Phase 1 - Ask

Business Task

```
Study the health-focused products to analyze the usage in order to know how the customers are using the products. The data will be used to get high level recommendations to see how it can help with influencing the market strategy.
```

Key Stakeholders

- Urška Sršen, Bellabeat’s co-founder and Chief Creative Officer
- Sando Mur, Mathematician and Bellabeat’s cofounder
- Bellabeat marketing analytics team

Phase 2 - Prepare

Data Credibility

- It is a public data from FitBit Fitness Tracker Data. It's a dataset created by collecting data from thirty fitbit users that includes minute-level output for physical activity, heart rate, and sleep monitoring. The database is segmented in several tables.

Loading Packages

In [1]:

```
library(tidyverse)
library(ggplot2)
library(dplyr)
library(lubridate)
library(readxl)
library(reshape2)
```

```
── Attaching core tidyverse packages ────────────────────── tidyverse 2.0.0 ──
✓ dplyr     1.1.2     ✓ readr     2.1.4
✓ forcats   1.0.0     ✓ stringr   1.5.0
✓ ggplot2   3.4.2     ✓ tibble    3.2.1
✓ lubridate 1.9.2     ✓ tidyr     1.3.0
✓ purrr     1.0.1
── Conflicts ──────────────────────────────────────── tidyverse_conflicts() ──
✕ dplyr::filter() masks stats::filter()
✕ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become erro
rs

Attaching package: ‘reshape2’


The following object is masked from ‘package:tidyr’:

    smiths
```

Before importing the datasets I cleaned the data using Google Sheets. I fixed the date and date-time formats.

Importing Datasets

In [2]:

```
#Daily Data

daily_intensity <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                              sheet = "dailyIntensities_merged") %>%
  rename(Date = ActivityDay)

daily_steps <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                          sheet = "dailySteps_merged") %>%
  rename(Date = ActivityDay)

daily_activity <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                               sheet = "dailyActivity_merged") %>%
  rename(Date = ActivityDate)


weight_log <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                         sheet = "weightLogInfo_merged")

daily_sleep <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                          sheet = "sleepDay_merged")

daily_cal <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Daily Data.xlsx",
                        sheet = "dailyCalories_merged") %>%
  rename(Date = ActivityDay)
```

```
#Hourly Data

hour_steps <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Hourly Data.xlsx",
                        sheet = "hourlySteps_merged")

hour_intensity <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Hourly Data.xlsx",
                        sheet = "hourlyIntensities_merged")

hour_cal <- read_excel("../input/d/anu6hav/fitbit-tracker-data/Hourly Data.xlsx",
                        sheet = "hourlyCalories_merged")
```

Exploring the dataset. Let's see the number of people participated for the tracking data.

In [3]:

```
n_distinct(daily_steps$Id)

n_distinct(daily_activity$Id)

n_distinct(weight_log$Id)

n_distinct(daily_sleep$Id)
```

```
33

33

8

24
```

We can see 33 participants in daily_steps and daily_activity datasets, 24 in daily_sleep and only 8 in weight_log. The weight_log is not having sufficient number of participants to make any recommendations or conclusions.

