Kinisi

The Apple Watch Form Correction Coach

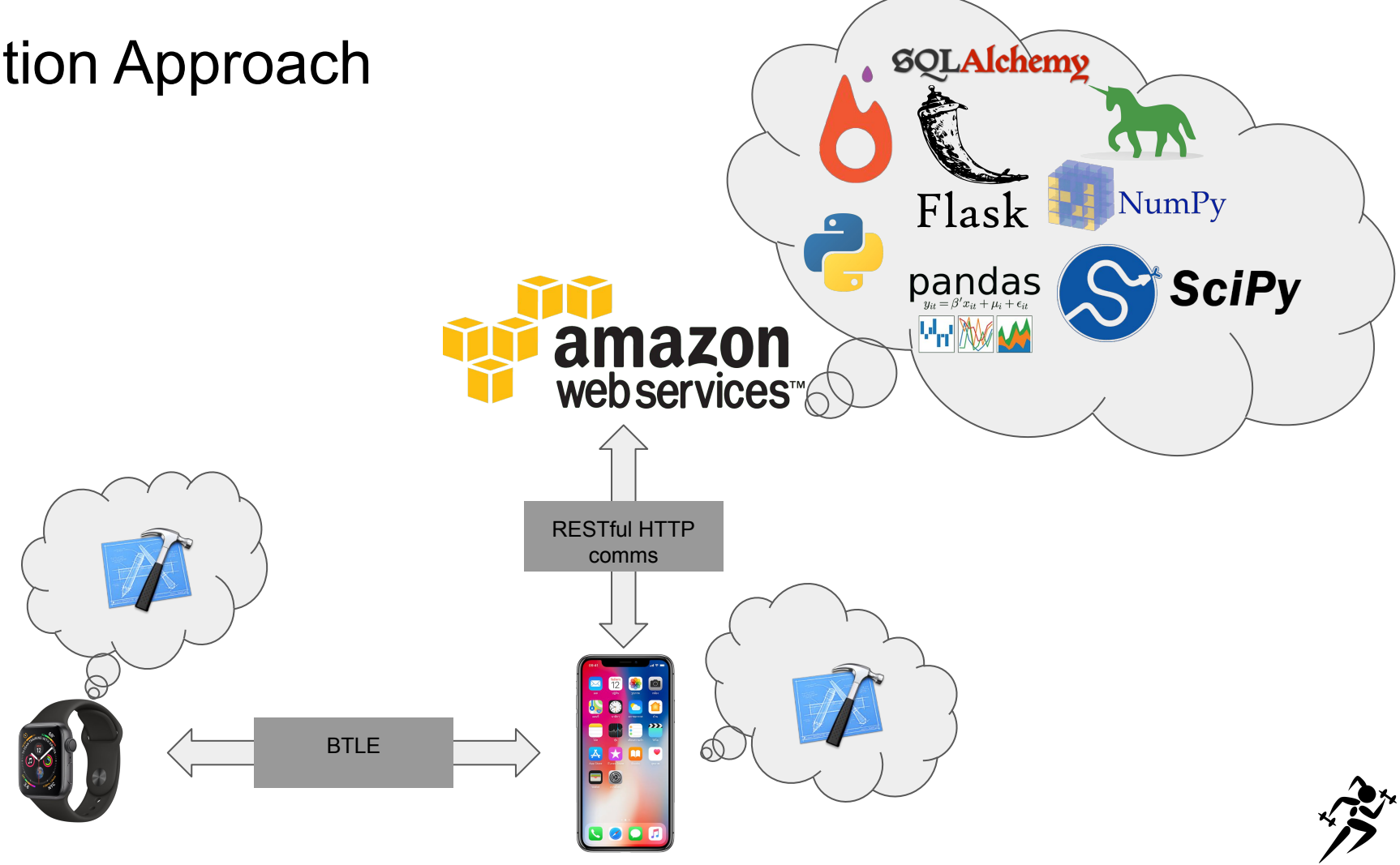
Team B3

Adrian Markelov and Kyle Jannak-Huang and Matt Spettel

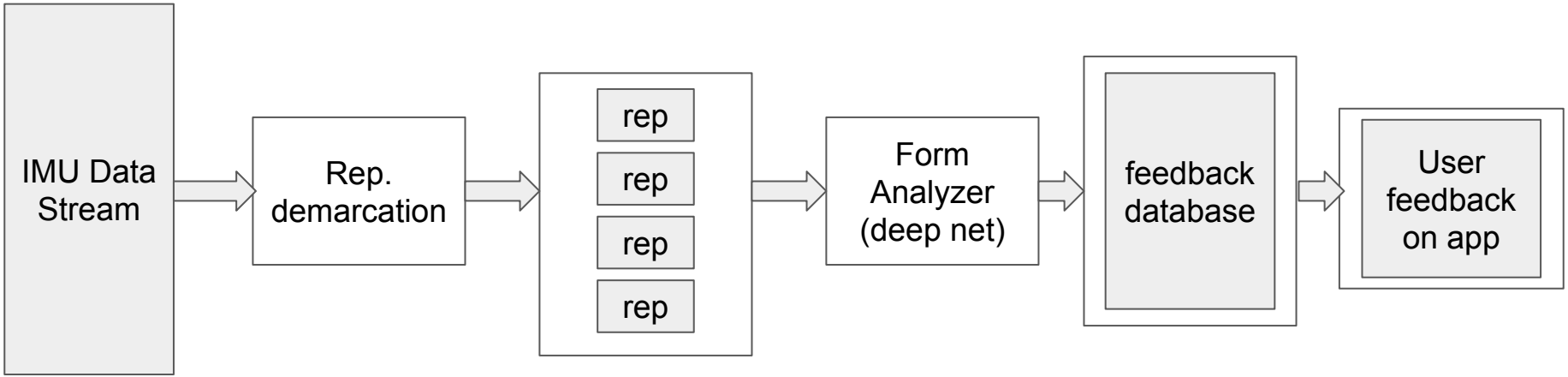
Use Case: Workout Form Correction Coach

- Problem Area:
  - Fitness, consumer tech, coaching, predictive and analytic tech
- Problem:
  - Personal training is prohibitively expensive ($60+ / hr).
- Kinisi:
  - Analyze a user’s form in live time
  - Diagnose issues they may have
  - Provide visual and instructional feedback to issue

