Department of Computer Science & Engineering
The University of Texas at Arlington

Project Charter
CSE 4316: Senior Design I
Spring 2021

Team SpaceTabs
LifeFit

Sushant Gupta
Naveen Narayan Gnanapazham
Krishna Khadka
Puskar Dev
Shishir Pathak

Revision History

Revision	Date	Author(s)	Description
0.1	04.12.2021	SG	document creation
0.2	04.15.2021	SG, NG, KK, PD, SP	complete draft
1.0	04.16.2021	SG	official release

CONTENTS

List of Figures

List of Tables

1 Introduction

Upon completion, our product will provide State Farm customers with an android mobile and web applications to track their health information in order to calculate a corresponding fitness level. This corresponding fitness level will then be used by State Farm to potentially provide life insurance to customers with a reduced rate. The person's heart rate, sleep, exercise, and location will be tracked by our product.

1.1 Purpose And Use

Our mobile application will be similar to a fitness app and will extract data from the FitBit web API in order to track the user's health information. The user should be able to log-in, view their health information, as well as their current fitness score. The health information that will be recorded will include the person's heart rate, steps, sleep, calories burned, and how often they exercise. Our application will also have GPS capabilities in order to track the user's location to calculate distance traveled during exercise.

1.2 Intended Audience

The intended audience of our product are the customers of State Farm. Specifically, the customers that are paying for life insurance at State Farm.

1.3 Features And Functions

As in Figure 1, the user will first see a Get Started screen with "State Farm Fit" title at the top. Also shown in Figure 1 the user will be able to view their health information such as their heart rate, steps, sleep, calories burned, etc. Our application will also have GPS capabilities in order to track the user's location. The user will also be able to view their current fitness score.