Let's have a look at summary statistics for the various datasets

In [4]:

```
#Active minutes based on the intensity

daily_activity %>%
  select(VeryActiveMinutes, FairlyActiveMinutes, LightlyActiveMinutes, SedentaryMinutes) %>%
  summary()

#calories burnt


daily_cal %>%
  select(Calories) %>%
  summary()

#sleep


daily_sleep %>%
  select(TotalMinutesAsleep, TotalTimeInBed) %>%
  summary()

#Weight


weight_log %>%
  select(BMI, WeightKg) %>%
  summary()

#Daily steps taken


daily_steps %>%
  select(StepTotal) %>%
  summary()
```

```
#Hourly steps taken

hour_steps %>%
  select(StepTotal) %>%
  summary()

#Hourly calories burnt

hour_cal %>%
  select(Calories) %>%
  summary()

#Hourly intensity

hour_intensity %>%
  select(TotalIntensity, AverageIntensity) %>%
  summary()
```

```
 VeryActiveMinutes FairlyActiveMinutes LightlyActiveMinutes SedentaryMinutes
 Min.   :  0.00    Min.   :  0.00      Min.   :  0.0        Min.   :   0.0  
 1st Qu.:  0.00    1st Qu.:  0.00      1st Qu.:127.0        1st Qu.: 729.8  
 Median :  4.00    Median :  6.00      Median :199.0        Median :1057.5  
 Mean   : 21.16    Mean   : 13.56      Mean   :192.8        Mean   : 991.2  
 3rd Qu.: 32.00    3rd Qu.: 19.00      3rd Qu.:264.0        3rd Qu.:1229.5  
 Max.   :210.00    Max.   :143.00      Max.   :518.0        Max.   :1440.0  

    Calories   
 Min.   :   0  
 1st Qu.:1828  
 Median :2134  
 Mean   :2304  
 3rd Qu.:2793  
 Max.   :4900  

 TotalMinutesAsleep TotalTimeInBed 
 Min.   : 58.0      Min.   : 61.0  
 1st Qu.:361.0      1st Qu.:403.0  
 Median :433.0      Median :463.0  
 Mean   :419.5      Mean   :458.6  
 3rd Qu.:490.0      3rd Qu.:526.0  
 Max.   :796.0      Max.   :961.0  
```

```
     BMI           WeightKg     
Min.   :21.45   Min.   : 52.60  
1st Qu.:23.96   1st Qu.: 61.40  
Median :24.39   Median : 62.50  
Mean   :25.19   Mean   : 72.04  
3rd Qu.:25.56   3rd Qu.: 85.05  
Max.   :47.54   Max.   :133.50  
```

```
  StepTotal    
Min.   :    0  
1st Qu.: 3790  
Median : 7406  
Mean   : 7638  
3rd Qu.:10727  
Max.   :36019  
```

```
  StepTotal      
Min.   :    0.0  
1st Qu.:    0.0  
Median :   40.0  
Mean   :  320.2  
3rd Qu.:  357.0  
Max.   :10554.0  
```

```
   Calories     
Min.   : 42.00  
1st Qu.: 63.00  
Median : 83.00  
Mean   : 97.39  
3rd Qu.:108.00  
Max.   :948.00  


TotalIntensity   AverageIntensity
Min.   :  0.00   Min.   :0.0000  
1st Qu.:  0.00   1st Qu.:0.0000  
Median :  3.00   Median :0.0500  
Mean   : 12.04   Mean   :0.2006  
3rd Qu.: 16.00   3rd Qu.:0.2667  
Max.   :180.00   Max.   :3.0000  
```

Data findings :

- The summary shows that the average sedentary time is 991.2 minutes which is a lot higher than the other categories. This can be improved.
- On an average a person sleeps 1 time and spends 7.64 hours for sleeping out of which the actual time the person is sleeping is 6.99 hours.
- The majority of people on an average are lightly active which can be for small activities like travelling, walking, cooking etc.
- The average steps taken a day are 7638 while the maximum is 36019. This can be adjusted based on the user goals.

Merging datasets

Before visualizing the data I will merge different datasets to find interesting trends and corelations. For merging I will use inner join.

