Краткое объяснение подхода к решению

В этом документе я постараюсь кратко объяснить в чем была идея моего решения с указанием конкретных классов.

Класс **RobotModel** представляет самого робота. У него есть координаты и направление которые можно менять использую сетеры и гетеры.

Абстрактный класс **Map** представляет абстрактную карту, которая имеет длину и ширину. Т к от карты нам нужно только узнавать свободна ли конкретная клетка – соответствующий метод единственный абстрактный. **MoonMap** представляет данную карту. Мы можем с легкостью заменить его на любой другой имеющий возможность проверить по запросу конкретную клетку.

Интерфейс *ICommandSource* предоставляет возможность с легкостью менять место, откуда поступают команды. В данном случае это класс **CommandParser**.

Интерфейс *IOutputDeclarator* и класс **OutputDeclarator** позволяют делать то же самое с местом вывода результата.

Перечисления **Directions CommandTypes** служат для отображения направлений и типов команд у робота соответственно.

Класс **Controller** собирает воедино все части и оперирует ими, выполняя команды с помощью метода executeCommands и выводит финальное состояние с помощью метода declareOutput.

Класс **Main** это начальная точка приложения.