарными. В следствии этого, получить достоверную информацию о прямых аналогах разрабатываемой ИС не представляется возможным. Тем не менее, можно воспользоваться информацией о косвенных аналогах, так как разрабатываемая ИС опирается на возможности схожих систем.

Таблица 1.1 Сравнение ключевых параметров

Критерии	JRC JUE-87-R	ГОНЕЦ АТ-МН-2.1	Furuno FELCOM 18	«Данная ИС»
Графический интерфейс	+	-	+	+
Встроенная радиолокация	+	+	+	-
Передача данных внешним службам	+	+	+	-
Хранение данных	-	-	+	+

Приведённые в таблице 1.1 системы являются судовыми земными станциями (СЗС), предназначенные исключительно для установки на судно для связи с другими судами и/или береговыми службами. Данная ИС разрабатывается исключительно для наземных радиостанций. Не смотря на разницу в итоговом развёртывании систем, данные примеры служат хорошим примером для сравнения функционала.

Отсутствие встроенной радиолокации не подразумевает его отсутствие. В приведённых примерах, радар является частью итоговой системы, в то время как в разрабатываемой ИС предполагается работа с уже имеющимся радаром при помощи плагинов. Также, отсутствие передачи данных внешним службам отмечено в таблице для выделения разницы работы с данными у приведённых систем, так как они базируются на спутниках, которые либо имеют доступ в интернет, либо подключены к требуемым службам. Само подключение к службам не является обязательным для морской радиолокационной системы,



так как оно может быть реализовано другими методами, и зачастую включается для уменьшения устанавливаемой аппаратуры.

## **1.6 Выделение необходимого функционала**

Поставленные перед разрабатываемой ИС задачи можно разложить на следующий функционал:

- Просмотр карты акватории;
- Отображение судов на карте акватории;
- Просмотр и слежение за судами;
- Запись перемещений судов и их данных;
- Просмотр записей;
- Поиск необходимых записей;
- Подключение радара посредством системы плагинов;
- Просмотр настроек радио;
- Изменение настроек радио.

Также, для технического обслуживания данной системы предполагается следующий функционал:

- Добавление записей;
- Удаление записей;
- Изменение записей;
- Диагностика соединений к периферии (Радиопередатчик, БД).

В следующей таблице приведен набор предполагаемого функционала, разделенного на категории от «Необходимые» функции до «Отсутствующие» (функции, которые не будут реализованы).

Таблица 1.2 Функционал проектируемой системы

Необходимые	Желательные	Возможные	Отсутствующие
Отображение судов на карте	Слежение за одним судом	Сжатие информации для экономии места	Отправка информации внешним службам
Запись информации о судах	Проверка поисковых запросов на валидность	Изменение настроек отображения карты	Коммуникация с экипажем судна
Хранение информации о судах	Настройка подключенного радиопередатчика	Слежение за судами в прямом эфире	Связь со спутниками
Просмотр записей о судах	Удаление просроченных записей	Сохранение изображений карты	Базовые или универсальные плагины
Поиск записей при помощи запросов	Диагностика подключений (к радару и БД)	Экспорт записей в виде репортов	Система разработки и редактирования плагинов
Подключение радара плагинами к системе			

Ключевой акцент сделан на добавляемых функциях, так как их внедрение требует отдельного внимания. Параллельно внедрению функций сохранения и просмотра информации также выделено отдельное место самой важной функции навигационных систем в мореплавании — непосредственная карта акватории.

### **1.7 Сроки реализации проекта**

В ходе проектирования данной ИС придётся решить широкий спектр проблем, на что потребуются определённые временные ресурсы. Для наглядности планируемых сроков реализации проекта была создана диаграмма Ганта, указанная на рисунке 1.1.

Диаграмма Ганта представляет из себя диаграмму, в которой иллюстрируется порядок и объём запланированных работ. В Диаграмму Ганта входят начало и концы сроков работ, длительность предполагаемых работ, последовательность и иерархию всех процессов. Предварительная подготовка перед началом работы это тоже основа для планирования. Диаграмма Ганта очень наглядно показывает процесс перехода между целями и результатами.

Хороший анализ предметной области и достаточное проектирование позволит точно определить все необходимые аспекты для учёта при реализации, что позволяет сократить время на реализации. Для ещё больших сокращений времени реализации, будет применено автоматическое тестирование, что позволит ещё сократить суммарное время реализации.

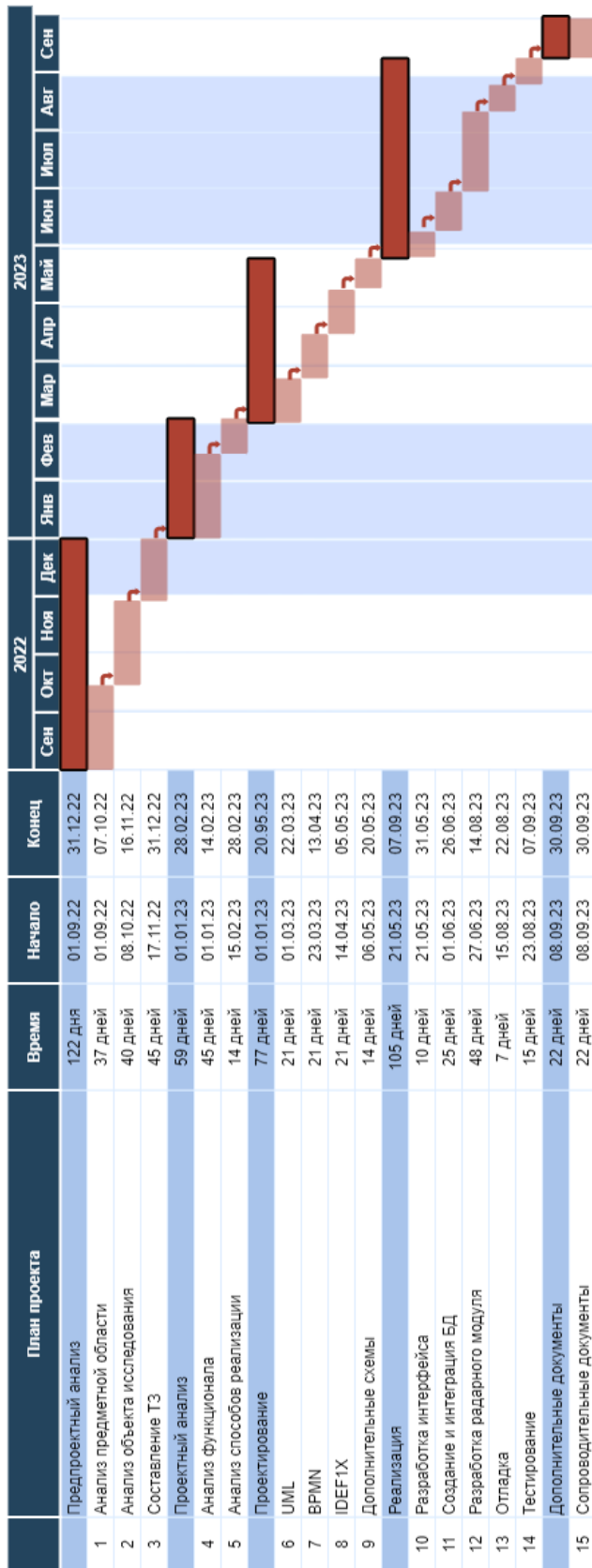


Рис 1.1 диаграмма Ганта

## 1.8 Техничко-экономические показатели

В рамках проектирования данной работы были рассчитаны предполагаемые экономические параметры, на основе которых были установлены планируемые затраты и прибыль.

Таблица 1.3 Оценки ожидаемых затрат на реализацию проекта

№	Наименование работ	Длительность работ (t чел.ч.)		
		$t_{min}$	$t_{max}$	$t_o$
1	Предпроектный анализ	560	720	624
2	Бизнес-моделирование	280	360	312
3	Проектирование	440	520	472
4	Реализация	460	580	508
5	Тестирование	60	180	108
<b>ИТОГО:</b>				<b>2024</b>

Для данных расчетов была использована следующая формула:

$$t_j^o = \frac{(3t_{min} + 2t_{max})}{5} \quad (1.1)$$

$t_j^o$  — ожидаемая продолжительность j-того этапа;

$t_{min}$  — наименьшая оценка по мнению эксперта;

$t_{max}$  — наибольшая оценка по мнению эксперта;

Данный метод оценки дает примерные расчеты, которые в следствии можно разложить на более подробные расчеты.

Примерную стоимость планируемых работ можно рассчитать относительно заработной платы исполнителя. В рамках данной работы будет применен единственный исполнительный, заработная плата которого 70 000 рублей в месяц после вычета норматива социальных, страховых взносов и НДФЛ.

