

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ

Математико-механический факультет

Кафедра системного программирования

Логистический портал Cargotime:

Разработка сервиса для отслеживания судов

ОТЧЕТ ПО УЧЕБНОЙ ПРАКТИКЕ

Маркус Владислав Александрович

1 курс, группа 23.M07-мм

Научный руководитель:

к.т.н., доцент кафедры информатики, Абрамов М.В.

Консультант:

старший преподаватель кафедры информатики, А.А. Корепанова

Санкт-Петербург, 2024

Оглавление

Введение	3
1. Постановка задачи	4
2. Обзор предметной области	6
3. Обзор смежных программных продуктов	7
4. Технологии и инструменты для работы над проектом	10
Заключение	12
Список источников и литературы.....	13

Введение

С развитием международной торговли логистика стала одной из важнейших вещей в современном мире. Сложно представить современную мировую экономику без цепочек поставок грузов, ежедневно заплывающих во все порты мира транспортных судов, а также фур, самолётов, товарных поездов и многочисленных служб доставки. Само собой разумеется, такая сложная и разветвлённая система не может существовать без какого-либо управления и систем коммуникации. И, если международные отношения регулируются государствами, то конкретные поставки контролируются локальными компаниями, и, в некотором смысле, конкретными логистами.

Любая пересылка товара описывается взаимоотношениями "X поставляет груз – Y его перевозит – Z организует приём". Для организации подобных взаимоотношений требуются специализированные средства коммуникации. Благодаря развитию интернета, взаимодействие перешло в online-плоскость, и реализуется через логистические порталы. Это позволяет участвовать в торговле как крупным корпорациям, так и крупному и малому бизнесу [1]. Особенно хорошо логистические порталы развиты в области частных автомобильных грузоперевозок, так как они наиболее доступны и закрывают потребности всех категорий бизнеса. Примеры таких порталов: Ati.su, 2082.ru.

В других сферах также существуют аналогичные решения. В сфере морских перевозок такой платформой является Cargotime.ru. Уже в данный момент она позволяет по номеру контейнера определить его текущее состояние. Однако, для выполнения ряда задач, необходимо создание функционала, позволяющего определить местоположение и пройденный маршрут конкретных судов.

1. Постановка задачи

Требуется реализовать сервис отслеживания текущего местоположения судна.

Задачами проекта являются:

- Ознакомиться с предметной областью и набором инструментов, необходимых для работы над проектом;
- Разработать вышеуказанный сервис, учитывая следующие пункты:
 - Данные о текущем местоположении по MMSI-коду или по названию судна необходимо извлекать из сервиса VesselFinder;
 - Используемые данные необходимо логировать;
 - Необходимо реализовать валидацию входных и выходных данных;
 - Также, необходимо использовать Docker-контейнер.

Первое применение этого сервиса — приложение для визуализации передвижения определенных лодок, которые исследуют морское дно. Информацию о судах необходимо собирать несколько раз в сутки, также необходимо реализовать API для извлечения истории перемещения лодок.

Ещё одна подзадача — улучшение ответов на пользовательские запросы трекинга контейнеров. Планируется выводить на карте в пользовательском интерфейсе пройденный путь, представление о котором может быть получено из ответов морской линии, и текущее местоположение судна, которое будет извлекаться из VesselFinder и его аналога. Сервис VesselFinder планируется использовать в качестве запасного варианта на случай отключения/поломки сервиса MarineTraffic, который при идентичном запросе предоставляет больше информации о судне.

Уникальность данной задачи состоит в том, чтобы объединить данные из различных источников и, для каждого судна, составить полный маршрут его перемещения. Это может потребовать дополнительных ресурсов для

хранения данных и утяжелить их отображение, но, с другой стороны, это позволит получать дополнительные сведения, как в исследовательских, так и в обывательских целях.