In [5]:

```
#Merging Calories and Activities datasets

merged_daily_1 <- merge(daily_cal, daily_activity, by = c('Id', 'Date', 'Calories'))

head(merged_daily_1)

merged_daily_2 <- merge(merged_daily_1, daily_sleep, by = c('Id'))

head(merged_daily_2)

#Merging Hourly Datasets

merged_hourly <- merge(hour_intensity, hour_cal, by = c('Id', 'ActivityHour'))

merged_hourly_2 <- merge(merged_hourly, hour_steps, by = c('Id', 'ActivityHour'))

head(merged_hourly)

head(merged_hourly_2)
```

A data.frame: 6 × 15

	Id	Date	Calories	TotalSteps	TotalDistance	TrackerDistance	LoggedActivitiesDistance	VeryActiveDistance	Mo
	<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<db
1	1503960366	4/12/2016	1985	13162	8.50	8.50	0	1.88	0.5
2	1503960366	4/13/2016	1797	10735	6.97	6.97	0	1.57	0.6
3	1503960366	4/14/2016	1776	10460	6.74	6.74	0	2.44	0.4
4	1503960366	4/15/2016	1745	9762	6.28	6.28	0	2.14	1.2
5	1503960366	4/16/2016	1863	12669	8.16	8.16	0	2.71	0.4
6	1503960366	4/17/2016	1728	9705	6.48	6.48	0	3.19	0.7

A data.frame: 6 × 19

	Id	Date	Calories	TotalSteps	TotalDistance	TrackerDistance	LoggedActivitiesDistance	VeryActiveDistance	Mode
	<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl
1	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87
2	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87
3	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87
4	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87
5	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87
6	1503960366	5/4/2016	1819	11100	7.15	7.15	0	2.46	0.87

A data.frame: 6 × 5

	Id	ActivityHour	TotalIntensity	AverageIntensity	Calories
	<dbl>	<chr>	<dbl>	<dbl>	<dbl>
1	1503960366	4/12/2016 1:00:00 AM	8	0.133333	61
2	1503960366	4/12/2016 1:00:00 PM	6	0.100000	66
3	1503960366	4/12/2016 10:00:00 AM	29	0.483333	99
4	1503960366	4/12/2016 10:00:00 PM	9	0.150000	65
5	1503960366	4/12/2016 11:00:00 AM	12	0.200000	76
6	1503960366	4/12/2016 11:00:00 PM	21	0.350000	81

A data.frame: 6 × 6

	Id	ActivityHour	TotalIntensity	AverageIntensity	Calories	StepTotal
	<dbl>	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	1503960366	4/12/2016 1:00:00 AM	8	0.133333	61	160
2	1503960366	4/12/2016 1:00:00 PM	6	0.100000	66	221
3	1503960366	4/12/2016 10:00:00 AM	29	0.483333	99	676
4	1503960366	4/12/2016 10:00:00 PM	9	0.150000	65	89
5	1503960366	4/12/2016 11:00:00 AM	12	0.200000	76	360
6	1503960366	4/12/2016 11:00:00 PM	21	0.350000	81	338