Относительно данной заработной платы рассчитана почасовая ставка по следующей формуле:

$$I_j = \frac{P}{K} = \frac{P}{168} \quad (1.2)$$

$I_j$  — почасовая ставка

$P$  — размер заработной платы

$K$  — количество часов (предполагаемое количество часов равно 168 в месяц в среднем)

На основе взятых чисел была проведена оценка общей стоимости проекта. Стоимость планируемых работ приведена в таблице ниже:

Таблица 1.4 Расчет ожидаемых оплат исполнителю

№	Наименование работ	$t_o$ чел.ч	Суммарные выплаты
1	Предпроектный анализ	624	259 958,40
2	Бизнес-моделирование	312	129 972,80
3	Проектирование	472	196 635,20
4	Реализация	508	211 632,80
5	Тестирование	108	44 992,80
<b>ИТОГО:</b>		2024	843 192

Также, в подобных расчетах необходимо учитывать норматив отчислений на страховые взносы на обязательное социальное, пенсионное и медицинское страхование, которое на момент начала проекта составляет 30%, а ставка НДФЛ 13%. МРОТ в рамках проекта равен 16 242 рублей в месяц

Расчетные значения норматива отчислений НДФЛ за месяц вычисляются по формуле:

$$H_{\text{НДФЛ}} = \frac{100 * Q}{87} - Q = Q * \left( \frac{100}{87} - 1 \right) \quad (1.3)$$

Расчетные значения норматива отчислений на страхование вычисляются по формуле:

$$H_{\text{ст}} = M * H_{\text{соц}} + (Q - M) * H_{\text{л}} \quad (1.4)$$

$H_{\text{соц}}$  — базовый норматив (в процентах) (30%)

$H_{\text{л}}$  — льготный норматив (в процентах) (15%)

$M$  — МРОТ

$Q$  — заработная плата

Таблица 1.5 Расчет затрат на социальные и налоговые выплаты

№	Наименование норматива	Сумма по нормативу, руб.
1	НДФЛ	135 977.01
2	Страховые взносы (медицинское, социальное, пенсионное)	168 171.90
<b>ИТОГО:</b>		<b>304 148.91</b>

Смета всех затрат на данный проект приведено в таблице ниже:

Таблица 1.6 Смета всех затрат на ВКР

№	Наименование статьи	Сумма, руб.
1	Заработная плата сотрудников	843 192
2	Отчисление на социальные нужды	304 148.91
3	Расходы на содержание и эксплуатацию оборудования	-
4	Материалы	-

5	Затраты по работам, выполняемым сторонними организациями	-
6	Амортизационные отчисления	-
7	Накладные расходы	-
8	Спецоборудование	-
		<b>1 147 340.90</b>

### 1.9 Выбор CASE-средств и их обоснование

Перед проектированием системы необходимо определиться с используемыми CASE-средствами, так как они упрощают процесс проектирования и позволяют достичь лучшего результата.

CASE-средства (Computer - Aided Software Engineering) являются набором методов и технологий, позволяющий проектировать информационные системы. Также, при помощи CASE-средств можно автоматизировать некоторые аспекты проектирования.

В рамках данного проекта будут использованы следующие CASE-средства:

- DbSchema
- Diagrams.net

Diagrams.net является универсальной платформой, предоставляющая все необходимые инструменты для создания различных диаграмм. В сравнении с другими CASE-средствами, мною был выбрана данная платформа из-за возможности настраивать каждый аспект диаграммы под моё видение.

Как средство для создания модели БД была выбрана программа DbSchema. В данном приложении возможно создание схем баз данных, с их последующим экспортом в язык SQL.



### 1.10 Подходы SWOT и VCM

Реализация ИС представляет определённые последствия, указанные при помощи SWOT анализа в таблице приведенной ниже.

Таблица 1.7 SWOT анализ

	Сильные стороны	Слабые стороны
Возможности	Улучшенный процесс документирования и ведения учета. Упрощение обслуживания и ремонта.	Приведение оборудования с морскими стандартами к обычным береговым понижает устойчивость и возможно надежность системы.
Угрозы	Более удобная и близкая человеку система отслеживания и связи. Уменьшение используемого дорогого и деликатного оборудования.	Уникальность системы может привести к непредсказуемым неполадкам, которые могут усложнить работу береговой охраны

SWOT анализ расшифровывается как сильные стороны (Strengths), слабые стороны (Weaknesses), возможности (Opportunities) и угрозы (Threats) и является методом определения всего перечисленного и их пересечения, которые будут влиять на будущий продукт и его эксплуатацию, помогающий определить стратегические цели.

Из данной таблицы выходит, что разрабатываемое решение хоть и выполняет те же задачи, что и косвенно схожие решения на рынке, отличается по внедрению на желаемый объект. Сведение функций дорогого оборудования

к более доступному и дешевому несёт указанные выше риски, однако данные риски окупаются положительными моментами.

### *VCM анализ*

#### 1 Оценка информационной емкости продуктов и процессов

- Данный продукт хранит идентификационную, рейсовую и навигационную информацию судов в течении месяца.
- Ключевые процессы в данном продукте состоят из считывания десятков и/или сотен сообщений от судов, одновременного сохранения и отображения информации из сообщений и возможного отображения сохраненной информации

2 В навигационной отрасли информационные технологии играют ключевую роль. Без них, морские сообщения становится гораздо менее надежными.

3 Способы, с помощью которых информационные технологии создают конкурентное преимущество:

- Скорость операций и обработки информации
- Возможность хранить большие объемы данных
- Удаленная передача данных
- Возможность автоматической корректировки данных

4 Способы, как информационные технологии могут создавать новое направление в навигационной сфере (бизнесах)

- Обработка Больших Данных

#### 5 План извлечения выгоды из использования ИС

- Увеличение доступности навигационных технологий
- Расширение возможностей обеспечения безопасности

- Уменьшение издержек на обучение персонала

Данный анализ позволил описать отличительные преимущества с помощью анализа всей цепочки видов деятельности в организациях, начиная от получения информации до конечной обработки и применения данных. Разрабатываемое решение является более доступным и дешевым как и в кратком, так и в долгосрочном плане.

## ГЛАВА 2. ПРОЕКТИРОВАНИЕ ИС

### 2.1 Моделирование предметной области

Первым важным шагом в проектировании данного проекта является определение общего вида всего функционала в системе. Для данной цели была разработана диаграмма вариантов использования, изображенная на рисунке 2.1

Данная диаграмма была создана при помощи UML (Unified Modeling Language, или унифицированный язык моделирования) и описывает отношения между актерами (это может быть человек или другая система) и вариантами использования (определённый порядок действий, выполняющий задачу). Язык UML предоставляет простое графическое представление таких необходимых компонентов как процессы, участвующие лица и/или системы, и взаимосвязи между ними. В языке UML основное внимание уделяется архитектуре; обобщение понятий и простота создания диаграмм в данном языке позволяет сосредоточиться непосредственно на проектировании.

В рамках данной работы используются следующие диаграммы:

- Диаграмма вариантов использования;
- Диаграмма последовательности;
- Диаграмма развёртывания;
- Диаграмма компонентов.

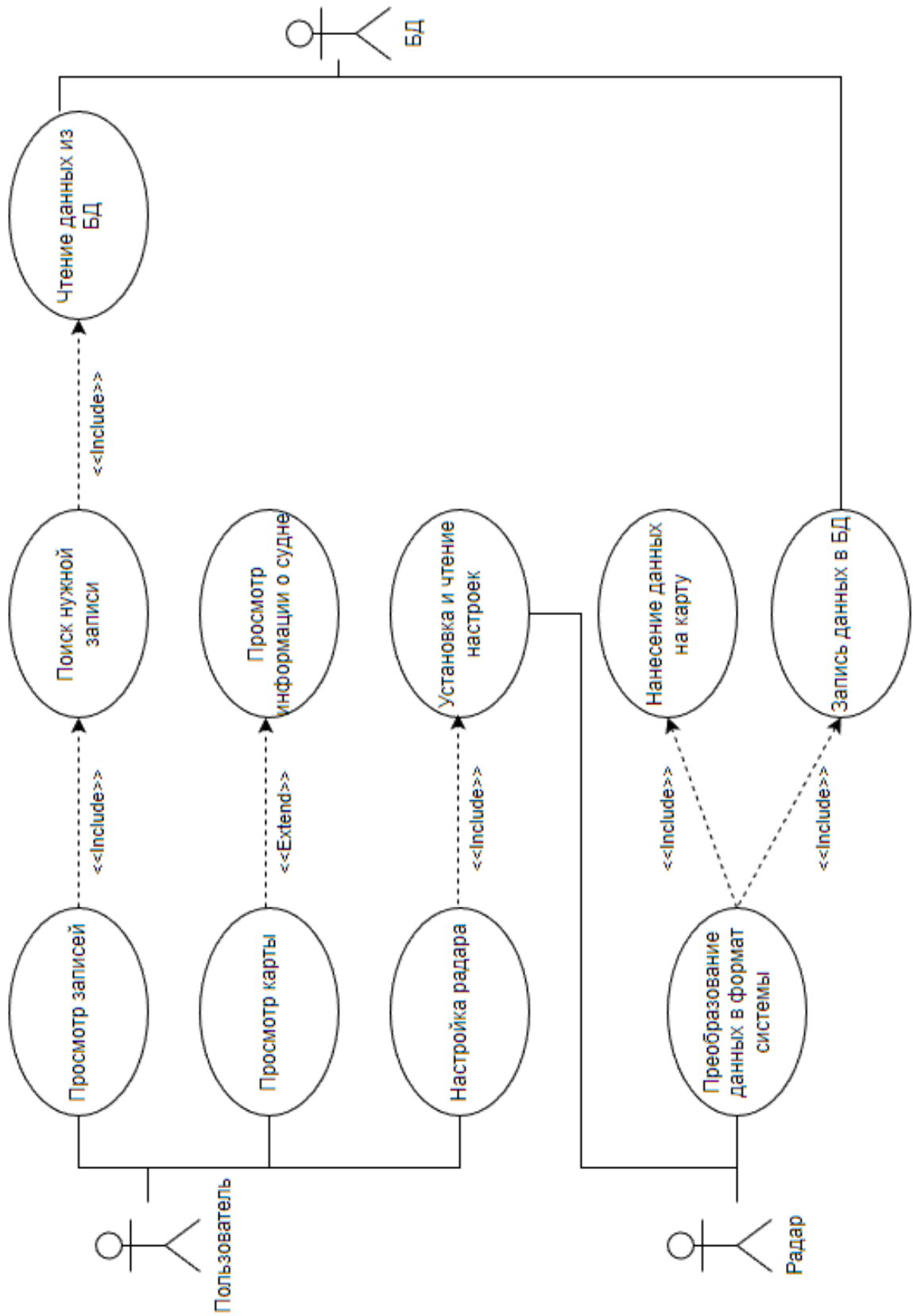


Рис 2.1 Диаграмма вариантов использования

В разрабатываемой ИС ключевыми сценариями выступают следующие варианты использования системы:

- Просмотр карты. Представляет собой окно с картой акватории, где и происходит слежка в прямом эфире.
- Просмотр записей. Состоит из окна результатов, куда будут выводиться результаты поиска, и диалогового окна, в которое вводится поисковый запрос.

Остальные сценарии также являются важной частью системы, и поддерживают главные варианты использования:

- Настройка радара. Окно с распознанными параметрами радара (как читаемыми, так и записываемыми). Основной функцией данного окна является подключение внешнего радара посредством системы плагинов. Также, в данном окне также есть возможность редактировать распознанные записываемые параметры (например у некоторых радаров есть возможность переключения рабочей частоты).
- Преобразование данных в формат системы. Данный сценарий подразумевает считывание потока данных с подключенного радара и преобразование нужных данных в необходимый для системы формат. Данный процесс состоит из двух подпроцессов, указанных на рисунке 2.1.
- Чтение данных из БД выведено к актёру БД для акцентирования факта, что в рамках разрабатываемого решения БД является отдельной развёрнутой системой, подключенной к компьютеру пользователя.

## **2.2 Сценарии на основе диаграммы вариантов использования**

Сценарии использования (USE-CASE) - это сценарий взаимодействия пользователя (или пользователей) с программным продуктом для достижения конкретной цели. Без правильно расписанных сценариев использование последующее проектирование и разработка превращаются в отчаянную стрельбу в небо. Они помогают разработчикам понять, как пользователи будут взаимодействовать с продуктом и какие функции им нужны для достижения своих целей. Без правильно составленных сценариев использования, разработчики могут потратить много времени и ресурсов на создание функций, которые никогда не будут использоваться. Кроме того, сценарии использования также помогают улучшить пользовательский опыт. Если продукт разработан с учетом того, как пользователи будут использовать его в реальной жизни, то он будет более интуитивно понятным и удобным в использовании.

На основе созданной UML диаграммы были разработаны предполагаемые сценарии использования разрабатываемой системы. В таблицах приведённых ниже указаны самые важные сценарии использования; остальные приведены в Приложении А.

Таблица 2.1 Главный раздел сценария «Просмотр карты»

<b>Главный раздел сценария выполнения варианта использования "Просмотр карты"</b>	
<b>Вариант использования</b>	Просмотр карты
<b>Актеры</b>	Пользователь
<b>Цель</b>	Пользователь открывает окно с картой акватории
<b>Тип</b>	Базовый
<b>Ссылки на другие варианты использования</b>	Расширение: Просмотр информации о судне

Таблица 2.2 Типичный ход сценария «Просмотр карты»

<b>Типичный ход событий сценария выполнения варианта использования "Просмотр Карты"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
1. Нажатие кнопки "Карта"	2. Загрузка карты
	3. Проверка соединения с радаром
	Исключение 1: отсутствует связь с радаром
	4. Получение информации судов
	Исключение 2: сбой соединения
	5. Нанесение данных на карту
6. Просмотр информации о судне	



Таблица 2.3 Исключения сценария «Просмотр карты»

<b>Исключения сценария выполнения варианта использования "Просмотр Карты"</b>	
Исключение 1: отсутствует связь с радаром	
<b>Действия актеров</b>	<b>Отклик системы</b>
	4. Сообщение "Отсутствует связь с радаром"
	5. Переключение на окно настройки радара
Исключение 2: сбой соединения	
<b>Действия актеров</b>	<b>Отклик системы</b>
	5. Сообщение "Сбой соединения"

Таблица 2.4 Главный раздел сценария «Просмотр записей»

<b>Главный раздел сценария выполнения варианта использования "Просмотр записей"</b>	
<b>Вариант использования</b>	Просмотр записей
<b>Актеры</b>	Пользователь
<b>Цель</b>	Просмотр и поиск нужных записей в базе данных
<b>Тип</b>	Базовый
<b>Ссылки на другие варианты использования</b>	Включает: Поиск нужных записей

Таблица 2.5 Типичный ход сценария «Просмотр записей»

<b>Типичный ход событий сценария выполнения варианта использования "Просмотр записей"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
1.Нажатие кнопки "Записи"	2.Подгрузка базы данных
	Исключение 1: Сбой соединения с БД
	3.Загрузка окна базы данных
4.Поиск нужной записи	

Таблица 2.6 Исключения сценария «Просмотр записей»

<b>Исключения сценария выполнения варианта использования "Просмотр записей"</b>	
Исключение 1: Сбой соединения с БД	
<b>Действия актеров</b>	<b>Отклик системы</b>
	4. Сообщение "Не удалось подключить БД"

Таблица 2.7 Главный раздел сценария «Настройка радара»

<b>Главный раздел сценария выполнения варианта использования "Настройка радара"</b>	
<b>Вариант использования</b>	Настройка радара
<b>Актеры</b>	Пользователь
<b>Цель</b>	Настройка пользователем изменяемых параметров работы радара
<b>Тип</b>	Базовый
<b>Ссылки на другие варианты использования</b>	Включает: Установка и чтение настроек

Таблица 2.8 Типичный ход сценария «Настройка радара»

<b>Типичный ход событий сценария выполнения варианта использования "Настройка радара"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
1. Нажатие кнопки "Настройки"	2. Установка и чтение настроек
	Исключение 1: отсутствует связь с радаром
	3. Отображение окна настроек
4. Установка/изменение настроек	
5. Нажатие кнопки "Применить"	
	6. Установка и чтение настроек
	Исключение 2: сбой при передаче данных

Таблица 2.9 Исключения сценария «Настройка радара»

<b>Исключения сценария выполнения варианта использования "Настройка радара"</b>	
Исключение 1: отсутствует связь с радаром	
<b>Действия актеров</b>	<b>Отклик системы</b>
	3. Сообщение "Технические неполадки"
Исключение 2: сбой при передаче данных	
<b>Действия актеров</b>	<b>Отклик системы</b>
	7. Отправить данные ещё раз
	Исключение 2.1 данные не отправлены
Исключение 2.1: данные не отправлены	
<b>Действия актеров</b>	<b>Отклик системы</b>
	8. Сообщение "Сбой при применении настроек"

Таблица 2.10 Главный раздел сценария «Поиск нужной записи»

<b>Главный раздел сценария выполнения варианта использования "Поиск нужной записи"</b>	
<b>Вариант использования</b>	Поиск нужной записи
<b>Актеры</b>	Пользователь
<b>Цель</b>	Поиск по запросу, составленного пользователем
<b>Тип</b>	Расширенный
<b>Ссылки на другие варианты использования</b>	

Таблица 2.11 Типичный ход сценария «Поиск нужной записи»

<b>Типичный ход событий сценария выполнения варианта использования "Поиск нужной записи"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
5. Введение запроса поиска	6. Проверка запроса
	Исключение 1: не валидный запрос
	7. Произведение поиска по запросу
	Исключение 2: не найдено ни одной записи
	8. Вывод результатов

Таблица 2.12 Исключения сценария «Поиск нужной записи»

<b>Исключения сценария выполнения варианта использования "Поиск нужной записи"</b>	
Исключение 1: не валидный запрос	
<b>Действия актеров</b>	<b>Отклик системы</b>
	7. Определение возможной ошибки пользователя
	8. Сообщение с предполагаемой ошибкой
Исключение 2: не найдено ни одной записи	
<b>Действия актеров</b>	<b>Отклик системы</b>
	8. Сообщение "По данному запросу 0 записей"

Таблица 2.13 Главный раздел сценария «Просмотр информации о судне»

<b>Главный раздел сценария выполнения варианта использования "Просмотр информации о судне"</b>	
<b>Вариант использования</b>	Просмотр информации о судне
<b>Актеры</b>	Пользователь
<b>Цель</b>	Просмотр подробной информации о наблюдаемом судне
<b>Тип</b>	Расширенный
<b>Ссылки на другие варианты использования</b>	

Таблица 2.14 Типичный ход сценария «Просмотр информации о судне»

<b>Типичный ход событий сценария выполнения варианта использования "Просмотр информации о судне"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
7. Выбор судна для слежки	8. Выделение судна
	9.Запрос данных с судна
	Исключение 1: нет связи с судном
	10. Выведение данных справа в окне

Таблица 2.15 Исключения сценария «Просмотр информации о судне»

<b>Исключения сценария выполнения варианта использования "Просмотр информации о судне"</b>	
Исключение 1: нет связи с судном	
<b>Действия актеров</b>	<b>Отклик системы</b>
	10. Запрос доступных записей из БД
	Исключение 1.2: данных о судне нет
	11. Выведение данных с предупреждением "Из БД"
Исключение 1.2: данных о судне нет	
<b>Действия актеров</b>	<b>Отклик системы</b>
	11. Выведение "нет связи" справа в окне

### 2.3 Модель данных

При любой работе с данными необходимо понимать в каком виде они собираются, хранятся и обрабатываются. Создание модели данных является обязательной частью разработки любого современного приложения. Отдельного внимания в этом процессе заслуживает проектирование базы данных.