2. Обзор предметной области

Уже в 20 веке количество кораблей в море стало довольно большим, и переговоры между отдельными судами и портами при помощи флажных сигналов и радиосообщения стали терять в эффективности. Возникла необходимость регуляции и контроля морского сообщения без непосредственного общения с командой корабля. В силу первичности развития технологий ЭВМ, такое решение было предложено только в 90-х годах.

Была создана система под названием AIS (Automatic Identification System) [2]. Она создавалась для решения двух задач: улучшение безопасности морской навигации и предотвращение столкновений. По сути, AIS функционирует как ЭРА-ГЛОНАСС в автомобилях, но с теми отличиями, что метка каждого корабля передаёт информацию не только о корабле и его координатах, а ещё и его текущую скорость, курс, и множество других данных [3]. С 2002 года IMO (International Maritime Organization) и SOLAS (Safety of Life at Sea) требуют установки AIS на все корабли водоизмещением больше 300 тонн, которые выходят в международные воды. Приёмниками сигнала выступают как наземные вышки, так и спутники. В силу несовершенства используемого протокола передачи данных, созданного на основе протокола TDMA (Time-division multiple access), полный переход на спутниковое сообщение остаётся невозможным.

Данные, предоставляемые AIS, публичны, что позволяет всем желающим посмотреть текущее местоположение и статус различных судов. Эти данные доступны для просмотра посредством использования различных сервисов, о которых пойдёт речь далее.

3. Обзор смежных программных продуктов

Как было замечено в прошлой главе, прямых бесплатных аналогов разрабатываемый продукт не имеет. Однако есть ряд продуктов, которые непосредственно связаны с разрабатываемым сервисом. В первую очередь, речь идёт о приложениях, позволяющих узнать текущее местоположение судна и некоторые его характеристики. Разберём каждый из таких продуктов более подробно.

Основным и самым популярным считается сервис [marinetraffic.com](https://www.marinetraffic.com), и, по причинам интерфейсных и функциональных сходств между им и остальными аналогичными сервисами, мы будем обозревать остальные сервисы, главным образом отталкиваясь от отличий в сравнении с [marinetraffic](https://www.marinetraffic.com).

1. [marinetraffic.com](https://www.marinetraffic.com)

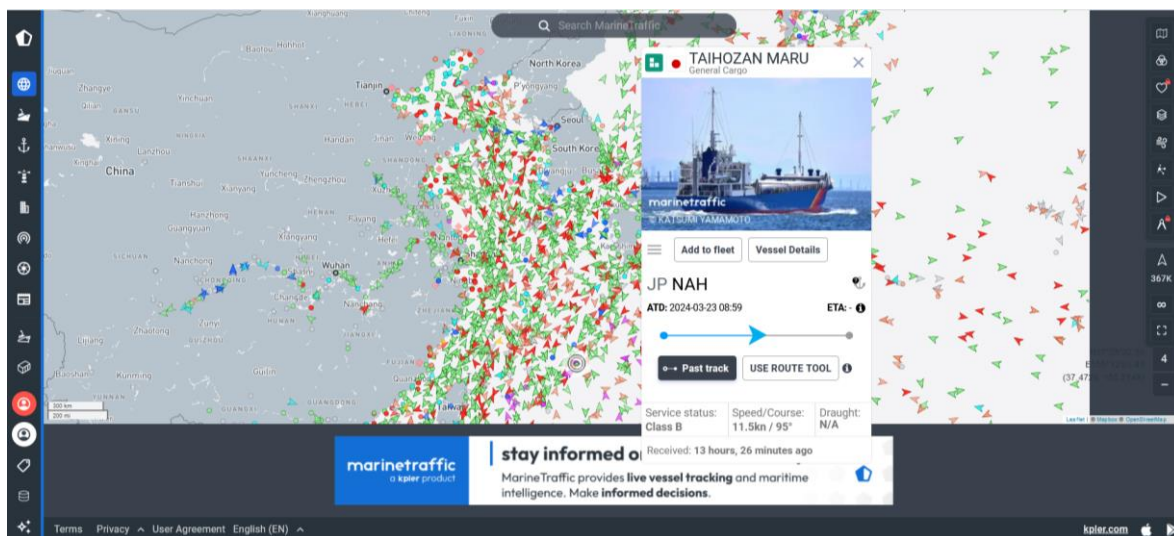


Рисунок 1 – Интерфейс сервиса [marinetraffic](https://www.marinetraffic.com)

