High Level Design

# Computer Application Component - DataLift

This component is used to collect and send data the

## Purpose

* Connects to two kinects
* Login
* Specify start and stop recording
* Identify the shortened version of the lift
  + Prompt the user to save it
  + Show the preview
    - Potential to allow them to grow it in either direction
  + Potential: tag before save
* Record and tag after on Bar None
* Serialize playback Kinect data
* Serialize video
* Stream the data to The Rack
* Delete temporarily created data on the PC with DataLift
* Dump video separate from all other info

# Website User Component – Bar None

## Purpose

* Lifts have to be organized
  + Folder structuring
  + Based on lift type, time, etc
* Add users to your lift database such that they can view the data
  + Read only users can be added/removed to your datasets
* New user option
* Login
* Can get raw video of each lift
* Skeleton video data of each lift
  + Both profiles
* Bar tracking
* Video play back
* Temp lift, if not tagged then delete the data after DAYS\_TILL\_EXPIRATION
  + Users can add or modify tags from a static amount of tags
* Dynamic display graph
  + This is to show accelerations
    - Bar speed/accel
    - Muscle/join stress
* Data synching between video playback
* Record deletion
* Playback is play until common point of lift
  + Play until sync point when comparing lifts
  + Move forward from this point to the next
  + Go to a specific point
    - Feet leave ground…
  + User given specific points
    - i.e. Highest calf stress, min knee angle, max bar height, bar accel, etc…
  + Pause and inspect
* Search lifts
* Display of the data
* User information
  + Profile: Age, weight, height, gender (body movements may differ)
    - Optional medical conds
    - Password change
    - Pass recovery email
* FAQ area
  + Help pages
* Lift comparisons
  + Compare points (this would be the static components or elements like bar data)

# Website Server Component – The Rack

* Perma-record all data
  + Profiles
  + Lift videos
  + Lift data
  + Lift tags
* Composable Rest API to fetch a large variety of data and update screen all at once
* Connected to Data recorder (DataLift) and the user component (Bar None)

Milestones

# Milestone 1 – November 20th

* Project Goals – Oct 6th
* Draft Sys Requirements – Nov 6th
* Reading Kinect Data
* Viewing Skeleton Data
* DataLift connecting to The Rack
  + DataLift needs to Insert
  + DataLift Login
* BarNone connects to The Rack
  + The Rack Needs to Query
* MVVM Architecture for DataLift
* Bar None Login & Add User
* UI – Bar None
  + Tree support for folder
    - Customizable tree components
  + Mock of dummy data for UI testing
  + Set up unit tests for the Unit tests
  + Video seeker
    - Only view Skeleton data for now
    - Skeleton rendering
  + Saving
* Dummy User
* Print tripod mounts
  + https://www.thingiverse.com/thing:5004
* EVERYTHING NEEDS UNIT TESTS

# Milestone 2

# Version 1

* Requires Clean and Jerk & Snatch

# Version 2

* Custom tags (One handed, poles, etc)