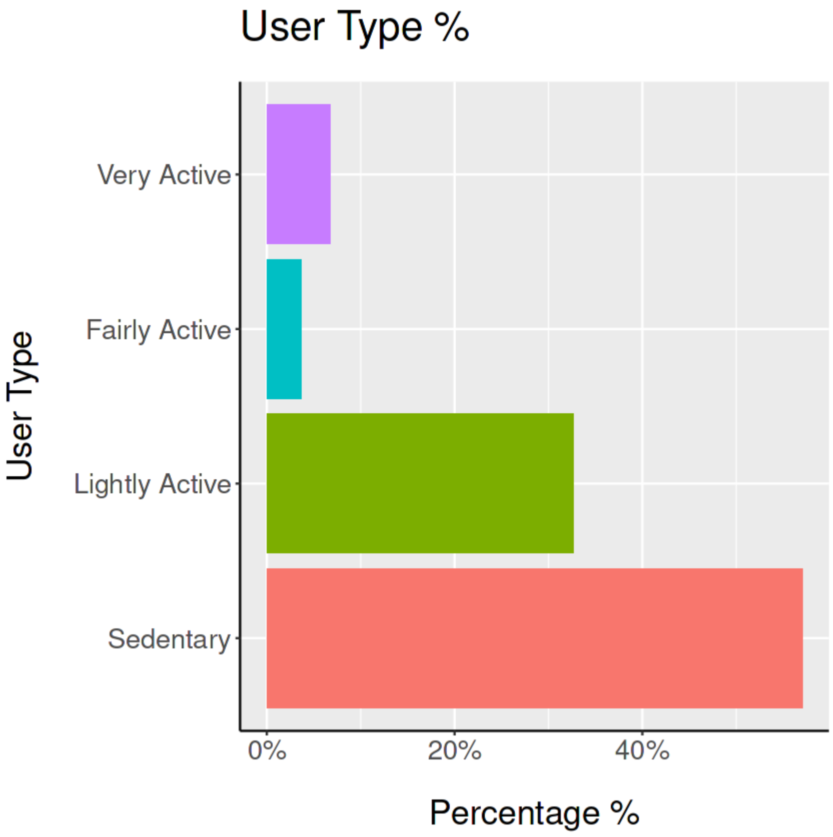
Phase - 3 Process

- Plot 1 - User Type %

In [6]:

```
user_type_per <- merged_daily_1 %>%
  summarise(
    user_type = factor(
      case_when(
        SedentaryMinutes > mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Sedentary",
        SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes > mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Lightly Active",
        SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes > mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Fairly Active",
        SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes > mean(VeryActiveMinutes) ~ "Very Active"
      ), levels= c("Sedentary", "Lightly Active", "Fairly Active", "Very Active")), .group=Id) %>%
  drop_na()


  user_type_per %>%
  group_by(user_type) %>%
  summarise(total = n()) %>%
  mutate(totals = sum(total)) %>%
  group_by(user_type) %>%
  summarise(total_percent = total/totals) %>%
  ggplot(aes(x = user_type, y = total_percent, fill = user_type)) +
    geom_col() +
    coord_flip() +
      xlab(label = "User Type") +
      ylab(label = "Percentage %") +
        scale_y_continuous(labels = scales :: percent) +
          theme(legend.position = "none") +
            labs(title = "User Type %") +
            theme(legend.position = "none" , text = element_text(size = 20),
                  plot.title = element_text(margin = margin(b = 20, l = 20)),
                  axis.title.x = element_text(margin = margin(20,20)),
                  axis.title.y = element_text(margin = margin(20, 20)),
                  axis.line = element_line(colour = "black"))
```

```
Warning message:
“Returning more (or less) than 1 row per `summarise()` group was deprecated in
dplyr 1.1.0.
ℹ Please use `reframe()` instead.
ℹ When switching from `summarise()` to `reframe()`, remember that `reframe()`
  always returns an ungrouped data frame and adjust accordingly.”
```