Сервис, позволяющий получить информацию о судах, портах, маяках. Позволяет получить такую информацию, как: название судна, фото (опционально), тип судна, порт отправления и порт назначения, навигационный статус, курс, скорость, и даже предполагаемое время прибытия в порт назначения. Также сервис предоставляет возможность построить маршрут из порта А в порт Б, используя актуальные морские маршруты, используемые в навигации в данный момент.

2. vesselfinder.com

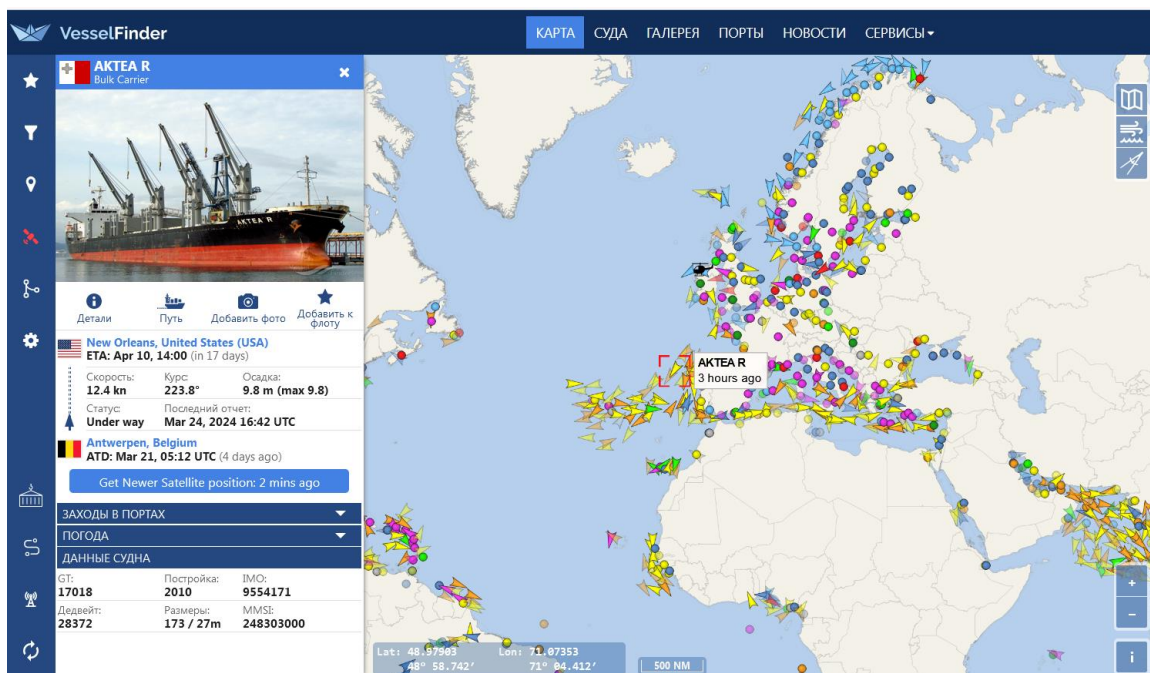


Рисунок 2 – Интерфейс сервиса vesselfinder

Данный сервис не обладает информацией о маяках, однако опционально предоставляет больше дополнительной информации о судне, например, текущие погодные условия и историю заходов в порты.

