Team Deliverable 1

Team 4: Ahmed Farid Khan, Bennett Blanco, Dhruv Shah, Yu-Chin (Alyssa) Chen

Introduction:

Our project focuses on building a cloud-based data pipeline to ingest and analyze activity data from Strava, specifically from individuals within our social circle. Through this initiative, we aim to help runners optimize their training routines by offering personalized insights. By leveraging Strava's detailed activity data, we can provide actionable feedback to improve a runner's performance, encouraging them to adjust their routines for better results. This not only enhances the user experience but could also attract more users to the app, leading to higher user retention and overall growth.

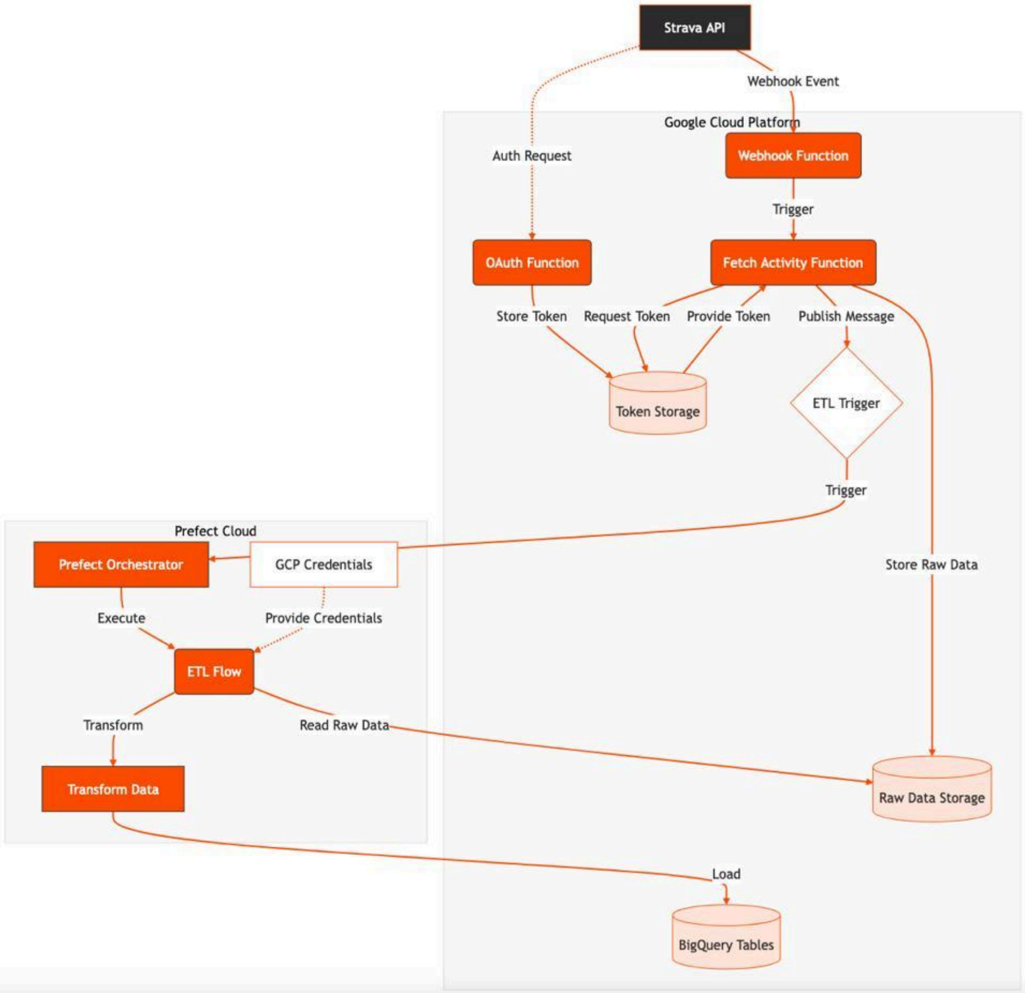
To achieve this, we are primarily using Strava's activity data, accessed via its API. This API provides a range of detailed metrics on runs, such as distance, speed, and duration, which are updated in real-time whenever a user completes an activity. Consequently, our database is consistently updated, ensuring we always have the most current information to support our analysis and insights.

Data Pipeline:

To support this process, we've designed an ETL pipeline that efficiently extracts, transforms, and loads the Strava data. An independent OAuth function either initiates the OAuth flow by generating a URL that redirects users to Strava's authorization page, or handles the callback after authorization. When Strava redirects the user back with an authorization code, the function exchanges the code for access tokens by making a POST request to Strava's token endpoint. Upon successful exchange, it retrieves and saves the tokens (which include access and refresh tokens) along with the athlete's ID in Google Cloud Storage. This allows the application to securely access the user's Strava data in the future without requiring reauthorization.

The workflow begins when a user completes a run, triggering a notification via the Webhook Function. Strava allows registered apps to subscribe to webhook events, which will return any POST committed by a user that is verified through OAuth. This sends a message to a PubSub which a data extraction cloud function is subscribed to, which leverages the information from the PubSub to make the necessary API requests from Strava. Once extracted, the raw data as JSON in a Cloud Storage bucket and sends a PubSub to a cloud function that triggers flows in Prefect Cloud via an HTTP request. It passes the athlete and activity data as parameters to the Prefect flow. The Prefect flow has two components, the deployment script ensures that the ETL flow is registered and ready to be triggered in the Prefect Cloud environment, enabling the automation of its execution and the ETL flow script is designed to automate the process of extracting, transforming, and loading Strava activity data. It begins by using Prefect tasks to load Google Cloud credentials and then extracts activity and lap data from Google Cloud Storage, specifically targeting JSON files related to an athlete's activities. The extracted data is then transformed into a structured format using pandas, ensuring proper data types, formatting dates, and handling key columns like speed, elevation, and geographical coordinates. After the data is cleaned and processed, it is loaded into Google BigQuery tables for further analysis. The flow is designed to run within the Prefect framework, enabling reliable and efficient data pipeline management.

Using BigQuery's powerful querying capabilities, we can generate insights such as personalized training recommendations, analyze the impact of performance, and produce detailed progress reports for users. These insights are accessible through cloud-based reporting tools, providing users with timely feedback that helps them refine their training.



Future Steps:

Currently BigQuery connects to Tableau for dashboarding and deriving insights via service account credentials just as a proof of functionality. We plan to extend this system by scaling the dashboard service and incorporating supervised and unsupervised machine learning models that will offer even deeper insights, such as predicting heart rate as runners often train within specific heart rate zones to target endurance, fat burning, or speed and a predictive model can help ensure that runners stay within the desired heart rate zone, maximizing the effectiveness of their workouts and preventing burnout and Clustering runs based on characteristics like distance, pace, elevation gain, and heart rate, users can identify patterns in their own performance. For example, a runner may realize they perform best on certain types of terrain or at specific times of the day. Additionally, we aim to develop interactive dashboards that will allow users to visually explore their progress and performance over time, enhancing user engagement and providing further value.

Ultimately, this automated pipeline is essential to our business objective of helping runners improve their training through timely, personalized insights. By providing unique value through data-driven training suggestions, we enhance user engagement and retention, while also laying the foundation for future features like personalized coaching and injury prevention. Machine learning models and interactive dashboards will be key future developments, further supporting the app's long-term goals.