VizFit: Computer Vision Application for Exercise

Sasha Lawson Sam Donaldson
Alana Matheny Noah Buchanan

February 18, 2022

Abstract

This project sets out to solve the following problems: Lack of an insightful fitness breakdown, inconsistency in maintaining a fitness routine, and the difficulty in properly exercising due to improper technique or a lack of general fitness knowledge. We propose the following solutions to each respective problem: First, by utilizing personalized user analytics that we aggregate over time we will provide insightful information about a user's personalized fitness journey. Second, providing an at-home substitution to going to the gym will allow for a better incentive to users in-order to maintain fitness consistency. Finally, to decrease the difficultly of properly exercising we will implement computer vision and machine learning concepts to determine the accuracy of a given fitness activity and maintain useful statistics such as rest times between exercises.

Our goal, on a much broader scope, is to make fitness easier. This will be accomplished through various functions such as computer vision and machine learning models, a front-end webpage which will provide personalized data and feedback, a database, and an API to move data amongst these entities. Our deliverables will include the Project Proposal, User Interface Design Graphics, Database Schema and Data, Deep Learning Model Code, Webpage Application Development Code, API Development Code, and the Final Report.

Department of Computer Science and Engineering
University of Arkansas – Fort Smith

Contents

Contents **2**

1 Introduction **3**
1.1 Problem Statement . . . 3
1.2 Objective . . . 4

2 Background **4**
2.1 Overview . . . 4
2.2 Technology . . . 5
2.3 Related Work . . . 7

3 Design **8**
3.1 Project Requirements . . . 8
3.2 Architecture . . . 9
3.3 Risks . . . 13
3.4 Tasks and Schedule . . . 14
3.5 Deliverables . . . 15

4 Project Members **15**
4.1 Team Members . . . 15
4.2 Departmental Advisors . . . 16

5 References **17**

1 Introduction

1.1 Problem Statement

Fitness is important. The benefits of proper fitness in increasing lifespan has been demonstrated in a plentiful number of prior research studies. However, nearly 80% of adults are not getting enough exercise. [1] There are three problems individuals are facing regarding fitness: One, the lack of an insightful fitness breakdown. Insight into a fitness routine allows one to better identify weaknesses and strengths in both the physical and technical sense. Second, consistency in maintaining a fitness routine. Failure in maintaining a consistent fitness routine leads to inconsistent progression and the risk of dropping the routine all together. Lastly, exercising correctly is important but difficult. For example, repetitive, overdone exercises can cause tearing, stress, and other negative effects on one's body. [2] Each one of these problems overlap in various ways, but stand out distinctly enough to be highlighted as key points to be improved upon in this project.

Exercising is a highly versatile activity. Depending on what exercise one does they net a different result. High intensity exercises that gets the heart pumping faster lower high levels of cholesterol while a peaceful walk soothes high blood-sugar levels. [3] This level of versatility demands personalization to get the most out of one's exercising and avoid overexercising parts of the body. This can be accomplished through insight into one's fitness and health data.

One of the many traditional New Year's Resolutions is to "get into shape". However, about two thirds of those who make any resolution at all quit within the first month [4]. This demonstrates a failure to maintain a consistent fitness routine. Diving deeper into this issue suggests that this inconsistency could be caused by having to go to the gym. Out of 2,000 Americans surveyed, 50% felt working out in a gym environment to be a "daunting" task. Many cited feeling intimidated by fit individuals and even the opposite sex. [5] Fear and laziness are the enemy of consistency in this situation causing many to abandon their fitness plan before making any progress.

Properly exercising is difficult. Setting aside all the issues already listed, safely and correctly exercising is critical. This means just doing the motion of the exercise does not equate to the exercise's intended result. Take a push-up for example, what seems like a straightforward act of going up and down using the strength of one's arms can be done incorrectly. Given the correct situation doing a push-up incorrectly can lead to potential upper body pain and injury [6]. Ensuring one's safety through proper technique is key to lessening the difficulty of properly exercising.

1.2 Objective

Our project sets out to solve three problems that modern fitness applications typically have. Firstly, our application will make fitness insightful. We will keep a variety of statistics for each user to provide invaluable information about their fitness goals and their rate of progression. This makes each user's fitness journey highly personalized.

The second problem we are attempting to solve is the matter of consistency required for physical fitness. Making it to the gym multiple times a week is a daunting task for busy individuals. This is where our application comes in. With our application you can exercise from any location that you have a computer and a webcam.