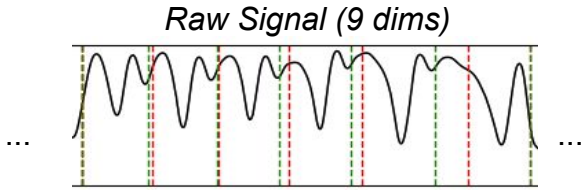
Solution Approach



System Specification: High Level Overview

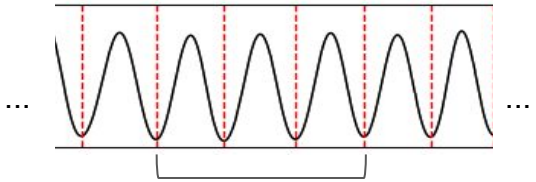


System Specification: Rep Demarcation

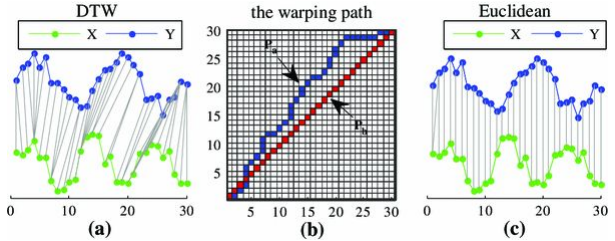


Heavy Gaussian Blur and Inversion

Simple Peak Detection



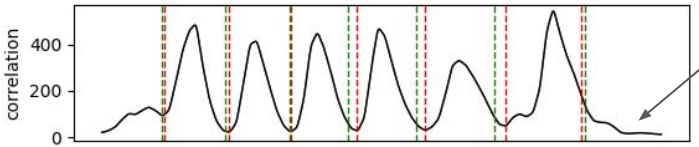
Extract middle rep



Sliding Dynamic Time Warping to Create Continuous Correlation Function

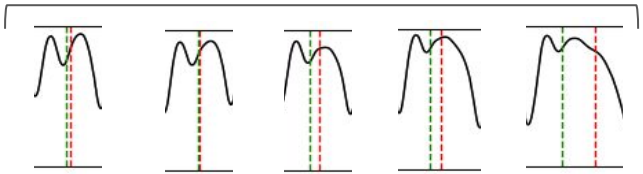
Simple Peak Detection

Place demarcations between peaks



Correlation function provides clean ends

Slice raw signal with demarcations

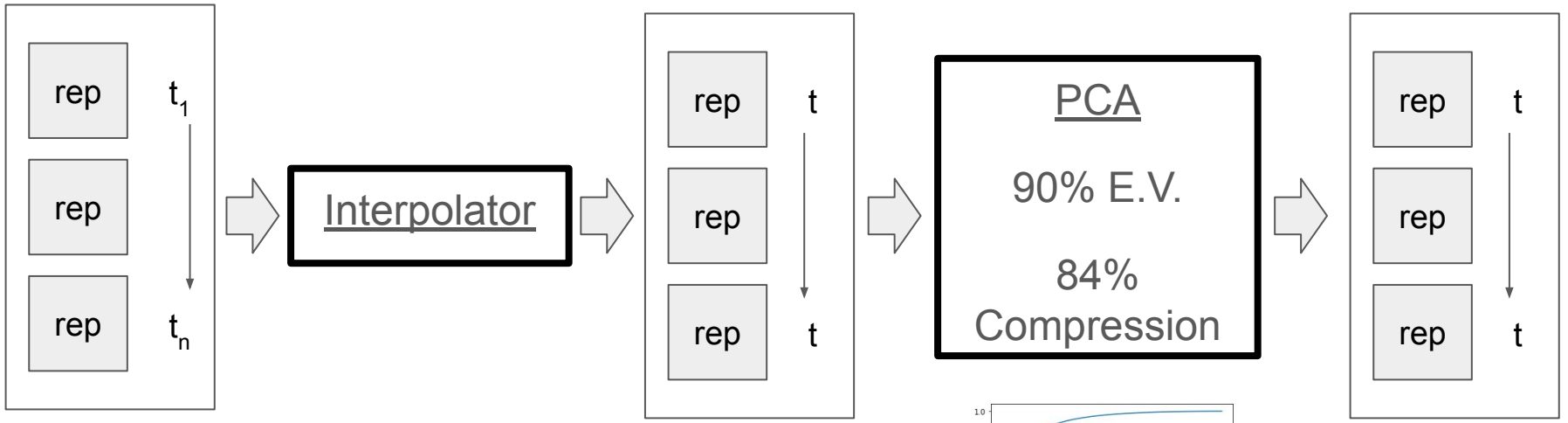


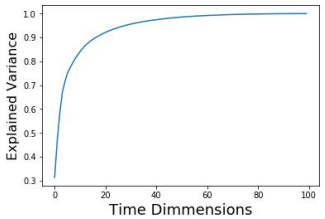
Solution Specification: Form Detection Preprocessing

$t_i \in$ {(random time len samples)*9DOF}
Dimensions = ???

t∈ {(100 time samples)*9DOF}
Dimensions = 900

t∈ {(16 time samples)*9DOF}
Dimensions = 144






```
In [17]:  1  np.cumsum(pca.explained_variance_ratio_)[16]
Out[17]: 0.8987758023556448
```

Apple Watch Sampler: 50hz

Avg Rep Len: 2 Sec = 100 samples

Scale: [First Sample → Last Sample] = [Start of Rep → End Rep]

System Specification: Form Classification

Method: Dilated Conv *(Yazdanbakhsh and Dick)*

Reasoning: Globally Biased Feature Extraction *(Yu and Kulton)*

Extracted features are similar to original time series. ex:

