SOFTWARE REQUIREMENTS SPECIFICATION

for

AARDVARK

Version 1.0 approved

Prepared by Hannah DeFazio, Nikolas Lamb, Ben Myrick, Jenna Ryan

Clarkson University

October 30, 2018

Contents

Revision History

Name	Date	Reason For Changes	Version
H DeFazio, N Lamb, B Myrick, J Ryan	9/27	Initial Revision	1.0

1 Introduction

1.1 Purpose

This document outlines the software requirements specification for the Automated Assistive Robot Doing Visual Analysis Running Kinects (AARDVARK). This document follows the IEEE standard for software requirements specification documents.

The purpose of this project is to produce an assistive robot with the motivation of helping elderly individuals. In a broad sense, the robot will have the capability of monitoring the user's movements and raising a support arm if it detects that the user is about to fall. This document will cover each of the system's intended features including all hardware, software, and other technical dependencies.

1.2 Document Conventions

VOIP: Voice over Internet Protocol
GUI: Graphical User Interface

1.3 Intended Audience and Reading Suggestions

This document is intended for several different groups of individuals. One such group is the developers who can review the project's requirements and determine the next steps for future development. Project testers can use this document as a template for finding bugs. This will make the process of testing more methodical and organized. Lastly, end users who want to learn in more depth how the project works and what the robot is capable of can read this document for further insight.

Readers who are interested in the technical features and aspects of the assitive robot should refer to parts 3 and 4. On the other hand, those who are interested in reading further into the non-technical aspects should read part 5, which includes information regarding safety, security, and other important aspects about the project.

1.4 Project Scope

The assistive robot will have the ability to not only detect that an individual is about to fall but also will extend a support arm and successfully catch the user. In the case that the robot fails to catch the user, backup protocols will be in place in order to ensure the safety of the user. The robot to be able to follow the user with numerous obstructions in its path.

1.5 References

The official website contains a brief description of the projects along with mock up drawings and case diagrams (INSERT WEBSITE LINK)

2 Overall Description

2.1 Product Perspective

This project is intended to act as a replacement for walkers, canes, and other walking tools. With AARDVARK, the user will gain more autonomy in their life. The robot does not need to be dragged alongside the user, thus freeing the user's hands and mind from using this tool.

2.2 Product Functions

The robot will follow the user and monitor their balance using a camera. When it is determined that the user is going to fall, the robot approaches the user and raises an arm in order to catch and balance the user. The robot stays like this until the user is stable enough to not require assistance. Once the user returns to a balanced state, the robot lowers the arm and continues to follow the user. The robot offers verbal and visual communications through a touch screen display. This display can display lists, such as grocery lists, communicate to emergency services if needed, and offer a visual face to the robot.

2.3 User Classes and Characteristics

The main user class is elderly individuals who are at risk of falling and seriously injuring themselves. This user class is also determined by physical characteristics; these individuals must be between 60 and 78 inches in height and weigh no more than 250 pounds.

Aside from the main user class, a secondary user class are the responders who come when they are contacted by the robot. Although responders aren't required to interact with the robot in the same sense that the main user does, they are still required to disable the emergency state of the robot.

2.4 Operating Environment

The robot shall only operate indoors and on a straight, fixed, flat surface. The robot shall not traverse stairs or any other hazardous conditions (extreme cold/heat, slippery floors, etc). Also, there will be no other obstructions while the robot is following the user.

2.5 Design and Implementation Constraints

Robot will provide a system by which developers are able to add their own modules. These modules may access data collected by the robot. Modules must be written in Python or C#, depending on the hardware that the module must access.

2.5.1 Robot Operational Noise

The robot should not cause irritating levels of noise as it moves, defined as a volume that would impede conversation or distract from tasks. In the process of designing the hardware of the robot, the movement of parts and running of the motors should work toward minimal noise, with a maximum volume of 50 decibel levels.

2.5.2 Overall Size of Robot

In order to not be an obstruction and for ease of robot navigation, the robot should be the size of a small shopping cart, about 40 inches tall, 30 inches deep, and 20 inches wide, and about 600 square inches in a square or rectangular shape for the horizontal cross section.

2.5.3 Robot Stability

The robot must be designed to remain standing if bumped into by an averaged sized human adult while walking at a leisurely pace of no more than 1.5 meters per second. It shall have four wheels arranged in a square at the corners of the base to provide stability.

