

Case Study: How Can a Wellness Technology Company Play It Smart?

Allan C Mathews

21 October 2021



bellabeat

1 Summary

Bellabeat is a high-tech manufacturer of health-focused smart products for women. The Bellabeat app provides users with health data related to their activity, sleep, stress, menstrual cycle, and mindfulness habits. This data can help users better understand their current habits and make healthy decisions and connects to their line of smart wellness products.

1.1 Objective

To focus on one of Bellabeat's products and analyze the smart device data to gain insight into how consumers are using their smart devices and with these insights could guide the marketing strategy for the company.

To answer the key business questions, **the six steps of the data analysis process: ask, prepare, process, analyze, share, and act** will be followed and explained step by step.

2 Ask phase :Ask Questions to Make Data-Driven Decisions

2.1 Business Task

- Identify trends in smart device usage
- How could these trends apply to Bellabeat customers?
- How could these trends help influence Bellabeat marketing strategy?

2.2 Key Stakeholders

- **Urška Sršen:** Bellabeat's cofounder and Chief Creative Officer
- **Sando Mur:** Mathematician and Bellabeat's cofounder
- **Bellabeat marketing analytics team:** A team of data analysts responsible for collecting, analyzing, and reporting data that helps guide Bellabeat's marketing strategy.

3 Prepare phase: Prepare Data for Exploration

3.1 Data source

FitBit Fitness Tracker Data (<https://www.kaggle.com/arashnic/fitbit>), This dataset is stored in Kaggle as a public domain and was made available through Mobius (<https://www.kaggle.com/arashnic>).

3.2 Accessibility and privacy of data

This dataset is open-sourced and were generated by respondents to a distributed survey via Amazon Mechanical Turk between 03.12.2016-05.12.2016. Thirty eligible Fitbit users consented to the submission of personal tracker data, including minute-level output for physical activity, heart rate, and sleep monitoring. It includes information about daily activity, steps, and heart rate that can be used to explore users' habits. Individual reports can be parsed by export session ID (column A) or timestamp (column B). Variation between output represents use of different types of Fitbit trackers and individual tracking behaviors / preferences.

3.3 Data Integrity

The dataset contains only data of thirty fitbit users without any demographic details,hence it cannot be determined if there is any sampling bias. Another problem we would encounter is that the dataset is not current and has only 2 months period data. That is why we will give our case study an operational approach

```
# Read Libraries here
library(tidyverse) #for data visualization and data manipulation
library(skimr) # to use functions provides a larger set of statistics
library(lubridate) # to work with date and time objects
```

```
# Read the data here
daily_activity <- read.csv('Fitabase_Data/dailyActivity_merged.csv')
sleep_log <- read.csv("Fitabase_Data/sleepDay_merged.csv")
weight_log <- read.csv("Fitabase_Data/weightLogInfo_merged.csv")
glimpse(daily_activity)
```

```
## Rows: 940
## Columns: 15
## $ Id <dbl> 1503960366, 1503960366, 1503960366, 150396036~<br>
## $ ActivityDate <chr> "4/12/2016", "4/13/2016", "4/14/2016", "4/15/~<br>
## $ TotalSteps <int> 13162, 10735, 10460, 9762, 12669, 9705, 13019~<br>
## $ TotalDistance <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~<br>
## $ TrackerDistance <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~<br>
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~<br>
## $ VeryActiveDistance <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~<br>
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~<br>
## $ LightActiveDistance <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~<br>
## $ SedentaryActiveDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~<br>
## $ VeryActiveMinutes <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~<br>
## $ FairlyActiveMinutes <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~<br>
## $ LightlyActiveMinutes <int> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~<br>
## $ SedentaryMinutes <int> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~<br>
## $ Calories <int> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203~
```

```
glimpse(sleep_log)
```

```
## Rows: 413
## Columns: 5
## $ Id          <dbl> 1503960366, 1503960366, 1503960366, 1503960366, 150~
## $ SleepDay    <chr> "4/12/2016 12:00:00 AM", "4/13/2016 12:00:00 AM", "~"
## $ TotalSleepRecords <int> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~
## $ TotalMinutesAsleep <int> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~
## $ TotalTimeInBed   <int> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~
```

```
glimpse(weight_log)
```

```
## Rows: 67
## Columns: 8
## $ Id          <dbl> 1503960366, 1503960366, 1927972279, 2873212765, 2873212~
## $ Date        <chr> "5/2/2016 11:59:59 PM", "5/3/2016 11:59:59 PM", "4/13/2~
## $ WeightKg    <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, ~
## $ WeightPounds <dbl> 115.9631, 115.9631, 294.3171, 125.0021, 126.3249, 159.6~
## $ Fat          <int> 22, NA, NA, NA, NA, 25, NA, NA, NA, NA, NA, NA, NA, ~
## $ BMI          <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25, ~
## $ IsManualReport <chr> "True", "True", "False", "True", "True", "True", "True"~
## $ LogId        <dbl> 1.462234e+12, 1.462320e+12, 1.460510e+12, 1.461283e+12, ~
```

4 Process phase: Process Data from Dirty to Clean

4.1 Data Cleaning

4.1.1 Data Formatting

- Format Id column from numeric to character type
- Format Date column from character to Date type

```
# Formatting
# Format Id column to character type
daily_activity <- transform(daily_activity,
                           Id = as.character(Id))
sleep_log <- transform(sleep_log,
                       Id = as.character(Id))

weight_log <- transform(weight_log,
                        Id = as.character(Id))
# Format Date column from character to Date type
daily_activity$ActivityDate <- as.Date(daily_activity$ActivityDate,format='%m/%d/%Y')
sleep_log <- sleep_log %>%
  separate(SleepDay, into=c("Date","SleepTime"), sep=" ",remove =TRUE )
sleep_log>Date <- as.Date(sleep_log>Date,format='%m/%d/%Y')
weight_log <- weight_log %>%
  separate(Date, into=c("Date","Time"), sep=" ",remove =TRUE )
weight_log>Date <- as.Date(weight_log>Date,format='%m/%d/%Y')
# Taking only Id,Date,WeightKg,BMI,IsManualReport fields
weight_log <- weight_log %>%
  select(Id,Date,WeightKg,BMI,IsManualReport)
# Summary of the dataset after formatting
skim_without_charts(daily_activity)
```