1.4 Product Interfaces

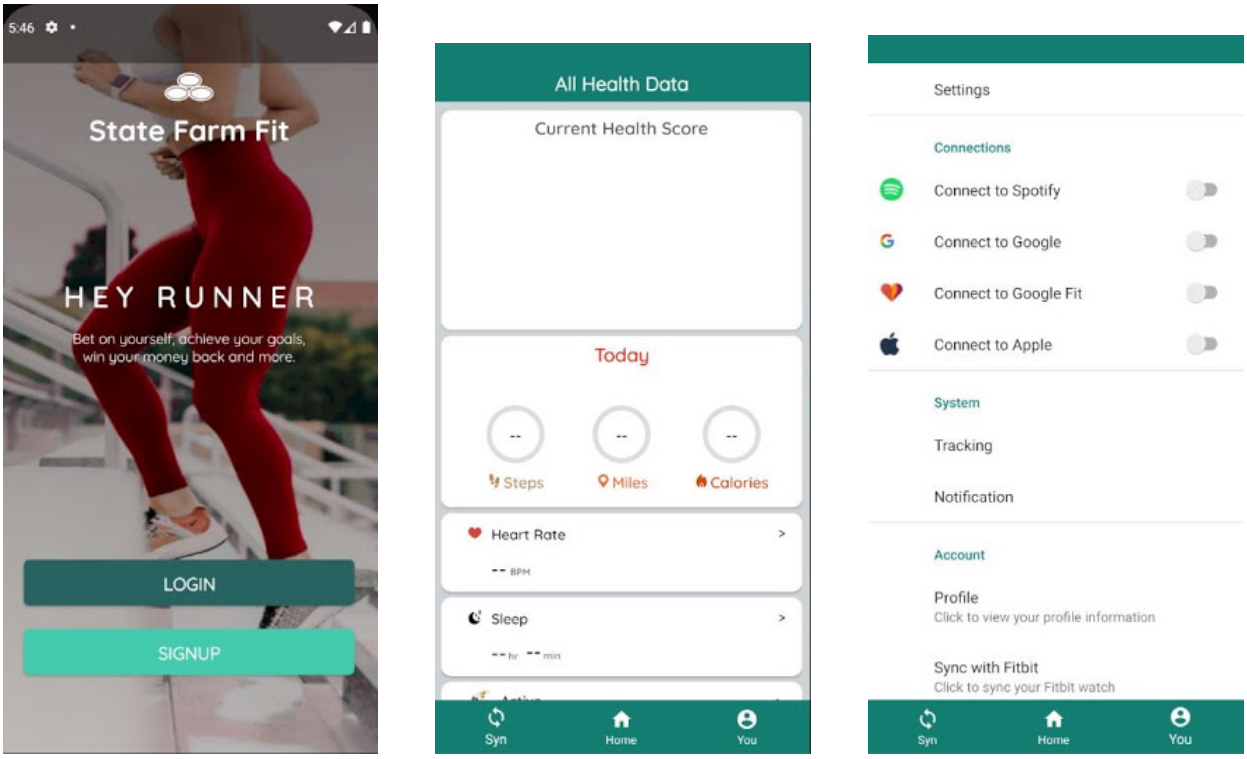

Figure 1: StateFarm Fit app Interfaces of Startup Screen, Dashboard, and Settings

2 System Overview

The structure of the service is comprised of three primary layers: Presentation, Business and Data.

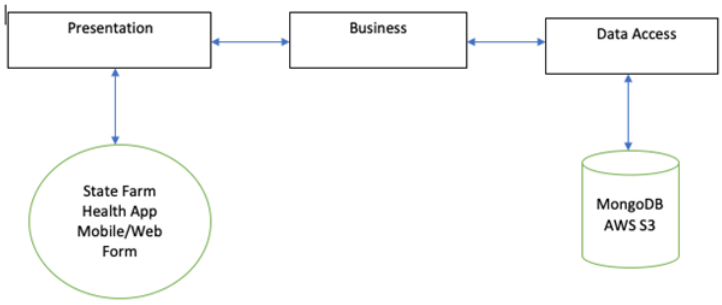

Figure 2: A simple architectural layer diagram

2.1 Presentation Layer Description

This layer will include all the presentation to the user in the form of a mobile app and a web interface. The mobile app and web interface will include login/sign up pages, screens for the user to view their health data and health status (fitness score) reports. This layer will directly capture data from the Fitbit watch and send it to the Fitbit Server which will in turn send the data back to the S3 bucket on AWS which will perform ML evaluation and send the results to the Web Dashboard and Mobile App separately.

2.2 Business Layer Description

The business layer involves the AWS S3 Bucket, ML Model, and the Fitbit Server. These 3 components are the key factors of the entire Mobile app/web interface for the Statefarm Health Project. This is where all the major computation, authentication, API Calls, Data Collection etc. occurs. Mobile App will make request to the Web Interface for health status reports, fitness score. The Web interface will sync with AWS and collect results of the ML model prediction for the Health Status prediction and collect Data from the S3 bucket to present data in a visually pleasing manner for the Users to have maximum understanding of their data. ML models like Random Forest Classifier, Ada Boosting and Bagging models will calculate a userâs health status and present them with reports of their improving or debleating health.

2.3 Data Access Layer Description

Data Access Layer will have AWS S3 Bucket for user Fitness Data Collection and MongoDB for user credential. This layer will interact with the Business layer. It will send and receive data from Mobile App and Web Server. The user subsystem will be where the user credentials are verified. The administrator subsystem will be where the administratorâs credentials are verified. The fitness data collected from the Fitbit Watch will be sent to Fitbit Server which in turn will be collected in the AWS S3 bucket where ML Models will work on Analysis of health data and Communicating individual data for particular user for Health Data visualization.

3 SUBSYSTEM DEFINITIONS & DATA FLOW

The data flow diagram shown below gives us clear picture of the flow of data from capturing it to process and saving it. It is a multi-tier system involving various segments of tech. API Calls, Syncs, UI/UX, Remote servers, User Login/Authentication, Cloud-Storage, ML Models, and some hardware (The Fitbit Watch). The Presentation layer has Mobile App Interface, Web Interface (WI) and Syncing Calls made to the watch. The mobile app and WI both display key data analysis and present the user with data visualization dashboards. The Fitbit Watch for example captures key features like heart rate, location, activities, calories burned, mets and sleep data. This Data will be stored on the Fitbit Server which we have to make a request from the Web App which will return the data. From there this data will be sent to S3 for Storage and for ML Models to work on Predicting the Userâs health Status and other important metric. This result will be sent to the Web Application and Shared with Web Users of the Statefarm Health App. The Web app will be shared with the Mobile App too, which will be presented in a pleasurable fashion to the users. The User Login and authentication for both Web/Mobile platforms will be controlled by MongoDB.