The last problem for most is that exercising is difficult. Our application aims to make it easier through two key innovations. First, our application will monitor your exercises and make sure you are performing them correctly. Additionally, our application will track your live progress and rest times throughout a workout. Both of these combined can make fitness magnitudes easier for those that find fitness difficult.

2 Background

2.1 Overview

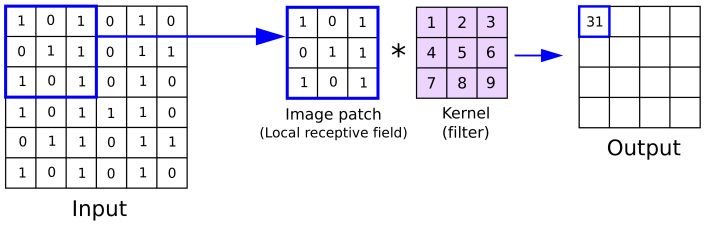
Convolutional Neural Network

The traditional neural network is a powerful tool that can be used in a variety of ways within limitations; one such limitation is the extremely costly computation of large amounts of data i.e images. Classifying multiple images in real time with millions, potentially billions, of parameters (depending on the resolution of the images and the amount of layers and neurons used) is not practical. This is where convolutional neural networks (CNNs) come in handy. CNNs do three specific things to make image classification practical:

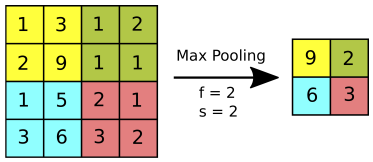
1. Reduce the number of input nodes and parameters.
2. Tolerate small shifts in where the pixels are in the image. Essentially meaning that through each filter that is applied, a band of width = $\text{floor}(\frac{\text{length}(f)}{2})$ in pixels, where f is the filter, will be removed from the edges of the image.
3. Take advantage of the correlations that are observed in complex images.

The first thing a CNN does is apply a filter to the input image. Filters are sometimes also called kernels. For this explanation the former will be used. A filter is a smaller matrix of pixels typically around the size of 3×3. The intensity of each pixel in the filter is determined by back propagation. To begin with the pixel values in the filter are randomized.

The filter is overlaid onto the image and the dot product of the image and the filter is calculated to get a value that we add a bias to and then put into a feature map.



We can say that the filter is "convolved" with the input after performing this which is what gives CNNs their name. The filter is then moved over one pixel or possibly more, this is up to the users discretion, and does that process over again. Once the feature map is finished we run it through an activation function, typically ReLU. Once the map has been run through ReLU we max pool the feature map to further minimize the size, max pooling meaning that the filter does not overlap itself, so for a 2×2 filter it would move 2 pixels. Max pooling selects the spots where the filter did the best job matching the input image; mean pool alternatively could be used where the mean of the filter is taken rather than the maximum value.



Now we convert the pooled layer into an $n \times 1$ input layer. These input nodes are then plugged into a normal neural network. [7]

2.2 Technology

MySQL MySQL is a relational database management system. MySQL remains free and open source providing high data scalability, security, and high performance speeds. [8] MySQL is widely used by companies such as Youtube, Twitter, Facebook, Netflix, Github, and Paypal. [9] Our primary method through which we will query and insert data will be through SQL calls via JDBC from our Java-based API.

Vue.js Vue is a self-described "progressive" framework for constructing user interfaces and front-ends. Vue's core library focuses on the view layer only, meaning one can start simple then build up in complexity from there. [10] The simplicity of Vue comes in the form of components which contain some data and/or logic. These components can be reactive based on logic within the component's script. Vue's easy to use framework will be used to build the webpage dashboard where users can track their health and fitness analytics.

Vuetify Vuetify is a user interface framework built to complement Vue.js. Vuetify is easy to learn and implement into any vue app. Vuetify has "meticulously" crafted components that are built to be responsive and work "out of the box" for any type of screen. [11] Vuetify's functionality will allow for easy web design development in our front-end system.

Java Java is an object-oriented programming language. Java runs using the Java virtual machine which allows Java to be run on any platform. [12] Java is widely by companies such as Google, Instagram, and Mojang. [13] Java's widespread use to develop API's makes it the perfect candidate to be used for building the application's API.

