Beats

Cameron Keene
Dillan Maraj
Thomas Pena
Charles Richardson

CIS4301 - Dr. Markus Schneider

University of Florida

Table of Contents

The overview and description of the application

Beats serves as a digital health tool for users seeking personalized insight into their daily routine. Beats's main purpose is to visualize the user's health vitals overlayed with their daily schedule. Inputs to Beats include health data coming from the user's wearable devices; as well as their digital calendar data, which would be manipulated and accessed from the user's digital devices. The calendar data would be simplified to improve finding trends, and the app will maintain synchronization with the calendar as data is manipulated or added.

The output of Beats is all visual. Primarily graphical representations of the trends revealed from the data. These outputs can be modified to specific timeframes, activities, and heart rate (HR) ranges. To ensure the user has the most seamless experience with Beats, it will be designed to work in a familiar environment to the user. Using the web to host this application is the most straightforward way of doing this. Furthermore, to maximize the efficiency of Beats and the productivity of the project, the simplest and most intuitive development stack will be used, explained later in this report. Ultimately, this application will improve the lives of its users by providing them with valuable personalized health data.

The motivation of the database needs of the application and the potential user interest in Beats

Databases are useful for handling large sets of structured data that need to be accessed methodically. The short sampling intervals necessary to produce clear pictures of user activity in our application present the need for this organization and ease of access in what will become a large set of data points. To draw trends among users, and perhaps derive more universal insights, complex trend queries will be utilized to make inferences beyond the single HR points for individuals.

While the majority of this team is composed of fitness-focused individuals, the world of personal health technology is a burgeoning industry that appeals to Ironman triathletes and average Joes alike. Our application can be used actively to optimize fitness, schedules, or passively to alert users to the possibility of early-onset health problems and to better inform themselves of the effects of their actions throughout the day. The users of this application are simply interested in their HR and what it can relay to them about their current health and fitness.

The description of the needed web-based user interface functionality

The web application will be developed using two different frameworks. By adopting this workflow the team will simplify the interaction between users and the management of the Oracle database. For the frontend, the team will rely on React to manage users interaction and data visualization. For the backend, the team will use the python-based Django framework that can be connected to the Oracle database.

This application will manage individuals' HR data and calendar events information, meaning the frontend will need to allow users to access their individual information. To achieve this, Beats will have a user authentication system that utilizes their google account, thus Beats will be able to access personalized calendar events. Moreover, users will also be able to sync their HR data, this can be achieved by connecting users' smartwatches or by uploading files in a specific format that allows Beats to read and store the HR data and timestamps to the database.

The web app will also be able to display interactive graphs that illustrate the different calendar events and the corresponding average, lowest and highest HR data. With detailed data on each event, users will have access to events that generate a spike in their HR and events that are impacting the stress level of users. Users will also have access to a graphical representation of the HR trend during the day, including physical activities, daily activities, and sleeping HR data. Each of these activities will be given to users to better plan their day and choose what kind of activities are better suited at a specific time of the day. With the users' calendar events, HR data, and the corresponding trend queries the web app will be able to simply and graphically represent data that can be used to optimize day planning.

The description of the application goals regarding trend analysis

The main goal of the project is to give users an insight into how their body vitals, strictly speaking, their heart rate adjusts to their schedule. With detailed information on how the heart rate adjusts to specific events at specific times of the day. Users will be able to better understand their days and optimize not only their routines, but also energy levels, sports performance, and detect what activities generate stress and what activities help them recover and reduce stress levels. We aim to successfully generate personalized trends that adjust to each user. Using their vitals we aim to detect how the activities we regularly perform impact our body, performance, mood, and stress.

We are looking to identify and classify activities by using heart rate data, meaning that each user's activity can give us details on how the body reacts. By analyzing physical activities, sleep schedules, and daily activities, trends regarding the length and time of day of each activity can be generated. Based on this, we also look to generate a recovery score that can tell users what events or routines better fit their bodies. Finally, we can analyze trends such as stress levels by identifying activities that spike heart rate activity.

The description of the real-world data forming the basis of the application and the complex trend queries

The real-world data relevant to this project is unique to each user. No two users will have the same data simply due to the nature of biology and the chaotic variability of human behavior. Data belonging to any particular user could range from highly volatile to extremely stable based on a variety of factors, however, this makes for a more diverse outcome, as well as more interesting computed trends. The specific data sources are as follows:

UserID

Each data point will belong to one specific user. There will not be any data in the database which belongs to multiple users, all users, or no users. This field will be required for every new entry, and consist of an integer.

User Name

To make Beats personable, the name of the user will be stored. This will be used to improve the UI by displaying the visuals and the name nearby, to ensure to the user that the data they are seeing is their own.

Timestamp

Each of the following data points will have a log of when it was recorded in real-time. Having this information tied to each tuple will be the foundation on which trends are found. This will also make manipulation of the visuals much easier for users. The specific format of the timestamp remains unknown, however, it will conform to a standard, making all timestamps syntactically similar. Once added to the database, this attribute is immutable. It will never be necessary to change the timestamp of a data point. This will be the primary key.