База данных (БД) — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных

База данных позволяет систематизировать необходимую информацию из рассматриваемой предметной области, проводя с ней различные операции. Не смотря на все прилагаемые риски, связанные с подключением отдельных систем и технологий, базы данных являются незаменимым решением вопроса хранения и систематизации данных.

Для выполнения одной из ключевых функций разрабатываемой системы внедрение базы данных является обязательным. Так как зачастую все снимаемые данные обрабатываются вместе и практически не применяются отдельно, целесообразно поделить их на предполагаемый характер хранения данных. На рисунке 2.2 указана схема базы данных

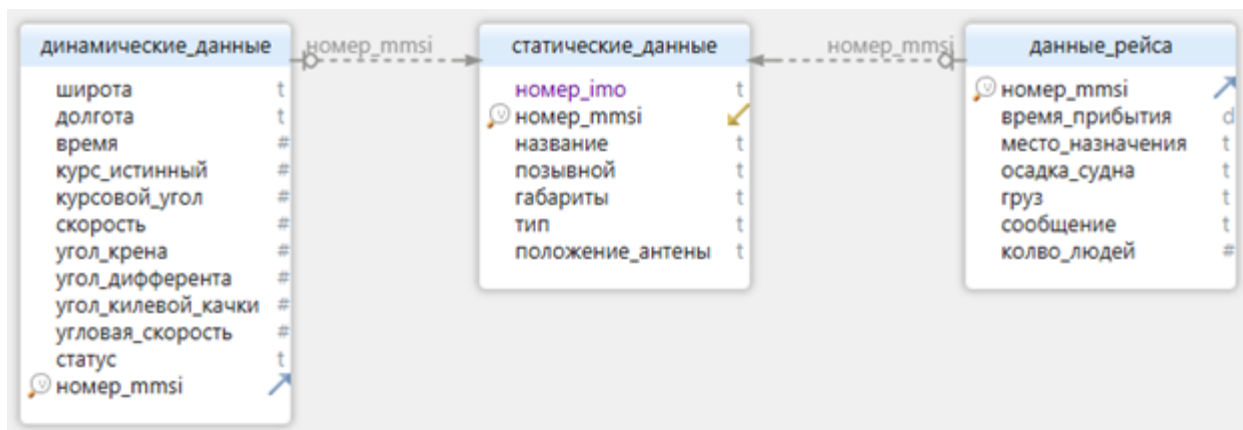


Рис 2.2 Схема базы данных

Данные судна передаются один раз в установленный промежуток времени. Этот временной промежуток в основном зависит от расстояния до береговой линии и/или ближайшего порта. Вся информация отправляется единым блоком, однако все данные можно разделить на три блока: статические данные, динамические данные и данные рейса.

Статические данные, как и предполагает название, являются данными, которые не меняются во время рейса. В подобные данные входят идентификационные номера и названия, габариты и тип судна и положение антенны передатчика на судне.

Данные рейса обычно не меняются в процессе самого рейса и определяют цель нахождения судна в открытых водах.

Динамические данные состоят из навигационных данных (в которые входят широта, долгота, курсы, всевозможные навигационные углы и угловая скорость) и текущего статуса судна, который представляет из себя электронный аналог семафорных сигналов в старые времена.



## 2.4 Диаграмма последовательности

Диаграмма последовательности — UML диаграмма, где на единой временной оси показаны жизненные циклы объектов и актеров внутри рассматриваемой системы. Вертикальной линией отображена непосредственно временная ось, поверх которой показываются объекты в виде прямоугольников. Они могут переходить с одной оси на другую, обозначая их участие в каком-то процессе. Все связи объектов между определёнными процессами и другими объектами обозначены стрелками.

На рисунках 2.3 и 2.4 изображены диаграммы последовательности для сценариев «Настройка радара» и совмещённый сценарий «Просмотр записей» и «Поиск нужной записи». Остальные диаграммы приведены в Приложении Б.

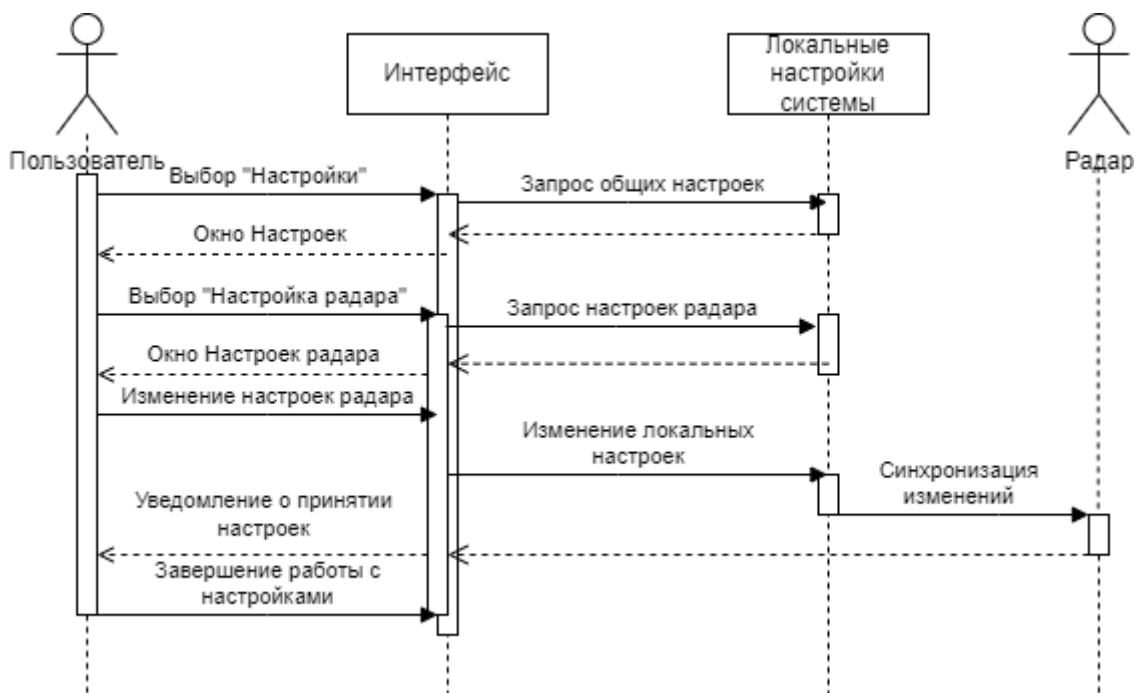


Рис 2.3 Диаграмма последовательности для сценария «Настройка радара»

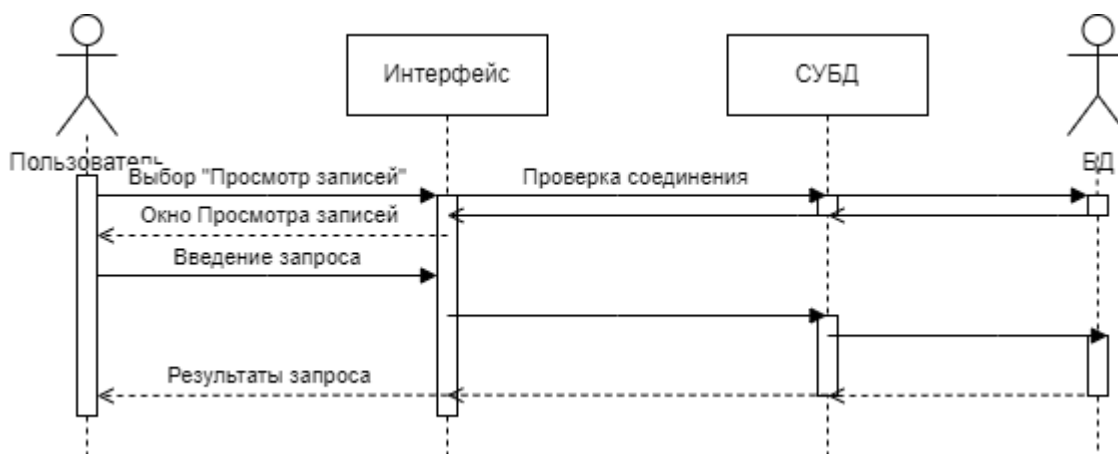


Рис 2.4 Диаграмма последовательности сценариев «Просмотр записей» и «Поиск нужной записи»

Данные диаграммы позволяют более подробно отобразить предполагаемые процессы как во времени, так и относительно связи компонентов системы. В процессе проектирования, подобные иллюстрации создаваемых интерпретаций идей являются незаменимым инструментом, показывающий создаваемую систему относительно одной из проекций практического применения (времени).