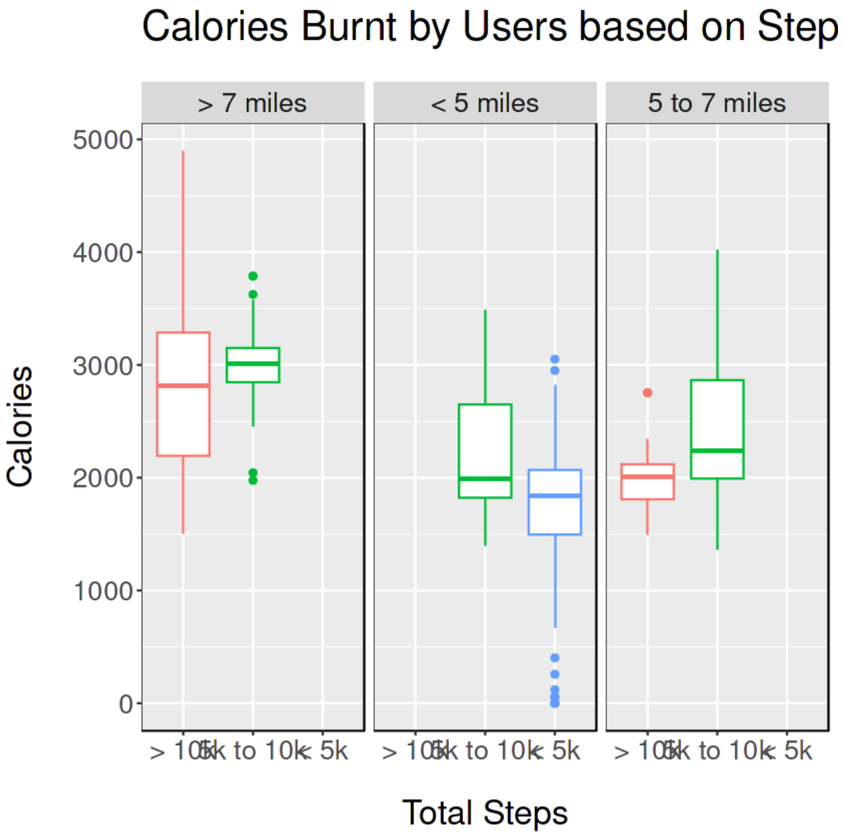


- Plot 2 - Calories Burnt by Users based on Steps/Distance

In [7]:

```
cal_burnt <- merged_daily_1 %>%
  summarise(
    total_steps = factor(
      case_when(
        TotalSteps < 5000 ~ "< 5k",
        TotalSteps >= 5000 & TotalSteps <= 10000 ~ "5k to 10k",
        TotalSteps > 10000 ~ "> 10k"), levels = c("> 10k", "5k to 10k", "< 5k")),
    total_distance = factor(
      case_when(
        TotalDistance < 5 ~ "< 5 miles",
        TotalDistance >= 5 & TotalDistance <= 7 ~ "5 to 7 miles",
        TotalDistance > 7 ~ "> 7 miles"), levels = c("> 7 miles", "< 5 miles", "5 to 7 miles")), Calor
ies)

cal_burnt %>%
  ggplot(aes(x = total_steps, y = Calories)) +
    geom_boxplot(mapping = aes(color = total_steps)) +
    xlab(label = "Total Steps") +
    ylab(label = "Calories") +
    theme(legend.position = "none") +
    labs(title = "Calories Burnt by Users based on Steps/Distance") +
    theme(text = element_text(size = 20),
          plot.title = element_text(margin = margin(b = 20, l = 20)),
          axis.title.x = element_text(margin = margin(20,20)),
          axis.title.y = element_text(margin = margin(20, 20)),
          panel.border = element_rect(colour = "black", fill = NA)) +
    facet_wrap(~total_distance)
```

```
Warning message:
“Returning more (or less) than 1 row per `summarise()` group was deprecated in
dplyr 1.1.0.
ℹ Please use `reframe()` instead.
ℹ When switching from `summarise()` to `reframe()`, remember that `reframe()`
  always returns an ungrouped data frame and adjust accordingly.”
```