Spring Spring is a RESTful API built on Java. Spring has "battle-tested" security that protects against attacks. Spring is commonly used so documentation is plentiful. Since Spring is built on Java, which is a language each of our team members is familiar with, each team member will be able to help develop the API. [14]

Python Python is an object-oriented, interpreted, high-level programming language. [15] In 2021 Python became the most popular language to use amongst developers, overtaking both C and Java. This popularity is easily accounted for by Python's simplicity and vast libraries. [16] Specifically, Python features a variety of machine learning and data science libraries. These libraries make Python necessary for the application's machine learning heavy aspects. Python will support the computer vision and deep learning functions of training and identifying a given exercise activity.

OpenCV-Python OpenCV-Python is an open-source python library package that provides various computer vision functionality. [17] OpenCV-Python is integral to the application as it provides the connection between the webcam and the machine learning program.

TensorFlow TensorFlow is a library package with various machine learning models that is used by both beginners and experts alike. [18] TensorFlow will be used to create neural networks that will be critical in the training process. Additionally, TensorFlow offers various training data that may prove useful.

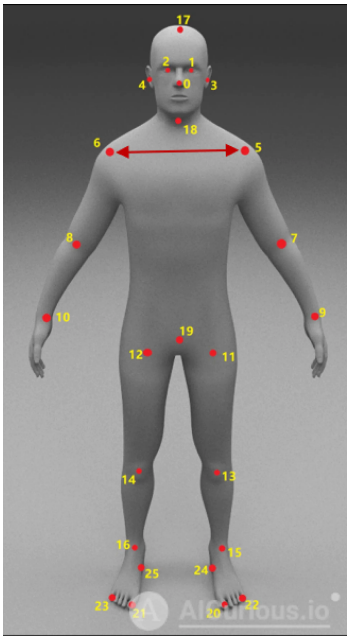
Matplotlib Matplotlib is a Python library package used to create useful data visuals. [19] Though data visuals may not be shown in this area of the application, Matplotlib will be essential for displaying the images received from the webcam for testing purposes.

Python Imaging Library Python Imaging Library, or PIL, is a Python library package used to open, save, and manipulate images across various file formats. [20] PIL will be another essential library so that images received from the webcam may be shown and manipulated whenever necessary.

2.3 Related Work

Push-up Counter

In [21] a utilization of OpenCV and Deep Learning was demonstrated to complete the task of counting push-ups. Their specific approach involved human keypoint detection (or human pose estimation). They used the MPII Human Pose Dataset and various push-up videos from Facebook as their training dataset. They used a unique metric to measure the quality of these keypoints, Percentage of Correct Keypoints (PCKs): "A detected joint is considered correct if the distance between the predicted and the true joint is within a certain threshold. The threshold here is chosen as 0.25 times of the distance between 2 wrists (or the distance between point 5 and point 6 in the image)."



This model for key point extraction is separate from the exercise recognition model. Signal processing from the keypoints is what is used to count repetitions. In their conclusion they stated they felt they could have achieved better results with more keypoints and leveraging the keypoints to correct wrong push-up poses. We will attempt to achieve both of these. The "sweet spot" of keypoints will likely be found through experimentation so we cannot report on that as of now. As for bad exercise poses, we intend to measure and fix these through the angles between specified keypoints (the best keypoints to measure will also likely be apparent through experimentation).

Recognizing Exercises and Counting Repetitions in Real Time

In [22] pose tracking and exercise recognition is used to count repetitions on pull-ups, push-ups, and squats. Their method involves three phases; pose tracker to identify and track users, exercise recognition to detect the name of the appeared exercises, and a counter to count and indicate the correct and incorrect repetitions. For pose tracking they use OpenPose. For exercise recognition however, the authors of this paper did not use a convolutional neural network rather a normal neural network; it is our teams' belief that we can achieve better results through a convolutional neural network. For repetition counting they first use a formula and the predictions of their neural network to predict the exercise as follows:

$$\text{performed\_exercise}(f) = most\_common(y(f-9), y(f-8), ..., y(f))$$

Where, f: frame, and y: predicted label from the neural network.

Once the exercise is detected, there are pre-selected parameters such as exercise range of motion, the major joint, and type of motion (push or pull) that will be used to count repetitions. We will likely use the same formula for detecting exercise as them. However, as stated before, they did not use a convolutional neural network; we will attempt to implement this and achieve better results.

