Research

- Balance
  - balance_control_module.cpp
  - Lot's of applied linear algebra needed
  - A lot to do with maybe not enough time.
  - Seems like a very difficult task. Large knowledge space that we have no experience in.
  - Nick says not doable
  - Brady says more doable, but hard
- Replay
  - Pose Mimic sends camera image to Open Pose, receives joint states to publish.
  - Would create data structure to hold joint states, so they can be played sequentially.
  - API approach would be ideal. Save replays as actions to action file.
  - If we can use the existing pose estimation module, shouldn't take super long to do this.
  - We should talk to Mourad about what "replication" of the pose estimation module means.
  - Open Pose seems to need an nVidia GPU.
- Defaults and Poses
  - Actions defined in motion_4095.bin
  - Initial pose is defined in ini_pose.yaml
  - Pose can be loaded from yaml and published whenever we want to go to init.
  - We could write a function that intercepts the action call, calling the init pose if needed depending on the action.
  - Yaml file contains target pose. Maybe we can programatically define the starting position?

Relevant Files

- Actions: ~/catkin_ws/src/ROBOTIS-OP3/op3_action_module/data/motion_4095.bin
- Action module to play actions: ~/catkin_ws/src/ROBOTIS-OP3/op3_action_module/src/action_module.cpp
- op3 base module: ~/catkin_ws/src/ROBOTIS-OP3/op3_base_module/src/base_module.cpp
  - This file seems to handle the dispatch of action names/pages to action module
- op3 manager: ~/catkin_ws/src/ROBOTIS-OP3/op3_manager/op3_manager.cpp