**Mtech NRMC Preliminary Task List**

* Mission Critical Tasks
  1. Implementation of the Remote Control System (aka the client) for manual control (Headed by Mack Sutherland)
  2. Implementation of a hardware interface for communication with equipment over USB (Headed by Abhimanyu Nath)
  3. Network Interface for communication with the Server (aka Robot) over a wireless network using UDP/IP (Headed by Josh Lee)
     + Updating communication protocol documentation
     + Creating the network interface for sending data over IP
       - Implementation in Java, C#, and Python are available
       - Needs to be implemented in C++
     + Creating the network-application interface
  4. Implementing the Manual Control System (MCS)
* Victory Critical Tasks
  1. Adding or enabling functionality in the Remote Control System for control of the Automatic Control System (ACS) (Headed by Mack Sutherland)
  2. First phase implementation of the Automatic Control System (ACS)
     + Pathfinding System (Headed by Josh Lee)
     + Localization System (Headed by Abhimanyu Nath)
     + Interfacing with the Hardware Interface
     + Interfacing with the General Hardware Query System (GHQS)
  3. Partial implementation of the General Hardware Query System (GHQS) for “Mission Critical” hardware
     + Examples: LIDAR, Pixies (electro optical sensors), Range Finders, other equipment for localization
* Auxiliary Tasks
  1. Implementation of Bryce’s wish list for the Remote Control System (Headed by Mack Sutherland)
  2. Full implementation of the General Hardware Query System (GHQS)
  3. Phase 2 of the Automatic Control System (ACS)
     + Implementation of an AI to optimize excavation routs
  4. Implementing the Augmented Reality Diagnostic System (ARDS)
     + A mobile app that can link into the robot and display real time data about the robot
  5. Other possible long shot ideas?
* Miscellaneous Tasks
  1. Website development
     + Last checked this counts to the social media and outreach score
     + If we get PTC students they would be responsible for front end dev
     + Our group would be responsible for back end dev
  2. Configuring the Atom
     + Switching to a Debian Headless Server or keep current Ubuntu Headless Server.
     + Configuring the server to start at System Boot.
       - Only achieved last year through Cron Jobs, but Cron Jobs start processes to slowly (ETA last year was 5mins)