# Разработка перечня задач проекта

Программный продукт «Программная система для навигации в акватории на основе кластеризации траекторий движения» состоит из следующих подсистем:

1. **UserInterface (UI)** – пользовательский интерфейс.
2. **LoadData (LD)** – модуль загрузки и обработки данных.
3. **DoMagic** **(DM)** – модуль вычислений (кластеризация и разбиение на полигоны).
4. **FindRoute** **(FR)** – модуль нахождения кратчайшего пути.
5. **ShowResult (SR)** – модуль отображения полученных данных.

## Задачи для подсистемы «Пользовательский интерфейс»

*Задача TASK\_UI\_001*

Название: Проектирование интерфейса.

Описание: основываясь на требованиях, разработать интерфейс. Минимальный функционал: окно карты в формате картинки, кнопка отображения всех маршрутов на карте, кнопка отображения части маршрутов на карте.

## Задачи для подсистемы «Модуль загрузки и обработки данных»

*Задача TASK\_LA\_001*

Название: Фильтр загружаемых данных.

Описание: После загрузки данных из файла, необходимо их обработать таким образом, чтобы в них не было повторяющихся строк, данных о судах с курсом 511, а также информации о судах со значением 0 в столбцах length и port – это могут быть статичные объекты, например буйки.

## Задачи для подсистемы «Модуль вычислений»

*Задача TASK\_* *DM \_001*

Название: Реализация DBSCAN.

Описание: В python алгоритмы кластеризации уже реализованы в различных библиотеках. Необходимо найти и подключить подходящую библиотеку, например, sklearn.

## Задачи для подсистемы «Модуль нахождения кратчайшего пути»

*Задача TASK\_FR\_001*

Название: Класс Graph.

Описание: Для того чтобы сделать реализацию алгоритма Дейкстры более лаконичной, создается класс Graph.

Реализуем граф как словарь Python. Ключи словаря будут соответствовать вершинам (полигонам), а его значения будут соответствовать рёбрам, где записывают расстояния до других вершин (полигонов) на графе.

## Задачи для подсистемы «Модуль отображения полученных данных»

*Задача TASK\_SR\_001*

Название: Реализация наложения на изображение.

Описание: В python огромное множество библиотек. Необходимо найти и подключить, подходящую для спроектированного в *TASK\_UI\_001* интерфейса, библиотеку, отображающую граф на изображении.

## Задачи для системы в целом

*Задача TASK\_SYS\_001*

Название: собрать программу.

Описание: сгенерировать .exe файл для упрощенного взаимодействия пользователя с интерфейсом.