Chapter 1

System Design

1.1 Requirements Specification

1.1.1 User Requirements

1.1.2 Functional Requirements

1.1.3 Non-Functional Requirements

1.2 System Modelling

1.2.1 Overview

1.2.2 Interaction Models

1.2.3 Core Class Diagrams

Class diagrams showing relationships between entities such as inheritance, each class diagram should be accompanied with an explanation. Core classes to model includes:

- Robot
- Proxy
- Planner
- Algorithm (abstract)
- DStarLite (inherits Algorithm)
- FieldDStar (inherits Algorithm)

1.2.4 Threaded Architecture

Since threaded programming is notoriously difficult to understand the threading model that the system uses will be covered here in detail, around 1-2 pages. Include a high level diagram and state how each thread interacts with the others. Likely to be around three threads:

- Main - User interaction
- Proxy - Robot Communications
- Planner - Path Planning

1.2.5 Communications Protocol

1.3 Interchangeable User Interfaces

1.4 Programming Platforms

1.4.1 Unix Flavours

1.4.2 Python3

1.4.3 Cython Modules

1.5 Source Control

1.5.1 Git

1.5.2 Open Repository

1.6 Hardware Agent Specification

1.6.1 Motion Model