На рисунке 2.3 делается акцент на том, что настройки радара вложены в общие настройки приложения, которые не имеют отношения к радару. Также, в правой части диаграммы указано, что связь с радаром ведётся только при применении настроек. Так как проектируемая система предполагается как закрытая, данный способ обмена информации является достаточным для манипулирования настройками, так как внешние взаимодействия с радаром не предусматриваются, а внутренних взаимодействий достаточно для полной реализации функционала радара.

На рисунке 2.4 отображена минималистичная, но достаточная последовательность процессов при поверхностном взаимодействии пользователя с базой данных. Предполагаемая последовательность подразумевает минимальную нагрузку на базу данных и СУБД.

## 2.5 Диаграмма компонентов

Диаграмма компонентов UML показывает компоненты системы и связи между ними. Данная диаграмма может показывать связи и отношения как и физических компонентов (таких как периферия, отдельные персональный компьютер или физические объекты), так и абстрактных (программные модули, иерархические и структурные единицы, интерфейсы и т.д.).

В разработке программного обеспечения, диаграмма компонентов UML чаще используется для отображения иерархии инкапсулированных классов, которые, в свою очередь, тоже могут состоять из классов.

Сама диаграмма составляется из двух простых частей – компонентов, представленных прямоугольниками, и связями, представленные линиями между прямоугольниками. Как и указано в определении, в прямоугольниках могут быть вложены другие прямоугольники, даже вместе с связями между ними. Сами прямоугольники подписаны соответствующим названием компонента, часто с припиской архетипа компонента (самой частой припиской является <<component>>, обозначающая непосредственно компонент системы, однако существуют и другие. Эта приписка может быть любой, но люди придерживаются самых частых). Связи отображаются либо стрелкой, указывающей «во что» вкладывается или от чего зависит компонент, однако есть и более деликатный способ изображения связи. Связь можно отобразить как окружность и полуокружность, «обнимающий» окружность. В данном случае, окружность является представляемыми аргументами или объектами, а полуокружность является интерфейсом.

На рисунке 2.5 представлена диаграмма компонентов разрабатываемой системы, где отображены все предполагаемые подсистемы и компоненты в итоговой системе.

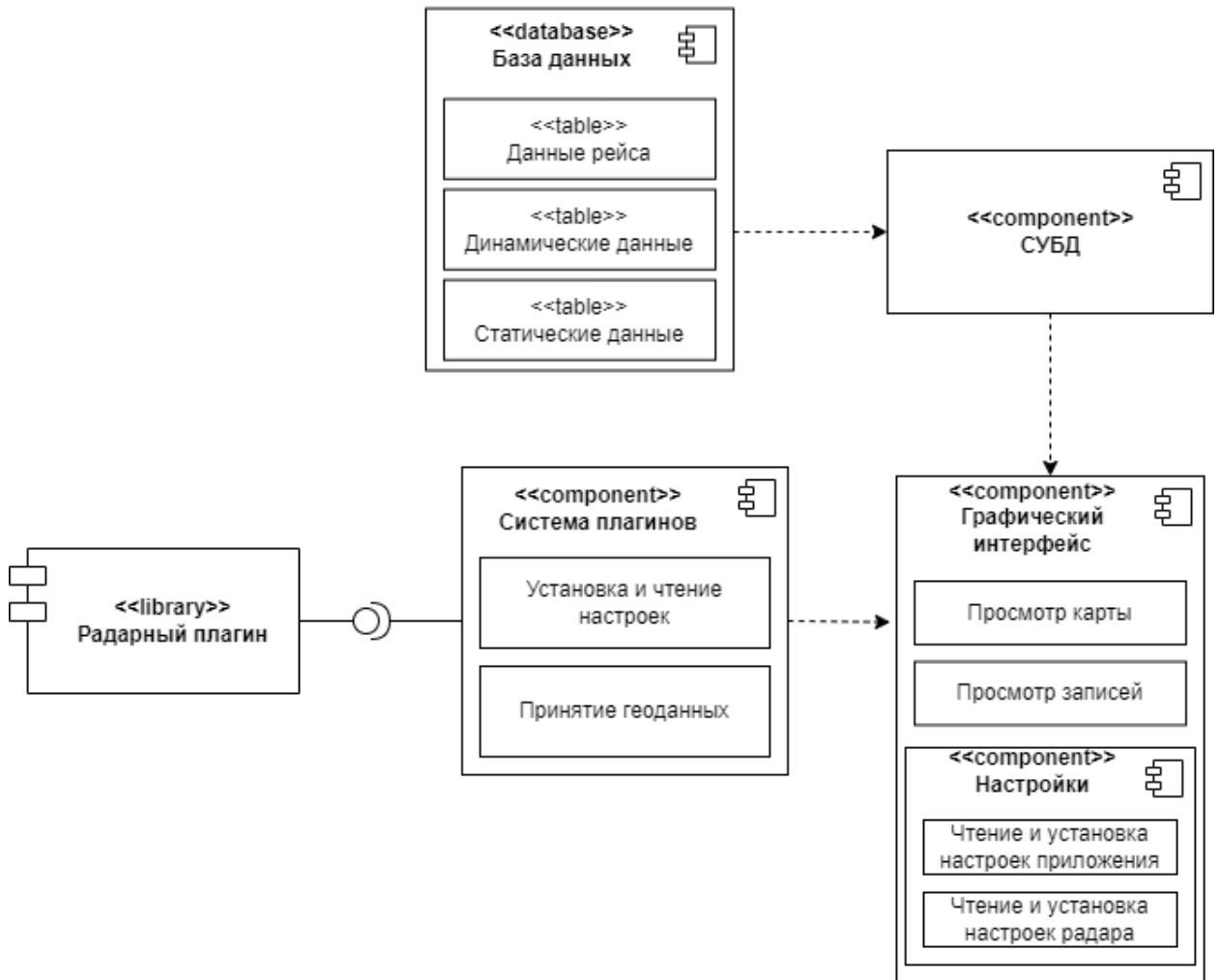


Рис 2.5 Диаграмма компонентов разрабатываемой системы

В рамках данной диаграммы представленные ранее части системы разделены на их практическую реализацию. Связь с базой данных будет реализована при помощи СУБД, в данном случае это MySQL. В самой базе данных поверхностно отмечена её структура.

Взаимодействие с радаром разделено на систему плагинов, посредством которой можно подключить любой радар к данной системе. Используемый функционал указан внутри системы плагинов. Левее отмечены подключаемые плагин и/или плагины как отдельный модуль системы, который можно подключать по требованию.

«Графический интерфейс», не смотря на название, предполагает разрабатываемый клиент, который будет использоваться пользователем. В

рамках данного названия указаны сегменты интерфейса, с которыми будет иметь дело пользователь.

## 2.6 Диаграмма развертывания

В диаграмме развертывания UML указывается непосредственно физическое развертывания системы. На данной диаграмме можно увидеть четкую иерархию физических компонентов и артефактов внутри них.

Физические «узлы» представляют собой аппаратные компоненты, на которых будет развернута система (например веб-сервер, сервер приложения, внешняя периферия и т.д.). Артефакты являются программными компонентами (например веб-приложение или база данных), которые располагаются каждый в своем узле и коммуницируют с собой различными методами (например REST или RMI).

Для полноценной реализации любой информационной системы, подобные диаграммы являются обязательной частью проектирования, так как ни одна ИС не существует в вакууме. Все разрабатываемые системы имеют физические компоненты, которые исполняют проектируемые процессы. На рисунке 2.6 указана система развертывания разрабатываемой системы

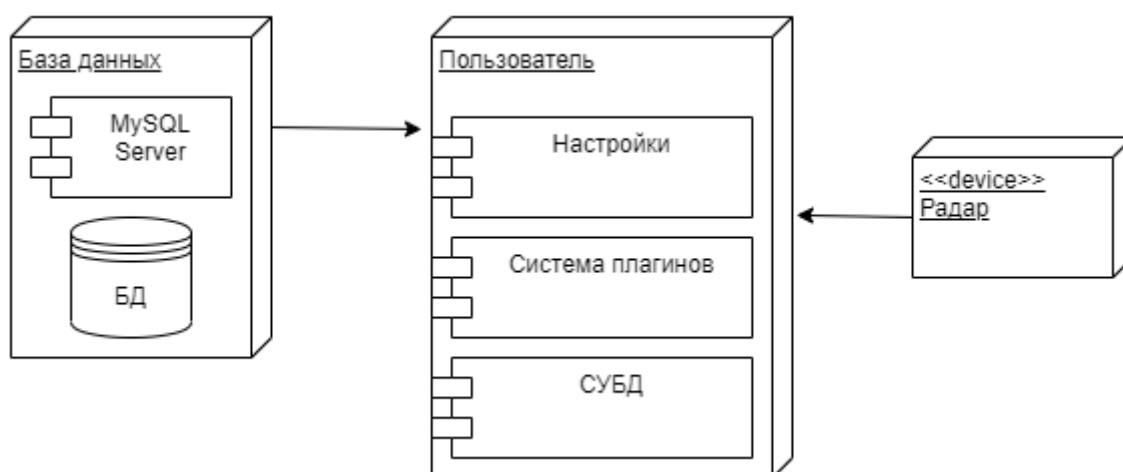


Рис 2.6 Диаграмма развертывания разрабатываемой системы

Разрабатываемая система требует лишь три физических узла для полноценной работы: клиент (представленный пользователем), с которым будет работать наш пользователь и где будет реализована наибольшая часть системы; база данных, где предполагается долгосрочное хранение данных; и непосредственно радар, который будет применяться для непосредственного нахождения объектов в акватории и слежения за ними.