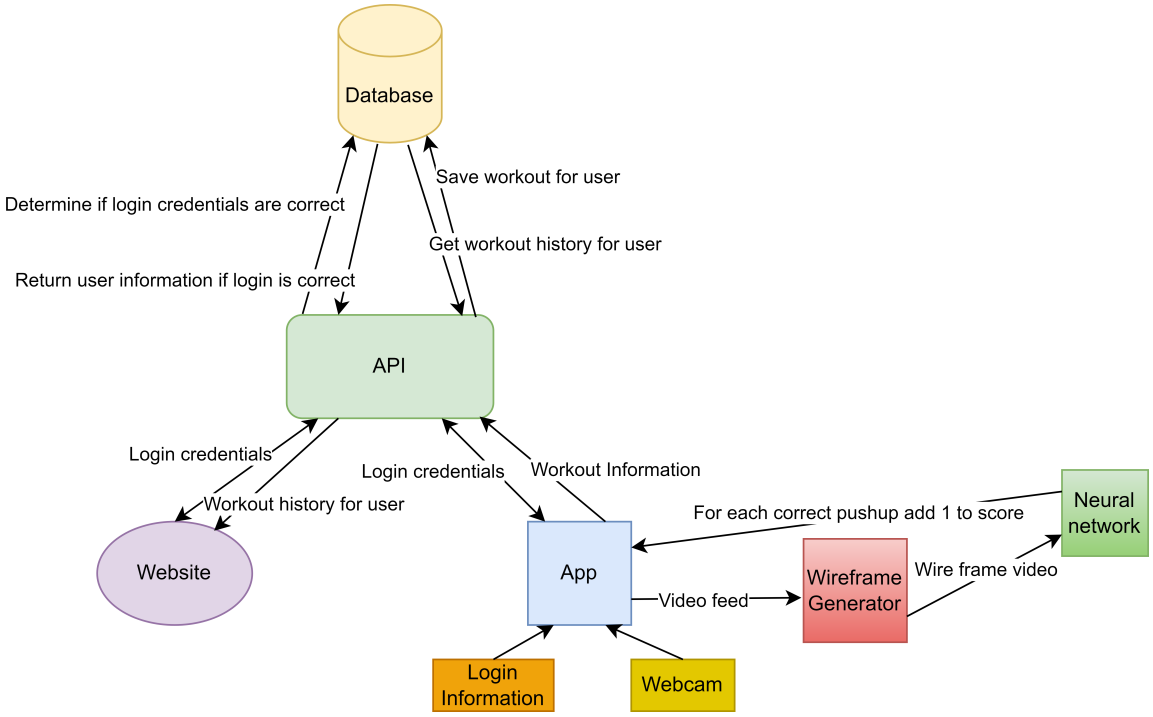
3 Design

3.1 Project Requirements

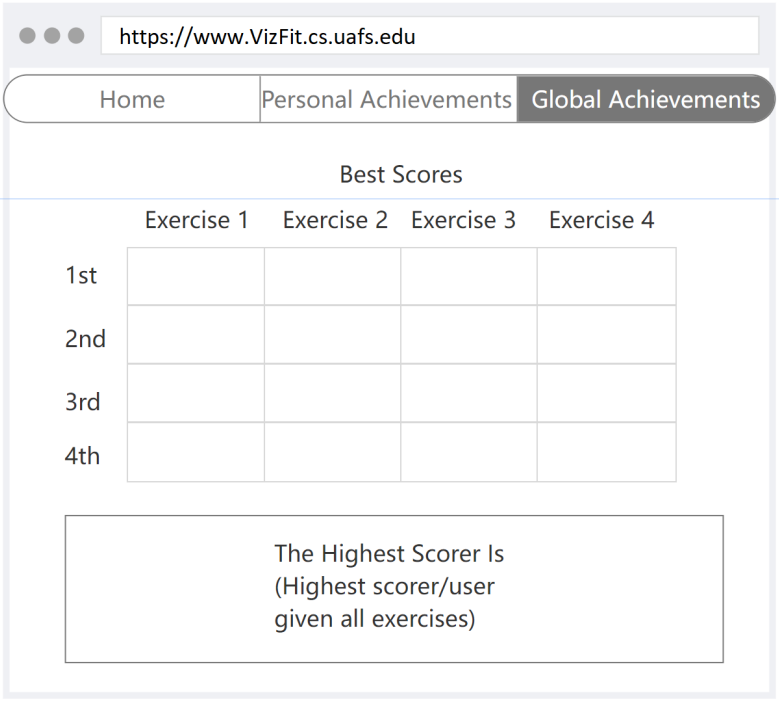
- A database to save and manage the progress made by our users.
- A user-faced app that uses a camera to keep track of how many reps and if the exercise is performed correctly.
- An API for the website and client to connect to to communicate with the database.
- A server to host the website, API, and database.