- Sharp edges (grads)
- Positional outliers

Decision: CNN's vs RNN's

RNN: Weight time important

CNN: Easier test and neutral to time

Important feature: position independent to time

Fast and slow reps are okay position only matters

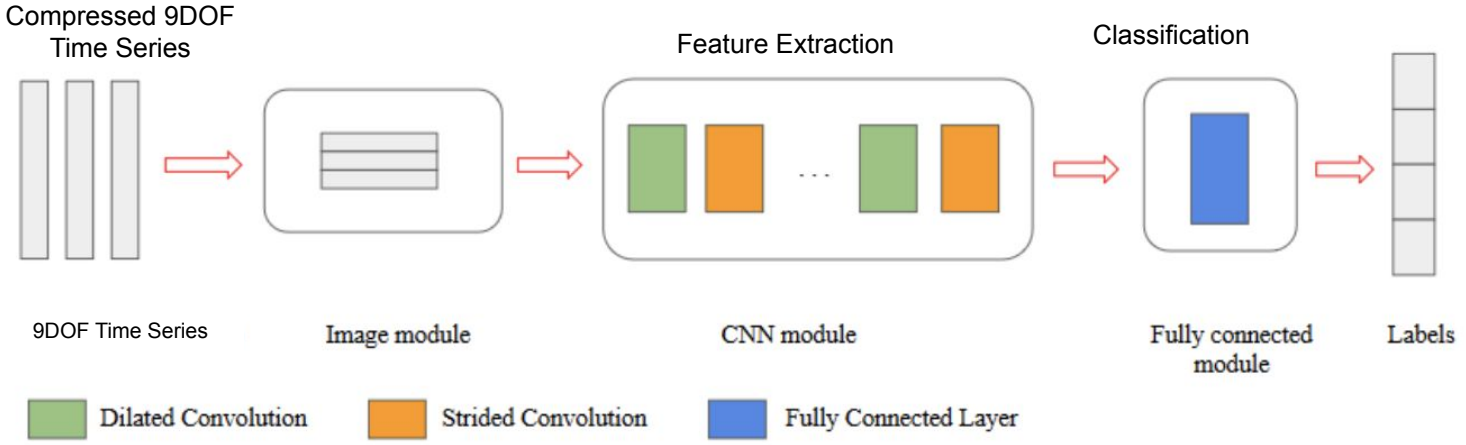
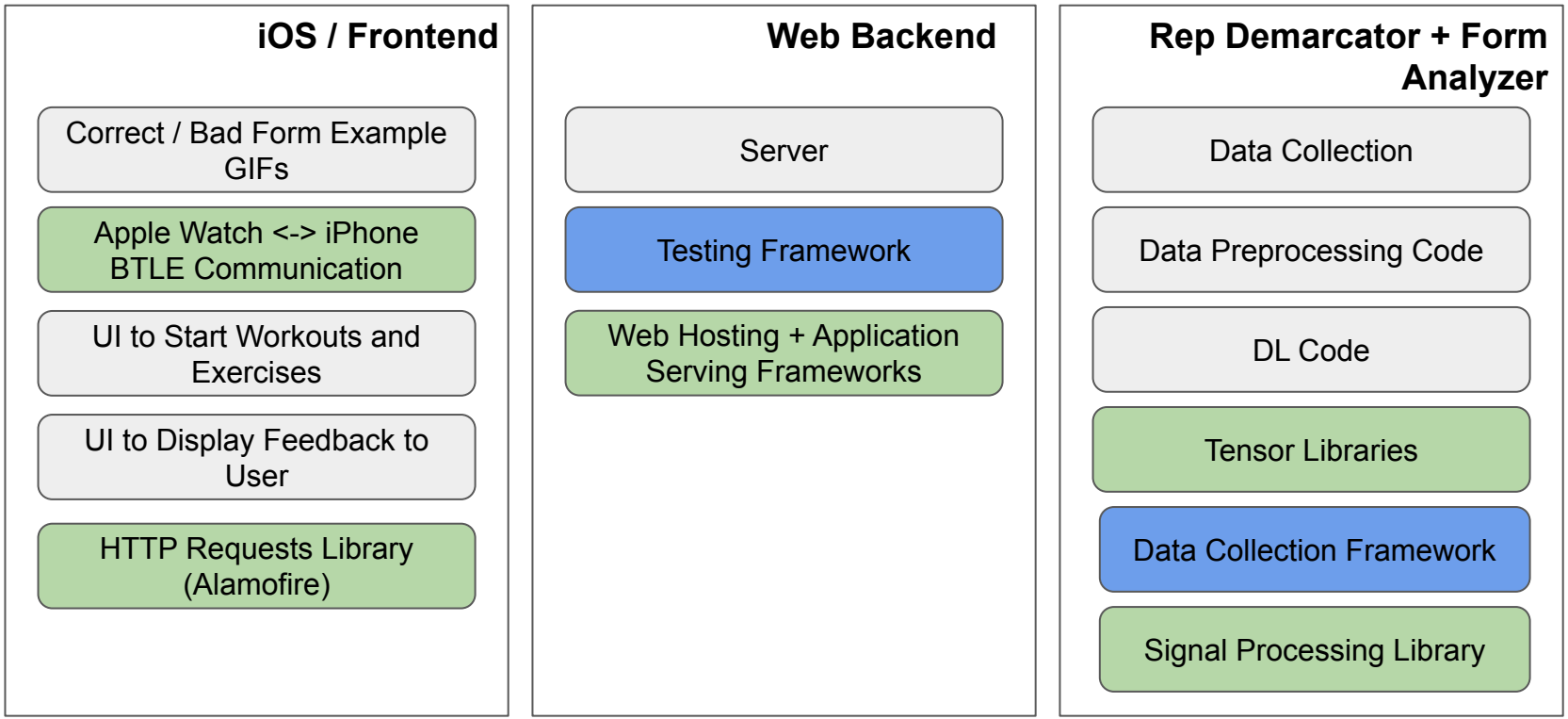


Figure 1. Block diagram of the proposed approach

Implementation Plan



Metrics and Validation

Requirement: Form analyzer must be accurate

- Each exercise typically contains 3 sets of 5-12 reps
- False Positives < 33% on sets (only 1 set per exercise can be messed up to avoid having the user lose confidence in the system)
- False Negatives < 80% on reps (1/5 reps must be flagged)
    - < 1% on sets (this is dangerous to the user)