Данная диаграмма является финальным штрихом в проектировании данной системы, что позволяет сделать переход к реализации спроектированной системы. Как правило, диаграмма развертывания является последним этапом проектирования, так как она представляет самый последний этап реализации и общий вид системы, в котором она будет внедрена в практику.

## ГЛАВА 3. РЕАЛИЗАЦИЯ ИС

### 3.1 Обоснование выбора средств разработки

Для реализации данной системы был выбран следующий инструментарий: Microsoft Visual Studio в качестве среды разработки, Python как язык программирования, библиотека PyQt6 и MySQL для реализации базы данных.

Microsoft Visual Studio — линейка продуктов компании Майкрософт, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework и Microsoft Silverlight.

Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и как отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как например, Subversion и Visual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования или инструментов для прочих аспектов цикла разработки программного обеспечения (например, клиент Team Explorer для работы с

Team Foundation Server).

В качестве языка программирования проекта были рассмотрены следующие языки: Python, Java и C#.

Python это высокоуровневый язык программирования, ключевым достоинством которого является простота синтаксиса и команд, а также большое кол-во библиотек, которые содержат уже написанный код. Данный язык пригоден для различных сфер программирования:

- Системное программирование: поиск электронных каталогов, запуск
- других программ.
- Графическое программирование: разработка приложений с веб-
- интерфейсом.
- Веб-сценарии: поиск, приём, передача, извлечение информации, загрузка
- веб-страниц, передача и обработка файлов и т.д.
- Создание прототипов.
- Создание программ для вычислений.
- Разработка робот-программ, игр и прочего.

Java – это язык программирования общего назначения, который следует парадигме объектно-ориентированного программирования и подходу «Написать один раз и использовать везде». Java используется для настольных, сетевых, мобильных и корпоративных приложений. Он используется в следующих сферах:

- Веб-приложениях.
- Правительственных веб-сайты.
- Технологии обработки больших данных.
- Java подходит и для научных проектов, особенно в области обработки
- естественного языка.
- Язык Java преобладал и в программировании для мобильных устройств,



- задолго до появления смартфонов — первые мобильные игры в начале 2000-х годов
- были написаны на Java.

C# — объектно-ориентированный язык программирования. Разработан в 1998—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft.NET Framework. Впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270. Он используется в следующих сферах:

- Разработка настольных приложений.
- Разработка web приложений.
- Разработка мобильных приложений.
- Разработка облачных приложениях.
- На C# разрабатываются компьютерные игры, программы
- математического моделирования, шлюзы платежных систем и многое другое.

Для разработки данной ИС мною был выбран язык программирования Python, так как он имеет доступные технологии, которые помогут реализовать требуемые функции. Также, простота Python позволяет сократить затрачиваемые на разработку ресурсы и упрощает процесс разработки приложения. Не смотря на выбранную среду разработки, Python отлично совмещается с ней посредством плагинов.

Основой для разрабатываемого клиентского приложения послужит библиотека PyQt6. Это библиотека для Python, позволяющая создавать приложения с графическим интерфейсом при помощи инструментария не безызвестного Qt, который является одним из самых известных в мировой практике. PyQt6 является надстройкой, позволяющей пользоваться всеми

модулями Qt на языке Python. С данной библиотекой можно реализовать практически любой функционал.

Для реализации базы данных были рассмотрены следующие опции: MS SQL Server, MySQL, PostgreSQL. Достоинства рассматриваемых решений представлены ниже:

Таблица 3.1 Сравнение СУБД

Критерий	Microsoft SQL Server	PostgreSQL	MySQL
Поддерживаемые операционные системы	Windows 10, Windows server 2016	Windows, семейство Unix, Mac OS	Windows, семейство Unix, Mac OS
Условия лицензирования	Коммерческий продукт с закрытым исходным кодом	Лицензия BSD Open Source	Коммерческая лицензия и GNU GPL
Использование в коммерческих проектах	Среднее	Среднее, но реже, чем MySQL	Среднее
Наличие графического ПО для конструирования и оптимизации запросов	Да (SQL Management Studio)	Да (PgAdmin4)	Да (MySQL Workbench)
Стоимость	212840	Бесплатно	Бесплатно

Для разрабатываемой ИС в качестве СУБД была выбрана MySQL. MySQL является относительно небольшой и быстрой реляционной СУБД, оптимальным решением для малых и средних по размеру приложений. MySQL является собственностью компании Oracle Corporation, получившей её вместе с поглощённой Sun Microsystems, осуществляющей разработку и поддержку

приложения. Распространяется под GNU General Public License или под собственной коммерческой лицензией. Основные преимущества СУБД MySQL:

- многопоточность, поддержка нескольких одновременных запросов;
- оптимизация связей с присоединением многих данных за один проход;
- гибкая система прав доступа;
- основанная на потоках, быстрая система памяти;
- максимальный размер таблицы ограничен 8 миллионами терабайт;
- легкость управления таблицей, включая добавление и удаление ключей и полей;
- гибкость, обеспечиваемая поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей.

Основная причина выбора MySQL в качестве СУБД для разрабатываемой ИС заключается в ее бесплатности, так как она распространяется под свободной лицензией с открытым исходным кодом. Так как предполагается, что с приложением будет работать единственный пользователь, то возможностей и производительности СУБД MySQL будет вполне достаточно.

Помимо MySQL в качестве СУБД также рассматривались и сравнивались между собой программные продукты других производителей.

Таким образом, MySQL является самым оптимальным вариантом. Бесплатный PostgreSQL соответствует основным требованиям, но под него сложнее разрабатывать ПО.

### **3.2 Метод реализации геолокационного функционала**

Краеугольным вопросом в реализации данной системы является метод реализации геолокационного функционала данной системы. В разных

системах данный вопрос решается по разному. Весь спектр решений относительно этого вопроса в рамках разрабатываемой системы можно свести к следующему вариантам:

- Наличие встроенного устройства, исполняющего роль геолокатора, например GPS/ГЛОНАСС модуль, встроенный в устройство.
- Связь с внешним геолокационным устройством. Это может быть внешний стационарный радар или спутник, подключенный к требуемому узлу системы.

В рамках данной системы, данный вопрос решается подключением внешнего радара посредством разработанного под него плагина. Для вычисления начальной позиции могут быть использованы как данные с самого радара, так и введенные вручную координаты требуемой точки (или выбор ее на предоставленной векторной карте).

Однако, геолокационный функционал подразумевает не только считывание и выдача координат, но и вычисления, связанные с обрабатываемыми геолокационными данными. В разрабатываемой системе, подобные вычисления входят в рисовании необходимой графики на карте («след» корабля, поверхностно отображающий курс и скорость корабля; предупреждения о возможных столкновениях и т.д.)

### **3.3 Описание реализации базы данных ИС**

Структура базы данных разрабатываемой ИС разработана в соответствии с пунктом 2.3. Ниже приведены фрагменты таблиц с описанием идентификаторов.

Таблица 3.2 Таблица «Данные рейса»

Поле	Тип данных	Описание
номер_mmsi	TINYTEXT	Номер Maritime Mobile Service Identity, состоящий из межнационального морского телефонного номера и временного идентификатора. Является внешним ключом
время_прибытия	DATETIME	Также известно как ETA. Планируемое время прибытия
место_назначения	TINYTEXT	Место назначения рейса
осадка_судна	DOUBLE(2,1)	Глубина погружения судна в воду
груз	TINYTEXT	Краткая классификация и описание груза на судне
сообщение	TINYTEXT	Сообщение для предупреждения и обеспечения безопасности грузоперевозки
колво_людей	MEDIUMINT	Количество людей на борту

Таблица 3.3 Таблица «Статические данные»

Поле	Тип данных	Описание
номер_imo	TINYTEXT	Номер Международной морской организации (ИМО). Используется в качестве избыточности для более точного определения судна

номер_mmsi	TINYTEXT	Номер Maritime Mobile Service Identity, состоящий из международного морского телефонного номера и временного идентификатора. Относительно этого номера в данной таблице идентифицируются судна
название	TINYTEXT	Название судна
позывной	TINYTEXT	Позывной, используемый в коммуникациях
габарит_д	DOUBLE(5,2)	Длина судна
габарит_в	DOUBLE(5,2)	Высота судна
габарит_ш	DOUBLE(5,2)	Ширина судна
тип	TINYTEXT	Тип судна
положение_антенны_ш	DOUBLE(3,5)	Положение антенны судна по ширине
положение_антенны_д	DOUBLE(3,5)	Положение антенны судна по долготе

Таблица 3.4 Таблица «Динамические данные»

Поле	Тип данных	Описание
широта	DOUBLE(3,5)	Местоположение судна по широте
долгота	DOUBLE(3,5)	Местоположение судна по долготе
время	DATETIME	Время занесения данных
курсовой_угол	DOUBLE(3,3)	Угол между диаметральной плоскостью судна и направлением на какой-либо наблюдаемый с судна объект
скорость	DOUBLE(3,3)	Скорость судна