2.5.4 Limitations on Robot Movement

The robot should be designed to only move due to commands specified by fatigue-assessment software. When not moving, it should be unable to be pushed by the user or used as a walker.

2.5.5 Power Management

For simplicity reasons, the robot shall be turned on at all times. In addition to this, the robot should have the capability of easily swapping batteries without shutting down the robot in the process. If the robot needs to change a battery, it shall notify the person that a change is needed.

2.6 User Documentation

Robot shall be packaged with a standard manual of parts and functions. Robot shall also come with tutorial for setup and common use cases.

2.7 Assumptions and Dependencies

The Kinect sensor that our system utilizes has ceased to be manufactured. With this in mind, in the future we will move to OpenPose to provide video and skeleton data for the fatigue-assessment software.

3 External Interface Requirements

3.1 User Interfaces

3.1.1 Assistive Arm

The robot arm which raises is designed for the user to grab onto, similar to a railing. To ensure a comfortable grip and soften the impact of sudden grabbing, the arm will be cushioned with soft material which will absorb some of the impact but remain firm enough to easily grasp for support.

3.1.2 Touch Screen Display

The robot will be equipped with a 10" touch screen display on top which will allow the user to sign in. It will be used to store lists, such as to-do, grocery, etc. The screen will offer start and reset buttons. The start button will tell the robot to begin following the user. The reset button will be pressed following the user receiving outside assistance after falling and the robot can proceed with its normal functions.

3.2 Hardware Interfaces

The robot utilizes a Raspberry Pi 3 in order to control the motors. A Pi Hat is used to control the stepper motor that moves the arm and the motors that move the wheels. A Python script talks to the motors and tells them when to turn on.

3.3 Software Interfaces

The computer of the robot will run Windows 10. The video recorder will run C# code. The motors will run Python code. The Python and C# code will communicate over a sever-client model. The Python code will use NumPy and Scikit-learn as dependency libraries. The C# code uses the Kinect SDK as a dependency library.

3.4 Communications Interfaces

In the event that the user has fallen and does not appear to be responding to the robot's attempts to contact them, the robot will use a VOIP protocol to contact the emergency contact on record. It will also broadcast a prerecorded message via speakers to the surrounding room.

4 System Features

Robot features are divided into three main operating states that are either reactive to user condition or proactive to maintain operation. Each operating state performs a vital function. The first state is default, that is other states will be enacted from this state and after they are completed the robot will return to the default state. Additional non-essential states may be added at a later date, as the system is designed to accommodate other functionalities depending on use case.

4.1 System Feature 1 - Passive Monitoring

4.1.1 Description and Priority

System procedure that mandates the robot stay near to the person at all times. The robot stays within approximately three strides of the person at all times. It is designed to accommodate the user as much as possible while ensuring their safety. This is the default robot state, i.e. from here the robot will move to all other states. This is a high priority state.

4.1.2 Stimulus/Response Sequences

The user should be going about normal activities while the robot stands by. In this state robot should remain passive unless user requires it. Robot will watch their actions and evaluate if they seem to be changing activities. Robot can also be called by user if they anticipate changing activities. If they appear to be changing activities, or they call the robot, robot will switch into active monitoring state. If user appears to be slipping, robot will move to intervention state.

4.1.3 Functional Requirements

REQ-1.1: Robot shall stay three (3) strides (6-8 feet) from the person during passive monitoring.
REQ-1.2: Robot shall change its position only if person moves more than three (3) strides from the robot (or if state changes).
REQ-2.1: Robot shall move into active monitoring state if called by user.
REQ-2.2: Robot shall move into active monitoring state if person appears to be changing activity.
REQ-3: Robot shall move into intervention state if person appears to be falling.

4.2 System Feature 2 - Active Monitoring

4.2.1 Description and Priority

System procedure that mandates the robot stay near to the person at all times. It is designed to move alongside the user while accommodating them as much as possible. This state can be reached from passive monitoring state or intervening state. This state may call passive monitoring state or intervening state. This is a high priority state.

4.2.2 Stimulus/Response Sequences

This state is active when the user is moving or walking. In this state robot should remain close to the user unless user requires an intervention or the person begins a stationary activity. Robot will watch the person's actions and evaluate if they seem to be falling or starting a stationary activity. If they appear to be starting a passive activity robot will switch into passive monitoring state. If user appears to be slipping, robot will move to intervention state.