3. myshiptracking.com

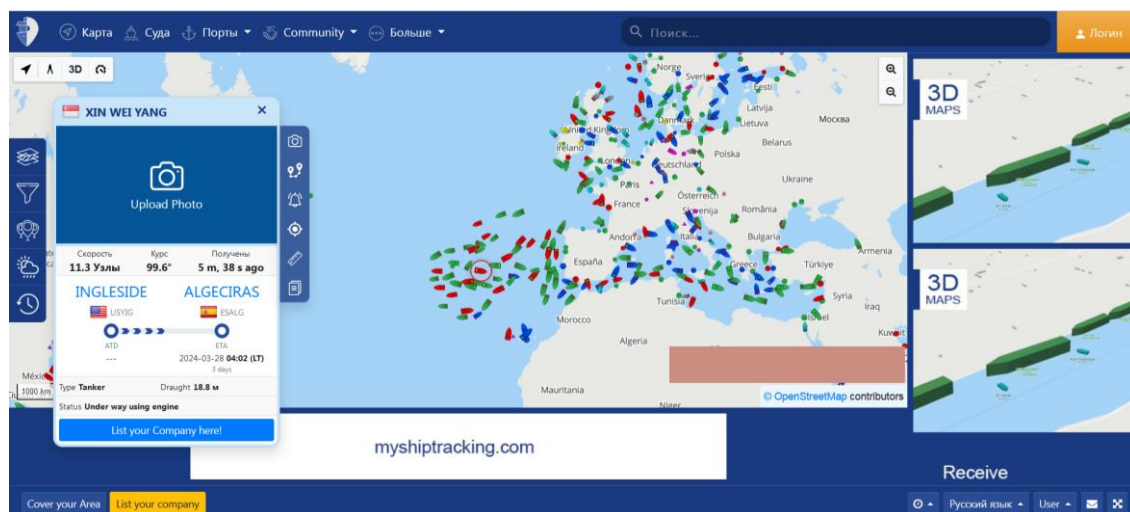


Рисунок 3 – Интерфейс сервиса myshiptracking

Схож с vesselfinder, но обладает также возможностью показывать предыдущие две точки местонахождения судна и встроенными 3D картами. Трёхмерные карты, однако, не функциональны в данный момент времени, и их необходимость остаётся под вопросом.

4. trackipi.com

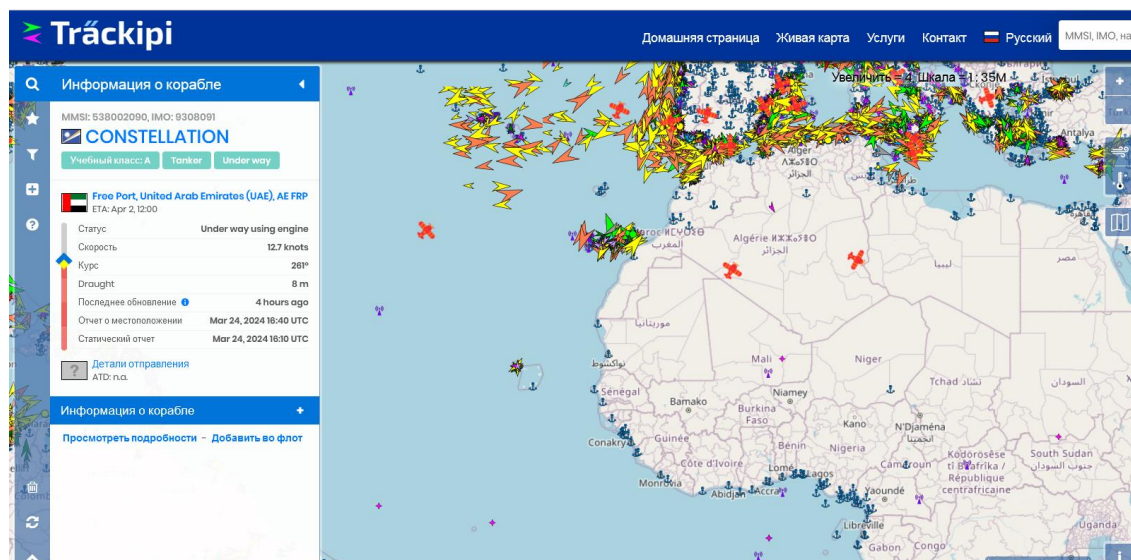


Рисунок 4 – Интерфейс сервиса trackipi

Данный сервис предоставляет минимум информации о судах, но содержит некоторые данные о поисково-спасательных самолётах.

