# ПРИЛОЖЕНИЕ Б Техническое задание

ВВЕДЕНИЕ

Программный комплекс **«NavRobot»**, далее по тексту **«NavRobot»**, предназначен для автоматизации процессов навигации и управления четырёхколёсным роботом. Комплекс упрощает выполнение задач по картографированию и локализации, избавляя от необходимости ручного контроля и снижая вероятность ошибок при обработке данных. Это повышает эффективность исследований и учебных лабораторных работ. Программный комплекс может быть использован студентами технических вузов, а также в научных и промышленных проектах, связанных с автономной робототехникой.

1. Основание для разработки

Основанием для разработки является приказ по институту «ФГБОУ ВО «МАИ (НИУ) - Московский Авиационный Институт (Национальный Исследовательский Университет)» №\_\_\_\_\_\_\_\_\_\_от «\_\_\_» \_\_\_\_\_\_\_\_202 г.

2. Назначение разработки

2.1. Функциональное назначение

Программный комплекс "NavRobot" предназначен для автоматизации процессов навигации и управления четырёхколёсным роботом, включая задачи картографирования, локализации и построения маршрутов в реальном времени.

2.2. Эксплуатационное назначение

"NavRobot" может быть использован:

* в учебном процессе для изучения основ робототехники и автономных систем;
* в научных исследованиях для тестирования алгоритмов SLAM;
* в промышленных приложениях для автономного мониторинга территорий.

3. Требования к программе

Программный комплекс должен быть разработан с использованием модульной архитектуры на базе ROS (Robot Operating System). Система должна обеспечивать сбор, обработку и визуализацию данных с датчиков робота.

3.1. Требования к функциональным характеристикам

Программный комплекс "NavRobot" должен выполнять следующие функции:

* Автономная навигация по заданным маршрутам
* Построение карты окружающей среды (SLAM)
* Обработка данных с лидара и камер
* Визуализация карт и траектории движения
* Ручное управление через джойстик
* Сохранение и загрузка карт местности

3.2. Организация входных данных

Входными данными для программного комплекса являются:

* Показания лидара (сканы окружения)
* Данные с инерциального измерительного модуля (IMU)
* Видеопоток с камеры
* Команды управления с джойстика
* Параметры конфигурации системы

3.3. Организация выходных данных

Выходными данными являются:

* Карта окружающей среды в формате occupancy grid
* Траектория движения робота
* Графики точности локализации
* Логи работы системы
* Визуализация в реальном времени

3.4. Требования к надежности

Для обеспечения нормального функционирования программного комплекса «NavRobot» необходимо предусмотреть обработку исключительных ситуаций, а именно:

* Контроль целостности данных с датчиков
* Обработку потери связи с оборудованием
* Восстановление после сбоев
* Защиту от некорректного ввода параметров

3.5. Условия эксплуатации

Для работы с системой не требуется специально подготовленный оператор. Интерфейс должен быть интуитивно понятным.

3.6. Требования к составу и параметрам технических систем

Минимальные требования:

* Процессор: Intel Core i5 или аналогичный
* ОЗУ: 8 ГБ
* Видеокарта: с поддержкой OpenGL 3.3
* Жесткий диск: 128 ГБ SSD
* ROS-совместимые датчики (лидар, камера, IMU)

3.7. Требования к информационной и программной совместимости

Необходимое ПО:

* ОС: Ubuntu 20.04 LTS
* ROS: Noetic Ninjemys
* Python: 3.8+
* Драйверыдля оборудования

Для разработки рекомендуется:

* Среда разработки: VS Code или PyCharm
* Система контроля версий: Git

4. Требования к программной документации

Состав программной документации:

Основные документы:

* "NavRobot". Техническое задание (полное описание требований к системе)
* "NavRobot". Пояснительная записка (детальное описание реализации)
* "NavRobot". Руководство пользователя (инструкции по эксплуатации)

Технические документы:

* "NavRobot".

Руководство программиста:

* Описание архитектуры системы
* Инструкции по установке и настройке
* API и интерфейсы

Дополнительные документы:

1. "NavRobot". Руководство системного администратора (при необходимости)
2. "NavRobot". Программа и методика испытаний
3. "NavRobot". Описание базы данных

5. Стадии и этапы разработки

5.1. Эскизный проект

* Проведение анализа существующих решений в области автономной навигации роботов
* Разработка концепции системы и выбор архитектурных решений
* Определение структуры входных (данные сенсоров) и выходных данных (карты, траектории)
* Создание предварительного описания алгоритмов SLAM и навигации
* Разработка технико-экономического обоснования проекта.

Срок окончания: \_\_\_

5.2. Технический проект

* Уточнение структуры входных и выходных данных.
* Разработка средств обработки ошибок и выдачи диагностических сообщений.
* Окончательное определение конфигурации технических и программных средств.
* Разработка пояснительной записки.

Срок окончания: \_\_\_\_\_\_

5.3. Рабочий проект

* Реализация программного комплекса на выбранной платформе
* Проведение модульного и интеграционного тестирования
* Отладка алгоритмов навигации и картографирования
* Разработка полного комплекта программной документации:

1. Руководство пользователя
2. Руководство программиста
3. Описание\_API

* Создание программы и методики испытаний
* Проведение предварительных испытаний системы
* Корректировка документации по результатам тестирования

Срок окончания: \_\_\_\_\_\_\_\_\_\_\_

5.4. Внедрение

Сдача выпускной квалифицированной работы бакалавра на кафедру.

Срок окончания: \_\_\_\_\_\_\_06.06.2024\_\_\_\_

6. Порядок контроля и приемки

Контроль и приёмка разработки оясуществляется в соответствии с программой и методикой испытаний, изложенными в пояснительной записке.

7. Перечень сокращений

ROS - Robot Operating System (операционная система для роботов)  
SLAM - Simultaneous Localization and Mapping (одновременная локализация и картографирование)  
IMU - Inertial Measurement Unit (инерциальный измерительный модуль)  
LiDAR - Light Detection and Ranging (лазерный сканер)  
API - Application Programming Interface (интерфейс программирования приложений)  
GUI - Graphical User Interface (графический интерфейс пользователя)  
RAM - Random Access Memory (оперативная память)  
CPU - Central Processing Unit (центральный процессор)  
GPU - Graphics Processing Unit (графический процессор)  
OS - Operating System (операционная система)  
IDE - Integrated Development Environment (интегрированная среда разработки