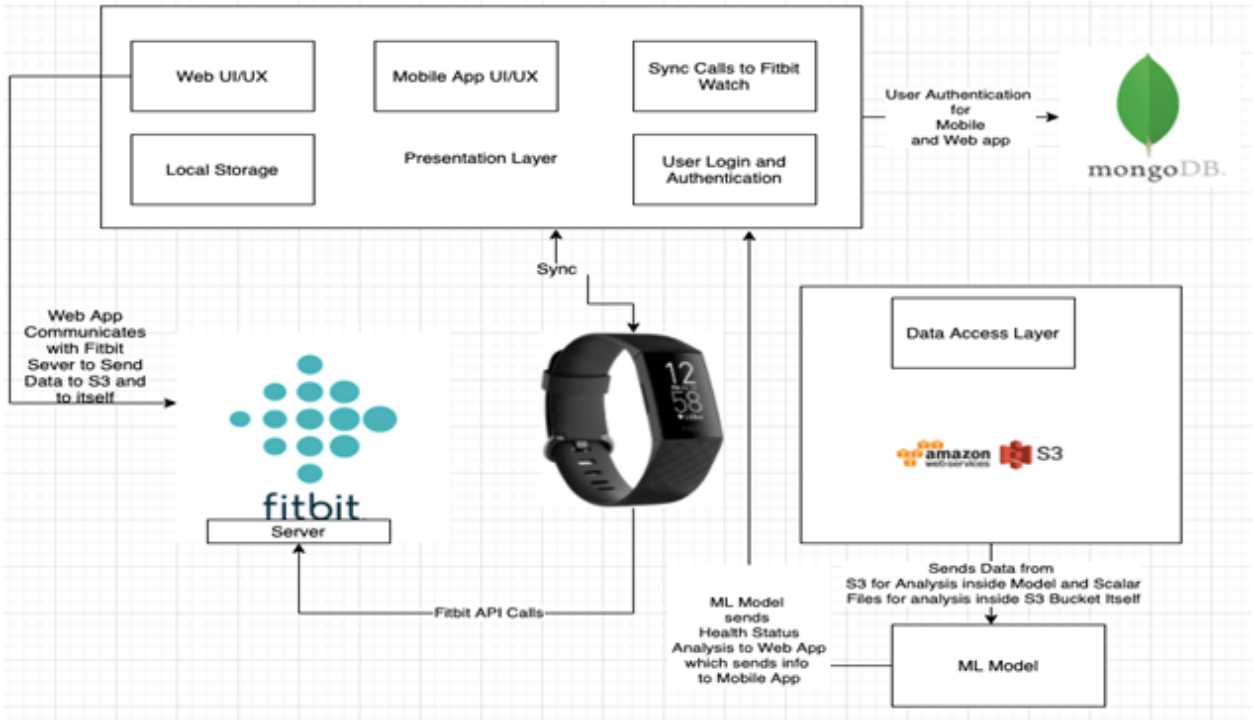

Figure 3: A simple data flow diagram

4 Presentation Layer

The Presentation layer consists of three subsystems consisting of Capture Subsystem, Android UI Subsystem, and Web UI Subsystem which runs on the Android device. The Capture system in Android device collects user location, heart rate, activities, sleep data, and other health related data of the users. The Android UI System allows user to connect to Fitbit device, login, register, modify their account and review their health status in the UI. The Web UI System also allows user to login, register, modify account and review health status in the web browser.