Data summary

Name	daily_activity
Number of rows	940
Number of columns	15

Column type frequency:

character	1
Date	1
numeric	13

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0	1	10	10	0	33	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
ActivityDate	0	1	2016-04-12	2016-05-12	2016-04-26	31

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	po	p25	p50	p75	p100
TotalSteps	0	1	7637.91	5087.15	0	3789.75	7405.50	10727.00	36019.00
TotalDistance	0	1	5.49	3.92	0	2.62	5.24	7.71	28.03
TrackerDistance	0	1	5.48	3.91	0	2.62	5.24	7.71	28.03
LoggedActivitiesDistance	0	1	0.11	0.62	0	0.00	0.00	0.00	4.94
VeryActiveDistance	0	1	1.50	2.66	0	0.00	0.21	2.05	21.92
ModeratelyActiveDistance	0	1	0.57	0.88	0	0.00	0.24	0.80	6.48
LightActiveDistance	0	1	3.34	2.04	0	1.95	3.36	4.78	10.71
SedentaryActiveDistance	0	1	0.00	0.01	0	0.00	0.00	0.00	0.11
VeryActiveMinutes	0	1	21.16	32.84	0	0.00	4.00	32.00	210.00
FairlyActiveMinutes	0	1	13.56	19.99	0	0.00	6.00	19.00	143.00
LightlyActiveMinutes	0	1	192.81	109.17	0	127.00	199.00	264.00	518.00
SedentaryMinutes	0	1	991.21	301.27	0	729.75	1057.50	1229.50	1440.00
Calories	0	1	2303.61	718.17	0	1828.50	2134.00	2793.25	4900.00

```
skim_without_charts(sleep_log)
```

Data summary

Name	sleep_log
Number of rows	413
Number of columns	6

Column type frequency:

character	2
Date	1
numeric	3

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0	1	10	10	0	24	0
SleepTime	0	1	8	8	0	1	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0	1	2016-04-12	2016-05-12	2016-04-27	31

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
TotalSleepRecords	0	1	1.12	0.35	1	1	1	1	3
TotalMinutesAsleep	0	1	419.47	118.34	58	361	433	490	796
TotalTimeInBed	0	1	458.64	127.10	61	403	463	526	961

```
skim_without_charts(weight_log)
```

Data summary

Name	weight_log
Number of rows	67
Number of columns	5

Column type frequency:

character	2
Date	1
numeric	2

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0	1	10	10	0	8	0
IsManualReport	0	1	4	5	0	2	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0	1	2016-04-12	2016-05-12	2016-04-27	31

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
WeightKg	0	1	72.04	13.92	52.60	61.40	62.50	85.05	133.50
BMI	0	1	25.19	3.07	21.45	23.96	24.39	25.56	47.54

