Requirement Specification
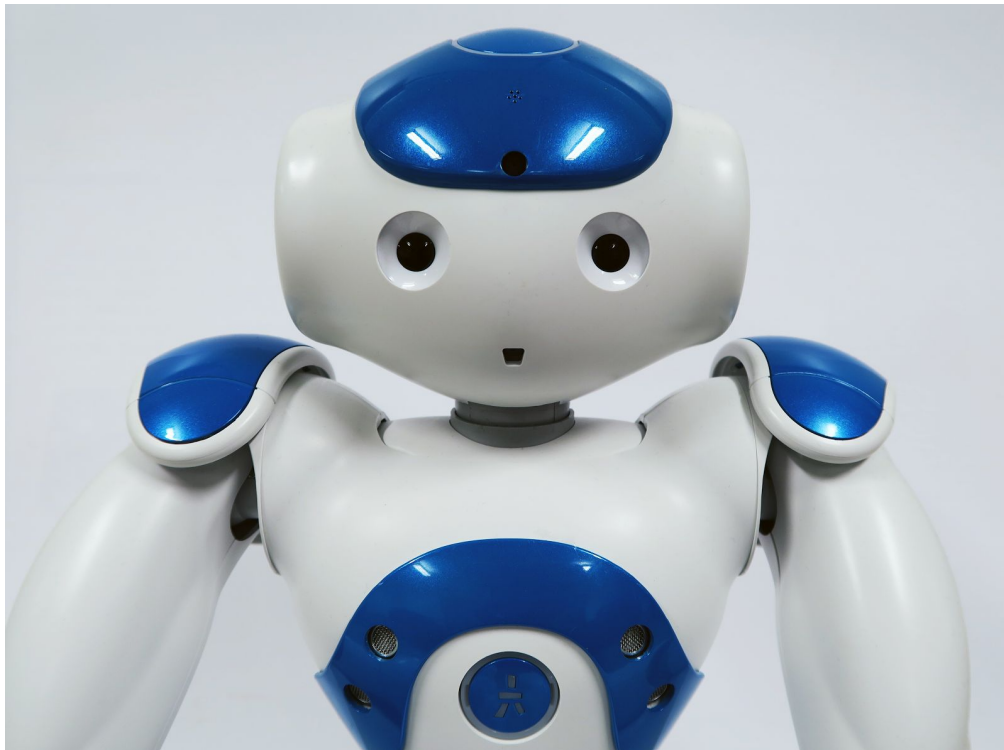
How to train your Nao

Version 1.0

TSBB11 HT2015

Project identity

Name	Responsibility	Phone	Email
Madeleine Stein	Project leader (PL)	0709383490	madst314@student.liu.se
Susanna Gladh	Quality assurance (QA)	0735030125	susgl621@student.liu.se
Anton Ågren	Testing (TES)	0768026950	antag110@student.liu.se
Fredrik Kvillborn	Design (DES)	0703050656	frekv134@student.liu.se
Elin Andersson	Scrum master (SM)	0736466026	elian253@student.liu.se
Richard Bondemark	Documentation (DOC)	0762464454	ricbo818@student.liu.se
Fredrik Löfgren	Nao Expert (NE)	0703571658	frelo223@student.liu.se

Customer: Michael Felsberg, LiU (ISY), michael.felsberg@liu.se
Project supervisor: Fahad Khan, LiU (ISY), fahad.khan@liu.se
Examiner: Michael Felsberg, LiU (ISY), michael.felsberg@liu.se

Contents

1 Introduction

This is the requirement specification of the project *How to Train Your Nao*, which is part of the course TSBB11, *Images and Graphics, Project Course CDIO* at Linköping University during HT2015.

1.1 Brief system description

The Nao robot will collect images and/or video using its two cameras. This data will be used either on its on-board computer or on an external computer to compute decisions. The decisions will be sent to the robot and acted upon. For more detail, see Project plan.

1.2 About this document

This document will describe the requirements on the system to be implemented. The requirements on the system are divided into software, performance, usage and installation requirements. This document will also list the documentation and delivery requirements of the project.

1.2.1 Requirement priorities

The requirements are prioritized in three levels:
1. Mandatory features, which must be delivered for the product to be considered finished.
2. Features which will be added if the mandatory features are completed.
3. If there is features with prio 1 and 2 are completed, requirements of priority three can be dealt with.

2 Software requirements

The robot should be able to detect and learn previously unknown objects and then be able to find them in an unknown environment.