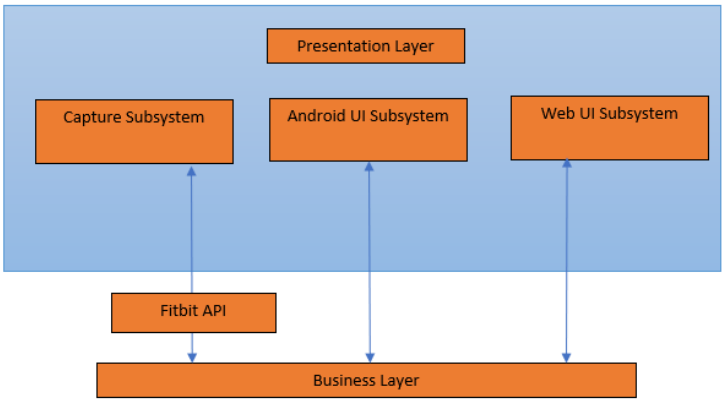

Figure 4: Presentation Layer

4.1 Capture Subsystem

Capture system runs on android device to collect data from the Fitbit API including heart rate, location, activities, sleep data and other health related data. The collected raw data will be stored on the phoneâs file system and the business layer will later upload the file system to the Mongo database in the AWS S3 bucket.

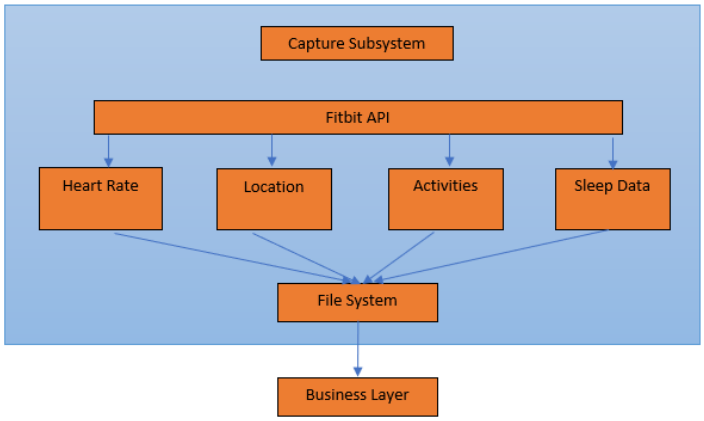

Figure 5: Capture System

4.1.1 Assumptions

- Assumption 1: User will use a Fitbit smart watch.

- Assumption 2: Data collected from the user will be stored in files on the file system in Android device.
- Assumption 3: A subsystem in the business layer will synchronize data between the file system and AWS S3 bucket.

4.1.2 Responsibilities

This subsystem is responsible to collect user health data from the Fitbit device and store the data on file system in the connected Android device.

4.1.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
01.	Collect the userâs data from Fitbit API	Username	User health data

Table 2: Subsystem interfaces

4.2 Android UI Subsystem

Android UI system running in android device allows user to view their health data collected from AWS S3 bucket and manage their account.

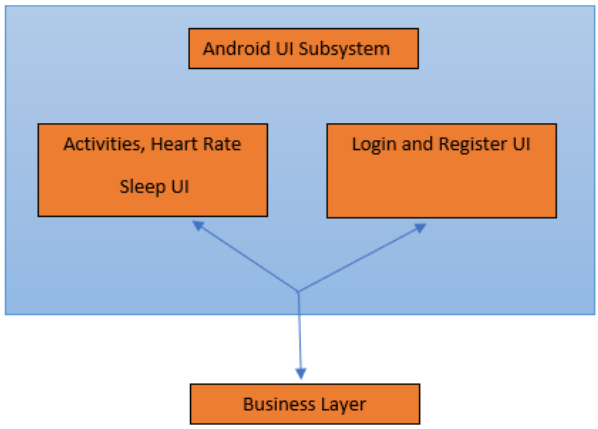

Figure 6: Android UI System

4.2.1 Assumptions

- Assumption 1: User can use this subsystem in android device and can view data when the device is online.
- Assumption 2: All the data will be stored in the file system.
- Assumption 3: A subsystem in the business layer will synchronize data between the file system and AWS S3 bucket.

4.2.2 Responsibilities

This subsystem is responsible to show user data in the UI and allow user to manage their account. The subsystem uses the local file system to display the data collected.