- Plot 3 - Sleep Quality Analysis

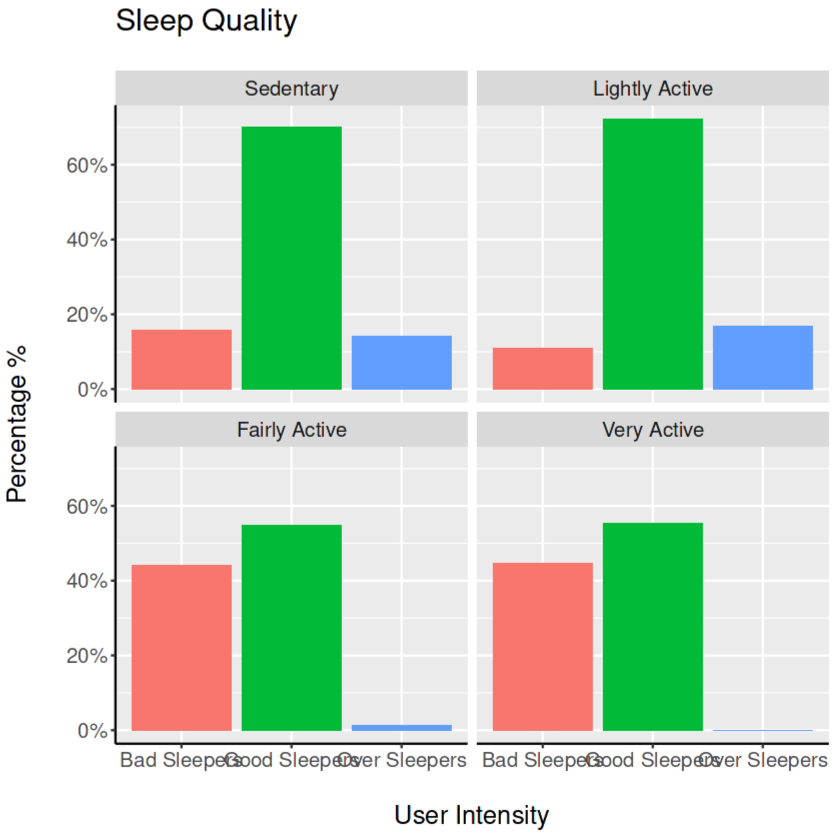
In [8]:

```
sleepQuality_user <- merged_daily_2 %>%
  group_by(Id) %>%
  summarise(
    user_intensity = factor(case_when(
      SedentaryMinutes > mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Sedentary",
      SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes > mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Lightly Active",
      SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes > mean(FairlyActiveMinutes) & VeryActiveMinutes < mean(VeryActiveMinutes) ~ "Fairly Active",
      SedentaryMinutes < mean(SedentaryMinutes) & LightlyActiveMinutes < mean(LightlyActiveMinutes) & FairlyActiveMinutes < mean(FairlyActiveMinutes) & VeryActiveMinutes > mean(VeryActiveMinutes) ~ "Very Active",
    ),levels=c("Sedentary", "Lightly Active", "Fairly Active", "Very Active")),
    sleep_quality = factor(case_when(
      mean(TotalMinutesAsleep) < 370 ~ "Bad Sleep",
      mean(TotalMinutesAsleep) >= 370 & mean(TotalMinutesAsleep) <= 490 ~ "Good Sleep",
      mean(TotalMinutesAsleep) > 480 ~ "Over Sleep"
    ), levels=c("Bad Sleep", "Good Sleep", "Over Sleep")), total_sleep = sum(TotalMinutesAsleep), .groups="drop") %>%
  drop_na() %>%
  group_by(user_intensity) %>%
  summarise(bad_sleepers = sum(sleep_quality == "Bad Sleep"),
            good_sleepers = sum(sleep_quality == "Good Sleep"),
            over_sleepers = sum(sleep_quality == "Over Sleep"), total = n(), .groups = "drop") %>%
  group_by(user_intensity) %>%
  summarise(
    "Bad Sleepers" = bad_sleepers / total,
    "Good Sleepers" = good_sleepers / total,
    "Over Sleepers" = over_sleepers / total,
    .groups = "drop"
  )

sleepQuality_user_melted <- sleepQuality_user %>%
  melt(id.vars = "user_intensity", value.name = "Percentage", variable.name = "Type")

head(sleepQuality_user_melted)

sleepQuality_user_melted %>%
  ggplot(aes(Type, Percentage, fill = Type)) +
  geom_bar(position = "dodge", stat = "Identity") +
  scale_y_continuous(labels = scales::percent) +
  facet_wrap(~user_intensity) +
  xlab(label = "User Intensity") +
  ylab(label = "Percentage %") +
  labs(title = "Sleep Quality") +
  theme(legend.position="",
        text = element_text(size = 15),
        plot.title = element_text(margin = margin(b = 20, l = 20)),
        axis.title.x = element_text(margin = margin(20,20)),
        axis.title.y = element_text(margin = margin(20, 20)),
        axis.line = element_line(colour = "black"))
```

```
Warning message:
“Returning more (or less) than 1 row per `summarise()` group was deprecated in
dplyr 1.1.0.
ℹ Please use `reframe()` instead.
ℹ When switching from `summarise()` to `reframe()`, remember that `reframe()`
  always returns an ungrouped data frame and adjust accordingly.”
```