Validation: Testing error of our form analyzer + field testing

- Test data will be collected from people who did not contribute to training data
- Test data will be at least 20% of the overall data
- The app will also be tested in the field on brand new data

Mitigating Risk: User can view correct form GIF regardless of classification

Metrics and Validation

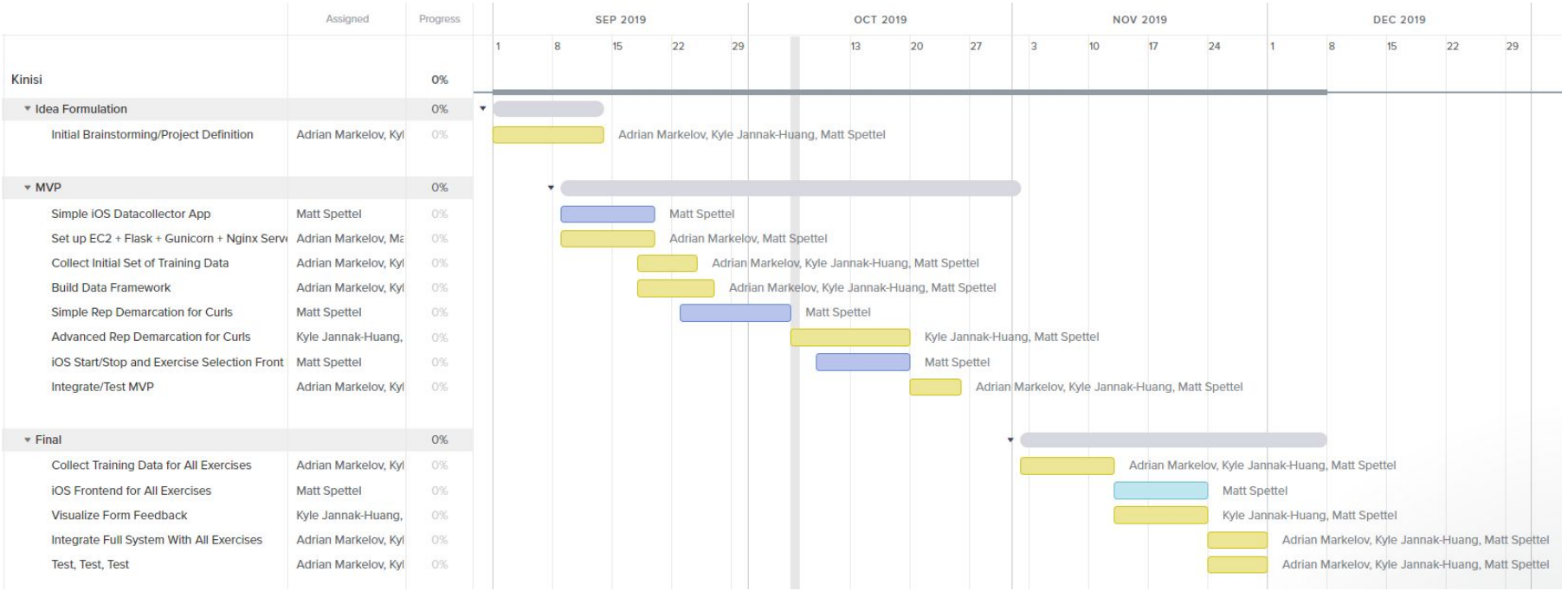
Requirement: Feedback must be available to user within 40 seconds after their set

- 60 seconds of rest time is the typical shortest rest time
- Allow the user 20 seconds of time to view visual form comparison, and read instructional feedback

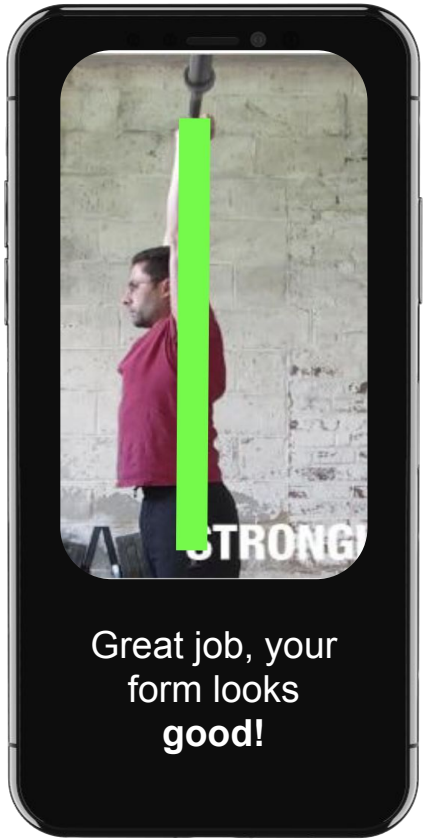
Validation: Time the duration from set complete to feedback appearing on the phone app