4.1.2 Clean and Manipulate

- Find and remove duplicated records
- Rename ActivityDate column to **Date**
- Create a new column with **Weekdays**

```
#Find duplicates
```

```
sum(duplicated(daily_activity))
```

```
## [1] 0
```

```
sum(duplicated(sleep_log))
```

```
## [1] 3
```

```
sum(duplicated(weight_log))
```

```
## [1] 0
```

```
#remove duplicates
sleep_log<-unique(sleep_log)

#Rename ActivityDate column to Date

colnames(daily_activity)[colnames(daily_activity) == "ActivityDate"] <- "Date"

#Add new column with weekdays

daily_activity$Weekdays<- wday(daily_activity$date, label = TRUE, abbr = FALSE)

#Verify changes made

glimpse(daily_activity)
```

```
## Rows: 940
## Columns: 16
## $ Id              <chr> "1503960366", "1503960366", "1503960366", "15~ 
## $ Date            <date> 2016-04-12, 2016-04-13, 2016-04-14, 2016-04-~ 
## $ TotalSteps       <int> 13162, 10735, 10460, 9762, 12669, 9705, 13019~ 
## $ TotalDistance    <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~ 
## $ TrackerDistance  <dbl> 8.50, 6.97, 6.74, 6.28, 8.16, 6.48, 8.59, 9.8~ 
## $ LoggedActivitiesDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~ 
## $ VeryActiveDistance   <dbl> 1.88, 1.57, 2.44, 2.14, 2.71, 3.19, 3.25, 3.5~ 
## $ ModeratelyActiveDistance <dbl> 0.55, 0.69, 0.40, 1.26, 0.41, 0.78, 0.64, 1.3~ 
## $ LightActiveDistance   <dbl> 6.06, 4.71, 3.91, 2.83, 5.04, 2.51, 4.71, 5.0~ 
## $ SedentaryActiveDistance <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~ 
## $ VeryActiveMinutes     <int> 25, 21, 30, 29, 36, 38, 42, 50, 28, 19, 66, 4~ 
## $ FairlyActiveMinutes    <int> 13, 19, 11, 34, 10, 20, 16, 31, 12, 8, 27, 21~ 
## $ LightlyActiveMinutes   <int> 328, 217, 181, 209, 221, 164, 233, 264, 205, ~ 
## $ SedentaryMinutes       <int> 728, 776, 1218, 726, 773, 539, 1149, 775, 818~ 
## $ Calories             <int> 1985, 1797, 1776, 1745, 1863, 1728, 1921, 203~ 
## $ Weekdays             <ord> Tuesday, Wednesday, Thursday, Friday, Saturday~
```

```
#Add new column with weekdays

sleep_log$Weekdays<- wday(sleep_log$date, label = TRUE, abbr = FALSE)

#Verify Sleep Day in date format

glimpse(sleep_log)
```

```
## Rows: 410
## Columns: 7
## $ Id              <chr> "1503960366", "1503960366", "1503960366", "15039603~ 
## $ Date            <date> 2016-04-12, 2016-04-13, 2016-04-15, 2016-04-16, 20~ 
## $ SleepTime        <chr> "12:00:00", "12:00:00", "12:00:00", "12:00:00", "12~ 
## $ TotalSleepRecords <int> 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ~ 
## $ TotalMinutesAsleep <int> 327, 384, 412, 340, 700, 304, 360, 325, 361, 430, 2~ 
## $ TotalTimeInBed    <int> 346, 407, 442, 367, 712, 320, 377, 364, 384, 449, 3~ 
## $ Weekdays          <ord> Tuesday, Wednesday, Friday, Saturday, Sunday, Tuesd~
```

```
#Add new column with weekdays

weight_log$Weekdays<- wday(weight_log$date, label = TRUE, abbr = FALSE)

#Verify changes made

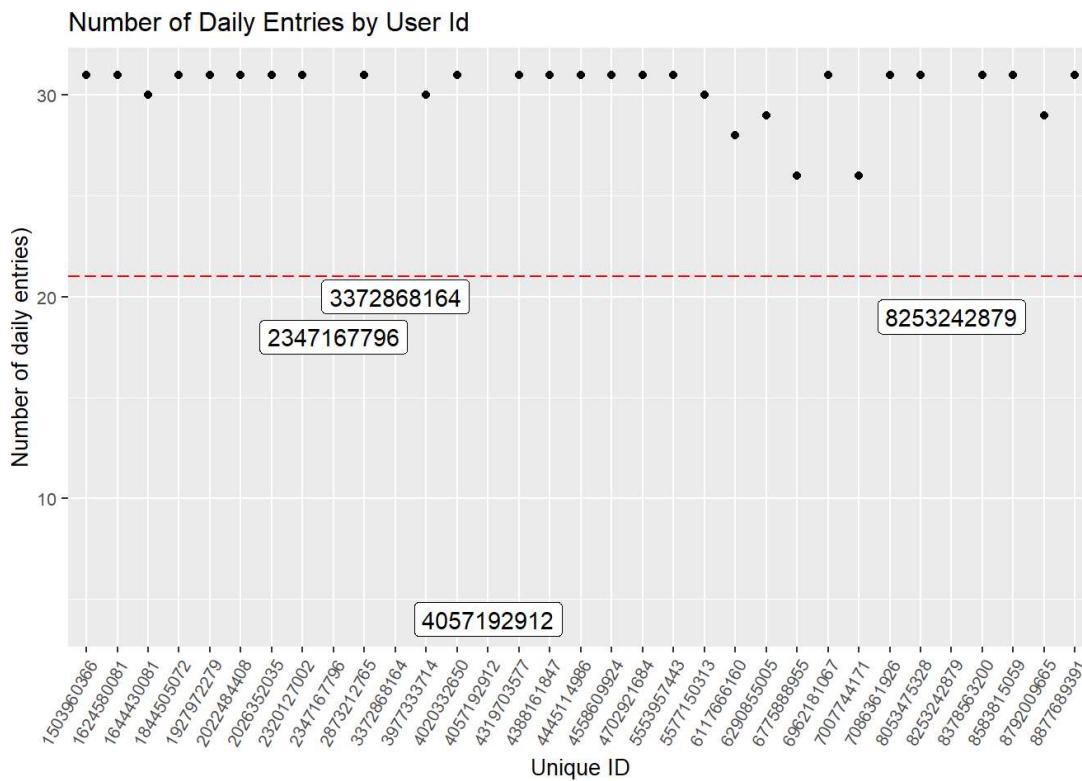
glimpse(weight_log)
```

```
## Rows: 67
## Columns: 6
## $ Id              <chr> "1503960366", "1503960366", "1927972279", "2873212765", ~
## $ Date            <date> 2016-05-02, 2016-05-03, 2016-04-13, 2016-04-21, 2016-0~
## $ WeightKg        <dbl> 52.6, 52.6, 133.5, 56.7, 57.3, 72.4, 72.3, 69.7, 70.3, ~
## $ BMI             <dbl> 22.65, 22.65, 47.54, 21.45, 21.69, 27.45, 27.38, 27.25, ~
## $ IsManualReport <chr> "True", "True", "False", "True", "True", "True", "True"~
## $ Weekdays         <ord> Monday, Tuesday, Wednesday, Thursday, Thursday, Sunday, ~
```

4.1.3 Check for any outlier

```
number_of_logs <- daily_activity %>%
  count(Id) %>%
  arrange(n)

ggplot(number_of_logs, aes(x=Id, y=n)) +
  geom_point()+
  geom_label(
    data=number_of_logs %>% filter(n<21), # To label the outlier
    aes(label=Id)
  )+
  ggtitle("Number of Daily Entries by User Id")+
  labs(x="Unique ID", y="Number of daily entries")+
  geom_hline(yintercept = 21, linetype="longdash", color="red")+
  theme(plot.title = element_text(size=12), text = element_text(size=10),axis.text.x = element_text(angle = 60, hjust = 1))
```



4.1.4 Remove Outliers

```
#Remove outliers from daily_activity
daily_activity <- daily_activity[!daily_activity$id %in% c("2347167796", "3372868164", "4057192912", "8253242879"),]

#Remove outliers from sleep_log
sleep_log <- sleep_log[!sleep_log$id %in% c("2347167796", "3372868164", "4057192912", "8253242879"),]

#Remove outliers from weight_log
weight_log <- weight_log[!weight_log$id %in% c("2347167796", "3372868164", "4057192912", "8253242879"),]
```

