

---

---

# **Lejos Especificaciones Suplementarias**

**Version <1.0>**

Lejos	Version: <1.0>
Especificaciones Suplementarias	Date: <15/03/16>
<identificador de documento>	

## Historia del documento

Fecha	Versión	Descripción	Autor
18/03/16	1.0>	Creación del documento	Medina Barrera Luis Alberto
28/03/16	2.0	Revisión para la primera iteración	Medina Barrera Luis Alberto

## Contenido

1. Introducción
  - 1.1 Propósito
  - 1.2 Alcance
  - 1.3 Definiciones, Acrónimos y Abreviaciones
  - 1.4 Referencias
  - 1.5 Visión General
2. Funcionalidad
  - 2.1 Generación de ejecutable Lejos
3. Usabilidad
  - 3.1 Utilización
  - 3.2 Lenguaje
4. Confiabilidad
  - 4.1 Disponibilidad
5. Rendimiento
  - 5.1 No aplica
6. Soporte
  - 6.1 No aplica
7. Restricciones de diseño
  - 7.1 Argo UML
  - 7.2 LibreOffice 5.0
  - 7.3 Git
  - 7.4 NetBeans 8.0
  - 7.5 Eclipse
  - 7.6 Microsoft Windows XP Service Pack 2 o superior
  - 7.7 Hardware base
8. Requerimientos de documentación del usuario y ayuda en línea

5  
5  
5  
5  
5  
5  
5

Lejos	Version: <1.0>
Especificaciones Suplementarias	Date: <15/03/16>
<identificador de documento>	

#### 8.1 Documentación

### 9. Componentes comprados

No aplica

### 10. Interfaces

- 10.1 Interfaz de Usuario
- 10.2 Interfaz de Hardware
- 10.3 Interfaz de Software
- 10.4 Interfaz de Comunicación

### 11. Requerimientos de Licencia

### 12. Estándares aplicables

Lejos	Version: <1.0>
Especificaciones Suplementarias	Date: <15/03/16>
<identificador de documento>	

# Especificaciones Suplementarias

## 1. Introducción

### 1.1 Propósito

El propósito de este documento es definir los requerimientos que no fueron capturados en los casos de uso. El modelo de casos de uso y las especificaciones suplementarias capturan completamente los requerimientos del sistema.

### 1.2 Alcance

La primera iteración tiene como objetivo determinar los principales requerimientos que deben ser solucionados en la aplicación, después de un análisis del proyecto. Los casos de uso serán modelados y descritos cada uno. Finalmente se tendrá un primer bosquejo de la documentación de nuestro proyecto

### 1.3 Definiciones, Acrónimos y Abreviaciones

Se puede consultar el documento Glosario, que funciona como referencia para todos los artefactos.

### 1.4 Referencias

- Plan de iteración
- Documento de Visión.
- Glosario.

### 1.5 Visión General

Será definido en futuras iteraciones

## 2. Funcionalidad

El sistema permitirá la “traducción” de código nativo de Lego Mindstorm NXT a la variante basada en Java llamada Lejos. Para cumplir esta acción es necesario encontrar una equivalencia entre ambos lenguajes. Algunos puntos claves son:

### 2.1 Generación de ejecutable Lejos

Se encarga de la interpretación del código objeto generado por el compilador Age2000, para después generar código objeto del lenguaje Lejos que puede ser ejecutado en un robot Lego Mindstorm NXT.

## 3. Usabilidad

En esta sección describen los requerimientos de usabilidad del sistema. Se detallan los aspectos relacionados con la utilización. Los usuarios dispondrán de una guía donde se detallan los procedimientos para la utilización del software.

### 3.1 Utilización

Se mantiene el objetivo que se plantea en la tesis, que el software es usado para la investigación en el área de inteligencia artificial, ya sea por investigadores o profesores, así como también por los alumnos interesados. Los usuarios dispondrán de una guía donde se detalla los procedimientos para la utilización del software.