Mitigating risk: User can view correct form GIF for their exercise immediately

Project Management



Putting a personal trainer right on your wrist



Use Case: Workout Form Correction Coach

- Problem Area:
  - Fitness, consumer tech, coaching, predictive and analytic tech
- Problem Currently:
  - Personal training is prohibitively expensive ($60+ / hr).
- Training Platform
  - Analyze a user’s form in live time
  - Diagnose issues they may have
  - Provide visual and instructional feedback to issue
- ECE Areas:
  - Signal Processing
  - Computer/Software Systems

Final Output: Joint Estimation and Correction

Requirements I

- User Interface - user can easily:
  - Begin and end a coaching session
  - Select a type of exercise to perform
  - Demarcate start and end of sets
  - View form feedback after every set
- Network - Apple Watch↔iPhone↔EC2
  - Transmitting packets over BTLE (Apple Watch↔iPhone)
  - Transmitting packets over HTTP (iPhone↔EC2)
  - System handles dropped packets without issues

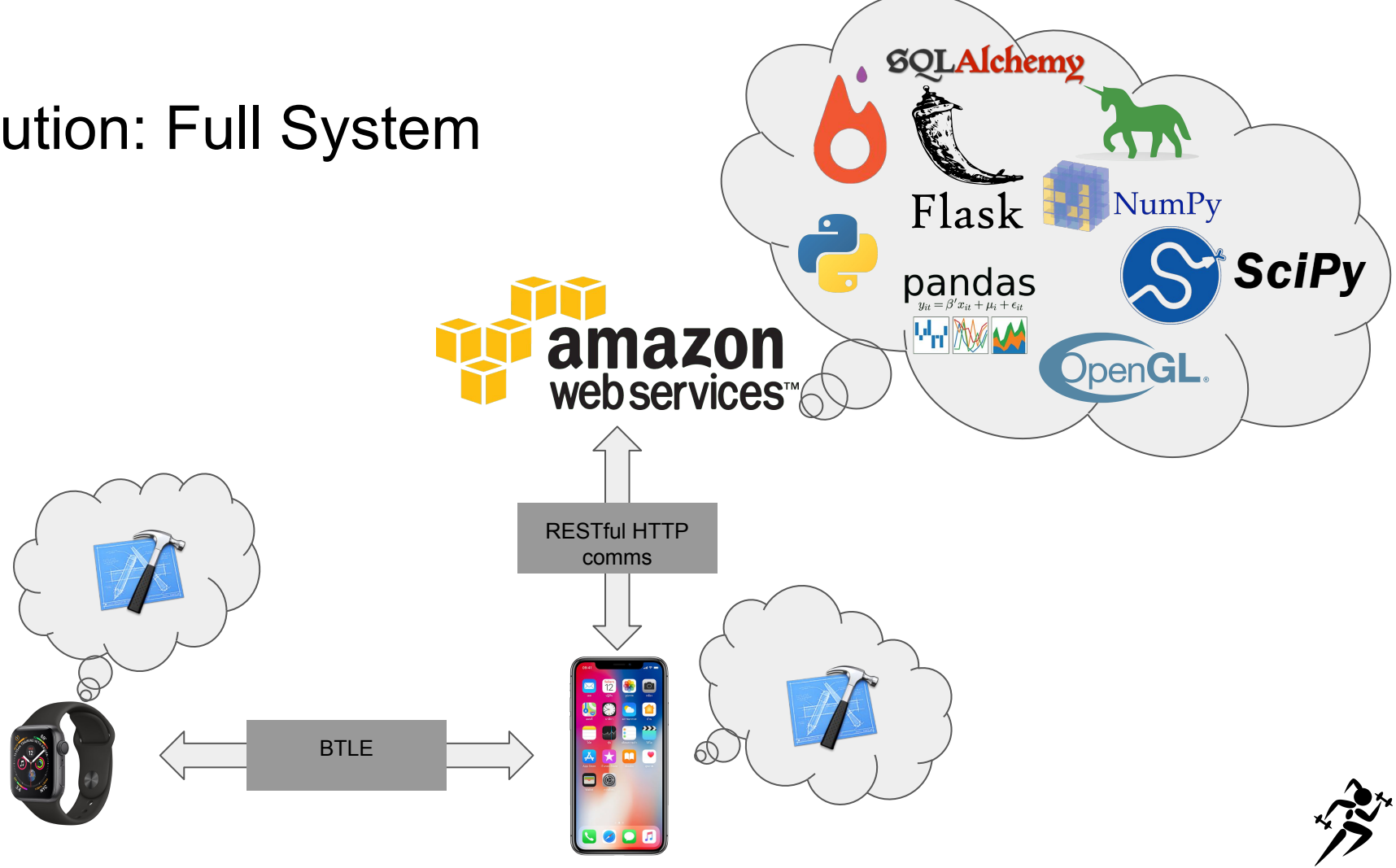
Requirements II

- Backend Management
  - Manage Users: ID, logins, personal training data etc in DB
  - Flask: General system management (HTTP, RESTful etc)