4.2.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
02.	Display userâs Fitbit data	Date	User health data

Table 3: Subsystem interfaces

4.3 Web UI Subsystem

Web UI subsystem is hosted on AWS server. The users and administrators will be able to view health data, health score, and account information of the users.

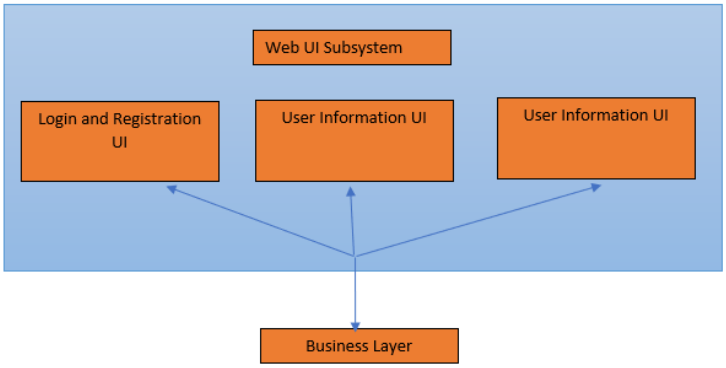

Figure 7: Web UI System

4.3.1 Assumptions

- Assumption 1: User can use this subsystem view their information in the web browser from AWS S3 bucket.
- Assumption 2: Administrator can use this subsystem to view the user information in the web browser from AWS S3 bucker

4.3.2 Responsibilities

This subsystem is responsible to show user data in the web browser to the users and administrators. The subsystem retrieves the usersâ data from the AWS S3 bucket.

4.3.3 SUBSYSTEM INTERFACES

ID	Description	Inputs	Outputs
03.	Login Interface	Username / password	Web Token
04.	Register Interface	User information	User Account
05.	Display User Information Interface	User id	User Information

Table 4: Subsystem interfaces

5 Data Access Layer

The main focus of this layer is to store data and to provide access to stored data. This layer is composed of four subsystems which are User, Administrator, Raw data and Analyzed data.

5.1 User

The User layer is where the credentials of the user are verified.

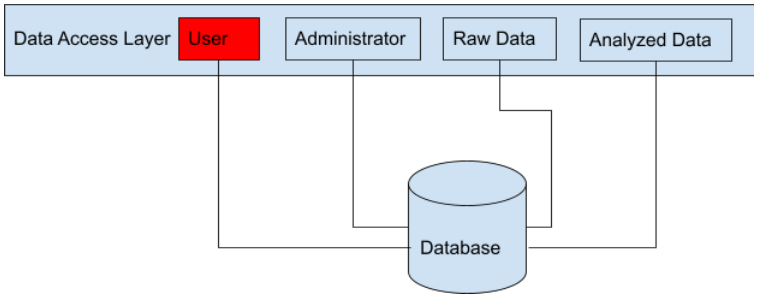

Figure 8: User Subsystem

5.1.1 Assumptions

- Assumption 1: User can access Web UI.
- Assumption 2: A subsystem in the business layer helps in data transfer between presentation to data access layer.
- Assumption 3: The layer has a proper connection with the database.

5.1.2 Responsibilities

The subsystem is responsible for verifying the credentials provided during sign in process with the stored credentials.

5.1.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
#06	Verify user login credentials	Username and Password	Boolean Value

Table 5: User Interfaces

5.2 Administrator

The administrator layer is used by the admin panel where the admin credential is verified.

5.2.1 Assumptions

- Assumption 1: Admin can access the Web UI.
- Assumption 2: A subsystem in the business layer helps in data transfer between presentation to data access layer.
- Assumption 3: The layer has a proper connection with the database.

