Tasks

## Assembling

FALL

* Assembling Link 1-3 11/15
* Assembling link 4-6 11/23
* 3D Print and set Camera mount on Knuckles 1/21
* 3D print End Caps for wiring 2/21
* Design and connect Knuckles to a plywood base : 2/6
* Group all wires and organize them in a labeled cables: 2/15
* Reroute wiring: 2/15

SPRING

* 3D print case for microcontroller 2/28
* 3D case for power distribution and stepper drivers: 2/28
* Laser cut plate for forearm: 2/15
* Reprint wrist in Red 2/17
* Print spare pulleys. 2/3
* Figure out how to mount end caps: 2/21
* Add Tactile pads on the gripper 2/28

## Motor and Power distribution

FALL

11/ 20

* Attaching motors to Link 1-3

11/27

* Attaching motors to link 4-6

11/30

* Connecting the stepper drivers and stepper controller (Arduino and Ramp 1.6)

11/30

* Test all 6 motors using different micro steps

12/4

* Test first prototype of fully built robot

SPRING

2/5

## Simulation

## FALL

## 11/15

## Robot represented in the simulation on RViz

1/5

* Associate virtual links to the actual physical motor on the robot

12/17

* Detect different April Tags including object names on Rviz

SPRING

## AprilTag Detection

FALL

2/17

* Camera detects AprilTag and labels object names on ROS

2/17

* Robot arm needs to move to keep AprilTag in center of the image. The arm can follow the tag.

SPRING

1/19 Rym and Matt

* Modify the micro step of each motors to have a smoother motions and reduce the current through the motors using switch dips and avoid motor overheating.

2/5

* When the object name is typed into the terminal as a command, the code compares that command to the array cell where the object name is saved, and then loads that pose for the robot. Thus the robot will have the apriltag in the center of the image.

2/8

* The robot arm approaches the object until at a certain z value, and then grabs the object

2/12

* The robot returns to a default position to give an object.

2/15

* Robot completes a full rotation to analyze its environment, and saves AprilTag locations during this rotation. Then, robot arm points at requested objects from the terminal, when the object name is typed in

2/18

* When the object is pulled on, the sensor value from the touch sensors is affected, so the gripper lets go. The arm waits until it returns to its default position to check this, or else it will drop the object prematurely.

## Voice Recognition

SPRING

2/5

* Connect the Speech Enable Development Kit to the Raspberry Pi3+. Create an Amazon Development Account to sync it to the Raspberry Pi 3+.

2/18

* Create a node in ROS to read voice-to-text from Alexa. Have a voice-to-text message on a ROS Shell.

2/26

* Create two publisher from the ROS-Alexa node one to the GUI and the other one the AprilTag node and the motor motion node. Create a link between action verbs (grab, take, hand me..." to motion; /jointsteps.

## GUI - PAOLA

SPRING

2/6

* Becker advised us to create a GUI in order to have a smooth transition towards voice recognition. The text recognition from the text command for the GUI will be later used on voice recognition. A completed designed graphical user interface (GUI). With 8 object and actions commands, Knuckles will be able to execute a total of 64 phrases. The GUI provides the user a scroll down list with each object and action commands. Also, have the fully designed GUI working properly on the Ubuntu command prompt.

February 2/8

* Create a node using rqtgraph for the GUI on Rviz, work on simulation process using GUI.

2/15

* Working alongside Matthew, have a fully operational GUI on Rviz, having Knuckles recognize the text of the action and the objects given from the list on the GUI

2/22

* Work alongside Rym, combine fully operational GUI from Rviz with microphone

2/28

* Fully operational voice recognition with a fully operational GUI