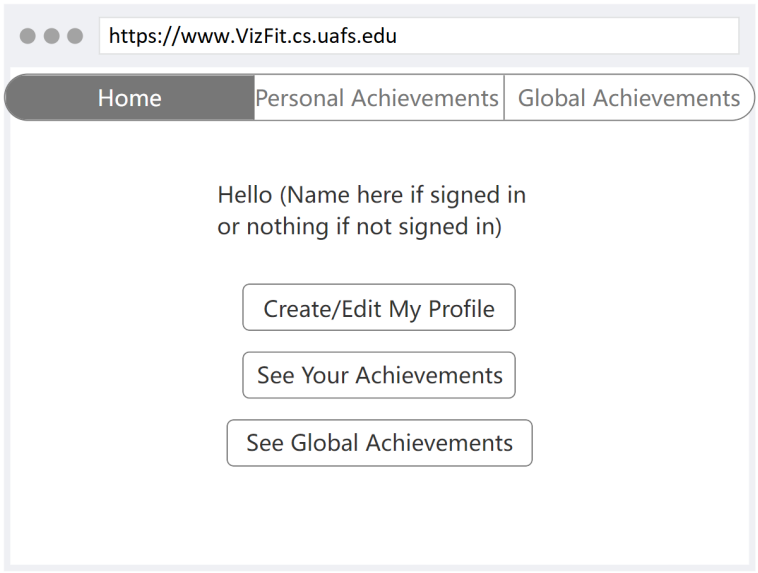
3.2 Architecture

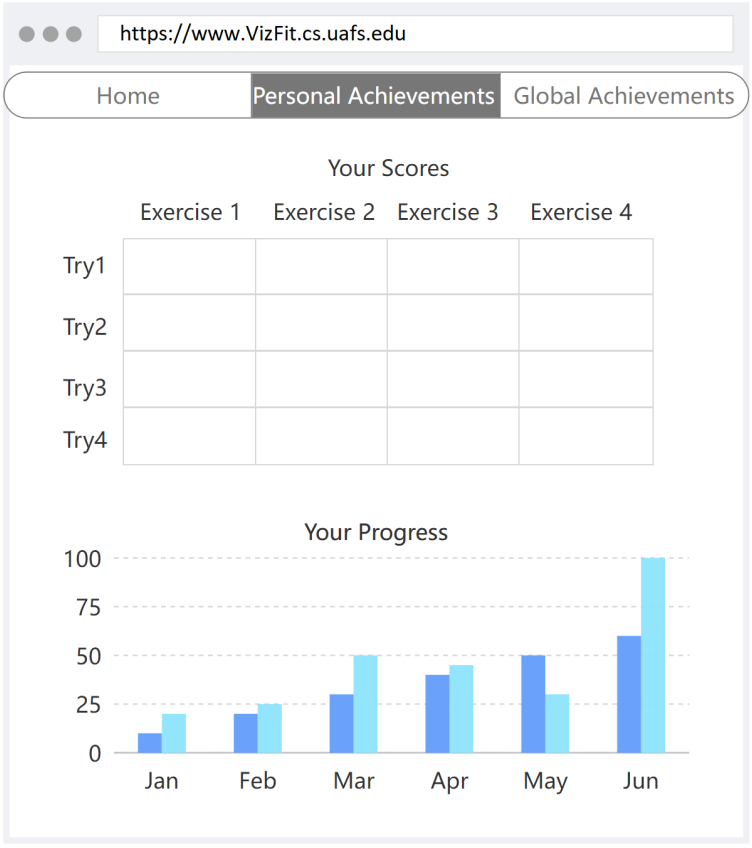
Our application is a basic program that captures video feed from a user's device. The application sends that video feed to an application that generates wire frames based on a human body. The application then sends the video of the wire frames to our state-of-the-art neural network. The network then determines if the wire frames, based on the user, are moving in such a way that the user is doing a correct push-up. If the network determines that a user has done a correct push-up it increases the user's score. We keep track of the number of correct push-ups the user has done and save that information based on the user's credentials to our database. The database is behind an API that allows us to easily select and insert information securely. Our website is also connected to this API and will display a user's information based on their login credentials.