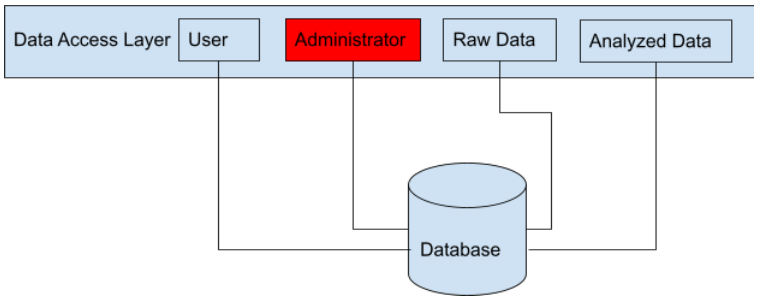

Figure 9: Admin Subsystem

5.2.2 Responsibilities

The subsystem is responsible for verifying the credentials provided during sign in process with the stored credentials.

5.2.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
#07	Verify user login credentials	Username and Password	Boolean Value

Table 6: Administrator Interfaces

5.3 Raw Data

In this layer, the communication between the raw data captured by FitBit watches in the presentation layer happens, which is then transferred through the business layer to the online database, MongoDB, which then can be viewed by the administrator.

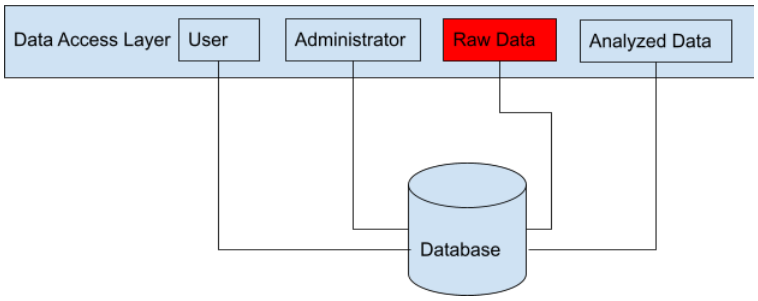

Figure 10: Raw Data Subsystem

5.3.1 Assumptions

- Assumption 1: User uses FitBit data.
- Assumption 2: Subsystem in business layer synchronizes data between user phone and online database.
- Assumption 3: Proper connection setup with the database.
- Assumption 4: Captured raw data is not modified.

5.3.2 Responsibilities

The data extracted from FitBit users are placed into the server database.

5.3.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
#08	Save user health data	Health Data	N/A
#09	Retrieve user health data	User ID	Health Data

Table 7: Raw Data Interfaces

5.4 Analyzed Data

The main purpose of this layer is to communicate the analyzed data such as fraud activity, fitness score, and risk factor from a subsystem of the business layer.

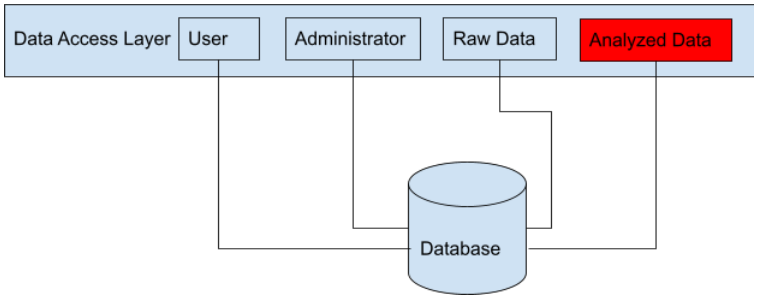

Figure 11: Analyzed Data Subsystem

5.4.1 Assumptions

- Assumption 1: A subsystem in the business layer which is responsible for producing analyzed data and then transferring to the Data access layer.
- Assumption 2: Proper connection with the AWS and MongoDB

5.4.2 Responsibilities

This subsystem is responsible for communicating the analyzed data such as, fraud activity, fitness score, and risk factor from a subsystem of the business layer, AWS to online database, MongoDB.

5.4.3 Subsystem Interfaces

ID	Description	Inputs	Outputs
#10	Analyze user health data	Raw Health Data	Analyzed Health Data

Table 8: Analyzed Data Interfaces

6 Business Layer

The main objective of the business layer is to host business logic and allow users to access data and its interpretation in multiple ways. Business layers comprises four major components which are AWS E2 instance, AWS S3 Bucket, Machine Learning Models, and the API gateways. This layer is responsible for all the major computations, user authentication, API Calls, Data Collection etc.