4.2 Detailed summary of cleaned Dataset

Data summary

Name	<u>daily_activity</u>
Number of rows	879
Number of columns	16

Column type frequency:

character	1
Date	1
factor	1
numeric	13

Group variables

None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0	1	10	10	0	29	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0	1	2016-04-12	2016-05-12	2016-04-27	31

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Weekdays	0	1	TRUE	7	Tue: 142, Wed: 140, Thu: 137, Fri: 116

Variable type: numeric

Data summary

skim_variable	n_missing	complete_rate	mean	sd	po	p25	p50	p75	p100
TotalSteps	0	1	7659.31	5174.03	0	3695.50	7412.00	10938.00	36019.00
TotalDistance	0	1	5.52	4.00	0	2.56	5.28	7.79	28.03
TrackerDistance	0	1	5.50	3.98	0	2.56	5.28	7.78	28.03
LoggedActivitiesDistance	0	1	0.12	0.64	0	0.00	0.00	0.00	4.94
VeryActiveDistance	0	1	1.52	2.71	0	0.00	0.20	2.14	21.92
ModeratelyActiveDistance	0	1	0.57	0.89	0	0.00	0.24	0.80	6.48
LightActiveDistance	0	1	3.35	2.07	0	1.94	3.37	4.83	10.71
SedentaryActiveDistance	0	1	0.00	0.01	0	0.00	0.00	0.00	0.11
VeryActiveMinutes	0	1	21.70	33.66	0	0.00	3.00	33.00	210.00
FairlyActiveMinutes	0	1	13.68	20.25	0	0.00	6.00	19.50	143.00
LightlyActiveMinutes	0	1	190.57	108.55	0	127.00	197.00	260.00	518.00
SedentaryMinutes	0	1	988.04	302.54	0	728.00	1054.00	1223.50	1440.00
Calories	0	1	2330.01	728.01	0	1826.50	2182.00	2822.50	4900.00

Name sleep_log

Number of rows 395

Number of columns 7

Column type frequency:

character 2

Date	1
factor	1
numeric	3

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0	1	10	10	0	23	0
SleepTime	0	1	8	8	0	1	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0	1	2016-04-12	2016-05-12	2016-04-27	31

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Weekdays	0	1	TRUE	7	Wed: 64, Tue: 63, Thu: 61, Sat: 56

Variable type: numeric

Data summary

skim_variable	n_missing	complete_rate	mean	sd	po	p25	p50	p75	p100
TotalSleepRecords	0	1	1.12	0.35	1	1	1	1.0	3
TotalMinutesAsleep	0	1	418.12	120.48	58	358	430	492.0	796
TotalTimeInBed	0	1	457.24	129.36	61	402	461	526.5	961

Name	weight_log
Number of rows	67
Number of columns	6

Column type frequency:	
character	2
Date	1
factor	1
numeric	2

Group variables	None
-----------------	------

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Id	0		1	10	10	0	8
IsManualReport	0		1	4	5	0	2

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
Date	0		1 2016-04-12	2016-05-12	2016-04-27	31

Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
Weekdays	0		1 TRUE	7	Mon: 12, Wed: 12, Thu: 11, Sun: 10

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
WeightKg	0		1 72.04	13.92	52.60	61.40	62.50	85.05	133.50
BMI	0		1 25.19	3.07	21.45	23.96	24.39	25.56	47.54

5 Analysis phase :Analyze Data to Answer Questions

We will analyze trends of the users of FitBit and determine if that can help us on BellaBeat's marketing strategy.

5.1 Type of users per activity level

since we don't have any demographic variables from our sample we want to determine the type of users with the data we have. We can classify the users by activity considering the daily amount of steps. We can categorize users as follows:

- Sedentary - Less than 5000 steps a day.
- Low active - Between 5000 and 7499 steps a day.
- fairly active - Between 7500 and 9999 steps a day.
- Highly Active - More than 10000 steps a day. Classification has been made per the following article
<https://pubmed.ncbi.nlm.nih.gov/14715035/> (<https://pubmed.ncbi.nlm.nih.gov/14715035/>)

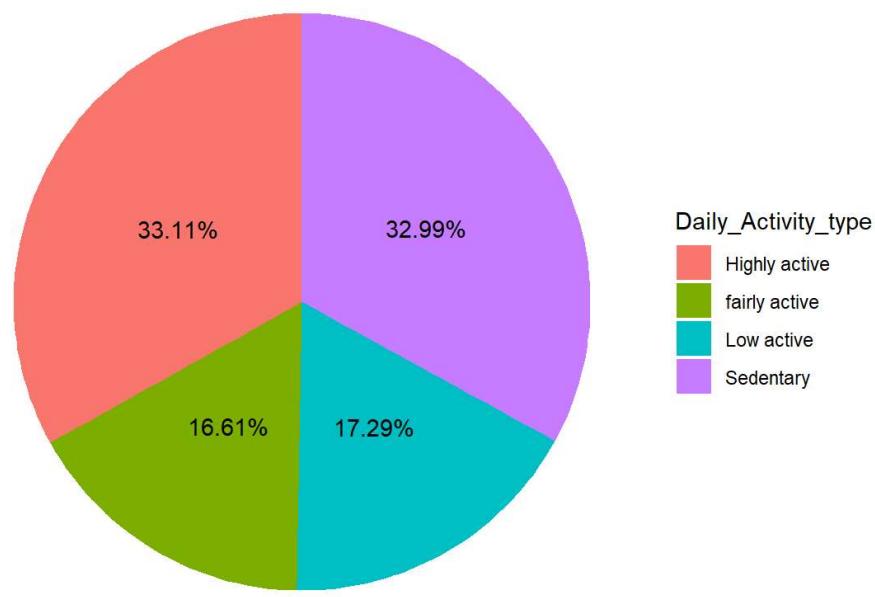
```
## # A tibble: 4 × 3
##   Daily_Activity_type total perc
##   <fct>              <int> <chr>
## 1 Fairly active      146 16.61%
## 2 Low active          152 17.29%
## 3 Sedentary           290 32.99%
## 4 Highly active       291 33.11%
```

