SIT374
Capstone Team Project (A)
Process Overview

- **.FIT Files:** FIT stands for **Flexible and Interoperable Data Transfer**. It is a data file format developed by Garmin to store and transfer fitness and health-related data from fitness devices such as Garmin and Wahoo watches, fitness trackers, and other sports and health monitoring devices efficiently. These files are binary files that need to be parsed to extract meaningful data.

- **SDK**: SDK stands for **Software Development Kit.** It is a collection of tools, libraries, and documentation that developers use to build software applications for specific platforms, purposes or services.

- **API:** API stands for **Application Programming Interface.** It is a set of rules and protocols that define how software components should interact with each other.

- The goal is to convert this data into a format that can be analysed and visualized for insights into athletic performance.

- We use APIs to fetch these .FIT files from the devices.

- Now, Garmin has an SDK that is designed to simplify the process of accessing and using data from Garmin devices. It provides the necessary tools and interfaces for developers to connect their applications with Garmin devices.

- Then we Use Garmin SDK and APIs to fetch the .FIT files from Garmin devices and extract fitness and health-related data from the .FIT files using the SDK's tools, simplifying data access and interaction.

- We then Parse the binary .FIT files to make the data readable and structured and convert the raw data into more accessible formats such as **CSV** or **JSON**, enabling easier manipulation, analysis, and storage. Tools that we use to do this are Garmin SDK, custom parsing scripts, or open-source libraries for FIT file parsing.

- Then we prepare the converted data for analysis by cleaning and transforming it by handling missing or incomplete data. We normalize and format the data consistently (e.g., ensuring uniform units of measurement like km or miles) and eliminating any anomalies or outliers that may skew analysis resulting in a clean dataset ready for deeper analysis.

- After the data has been cleaned and prepared, the next step is to upload it to a central, scalable, and accessible platform. This is achieved by uploading the cleaned data to Google Cloud Storage (GCS), where it can be easily accessed by the data analysis team. The data is uploaded using Google Cloud Storage's bucket.

- Once the data is securely stored, it can be analysed to uncover insights into athletic performance. Tools such as **Python** (with libraries like **Pandas** and **NumPy**) or **R** can be used to perform statistical and performance analysis on the data. The results of the analysis are then visualized using platforms like **Tableau**, **Google Data Studio**, or **Matplotlib** to create charts, graphs, and trends. This step helps identify key performance metrics, such as heart rate trends, calories burned, and distance covered, ultimately providing a detailed understanding of user fitness data and performance trends over time.

- After the analysis is complete, the findings need to be communicated effectively to stakeholders. This can be done by generating **reports** or **dashboards** based on the analysed data. These insights, along with visualizations, are shared with the team, athletes, or clients to support decision-making and performance improvements. By presenting the data in a clear and actionable format, stakeholders can easily understand and apply the insights.

Step 1: Generating FIT files from devices

.FIT files are generated from devices like Garmin or Wahoo, which contain fitness and health-related data.

Step 2: Retrieving the data from FIT files through APIs

The data from the .FIT files is retrieved using APIs provided by the Garmin SDK, which simplifies accessing and using the data.

Step 3: Converting the data into CSV or JSON format

The retrieved data is then converted into CSV or JSON format to organize it for storage and further processing.

Step 4: Pre-processing/Cleaning the data

The data is pre-processed or cleaned to ensure it is ready for analysis by handling missing values, normalizing, and ensuring data consistency.

Step 5: Uploading the data to Google Cloud Platform (GCP)

After cleaning, the data is uploaded to Google Cloud Storage, making it accessible for the data analysis team.

Steps to upload a file to Google Cloud Storage

1. **Created a Google Cloud Account**
   I started by logging into my Google Cloud account and accessed the console.
2. **Created a New Project**
   After logging in, I created a new project in the console and named it **sampleproject**.
3. **Selected the Project**
   Once the project was created, I selected **sampleproject** from the project dropdown menu to make it my active project.
4. **Navigated to Resources**
   In the sampleproject dashboard, I went into the **Resources** section to manage the project resources.
5. **Accessed the Storage Option**
   From the **Resources** section, I selected **Storage** to manage the storage resources for the project.
6. **Created a New Bucket**
   After selecting Storage, I clicked on **Create Bucket**. I configured the necessary options for the bucket and created it successfully.
7. **Uploaded the File**
   Once the bucket was created, I selected the **Upload** option, chose the file from my local device, and uploaded it to the bucket.

➪ ***Here is a video in which all the above steps regarding the bucket creation and uploading the content are performed:***
https://drive.google.com/file/d/1Pb67oumMMMA13SgKwlaRK_NI23G4T7An/view?usp=sharing

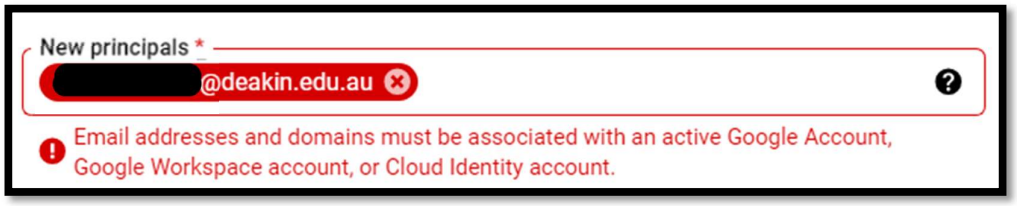
Steps to Grant Access to the Google Cloud Storage Bucket

1. **Access Google Cloud Console**:
   - I opened the **Google Cloud Console**.
   - Made sure I was working in the correct project, which in this case was the **Redback Operations project**.
2. **Navigate to IAM & Admin**:
   - I searched for **IAM** in the top search bar and clicked on the **IAM & Admin** option from the results.
   - By default, the IAM page loaded, showing all the members who already had access to the project.
3. **Open IAM Section**:
   - From the left-hand side column, I clicked on the **IAM** option (which was already selected by default).
4. **Grant Access**:
   - I clicked on the **+ Grant Access** button at the top of the page.
5. **Add New Member**:
   - In the **New Principals** section, I added the email address of **Viet Song**, the head of the Data Analysis team.
   - **Note**: Only email addresses of people who have a **Google Cloud Platform account** can be added.

     If the person you're trying to add does **not** have a Google Cloud, Google Workspace, or Cloud Identity account, you'll see this message:

     "Email addresses and domains must be associated with an active Google Account, Google Workspace account, or Cloud Identity account."

     This means the person you're adding must have one of these accounts to be granted access.



6. **Assign Role**:
   - From the **Select a Role** dropdown, I assigned **Actions Admin** to Viet so he would have the necessary permissions to manage actions.
7. **Save Changes**:
   - Finally, I clicked the **Save** button to apply the changes.

Now, **Viet Song Dang** has the role of **Actions Admin** and can handle the necessary tasks for the data analysis team.

⇨ ***Here is a video in which all the above steps regarding the granting the access are performed:***

https://drive.google.com/file/d/1DhY-jSfl2K49ZKospZKd71nX-aAWpczz/view?usp=sharing

Here in the video, I am granting access to **Daniel Greg Chen Smith**, one of my teammates who has a Google Cloud account.

⇨ ***Here is the link, through which authorized personnel can access the bucket:*** https://console.cloud.google.com/storage/browser/redback_data