### 3.2 Lenguaje

El sistema puede ser utilizado en dos idiomas, inglés y español. Además el software cuenta con una función para la traducción inglés-español y viceversa del código introducido en el IDE.

Lejos	Version: <1.0>
Especificaciones Suplementarias	Date: <15/03/16>
<identificador de documento>	

## 4. Confiabilidad

Propiedad que tiene el sistema de realizar las funciones para las que fue diseñado sin fallos. Se detallan los aspectos relacionados con la disponibilidad.

### 4.1 Disponibilidad

El sistema debe estar disponible en cualquier momento que sea necesario utilizarlo y además debe mantenerse funcionando correctamente durante todo el tiempo de utilización.

## 5. Rendimiento

Será definido en futuras iteraciones

## 6. Soporte

Será definido en futuras iteraciones

## 7. Restricciones de diseño

El proyecto se desarrollará bajo dos entornos, NetBeans 8 y Eclipse, además será programado en Java. El proceso de desarrollo de software, especialmente la parte de documentación, utilizará parte de la metodología divide y vencerás. A continuación se enuncian los requisitos de diseño para el desarrollo de las actividades del proyecto:

### 7.1 Argo UML

Es la herramienta CASE que será usada durante el proyecto. Soporta de forma completa la especificación de UML permitiendo crear un modelo completo que representa el dominio del problema y el sistema software a desarrollar.

### 7.2 LibreOffice 5.0

Empleado en la redacción de los documentos presentados durante el desarrollo del proyecto.

### 7.3 Git

Es el sistema de control de versiones que será utilizado durante el desarrollo para facilitar el intercambio de los archivos recién modificados.

### 7.4 Netbeans 8.0

Es el IDE utilizado para el desarrollo del software.

### 7.5 Eclipse

Es el segundo IDE utilizado para el desarrollo del software.

### 7.6 Microsoft Windows XP Service Pack 2 o superior

Los requerimientos del desarrollo exigen la utilización de una versión mínima para el sistema operativo sobre el cual se lanzará la aplicación.

### 7.7 Hardware base

Procesador: Intel Pentium III o equivalente a 800 MHz

Memoria: 512 MB

Espacio de disco: 750 MB de espacio libre en el disco

Lejos	Version: <1.0>
Especificaciones Suplementarias	Date: <15/03/16>
<identificador de documento>	

## **8. Requerimientos de documentación del usuario y ayuda en línea**

### **8.1 Documentación**

Elaborar la documentación técnica y de usuario, de modo que sea clara y fácilmente entendible e interpretable por todos los usuarios que van a tener interacción con el sistema. Estos manuales deben ser difundidos a todas las áreas de interés y deben ser de fácil acceso para todos los interesados.

## **9. Componentes comprados**

No aplica

## **10. Interfaces**

Esta sección definen y describen las interfaces soportadas por este sistema. Las interfaces de Software son las definidas con otros subsistemas incluidos en el proceso mientras que las interfaces de comunicación describen las relaciones con sistemas externos. En sucesivas iteraciones se explicará con más detalle las interfaces de usuario empleadas.

### **10.1 Interfaz de Usuario**

Se definirán en próximas iteraciones

### **10.2 Interfaz de Hardware**

Se definirán en próximas iteraciones

### **10.3 Interfaz de Software**

Se definirán en próximas iteraciones

### **10.4 Interfaz de Comunicación**

Se definirán en próximas iteraciones

## **11. Requerimientos de Licencia**

Para realizar los cambios necesarios en la aplicación no será necesaria la compra de la licencia de ningún producto, ya que usamos únicamente software libre. Por otra parte, la utilización de librerías de código para la ejecución de la misma tampoco hace necesaria la adquisición de ninguna licencia comercial ni se imponen restricciones de uso.

## **12. Estándares aplicables**

Será definido en futuras iteraciones