- Signal Processing - Data analysis system will be able to:
  - Identify demarcations between reps from a set of the desired exercises
  - Count the number of reps performed with an average of >95% accuracy
  - Process an ‘exercise set’ of data and recognize form issues with an average >95% accuracy
- User feedback -
  - Rendering user’s motion, visualizing ideal motion
  - Explain issue to user from a set of pre-allocated typical problems (Descriptions are pre-made and stored in DB)

Key Challenges

- User Interface:
  - Designing for ease of use
- Network:
  - System reliability - no crashes or performance drops
- Backend Management:
  - Orchestrating backend systems including: databases, deep nets and graphics generators without clogging the system or crashing
- Signal Processing:
  - Demarcating exercise repetitions accurately
  - Training CNN to find faulty form
  - Estimating position of user’s limbs from only IMU data

Solution: Full System



Data Processing Solution: Server Side

Backend Management

SQLAlchemy

Flask

Complete Backend Orchestration

Data Analyzing System

Classify form issue

SciPy

NumPy

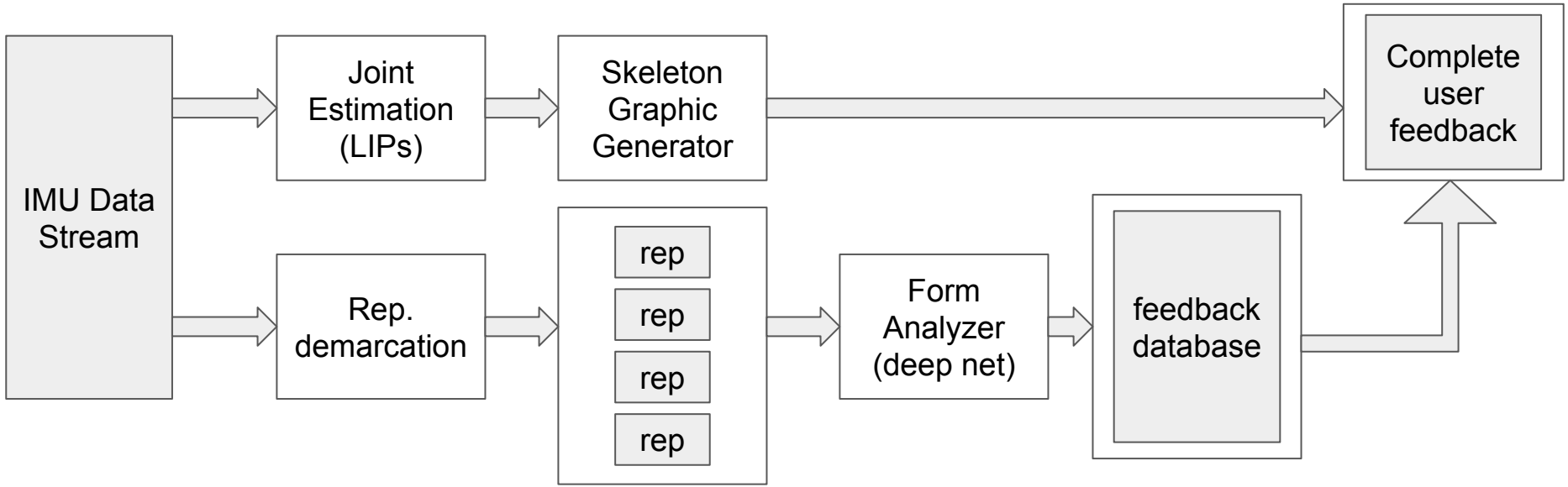
pandas

IMU data preprocessing and Position Estimation (inverse problem)