4.2.3 Functional Requirements

REQ-1.1: Robot shall stay three (3) strides (6-8 feet) from the person during active monitoring.
REQ-1.2: Robot shall remain this distance directly in front of person.
REQ-1.2: Robot shall keep pace with the person while they are walking, moving not more than 8mph.
REQ-2.1: Robot shall move into passive monitoring state if user begins stationary activity.
REQ-3: Robot shall move into intervention state if person appears to be falling.

4.3 System Feature 3 - Intervention

4.3.1 Description and Priority

System procedure that activates when a person appears to be falling. This state can be reached from passive monitoring state or active monitoring state. This state may call passive monitoring state, active monitoring state, or emergency response state. This is a high priority state.

4.3.2 Stimulus/Response Sequences

This state is active when the user appears to be falling. In this state, the robot will quickly move to provide support to the person. As it reaches the person, the robot will raise an arm to provide support. If the fall is interrupted and the person appears to have recovered, the robot will move to active or passive monitoring. If fall is not interrupted the robot will move to emergency response state.

4.3.3 Functional Requirements

REQ-1.1: Robot shall move towards the person.
REQ-1.2: Robot shall stop when it reaches 1.5 ft from the person.
TBD: Robot move speed shall not exceed ¡¿mph
REQ-2: Robot shall raise arm to provide support
REQ-3.1: If fall is interrupted, robot shall not move until person has released support arm. Robot shall then return to previous state.
REQ-3.2: If fall is not interrupted, robot shall not move unless prompted by user. Robot shall enter emergency response state.

4.4 System Feature 4 - Emergency Response

4.4.1 Description and Priority

System procedure that activates after a person has fallen and their fall has not been interrupted. This state can be reached from intervention state. This state may call passive monitoring state or active monitoring state. This is a high priority state.

4.4.2 Stimulus/Response Sequences

This state is active when the user has fallen. In this state, the robot will attempt to analyze the user's condition. If contact can be made with the user, robot will call for assistance if user appears to be injured, or revert to active or passive monitoring if user is physically sound. If contact cannot be made with the user, robot will call for assistance. Robot may issue an audio message to the room and also may contact emergency responders or emergency contacts. Additionally if contact cannot be made with the user the robot will record their condition via video.

4.4.3 Functional Requirements

REQ-1: Robot shall not move unless prompted by user.
REQ-2: Robot shall attempt to contact the user by playing the message: "Are you okay?".
REQ-3.1: If the user responds and indicates they are physically sound, robot shall move to active or passive monitoring.
REQ-3.2: If the user responds and indicates they are not physically sound, robot shall play the message: "Assistance required.". Simultaneously the robot shall contact the stored emergency contact via video call and stream video of the user. Simultaneously the robot shall record video of the user.
REQ-3.3: If the user does not respond, robot shall play the message: "Assistance required.". Simultaneously the robot shall contact the stored emergency contact via video call and stream video of the user. Simultaneously the robot shall record video of the user.

REQ-3.3: Unless otherwise specified, robot will not exit this state unless manually reset by emergency contact.

4.5 System Feature 5 - Power Management

4.5.1 Description and Priority

System procedure that activates when the robot is about to run out of power. This state can be reached from any monitoring state. This is a high priority state.

4.5.2 Stimulus/Response Sequences

This state is active when the robot is about to run out of battery. When the battery is getting low, the robot will verbally alert the user of the fact. If the user responds, the robot will move to a charging station to change batteries. If the user does not respond, the robot will revert to it's previous state and continue to alert the user every ten minutes, then every minute when the battery level becomes critical.

4.5.3 Functional Requirements

REQ-1: When in monitoring mode, robot shall alert the user it is low on battery when the battery has enough power for 30, 20, or 10, 5, 4, 3, 2, 1 minutes of operation by playing the message: "Low battery." The robot shall then revert to its previous mode.
REQ-2: If the robot is allowed to recharge, it shall move to a predefined charging station and swap batteries from the station.
REQ-3: The robot shall then return to its previous position and revert to its previous mode.

4.6 System Feature 6 - Initial Setup

4.6.1 Description and Priority

When the robot arrives to the user, the technician will take the user through the below steps to setup the robot. This is a high priority state.

4.6.2 Stimulus/Response Sequences

This state is active when the robot arrives and it turned on for the first time. The user will be asked to enter an emergency contact. The user will be asked to enter some kind of unique identification. The user will be asked to identify a static charging station. The robot will then enter passive monitoring state.