угол_крена	DOUBLE(3,3)	Угол наклона судна в бок
угол_дифферента	DOUBLE(3,3)	Угол отклонения поперечного сечения корабля от вертикали
угол_килевой_качки	DOUBLE(3,3)	Угол наклона судна вперед или назад
угловая_скорость	DOUBLE(3,3)	Угловая скорость судна
статус	TINYTEXT	Навигационный статус, изложенный в краткой форме (например: Ограничен в возможности маневрировать)
номер_mmsi	TINYTEXT	Номер Maritime Mobile Service Identity, состоящий из межнационального морского телефонного номера и временного идентификатора. Является внешним ключом

### 3.4 Описание технологии работы с ИС

При запуске приложения, открывается начальное окно, представленное на рисунке 3.1

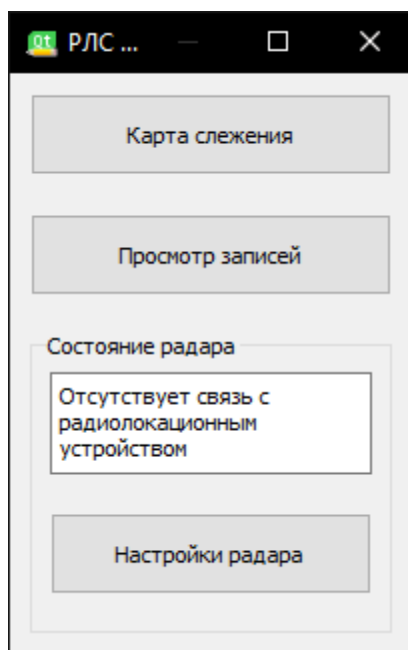


Рис 3.1 Начальное окно приложения

Данное окно было создано чтобы не подгружать все подсистемы одновременно, что ускоряет время запуска программы. Из этого окна можно перейти к окнам «Карта слежения», «Просмотр записей», и при отсутствии связи с радаром нижняя секция окна будет выглядеть как на рисунке, что позволяет перейти к окну «Настройки радара». В ином случае, внизу будет отображено сообщение о штатной работе подключенного радара.

На рисунке 3.2 представлена структура окна «Карта слежения»





Рисунок 3.2 Окно «Карта слежения»

На данном рисунке приведен примерный принцип работы данного окна. Наибольшую часть окна занимает карта, на которой полым квадратом указан используемый радар, а закрашенными квадратами представлены суда. Линия за судном представляет «след» судна, по которому можно представить истинное направление судна и его скорость, в то время как стрелка показывает планируемое направление судна. Справа находится сводка информации, куда будут выводиться все данные при выборе судна на карте. Также имеется возможность переключиться на другие окна при помощи кнопок на верхней панели.

На рисунке 3.3 представлено окно «Просмотр записей»

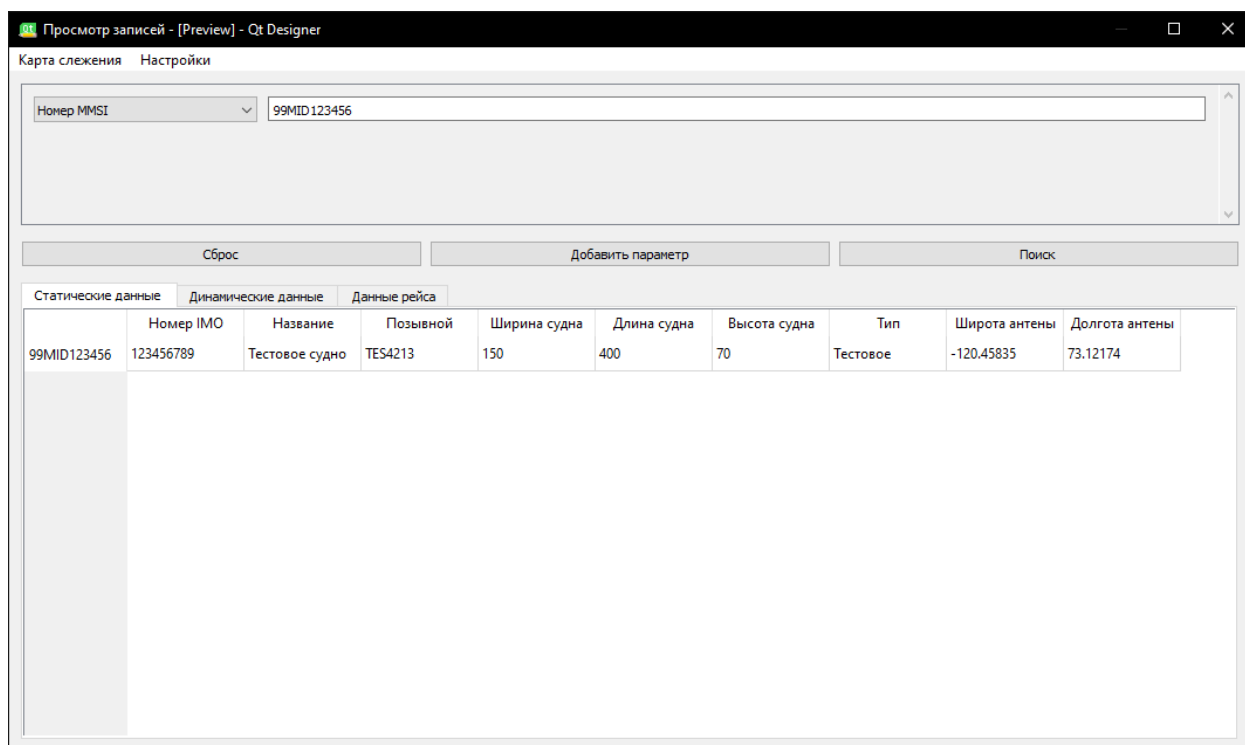


Рис 3.3 Окно «Просмотр записей»

При помощи данного окна пользователь сможет найти необходимую записанную информацию. Сверху представлена зона для ввода параметров, где указываются необходимые параметры для поиска и их значение. Параметр всегда как минимум один, и относительно первого параметра ведется поиск. Количество параметров можно сбросить или увеличить при помощи кнопок ниже. Кнопка «Поиск» запускает процедуру поиска по введенному запросу. Ниже имеются три вкладки, представляющие три таблицы базы данных. В каждой из них будет выводиться найденная информация. Также, при помощи кнопок на верхней панели можно переключиться на другие окна.

## ЗАКЛЮЧЕНИЕ

Тема дипломной работы была раскрыта в трех основных разделах:

В первом разделе были описаны решаемые задачи разрабатываемой системой и истекающий из данных задач функционал. Были сделаны расчеты как и временных ресурсов на реализацию всех необходимых работ, так и экономических ресурсов на оплату необходимых материалов, работу исполнителей и необходимые отчисления. Был сделан вывод о необходимости реализации данной ИС и список необходимых требований к разрабатываемой ИС. Также, были исследованы представленные на рынке альтернативы и описана необходимость разработки собственного решения.

Во втором разделе были спроектированы предполагаемые бизнес-процессы, сам проектируемый функционал был рассмотрен с абстрактной, семантической, технической и пользовательской точки зрения. Был составлен эскиз необходимой базы данных и структура реализации разрабатываемой системы.

В третьем разделе был произведен более подробный анализ инструментария непосредственной реализации системы и подчеркнут его выбор. На основе выбранных средств была разработана итоговая система

На основании результатов выполнения дипломной работы можно сделать следующие выводы:

1. Был произведен анализ исходных данных.
2. Исследованы существующие методы проектирования информационных систем, в том числе геоинформационных систем.
3. Проведен анализ существующих на рынке решений.
4. Разработана достаточная модель данных для выполнения необходимых функций.

5. В целях реализации поставленной задачи было проведено проектирование ИС, с расчетом необходимых экономических показателей.

Из данных выводов следует, что цели дипломной работы достигнуты.

## СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Проектирование информационных систем: учеб. пособие / Н.З. Емельянова, Т.Л. Партыка, И.И. Попов. - М.: Форум, 2009. - 432 с.: ил.; 60x90 1/16. - (Профессиональное образование).
2. Как правильно написать реферат, курсовую и дипломную работы / Юлия Ивановна Бушенева. - Москва : Издательско-торговая корпорация "Дашков и К", 2013.
3. Организация выполнения и защиты дипломного проекта (работы) и выпускной квалификационной работы бакалавра: Учебное пособие / Юлия Олеговна Толстых, Николай Яковлевич Кузин, Татьяна Владимировна Учинина. - Москва : ООО "Научно-издательский центр ИНФРА-М", 2012.
4. Рефераты, курсовые и дипломные работы. Методика подготовки и оформления / Игорь Николаевич Кузнецов. - 7. - Москва : Издательско-торговая корпорация "Дашков и К", 2013.
5. Афонин В.В. Моделирование систем: учебно-практическое пособие / В.В. Афонин, С.А. Федосин. - М.: Интуит, 2016. – 231 с.
6. Баженова И.Ю. Основы проектирования приложений баз данных (2 изд). – М.: Интернет-университет информационных технологий, 2016. – 237 с.
7. Венделева М.А. Информационные технологии в управлении. Учебное пособие для бакалавров / М.А. Венделева, Ю.В. Вертакова. - Люберцы: Юрайт, 2016. – 462 с.
8. Валитов Ш.М. Современные системные технологии в отраслях экономики: Учебное пособие / Ш.М. Валитов, Ю.И. Азимов, В.А. Павлова. - М.: Проспект, 2016. – 504 с.
9. Вдовин В.М. Теория систем и системный анализ: Учебник для бакалавров, 3-е изд.(изд:3) / В.М. Вдовин, Л.Е. Суркова. - М.: ИТК Дашков и К, 2016. – 644 с.