6 Share phase: Share Data Through the Art of Visualization

6.1 Users daily Lifestyle

- Half of the User's Daily Activity record haven't reached 8000 steps per Day
- Only One-third of the Users Daily activity have reached 10000 steps and more

Daily Lifestyle type distribution



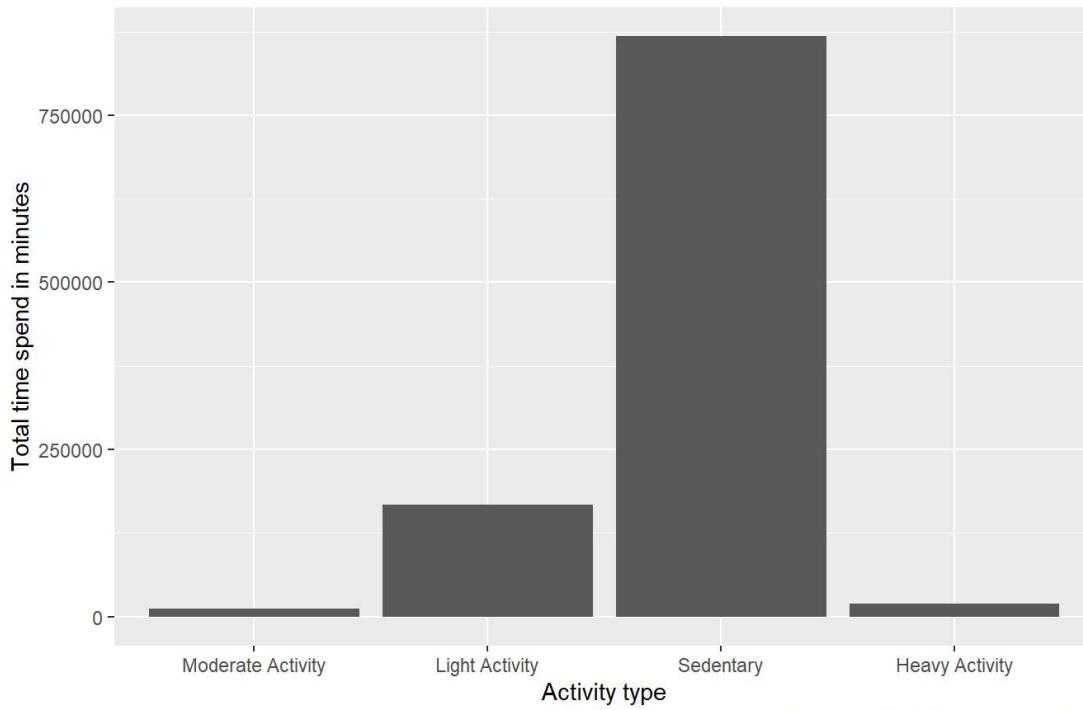
Data Source: FitBit Fitness Tracker Data

6.2 Daily Activity type distribution in minutes

- Majority Users are sitting all day.

We can see that a majority of the day (81%) users are **sedentary**. There are serious health effects of too much sitting as per the following article <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3404815/> (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3404815/>)

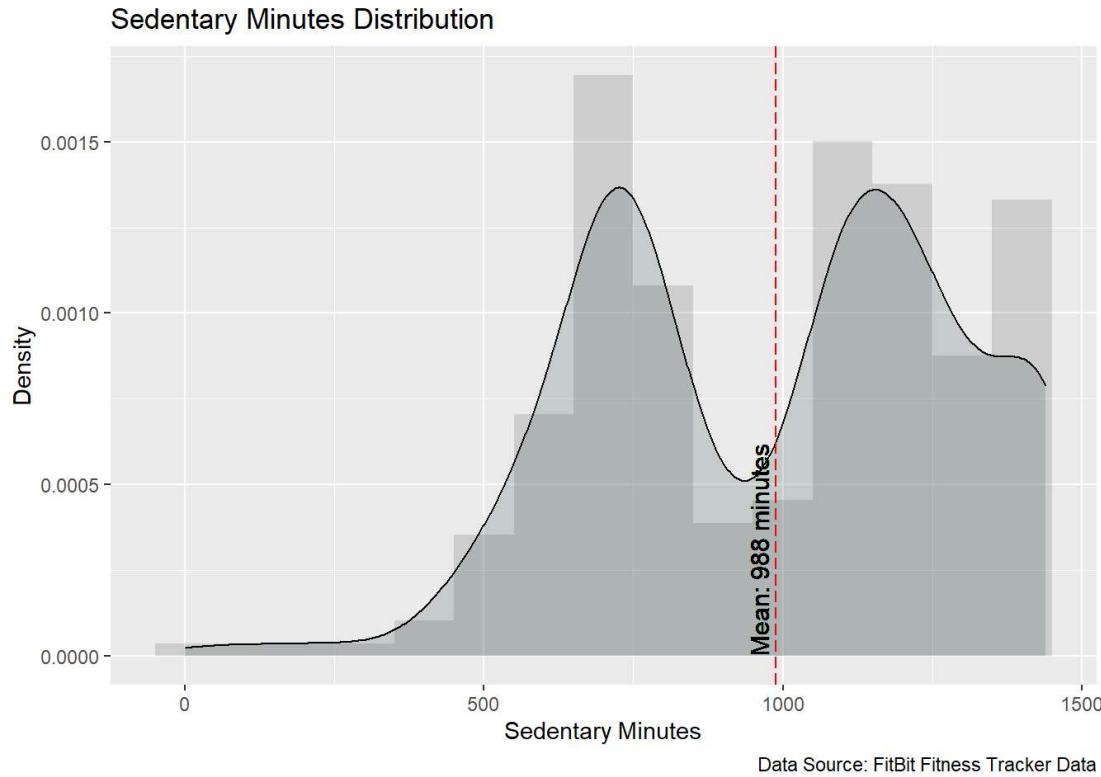
Daily Activity type distribution in minutes