6.1 AWS EC2 Instance

EC2 instance is a virtual machine/server in Amazon Web services that will allow us to host our server and get the necessary computing resources required to run the business logic.

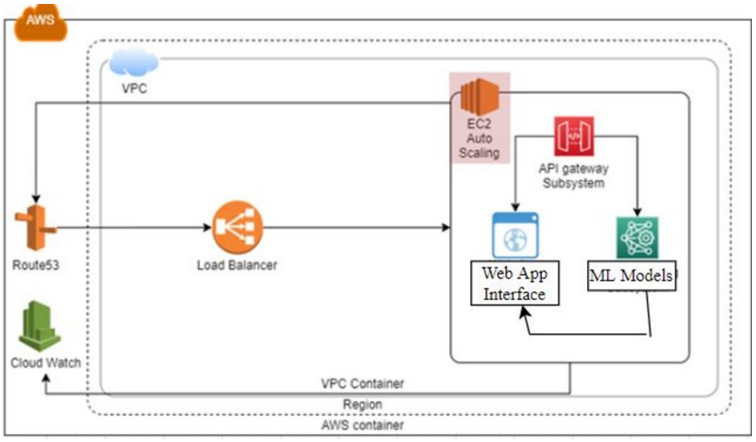

Figure 12: AWS System Diagram

6.1.1 Assumptions

- Assumption 1: AWS ELB will be used to create AWS EC2 instances.
- Assumption 2: AWS ELB will be able to auto scale (using EC2 Auto Scaling) feature.

6.1.2 Responsibilities

EC2 instance is responsible for hosting the server, running docker environments, and scale automatically as the server need grows.

6.1.3 Route 53

Route 53 connects user requests to infrastructure running in AWS â such as Amazon EC2 instances, load balancers, or Amazon S3 buckets. This is the EC2 interface. It inputs a HTTP/HTTPS request from a user and sends HTTP/HTTPS response.

6.2 API Gateway

This will source a primary point of contact to the VPC (Virtual Private cloud) through HTTP/HTTPS traffic routed via Route53. This service will be responsible for handling all requests sent to the server.

6.2.1 Responsibilities

All requests resolved from Route53 to the server will be handled by this service. It will be responsible for managing the API requests from Android app and Web app to handle user operations such as creating accounts (login/signup), Data analysis on Fitbit data and render score. This component is also responsible to interact with the Data access layer to retrieve the user data for analysis.

6.2.2 API Calls

The server receives API calls through various interfaces such as: Android app, Web App, Machine Learning interface to pull data for analysis and Data Access Interface to pull the data from FitBIt server. The type of requests and response for these API calls will be in JSON format.

6.3 Web Application

This will source a primary point of contact to the VPC (Virtual Private cloud) through HTTP/HTTPS traffic routed via Route53. This service will be responsible for handling all requests sent to the server.

6.3.1 Assumptions

- Assumption 1: Admin access may be required (if logged in as Admin)

6.3.2 Responsibilities

The main responsibility of the web application is to provide the user and admins to access various information including userâs health score, reasoning behind score update, userâs data visualization in charts and graphs, userâs health statistics and recommendations. Admin will be able to access system status, data analytic on all users.

6.3.3 Web Application API Calls

Training data and testing data will be separated. Training data will be used to train the model and testing to test the model. Various ML models will be used and the model with highest test accuracy will be selected. The ML models will classify a user and assign a health score. Score reasoning feature will be implemented that will allow to know the user why their score changed.

6.4 Data Analysis

This stack will be responsible for analyzing the user data. Machine Learning models such as Random Forest Classifier, Ada Boosting and Bagging models will be used to classify the user and assign a health score.

6.4.1 Assumptions

- Assumption 1: ML libraries to run the model will be imported beforehand.
- Assumption 2: Data is already present in the S3 Bucket and ready for retrieval.

6.4.2 Responsibilities

Training data and testing data will be separated. Training data will be used to train the model and testing to test the model. Various ML models will be used and the model with highest test accuracy will be selected. The ML models will classify a user and assign a health score. Score reasoning feature will be implemented that will allow to know the user why their score changed.

References