4.6.3 Functional Requirements

REQ-1: The robot shall first prompt the user to enter an emergency contact.
REQ-2: The robot shall then prompt the user to enter some kind of unique identifier.
REQ-3: The robot shall then prompt the user to move it to a charging station.
REQ-4: Finally, the robot shall enter passive monitoring mode.

5 Other Nonfunctional Requirements

5.1 Performance Requirements

When a fall is perceived to be occurring, the robot must reach the specified distance (0.5m) from user and raise its arm in less than 0.5 seconds. Additionally when this operation occurs, the robot must not end up any closer than 0.5m from the user.

Robot must be prepared to assist user at all times, for an indefinite amount of time, except when changing batteries. Though normal wear and tear is expected, the robot shall notify the user if it detects any parts may be degrading, and must continue operation until it is physically impossible.

5.2 Safety Requirements

Robot will have an emergency contact programmed in as part of setup process which will be the resource called when the robot identifies a need for outside assistance.

No part of the robot should pose danger in a collision or during the intervention process. To accomplish this, the robot should have no moving parts enclosed and be free from sharp corners and edges. It will be designed using principles of soft robotics, ensuring that contact between the robot and the user will not pose a risk of injury. Touch screen display should be located in a position away from where the person would grab for support and should be made of durable glass which does not pose a risk of shattering.

5.3 Security Requirements

User profiles will be specified using an off-the-shelf fingerprint reader. Given the limited scope the robot is intended for, data will not be encrypted, and additional security measures are intended more to ensure the robot imprints to the right person, instead of ensuring against hostile entities.

When sending video to the emergency contact, the feed must be accessible only to the designated location and should not be able to be streamed to any other person. In addition, the only time the video should be able to be seen is when the robot reaches out to the contact; the video can be made available to the emergency contact, but that person or any other entity cannot independently contact the robot for video.

6 Other Requirements

6.1 Appendix A: Analysis Models

State Transition Diagram:

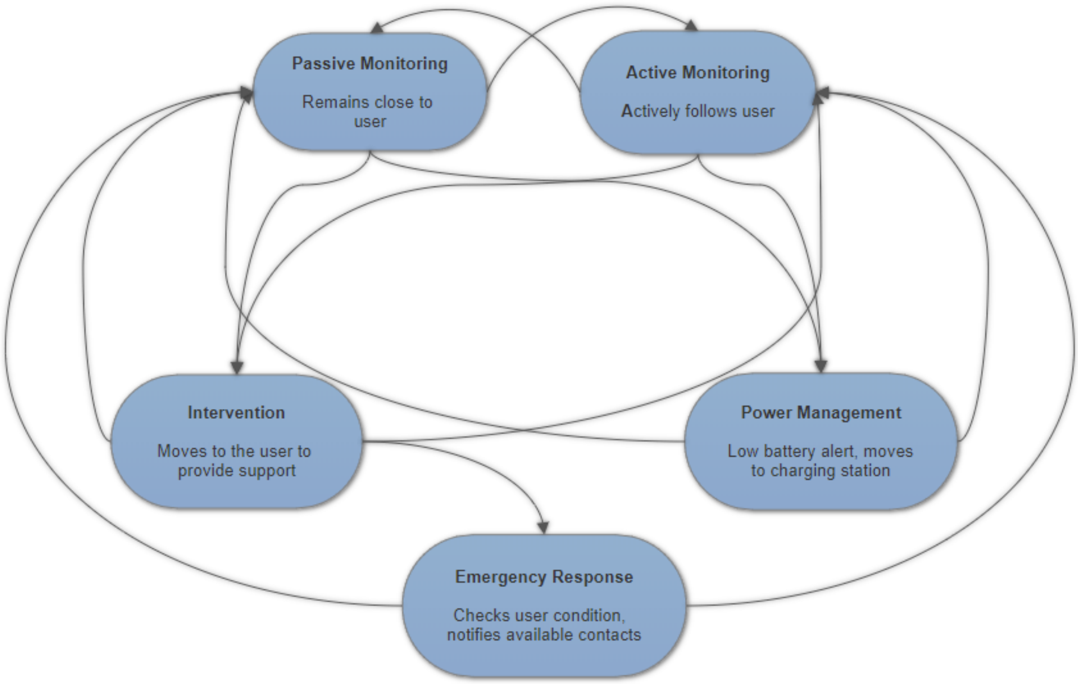

Figure 6.1: State Transition Diagram. Shows interactions between possible protocol states (excluding initial setup state which is a single-occurrence state)