Data Source: FitBit Fitness Tracker Data

6.2.1 Users display long sedentary minutes throughout the day.

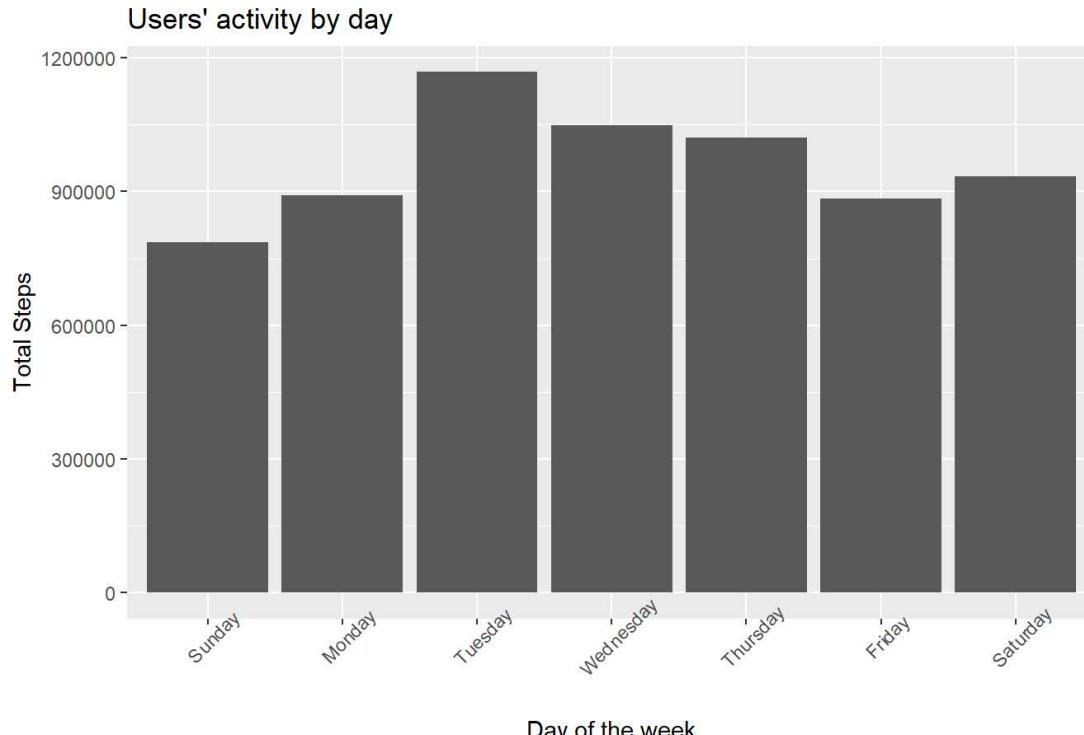
The average daily sedentary minutes for all participants is **988** minutes, or **16.47** hours.



6.3 Users' activity by day

6.3.1 Users are more motivated to work out at the beginning of the week.

Users start off the week active and slow down mid-week(Highest activity is seen on Tuesday). This could be due to busy work schedules. There tends to be a pick up by Saturday which could be due to the weekend users spending time outdoors.

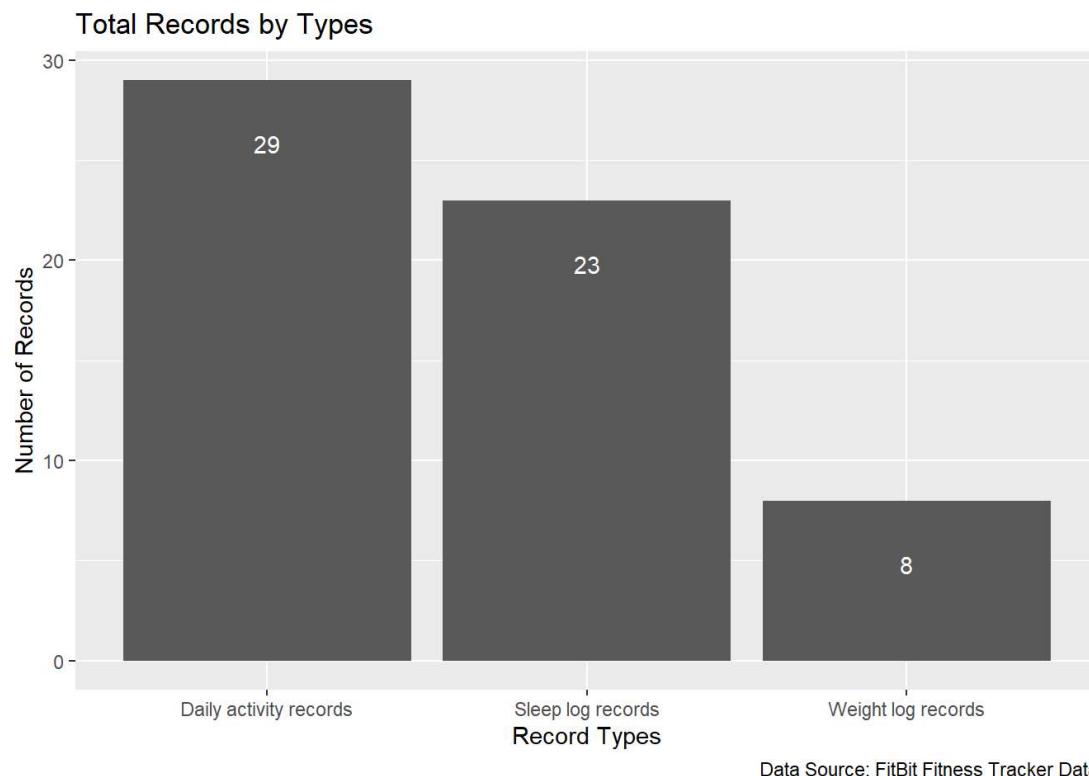


6.4 Customers use Bellabeat smart gadgets to record the daily activity the most

- Daily Activity
 - 29 Users
 - 879 records
- Daily Sleep
 - 23 Users
 - 395 records

It is most common for the Users to use fitness trackers to record daily activity. However, users do not always record their daily sleep and weight.