5. shiplocation.com

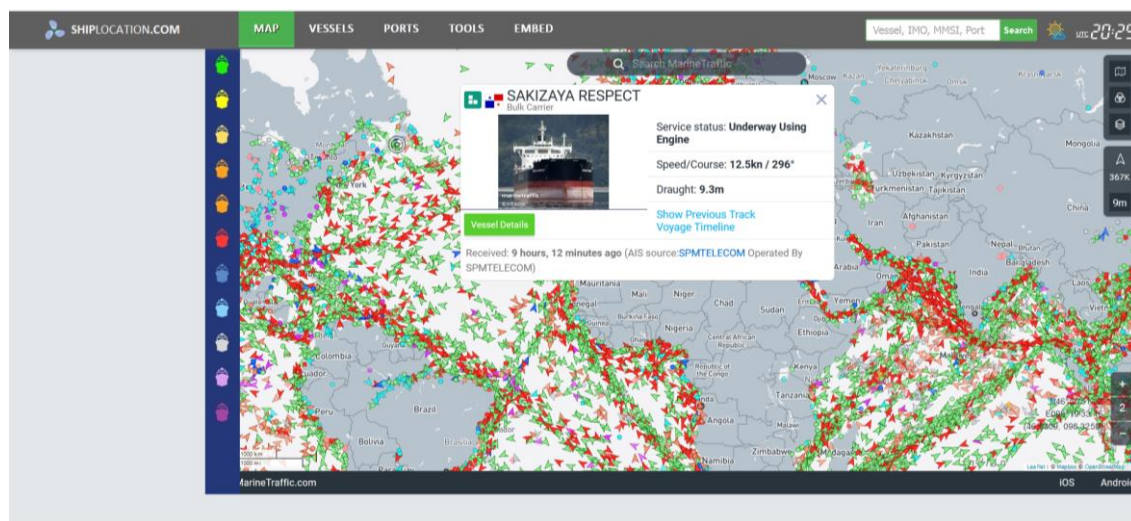


Рисунок 5 – Интерфейс сервиса shiplocation

Сервис подкачивает всю информацию, в том числе и интерактивную карту, с marinetraffic.com, урезая часть функционала, и при запросе дополнительной информации перенаправляет на оригинальный сайт.

Резюмируя, можно утверждать, что все вышеперечисленные продукты, в том или ином виде, предоставляют схожий функционал, не удовлетворяющий потребностям, отображённым в задачах данного проекта, однако могут послужить источниками информации о судах.

4. Технологии и инструменты для работы над проектом

В рамках работы над данным проектом предполагается использование следующего набора технологий/инструментов.

Таблица 1. – Технологии и инструменты для работы над проектом

Название инструмента	Назначение использования	Краткое описание
NestJs	Основная платформа для реализации сервиса	NestJS это фреймворк для создания серверных приложений. Он написан на TypeScript и поддерживает его использование, но приоритетным ЯП для работы считается Javascript. NestJs позволяет работать в таких парадигмах программирования, как объектно-ориентированное программирование, функциональное программирование и функционально-реактивное программирование [4].
Redis	Инструмент для эффективной работы с БД и кэширования данных	Redis (Remote Dictionary Server) — это нереляционная резидентная СУБД, хранящая данные в виде пар «ключ-значение». От реляционных БД Redis отличается: 1) более высокой производительностью, благодаря хранению данных в оперативной памяти сервера; 2) отсутствием языка SQL; 3) гибкостью, благодаря использованию более удобных структур хранения данных; 4) лучшей масштабируемостью [5].
RabbitMQ	Брокер сообщений для работы с запросами пользователей	RabbitMQ – это брокер сообщений с открытым исходным кодом, который маршрутизирует сообщения, отправленные приложением-