The following notation will be used:
- ***Basic environment***: This is a very controlled and predictable environment, with a single color, homogenous background, which only contains the desired objects. The environment has a constant and specific level of illumination.
- ***New environment:*** Same properties as basic environment but with some changes, either to the background, to the present objects in the scene, the illumination or a combination of some of them.
- ***Advanced environment:*** This is a highly changeable environment, with an arbitrary background which might contain many moving objects. The lighting of the scene might be changing.
- ***Basic object:*** An object with a simple shape and clear uniform color for example a ball or a standard post-it note.
- ***Normal object:*** An everyday object that does not necessarily have a simple shape nor clear color. For example a paper coffee cup, a phone or a shoe.
- ***Advanced object:*** An object that can change its shape. The main object here is humans.

Example of environments can be found in the project plan under *Common scenarios and backgrounds*.

2.1 Object recognition

To learn to recognize objects, the Nao will first observe the background without the object and then the object will be placed in front of the Nao. The object should then be recognized when inside the Nao's vision field. When not finding a specific object in the Nao's vision field the Nao may search the near environment by walking round in some chosen pattern to try to find it.

Table 1: Object recognition requirements and corresponding priority, where 1 is highest priority. The requirements should be met in accordance with the description above. See section 2 for notation description.

Req.	Description	Priority
2.1.1	Recognize a *basic object* in a *basic environment*	1
2.1.2	Recognize a *basic object* in a *new environment*	1
2.1.3	Recognize a *basic object* in an *advanced environment*	2

2.1.4	Recognize a *normal object* in a *basic environment*	1
2.1.5	Recognize a *normal object* in a *new environment*	2
2.1.6	Recognize a *normal object* in an *advanced environment*	3
2.1.7	Recognize an *advanced object* in a *basic environment*	1
2.1.8	Recognize an *advanced object* in a *new environment*	2
2.1.9	Recognize an *advanced object* in an *advanced environment*	3
2.1.10	Go looking for non-found objects	1

2.2 Following objects

In order to follow objects the robot should be able to take into consideration that while moving, the background as well as the object will likely move and change shape/size.

The object identification algorithm should be fast enough so that the robot does not get stuck in heavy calculations and left without any new decisions. The estimation of the position of objects is of importance since the robot might collide with objects if the position is completely unknown. The task of following an object can be done without position estimation.

Table 2: Requirements for following objects, and corresponding priority, where 1 is highest priority. The requirements should be met in accordance with the description above.

Req.	Description	Priority
2.2.1	Be able to follow a known object	1
2.2.2	Estimate position of object in local coordinates using the cameras	2

2.3 Object association

The robot should be able to distinguish different classes of objects. Simple object classification can be any kind of classification, such as color or shape. The robot will have the feature to link objects to each other with some kind of relation, for example ownership of an object.

Table 3: Object association requirements and corresponding priority, where 1 is highest priority. The requirements should be met in accordance with the description above.

Req.	Description	Priority
2.3.1	Simple object classification	1
2.3.2	Distinguish humans from objects	2
2.3.3	Link objects to each other	2

3 Documentation

Apart from this document, the following documentation will be produced during the project:
- **Project plan:** This document gives more in depth description of the project and the first version of the project backlog.
- **User manual**: Documentation is a guide to help the user how to install and use the product.
- **Scrum review document:** After every sprint review, a description of the result and the backlog will be documented.
- **Technical documentation:** This is the final report, which will include the software developed as well as a more in depth description of the features available. It will also provide some insight to how the project group went about to fulfil the requirements. Also, some possible technical improvements that might be implemented later on will be discussed.
- **Reflection document:** this contains the scrum retrospective documents, which will be produced after each sprint.

3.1 Documentation requirements

Table 4: Documentation requirements and corresponding priority, where 1 is highest priority. The documents should be made in accordance with the descriptions in section 3.

Req.	Description	Priority
3.1.1	Deliver the project plan	1
3.1.2	Deliver the written technical documentation	1
3.1.3	Deliver the user manual of the product	1
3.1.4	Deliver the reflection document	1

4 Delivery

The project will be delivered in two steps: a mid-term checkpoint and a final delivery. The mid-term delivery should include a demo of the current, not yet finished, product so the customer can see the progress of the project.

The finished product will be delivered in form of a demo, showing that at least the requirements specified with priority 1 have been achieved. The product will be delivered along with the documentation mentioned in section 3 as well as a poster showing a summary of the project and a website describing the project.

4.1 Delivery requirements

Table 5: Delivery requirements and corresponding priority, where 1 is highest priority.

Req.	Description	Priority
4.1.1	Create a website	1
4.1.2	Create a poster	1
4.1.3	Presentation	1
4.1.4	Final product	1

4.2 Delivery dates

The delivery dates are presented in the table below.

Table 6: Deadline dates for deliveries to be made.

Document	Deadline date
Requirement specification	2015-09-21
Project plan	2015-09-21
Midterm	2015-10-21
User manual	2015-12-12
Draft of technical documentation	2015-12-12
Final product	2015-12-12

Poster	2015-12-21
Presentation	2015-12-21
Website	2015-12-21
Reflection document	2015-12-21
Technical documentation	2015-12-21