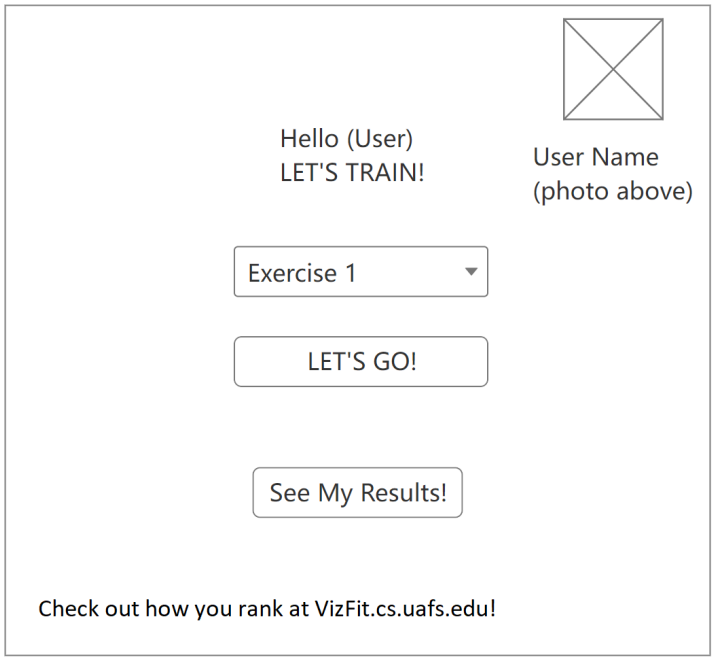
Web Mockups

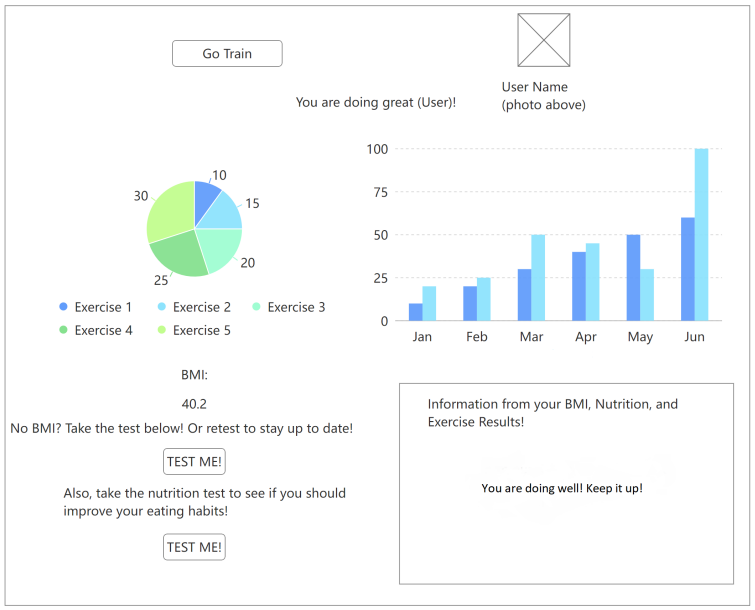

https://www.VizFit.cs.uafs.edu
Home
Personal Achievements
Global Achievements
Best Scores
Exercise 1
Exercise 2
Exercise 3
Exercise 4
1st
2nd
3rd
4th
The Highest Scorer Is
(Highest scorer/user
given all exercises)


https://www.VizFit.cs.uafs.edu
Home
Personal Achievements
Global Achievements
Hello (Name here if signed in
or nothing if not signed in)
Create/Edit My Profile
See Your Achievements
See Global Achievements


https://www.VizFit.cs.uafs.edu
Home
Personal Achievements
Global Achievements
Your Scores
Exercise 1
Exercise 2
Exercise 3
Exercise 4
Try1
Try2
Try3
Try4
Your Progress
100
75
50
25
0
Jan
Feb
Mar
Apr
May
Jun

Application Mockups





3.3 Risks

Building a fitness application comes with many risks. Given below are some of the common risks and the steps we are taking to mitigate them.

1. **Data Security** - Data security for users is very important to us. To assure privacy for our users' data, we will implement appropriate and thorough encryption to their data through a secure API and database. We will also make the data we collect as anonymous as possible, and the user will own their data. For example, the data we collect will not be sold or given to anyone and we will not save any recordings we take of a user doing their exercises.