Absolute HR

The most relevant and sensitive data to the user is their HR. This will be an integer with a minimum of 0 and an upper limit of 300. The upper limit is much higher than the usual maximum HR of an average user, however, there is a possibility for outliers, such as users with abnormal HRs and underlying health conditions. Similar to the timestamp, this data is immutable. Changing a recorded HR field would invalidate the sample, as it is impossible to “correct” a HR reading that was recorded in the past by a single device.

Scheduled Activity

As a user navigates through their day, they will undergo routine activities such as eating and attending classes, which will affect their HR. To limit the complexity of the data, this attribute will be limited to several fixed values. The values are:

- Work/School
- Fitness
- Social
- Sleep/Rest
- Eating
- Other

These 6 generalized activities account for all possible activities a user could be participating in and simplify the visualization of data in Beats. A downside to limiting the activities to 6 options is the fact that some activities may be more physically demanding than others. Using this generalized set, the context of the activity is lost. For example, attending a neighbors bonfire and going to the club would both classify as a *social* activity. At the bonfire, the user's HR could be 70 beats per minute, whereas, at the club, their HR could be 140 beats per minute. This large range could lead to unwanted results.

Scheduled Activity Start Time

Since there can be many activities starting at the same time, it is necessary to include a field for the start time of each activity rather than depending on the timestamp attribute. For example, a scheduled *school* activity can start at 12:00pm, and at the same time, the user could be *eating*. Therefore, each activity must have its start time attribute. The format of this attribute remains undetermined, but chances are it will be the same format as the timestamp

Scheduled Activity Duration (or End Time)

There is no fixed value for each activity. Some activities may last for 30 minutes, while others could last for 5 hours. The particular value of this attribute is yet to be determined. It could be an integer, representing the number of seconds that the activity spans, or it could be a timestamp format, and the duration could be computed by taking the difference in end time with the start time.

Five colloquial complex trend queries and their explanation

Avg/High/Low for each class of activity

Based on the corresponding categories of classes the average, the high, and the low HR will be calculated and displayed to the user. The user will also be allowed to specify the time scale of the HRs. Allowing the user to pick from a selection of default time scales or define their own, showing modified trends based on the specific time scale. The HRs will be displayed on the y-axis and time will be displayed on the x-axis. This charting trend will follow throughout the rest of the trends.

Effect of activity sequencing

This trend will address the effect of activity sequencing. Showing the different effects of what you do in a day and how it affects the rest of your day. An Example of this is the position of a workout relative to the rest of the activities in your day. Determining if there's a difference between doing a workout first thing in the morning or at night. This can be extrapolated across the rest of the classes to see the effects different activities have on the rest of your daily activities.

Duration of activity to demand on HR

This trend aims to examine the trend between the time of an activity and the impact it has on HR. Users will be able to determine the effect that the duration of an activity has on their HR. This will also be extrapolated across the different activities to allow users to draw trends between activities and also allow them to help in the sequencing of activities. An example of this can be seen when watching the trend in fitness activity and increasing in periods. As the overall duration of a fitness activity increases cardiac creep starts to be induced, cardiac creep refers to the increase in HR over time even if the effort level of the user remains the same.

Sleep schedule effects on HR

This trend is used to show trends in the details of a sleep schedule and the effect on the HR of the user. The amount of sleep that a user gets has a large impact on their health. Changes to a user's sleep pattern such as the duration of sleep or the times that they were asleep could potentially have an impact on their HR, more specifically their resting HR. An example of this could be that a decrease in the overall time asleep could result in an elevated resting HR or sleeping too much could also result in a decrease in resting HR.

Recovery score based on a moving average resting HR.

This trend will be based on calculating a recovery score for each user. The recovery score will be based on a moving average resting HR trend. This will be a moving window that is continually updated to reflect the HRs based within a certain period, for example, a two-week window. The HR score will be based upon the average resting HR in a given time and that day's resting HR data. The recovery score will be used to quantify the level of tiredness that a user might be feeling. This will then be made into trends over specified periods to analyze if a user is maintaining, downwards trending, or upwards trending with their recovery score. An example of a downwards trending recovery score could be if a user has a week of exams. They could be stressed over their exam and not get a lot of sleep. The combination of these two events could lead to an overall downtrend in their recovery score.

Acute stress relationship in daily activities

This trend will be used to determine the relationship between acute stress levels and their daily activity. Different activities that a user performs will have different impacts on their overall stress level. Users can map different activities to determine which activities induce stress or which activities relieve stress. An example of this could be a user going for a walk, which will fall under the fitness category. After the walk, the stress level of the user could be decreased. Thus, showing a positive increase in fitness and stress.

The description of the intended use of public domain and/or proprietary software

We determined the best software to use will be a combination of Python and ReactJS. Specifically, we will use ReactJS for the front-end of the website, combined with the Django framework on top of python to manage the backend of the project. For the database, we will use the required software, CISE Oracle SQL Database, which is provided for us. To graph the data, a graphing library such as canvasJS would be used to create interactive and informative charts.

To get all of the data for the project we will use a multitude of sources. The first source is Apple HealthKit data, which can be achieved by exporting user profiles from the health app on Apple Devices. Once exported we parse the XML file and look for the Apple Healthkit data relating to HR. Another data source that we would use to grab health data would be using the Garmin Connect API. To collect the calendar data we would use the following APIs: Outlook calendar API, Google Calendar API, and the Apple Eventkit API to collect calendar info.