OpenGL

Render Basic Skeleton Graphic of good vs bad form

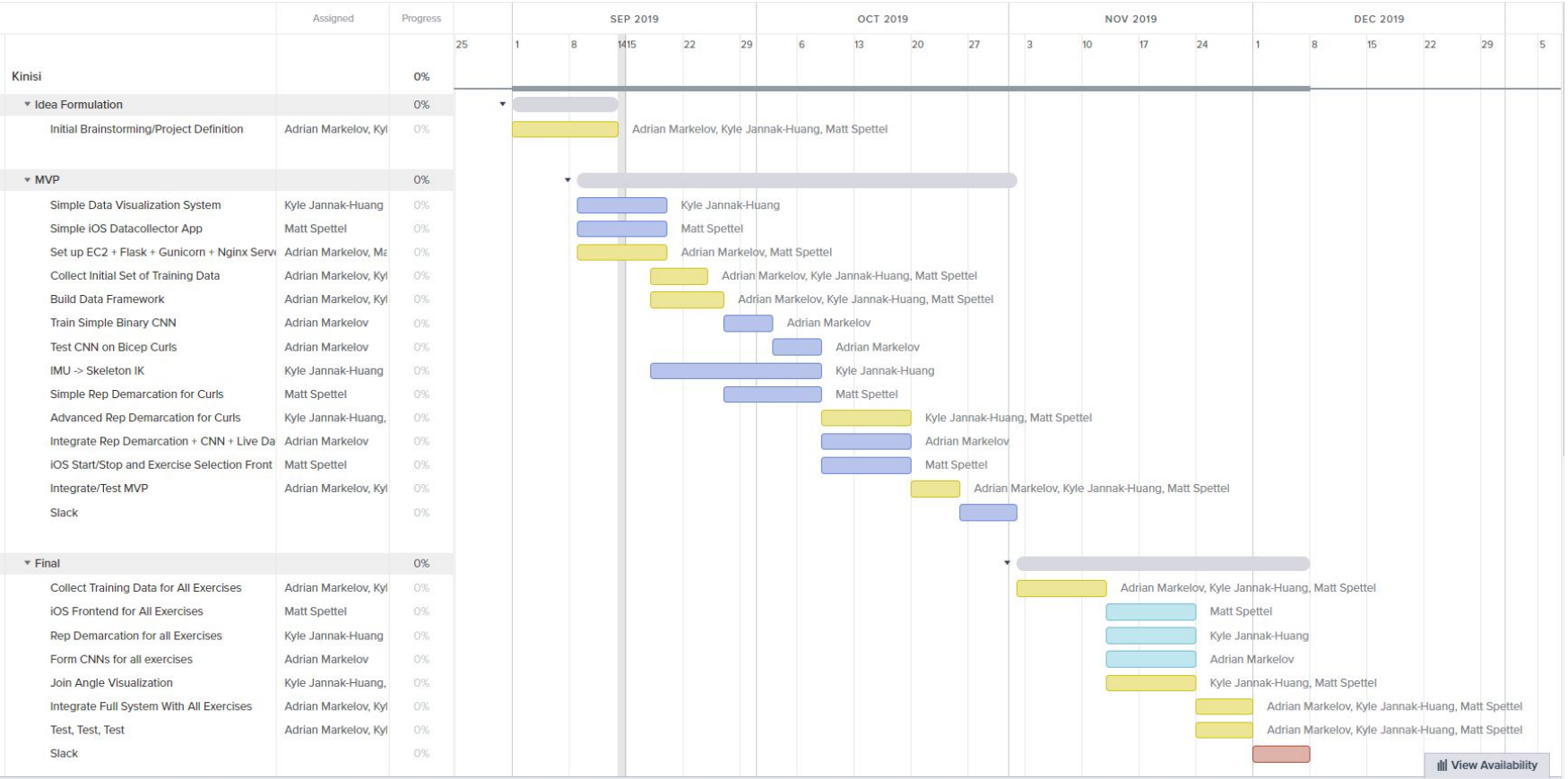
Data Processing Model



Task Partitions

- Adrian:
  - Backend server management
  - Faulty form detection and classification with CNN
- Kyle:
  - IMU data processing
  - Repetition demarcation
  - User position estimation
  - Graphics generation
- Matt:
  - Backend server management
  - iOS (UI + Networking)
  - Repetition demarcation

Schedule (Gantt Chart)



Testing and Requirement Success Metrics

User Interface:

- If a user can navigate the app and understand the feedback well enough to correct their form without external guidance, the UI is effective.

Rep Demarcation:

- Run demarcation algorithm on every set of training data.
  accuracy = 1 - abs((repsCounted - totalReps) / totalReps)

Faulty Form Detection

- Run form detection CNN on each rep of training data
  accuracy = correctlyLabeledReps / totalReps