2. **Issues With The Systems** - Many things could go wrong when implementing the systems that would prevent users from using the application. For example, if the server goes down, the API can not connect to the database, and if the API can not connect to the application and/or website the user would not be able to save their exercise data or retrieve their saved data. To prevent this issue from occurring the database, API, application, and website will be thoroughly tested for bugs, constructed properly with dependability in mind, and the appropriate security will be implemented to ensure privacy for the user. One other issue that would prevent users from using the application would be if the machine learning model is constructed and integrated into the application incorrectly. This issue could lead to not being able to detect the user or correctly interpret the data it collects. To ensure this is prevented, we will do extensive testing when integrating the model into the application and have the model inspected by every team member.

3. **Bias Datasets** - When using neural networks to learn to identify humans and the specific movements they do, bias or incorrect datasets while training or after training could result in unfair results depending on the user. To prevent bias datasets, we will implement well-tested, pre-trained datasets to track users and the specific movements they do. In looking for datasets we will attempt to find datasets where the set will include different body types, ages, genders, etc. By using different groups of people. This will help standardize results and mitigate bias. One of the other ways we will implement a bias-free model will be to create a stick figure representation of a human to train the model to so differences in users body type, clothing, and other factors will be mitigated.

3.4 Tasks and Schedule

Task	Dates
Create, revise, and finish proposal	1/10 - 1/22
Research and find good exercise datasets	1/17 - 1/30
Implement code for initial dashboard webpage design	1/24 - 1/30
Implement code for initial machine learning application GUI	1/24 - 1/30
Setup and configure database	1/24 - 1/30
Create and implement database schema	1/24 - 1/30
Implement wireframe manipulation	1/31 - 2/6
Create first neural network model	1/31 - 2/6
Create presentation for Week 5	1/31 - 2/6
Create preliminary back-end API	1/31 - 2/13
Create controller, readers, and writers classes for back-end API	1/31 - 2/13
Connect dashboard webpage API to back-end API	2/7 - 2/13
Create functionality for users to create/delete their account	2/7 - 2/27
Improve, rebuild, and train neural network model	2/7 - 2/27
Improve visual design of dashboard webpage	2/14 - 2/20
Finish GUI for machine learning application	2/21 - 2/27
Create user authentication for both front-end applications	2/21 - 2/27
Create presentation for Week 8	2/21 - 2/27
Improve, rebuild, and train neural network model	2/28 - 3/13
Finish front-end application designs	2/28 - 3/6
Ensure back-end API sends/receive data from machine learning application API	3/7 - 3/13
Attempt to finalize neural network model	3/14 - 3/20
Fix issues with front-end applications' designs	3/14 - 3/20
Create presentation for Week 11	3/14 - 3/20
Review all system code and correct bugs as needed	3/21 - 3/27
Discuss adding extra features	3/21 - 3/27
Implement complete user testing and fix bugs as needed	3/28 - 4/10
Finalize any project paperwork	4/4 - 4/10
Receive project review and make adjustments as needed	4/11 - 4/17
Final presentation	4/18 - 4/24
Project defense	4/24 - 5/1

3.5 Deliverables

- Project Proposal
- User Interface Design Graphics and Flowcharts
- Database Schema and Data
- Deep Learning Model Code
- Web page Application Development Code
- API Development Code
- Final Report

4 Project Members

4.1 Team Members

Alana Matheny (4023) Alana Matheny is Computer Science major with concentration in Data Science and Artificial Intelligence and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas - Fort Smith. She has completed relevant coursework for the proposed project by completing CS 3113 - Artificial Intelligence, CS 2033 - Web Systems, CS 2043 - Database Systems II, CS 3003 - Distributed Systems, CS 4003 - Software Engineering, CS 4033 - Ethics in Professional Practice, CS 3323 - Computer Graphics, CS 4043 - Formal Languages, CS 4323 - Data Analytics, and CS 4373 - Information Retrieval. Her responsibilities will include the development of a machine learning model, front-end development of the website, front-end development of the application, back-end development of the database, and back-end development of the API.

Sasha Lawson (4023) Sasha Lawson is a Computer Science major with concentration in Data Science and Artificial Intelligence and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith. He has completed relevant coursework for the proposed project by completing CS 3113 – Artificial Intelligence, CS 3323 Computer Graphics, and CS 4333 – Machine Learning. He also obtained relevant experience through his internship at ArcBest Technologies as a Software Developer Intern. His responsibilities will include the development of a machine learning model to learn the proper actions for a given exercise. Additionally, he will develop required back-end systems to support the dashboard webpage.

