Document Title

Table of Contents

[Requirements 2](#_Toc155086100)

[Functionality 2](#_Toc155086101)

[Usability 2](#_Toc155086102)

[Reliability 2](#_Toc155086103)

[Performance 2](#_Toc155086104)

[Supportability 2](#_Toc155086105)

[Use Case: 2](#_Toc155086106)

[Title 2](#_Toc155086107)

[Primary Actor 2](#_Toc155086108)

[Success scenario: 2](#_Toc155086109)

[Use Case Diagram 2](#_Toc155086110)

[Class relationships 3](#_Toc155086111)

[Conceptual Object Model 4](#_Toc155086112)

[Identifying objects, relationships and responsibilities 4](#_Toc155086113)

[UML Class Diagram 5](#_Toc155086114)

[Visibility 5](#_Toc155086115)

[- private 5](#_Toc155086116)

[+ public 5](#_Toc155086117)

[Static will have an underline. 5](#_Toc155086118)

[Inheritance 6](#_Toc155086119)

[Abstract class 6](#_Toc155086120)

[Interface 7](#_Toc155086121)

# Requirements

## Functionality

Refers to the capabilities and features of the app. This is the core of what the customer wants.

## Usability

Affects the person who will be using the program. Is it easy on the eyes? Is it intuitive to use? Is the documentation accurate and complete?

## Reliability

You must know how much system downtime is acceptable to define reliability requirements. If the failures are predictable, how can the system be recovered?

## Performance

Requirements dictate the application's response time throughput. And they put limits on the system resources it can use.

## Supportability

Ensure the application can be tested, extended, serviced, installed and configured.

# Use Case:

## Title

Heat Meal

## Primary Actor

Astronaut

## Success scenario:

Astronaut inserts meal package.

The system identifies the type of meal.

The system heats the package for the time required for meal type.

The system notifies astronauts that the meal is ready via space pager.

The astronaut removes the package from the system.

# Use Case Diagram

2 actors

3 scenarios in the box to illustrate the software.

1 external actor but not human, so put in a rectangle.

A whiteboard with text on it

Description automatically generated

# Class relationships

A white board with blue writing

Description automatically generated

# Conceptual Object Model

A black background with white text

Description automatically generated

# Identifying objects, relationships and responsibilities

A screenshot of a computer screen

Description automatically generated

# UML Class Diagram

## Visibility

### - private

### + public

A screenshot of a computer program

Description automatically generated

## Static will have an underline.

A screenshot of a computer

Description automatically generated

## Relationships

A black screen with white text

Description automatically generated

A screenshot of a computer program

Description automatically generated

## Inheritance

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

## Abstract class

Use Italic font to mark class or method as abstract.

## Interface

A diagram of a program

Description automatically generated

## Aggregation

### Has a – attribute

Fleet has spaceship\s

A screen shot of a black background

Description automatically generated

## Composition

### Owns a – attribute

Spaceship owns an engine. (0, 1 or more)

The Engine does not exist without a spaceship.

A screenshot of a computer

Description automatically generated