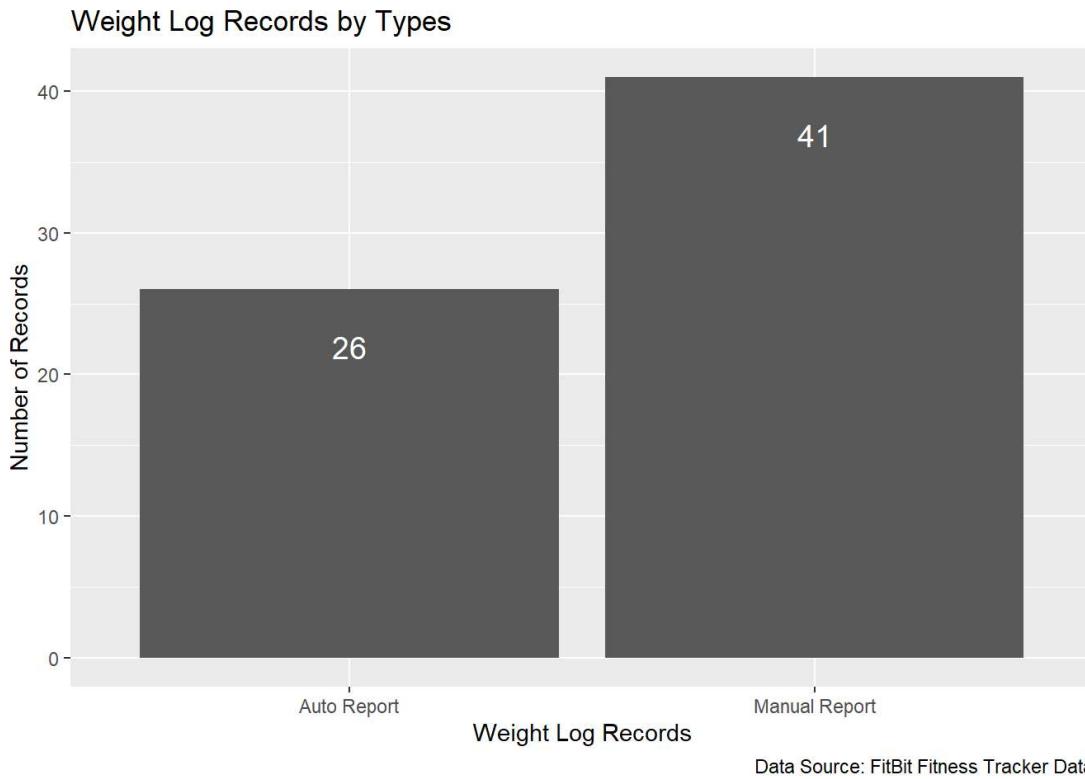
This could be due to lack of comfort to sleep with their fitness trackers at night and therefore taking them off before getting in bed. Consider reviewing the design of fitness tracker to make it more comfortable.



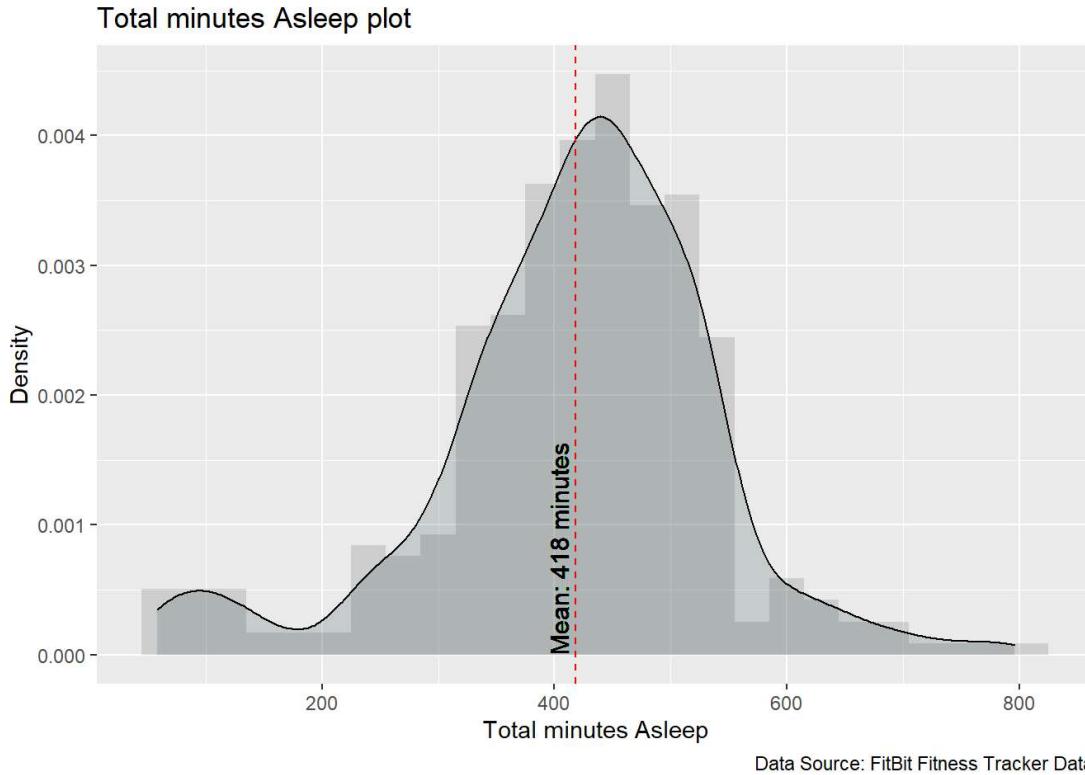
- Daily Weight
 - 8 Users
 - 67 records
- Only 28% of the users recorded their weight Only **8** out of **29** users recorded their weight. Due to the small sample set, I would not be able to gather significant statistical analysis from the data.