Sam Donaldson (4023) Sam Donaldson is a Computer Science major with concentration in Data Science and Artificial Intelligence and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith. He has completed relevant coursework for the proposed project by completing CS 3113 – Artificial Intelligence, CS 4343 – Natural Language Processing, CS 2033 – Web Systems, CS 4003 – Software Engineering, CS 3323 - Computer Graphics, and CS 2043 – Database Systems II. He also obtained relevant experience through his internship at ArcBest Technology as a Software Developer Intern. His responsibilities will be the development of a machine learning model, front-end development of the website, and back-end development of the database and API.

Noah Buchanan (4023) Noah Buchanan is a Computer Science major with concentration in Data Science and Artificial Intelligence and a minor in mathematics in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith. He has completed relevant coursework for the proposed project by completing CS 3113 – Artificial Intelligence, CS 4333 – Machine Learning, CS 2033 – Web Systems, CS 4003 – Software Engineering, and CS 2043 – Database Systems II, CS 4373 – Information Retrieval, CS 4363 – Internet of Things Development. His responsibilities will include the development of our proposed model, front-end development, back-end development, and development of our API.

4.2 Departmental Advisors

Our project was supervised by Professor Israel Cuevas and Professor Andrew Mackey in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith.

5 References

[1] "The physical activity guidelines for americans." [Online]. Available: https://jamanetwork.com/journals/jama/article-abstract/2712935

[2] "7 ways exercise can go wrong for your mind and body." [Online]. Available: https://www.psychologytoday.com/us/blog/fulfillment-any-age/201409/7-ways-exercise -can-go-wrong-your-mind-and-body

[3] "National institutes of health," Aug 2020. [Online]. Available: https://newsinhealth.nih.gov/2020/07/personalized-exercise/

[4] J. M. Dickson, N. J. Moberly, D. Preece, A. Dodd, and C. D. Huntley, "Self-regulatory goal motivational processes in sustained new year resolution pursuit and mental wellbeing," *International Journal of Environmental Research and Public Health*, vol. 18, no. 6, 2021. [Online]. Available: https://www.mdpi.com/1660-4601/18/6/3084/

[5] "Gymtimidation is real: Half of americans afraid to work out around others," Sep 2021. [Online]. Available: https://swnsdigital.com/us/2019/03/gymtimidation- is-real-half-of-americans-afraid-to-work-out-around-others/

[6] A. Malaythong, "Proper pushup form and technique: Nasm guide to push-ups." [Online]. Available: https://blog.nasm.org/nasm-guide-to-push-ups/form-and-technique/

[7] A. H. Reynolds, "Convolutional neural networks (cnns)," Oct 2017. [Online]. Available: https://anhreynolds.com/blogs/cnn.html/

[8] T. Branson, "8 big advantages of using mysql," May 2021. [Online]. Available: https://www.datamation.com/storage/8-major-advantages-of-using-mysql/

[9] "Mysql." [Online]. Available: https://www.mysql.com/

[10] Vue.js, "Introduction - vue.js." [Online]. Available: https://vuejs.org/v2/guide/

[11] "Why you should be using vuetify." [Online]. Available: https://vuetifyjs.com/en/introduction/why-vuetify/

[12] R. A. S, "What is java api, its advantages & need for it: Simplilearn," Nov 2021. [Online]. Available: https://www.simplilearn.com/tutorials/java-tutorial/java-api/

[13] "Why developers like java." [Online]. Available: https://stackshare.io/java/

[14] "Building rest services with spring," 2022. [Online]. Available: https://spring.io/guides/tutorials/rest/

[15] "What is python? executive summary." [Online]. Available: https://www.python.org/doc/essays/blurb/

[16] B. Borisov, "Python has become the most popular language, according to tiobe," Oct 2021. [Online]. Available: https://linuxiac.com/python-the-most-popular-programming-language/

[17] "Opencv." [Online]. Available: https://docs.opencv.org/4.x/d1/dfb/intro.html

[18] "Tensorflow 2 quickstart for beginners : Tensorflow core." [Online]. Available: https://www.tensorflow.org/tutorials/quickstart/beginner/

[19] "Visualization with python." [Online]. Available: https://matplotlib.org/

[20] "Pillow." [Online]. Available: https://pillow.readthedocs.io/en/stable/

[21] V.-A. Nguyen, "Build a pushup counter app with opencv and deep learning," Feb 2021. [Online]. Available: https://aicurious.io/posts/2021-02-15-build-a-pushup-counter/

[22] T. Alatiah and C. Chen, "Recognizing exercises and counting repetitions in real time," May 2020. [Online]. Available: https://arxiv.org/abs/2005.03194/