A data.frame: 6 × 3

	user_intensity	Type	Percentage
	<fct>	<fct>	<dbl>
1	Sedentary	Bad Sleepers	0.1572872
2	Lightly Active	Bad Sleepers	0.1099158
3	Fairly Active	Bad Sleepers	0.4403292
4	Very Active	Bad Sleepers	0.4470990
5	Sedentary	Good Sleepers	0.7012987
6	Lightly Active	Good Sleepers	0.7231057



- Plot 4 - User Intensity Analysis

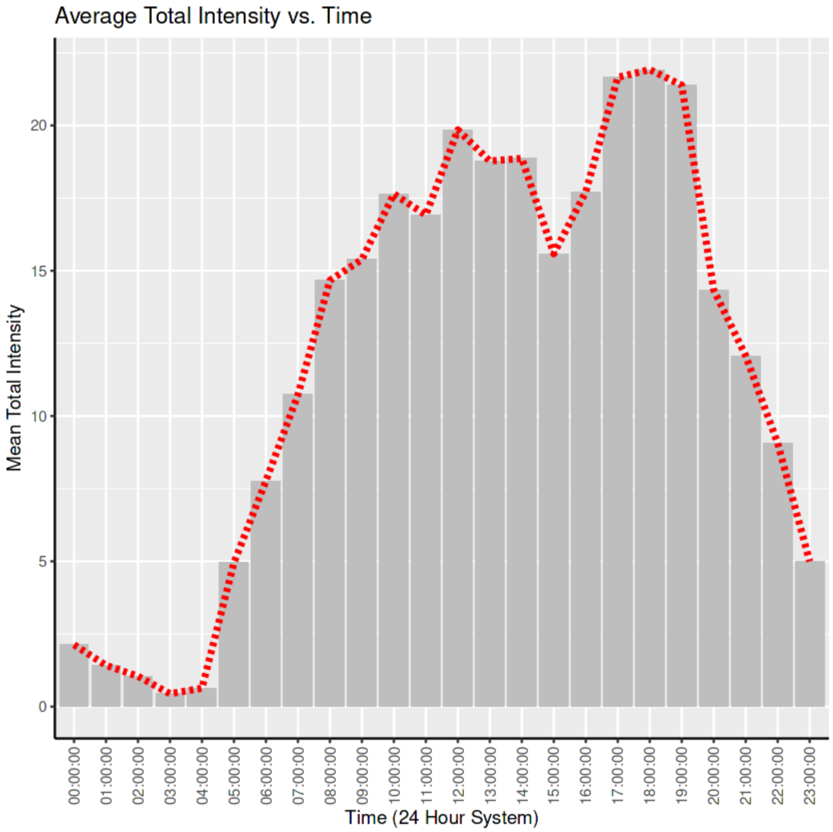
In [9]:

```
hour_intensity$ActivityHour = as.POSIXct(hour_intensity$ActivityHour, format="%m/%d/%Y %I:%M:%S %p", tz=Sys.timezone())
hour_intensity$time <- format(hour_intensity$ActivityHour, format = "%H:%M:%S")
head(hour_intensity)

Intensity_by_Time <- hour_intensity %>%
  group_by(time) %>%
  drop_na() %>%
  summarise(Mean_Total_Intensity = mean(TotalIntensity))

Intensity_by_Time %>%
  ggplot(aes(time, Mean_Total_Intensity, group = 1)) +
  geom_histogram(stat = "identity", fill = "gray") +
  geom_path(size = 1.5, colour = "red", linetype = "dashed") +
  xlab(label = "Time (24 Hour System)") +
  ylab(label = "Mean Total Intensity") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5),
        axis.line = element_line(colour = "black")) +
  labs(title="Average Total Intensity vs. Time")
```