However, it would be worthwhile to explore why users are not logging their weight as much as daily activity and sleep.

- Nearly double the amount of entries were manually reported

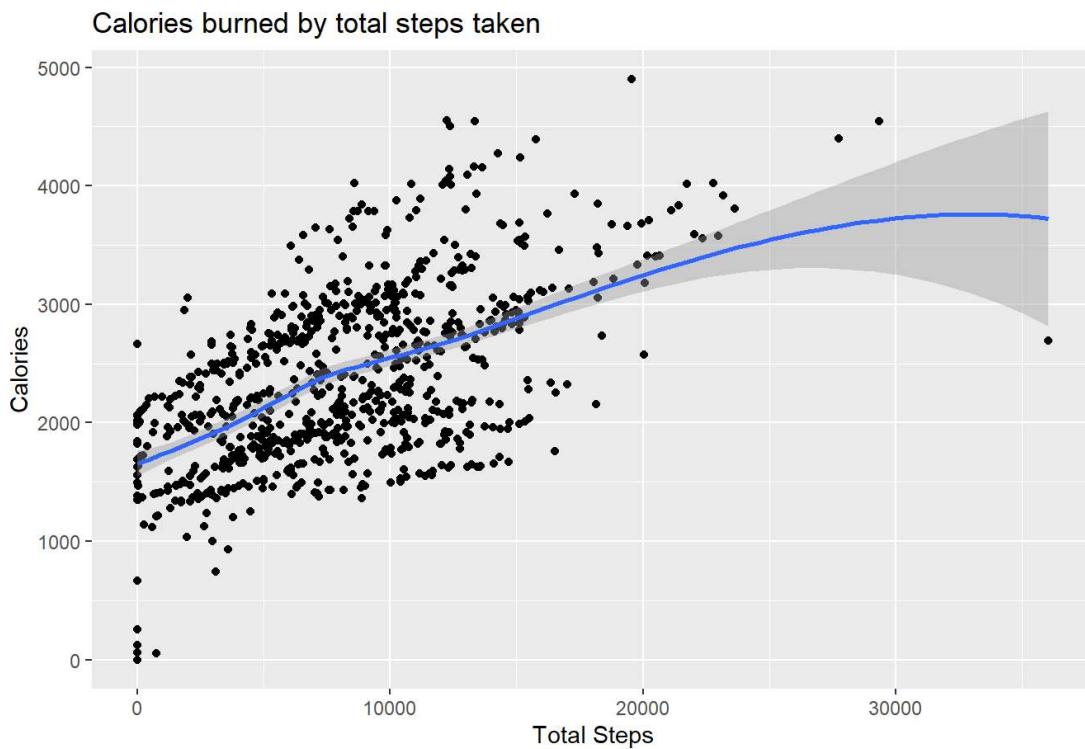


6.5 Users have adequate sleep



The data shows that the sleep time is normally distributed among all users. The average sleep time is 418 minutes (~7 hrs Sleep), which means on average, participants have adequate sleep.

6.6 Calories burned by steps



Data Source: FitBit Fitness Tracker Data

There is a clear positive correlation between these two variables: the more steps taken in a day, the more calories burned. To make sure of this assumption, we can check the correlation between these two variable using the *Pearson Correlation Coefficient*.

Simply put, the Pearson Correlation Coefficient is a measure of linear correlation between 2 variables. For more information, click here (https://en.wikipedia.org/wiki/Pearson_correlation_coefficient)

```
##
## Pearson's product-moment correlation
##
## data: daily_activity$TotalSteps and daily_activity$Calories
## t = 21.754, df = 877, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5473217 0.6333507
## sample estimates:
##      cor
## 0.5920201
```

With a confidence level of 95%, the correlation between the variables is almost 0.6. This means that there is a strong relationship between the variables.

7 Act Phase(Conclusion): Time to act on your data

7.1 Daily notification on the total steps

We have seen a correlation between steps and calories So Calories are burnt by the steps taken daily. Based on users' objectives, Bellabeat could recommend a minimum number of 8,000 to 10,000 steps is a decent goal for daily walking. sending them alarms if they haven't reached the steps and also creating posts on our app explaining the health benefits of reaching that goal.

7.2 Alert notification for too much sitting

From the Users Daily Activity, We have seen that around 81% of the time spend is Sedentary. As recent study shows sitting all day causes serious Health effects, Bellabeat could send alert notification when user is showing Sedentary behaviour for a given amount of time and also receives a task to do like take a walk or any exercise customized by the User with a side note Bellabeat should remind the User the reasons why sitting too much is bad for health

7.3 Keeping track on the Users Daily Sleeping hours

We have seen that there is a very few records of Uses Daily sleep Log. It signifies User removes the gadget before sleep. Keeping track of their daily sleeping hours and issue an alert next day if the User didnt have sufficient sleep. Also offer helpfull resources to help customers sleep - ex. breathing advises, podcasts with relaxing music, sleep techniques. Based on Users profile, recommend the best sleeping hours range for health growth and well being. Users could set up a desired time to go to sleep and receive a notification minutes before prepare to sleep.

7.4 Reward system

- We are aware that some people don't get motivated by notifications so we can classify them into different levels/categories and will lose/gain points based on their daily activity and all these points will be accumulated by Week/Year which would inturn reflect the Level they are.
- The more points they receive the more they can avail the discounted prices of any Bellabeat products.
- Also the merits accumulated is to be used before the end of the Year after that it will be reset to zero. This will encourage Users to wear Bellabeat gadgets more frequently which would help the Bellabeat app to gain more valuable insights from their daily activities and also improve Bellabeat advertising. Two birds with one stone