10. Гаврилов М.В. Информатика и информационные технологии: Учебник / М.В. Гаврилов, В.А. Климов. - Люберцы: Юрайт, 2016. – 383 с.
11. Замятина О.М. Вычислительные системы, сети и телекоммуникации. моделирование сетей. Учебное пособие для магистратуры / О.М. Замятина. - Люберцы: Юрайт, 2016. – 159 с.
12. Ерохин В.В. Безопасность информационных систем: учеб пособие / В.В. Ерохин, Д.А. Погonyшева, И.Г. Степченко. - М.: Флинта, 2016. – 184 с.
13. Долганова О.И. Моделирование бизнес-процессов: Учебник и практикум для академического бакалавриата / О.И. Долганова, Е.В. Виноградова, А.М. Лобанова. - Люберцы: Юрайт, 2016. – 289 с.
14. Дарков А.В. Информационные технологии: теоретические основы: Учебное пособие / А.В. Дарков, Н.Н. Шапошников. - СПб. Лань, 2016. – 448 с.
15. Грошев А.С., Закляков П. В. Информатика: учеб. для вузов – 3-е изд., перераб. и доп. – М.: ДМК Пресс, 2015. – 588 с.
16. Аглицкий Д.С., Аглицкий И.С. Рынок информационных технологий: проблемы и решения. - М.: 2019
17. Алексеева О.Г. Методические указания по экономическому обоснованию выпускных квалификационных работ бакалавров: Метод. указания, СПб.: Изд-во СПбГЭТУ “ЛЭТИ”, 2013. С
18. Крэг Ларман. Применение UML 2.0 и шаблонов проектирования = Applying UML and Patterns : An Introduction to Object-Oriented Analysis and Design and Iterative Development. — 3-е изд. — М.: Вильямс, 2006. — 736 с.
19. Джозеф Шмуллер. Освой самостоятельно UML 2 за 24 часа. Практическое руководство = Sams Teach Yourself UML in 24 Hours, Complete Starter Kit. — М.: Вильямс, 2005. — 416 с.
20. Грейди Буч, Джеймс Рамбо, Айвар Джекобсон. Язык UML. Руководство пользователя = The Unified Modeling Language user guide. — 2-е изд. — М., СПб.: ДМК Пресс, [Питер](#), 2004. — 432 с.

21. Буч Г., Якобсон А., Рамбо Дж. UML. Классика CS / С. Орлов. — 2-е изд.. — СПб.: Питер, 2006. — 736 с.
22. Леоненков А. В. Самоучитель UML 2. — СПб.: БХВ-Петербург, 2007. — 576 с.

## ПРИЛОЖЕНИЕ А

Таблица 1 Главный раздел сценария «Установка и чтение настроек»

<b>Главный раздел сценария выполнения варианта использования "Установка и чтение настроек"</b>	
<b>Вариант использования</b>	Установка и чтение настроек
<b>Актеры</b>	Радар
<b>Цель</b>	Применение настроек пользователя к радару. По такому же сценарию проводится чтение настроек
<b>Тип</b>	Расширенный
<b>Ссылки на другие варианты использования</b>	

Таблица 2 Типичный ход сценария «Установка и чтение настроек»

<b>Типичный ход событий сценария выполнения варианта использования "Установка и чтение настроек"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
	3. Чтение локальных настроек
	4. Вывод окна настроек с локальными настройками
5. Изменение настроек	
	6. Преобразование и отправка введенных настроек
	Исключение 1: отсутствует связь с радаром
	7. Получение подтверждения изменений



	Исключение 2: отсутствует подтверждение вхождения изменений в силу
	8. Вывод подтверждения

Таблица 3 Исключения сценария «Установка и чтение настроек»

<b>Исключения сценария выполнения варианта использования "Установка и чтение настроек"</b>	
Исключение 1: отсутствует связь с радаром	
<b>Действия актеров</b>	<b>Отклик системы</b>
	7. Сообщение "Технические неполадки"
Исключение 1: отсутствует подтверждение вхождения изменений в силу	
	8. Сообщение "Не удалось применить настройки"

Таблица 4 Главный раздел сценария «Нанесение данных на карту»

<b>Главный раздел сценария выполнения варианта использования "Нанесение данных на карту"</b>	
<b>Вариант использования</b>	Нанесение данных на карту
<b>Актеры</b>	Радар
<b>Цель</b>	Нанесение набора данных собранных за один момент времени на карту
<b>Тип</b>	Расширенный
<b>Ссылки на другие варианты использования</b>	

Таблица 5 Типичный ход сценария «Нанесение данных на карту»

<b>Типичный ход событий сценария выполнения варианта использования "Нанесение данных на карту"</b>	
<b>Действия актеров</b>	<b>Отклик системы</b>
6.Отправка полученной информации	
	7.Парсирование информации
	8.Занесение обработанной информации в базу данных
	Исключение 1: отсутствует связь с базой данных
	9.Нанесение данных на карту слежения
	Исключение 2: нет целевого объекта на карте

Таблица 6 Исключения сценария «Нанесение данных на карту»

<b>Исключения сценария выполнения варианта использования "Нанесение данных на карту"</b>	
Исключение 1: отсутствует связь с базой данных	
<b>Действия актеров</b>	<b>Отклик системы</b>
	7. Добавление в названии окна "Отсутствует связь с базой данных" вместе с оповещением
Исключение 2: нет целевого объекта на карте	
	10.Создание судна как объекта
	11.Нанесение судна и его курса и направления на карту
	12.Нанесение данных на карту

## ПРИЛОЖЕНИЕ Б

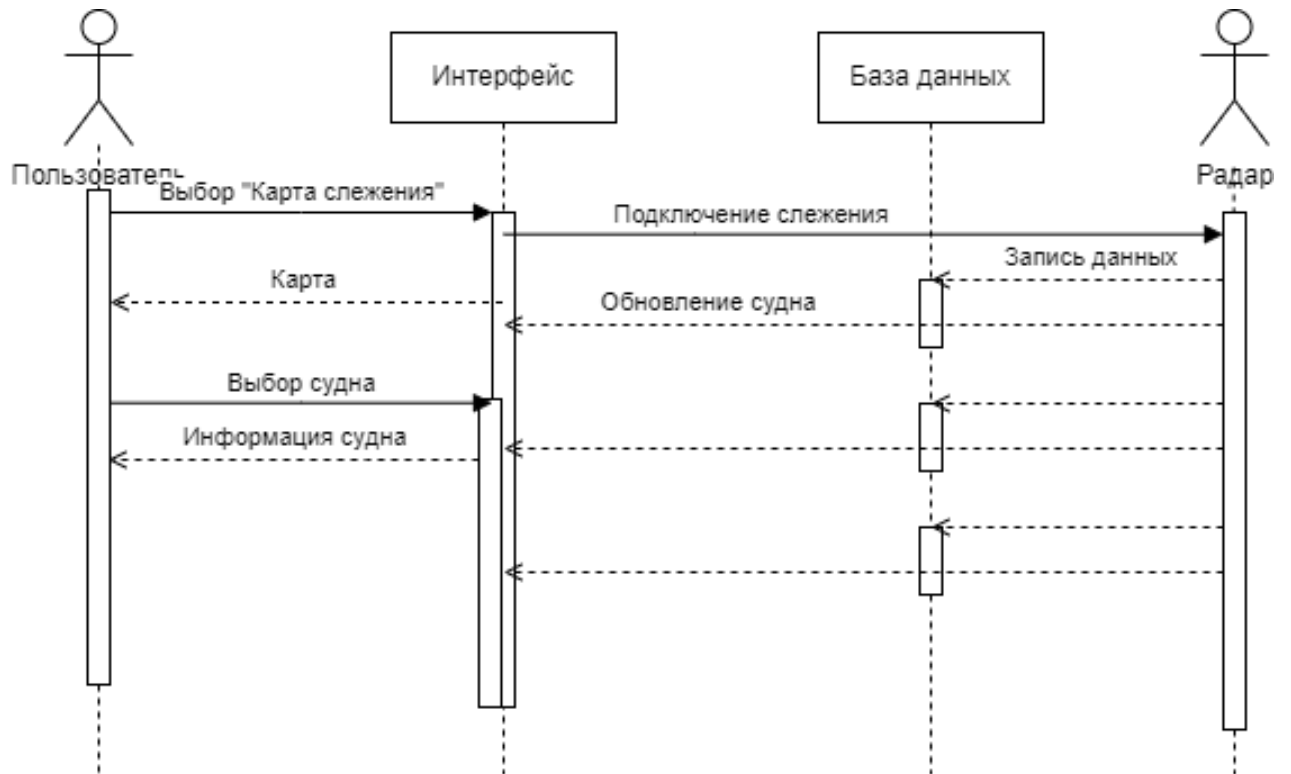


Рис 1. Диаграмма последовательности сценария «Просмотр карты» и частично сценария «Нанесение данных на карту»