A tibble: 6 × 5

Id	ActivityHour	TotalIntensity	AverageIntensity	time
<dbl>	<dttm>	<dbl>	<dbl>	<chr>
1503960366	2016-04-12 00:00:00	20	0.333333	00:00:00
1503960366	2016-04-12 01:00:00	8	0.133333	01:00:00
1503960366	2016-04-12 02:00:00	7	0.116667	02:00:00
1503960366	2016-04-12 03:00:00	0	0.000000	03:00:00
1503960366	2016-04-12 04:00:00	0	0.000000	04:00:00
1503960366	2016-04-12 05:00:00	0	0.000000	05:00:00

```
Warning message in geom_histogram(stat = "identity", fill = "gray"):
“Ignoring unknown parameters: `binwidth`, `bins`, and `pad`”
Warning message:
“Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead.”
```


Average Total Intensity vs. Time
Mean Total Intensity
0
5
10
15
20
00:00:00
01:00:00
02:00:00
03:00:00
04:00:00
05:00:00
06:00:00
07:00:00
08:00:00
09:00:00
10:00:00
11:00:00
12:00:00
13:00:00
14:00:00
15:00:00
16:00:00
17:00:00
18:00:00
19:00:00
20:00:00
21:00:00
22:00:00
23:00:00
Time (24 Hour System)

Phase - 4 Analyze

- Plot 1:
  - This chart shows the different types of users based on how active they are. You can see that the largest percentage is sedentary and fairly active are having the lowest percentage.
- Plot 2:
  - The boxplot tells the relation between steps taken, distance travelled and calories burnt in the process. The interesting part is that the steps taken in the range of 5000 to 10000 are having the most amount of fat burn which can be because of higher intensity like running which resulted in more distance travelled in less number of steps.
  - The chart also shows that distance travelled with high number of steps which might be because of various forms of cardio like jogging, swtiching between running and walking which resulted in large amount of calories burnt.
- Plot 3:
  - The bar chart depicts the relationship between sleep quality and activeness. The highest percentage of bad sleepers are among the sedentary users which tells that good quality of sleep is dependant on how active you are throughout the day.
  - The very active category is having the highest good sleepers. The interesting part is how the number of over sleepers reduce going from sedentary to very active which tells that the users who are active are less likely to stay in bed for too long.
- Plot 4:
  - The histogram tells us about the intensity levels throught the day. The intensity levels are high in the timeframe (14:30 - 19:30) which can be the time when people are involved in some kind of fitness activity like running, weight lifting.
  - Also the intensity levels start rising after 4 which tells that some users wake up around 5 in the morning.
  - The intensity levels are lowest in the timeframe (02:00 - 04:00) which can be due to users being in deep sleep.

Phase - 5 Share

Key Objectives

There are some interesting insights that would be beneficial when trying to create marketing strategies which will result in more customers.

- Increasing the awareness of benefits of logging the daily activity and sleep time which can increase the number of interested people and potential customers. Logging the data can help in motivating people to do physical activites that will help them with achieve better physical fitness and sleep.
- The data shows that sleep quality is having positive trend with activity intensity. Better sleep would result in weight loss and less stress which will improve quality of life. Spreading awareness for benefits of doing physical activities and include presets for activites in the devices based on intentisity that the customers can follow.
- The sedentary catergory were having the most percentage and it can be reduced by having alerts in the device which will trigger if the customer is sedentary for a long time.