		продюсером согласно принципам протокола AMQP (Advanced Message Queuing Protocol) [6].
Elastic- Logstash- Kibana	Стек для осуществления распределённого логирования	ELK стек обеспечивает удобное централизованное логирование (ведение журналов) с разных серверов. Он позволяет надёжно и безопасно получать данные из любого источника во всех форматах и работать с этими данными: осуществлять поиск по ним, анализировать и визуализировать их в режиме real-time (NRT) [7].
Docker	Платформа для развёртывания сервиса	Docker — это платформа, которая предназначена для разработки, развёртывания и запуска приложений в контейнерах. В каком-то смысле, Docker является логическим продолжением концепции виртуальных машин, но контейнеры Docker-а ресурсно легче, обладают лучшей переносимостью и запускаются быстрее [8].

Необходимость использования именно данного набора инструментов обусловлена текущей архитектурой проекта, в которую будет внедрён разрабатываемый сервис.

Заключение

В рамках учебной практики были выполнены следующие задачи:

- Проведён обзор существующих аналогичных сервисов, которые могут послужить вспомогательными инструментами при работе над проектом;
- Изучена предметная область, выделены основные понятия и принципы, необходимые для реализации проекта
- Определены инструменты для работы над проектом, разобран принцип их работы;

Далее планируется произвести последовательную разработку всех необходимых компонент сервиса, с использованием указанных инструментов. Также планируется произвести тестирование и отладку разработанного сервиса и внедрить его в уже действующую платформу Cargotime.ru.

Список источников и литературы

1. Логистика в современном мире — основные принципы и успешные стратегии // Logists [Электронный ресурс]. – 2024 – URL: <https://logists.by/blog/logistika-v-sovremennom-mire-osnovnye-printsipy-i-uspeshnye-strategii?ysclid=lu5pxudmrm561007028> (дата обращения 13.01.24).
2. Необыкновенные приключения AIS в Китае, или как найти корабль в море // Habr [Электронный ресурс]. – 2024 – URL: <https://habr.com/ru/articles/591557/> (дата обращения 13.01.24).
3. Спутниковое слежение за транспортом: что представляет собой система ГЛОНАСС // ТрансТехСервис [Электронный ресурс]. – 2024 – URL: <https://izhevsk.tts.ru/blog/tyuning/sputnikovoe-slezhenie-za-transportom-glonass/?ysclid=lu6xsj6v1a351747335/> (дата обращения 25.01.24).
4. Введение к документации | NestJs // NestJs [Электронный ресурс]. – 2024 – URL: <https://nestjs.ru/?ysclid=lu8c9zx4up999020904> (дата обращения 08.02.24).
5. Знакомимся с Redis // Skillbox [Электронный ресурс]. – 2024 – URL: https://skillbox.ru/media/code/znakomimsya_s_redis/?ysclid=lu8chdaktb772428144 (дата обращения 09.02.24).
6. RabbitMQ для аналитика: практический ликбез // BabokSchool [Электронный ресурс]. – 2024 – URL: <https://babok-school.ru/blog/rabbitmq-for-analyst/?ysclid=lu8ctcr3oy361753320> (дата обращения 09.02.24).
7. Что такое ELK-стек: установка и настройка Elasticsearch, Logstash и Kibana // Dzen [Электронный ресурс]. – 2024 – URL: <https://dzen.ru/a/YqnG8Y9PGk46pXTa> (дата обращения 11.02.24).
8. Изучаем Docker // Habr [Электронный ресурс]. – 2024 – URL: <https://habr.com/ru/companies/ruvds/articles/438796/> (дата обращения 11.02.24).