Harmony, mobile application for well-being monitoring using HRV

Qiang Chen

Resumen– Este proyecto consiste en desarrollar una aplicación Android que utiliza un dispositivo para detectar las frecuencias cardíacas y calcular la variabilidad de la frecuencia cardíaca y atreves de estos valores se calcula el nivel del estrés y la calidad de sueño del usuario y el objetivo de la aplicación es recomendar ejercicios de respiración para recuperar del estrés. Y para alcanzar este objetivo de la aplicación Android, se implementaron varios prototipos aparte, uno de ellos es el servidor remoto donde contiene la base de dato, mediante este módulo se permite usuario guarda datos en la red y de esta forma se convierte la aplicación en multiusuario. Y los otros módulos del proyecto es el generador de valores de HRV, que estos valores irán directo a la base de dato. Y el generador del modelo JSON que utilizan los datos de base de dato para poder generar los modelos, los modelos JSON generado se utilizaran en la aplicación Android para poder simular valores de HRV. Estos dos generadores se implementaron básicamente para poder simular un dispositivo que detecta las frecuencias cardíacas ya que a final no podemos conseguir este dispositivo.

Palabras clave– Android, calidad de sueño, estrés, frecuencia cardíaca, generador de HVR, Laurea, MySQL, PHP, Polar, red bayesiana, UAB, XAMPP.

Abstract–This project consists of developing an Android application that uses a device to detect the heart rates and calculate the variability of the heart rate and through these values the stress level and the sleep quality of the user is calculated and the objective of the application is recommend breathing exercises to recover from stress. And to achieve this goal of the Android application, several prototypes were implemented separately, one of them is the remote server where it contains the data base, through this module it allows users to save data in the network and in this way the application becomes multi-user And the other modules of the project is the generator of HRV values, which these values will go directly to the data base. And the generator of the JSON model that uses the data base data to be able to generate the models, the generated JSON models will be used in the Android application to be able to simulate HRV values. These two generators were basically implemented to be able to simulate a device that detects heart rates since in the end we can not get this device.

Keywords– Android, sleep quality, stress, heart rate, HRV generator, Laurea, MySQL, PHP, Polar, Bayesian network, UAB, XAMPP.

1 INTRODUCTION

In some technical terminologies stress can be defined as a set of physiological rations that occurs when a person suffers a state of nervous tension, which can occur due to many situations in work or personal areas, for example overwork, poor relationship with families, etc.

According to studies the human body, stress activates the nervous system and certain hormones. These hormones increase heart rate, respiratory rate, blood pressure and metabolism.

If the stress produces in the long term can damage the human body, for example the heart that triggers cholesterol, blood pressure and the presence of triglycerides, the lungs that decreases their immune response, the brain and many other human organs.

One of the best ways to recover from stress is rest, perform different activities that improve rest for the human body.

A group of professors and master students from the Laurea cite laurea University of Applied Sciences have done

- E-mail de contacte: qiang.chen@e-campus.uab.cat
- Menció realitzada: Enginyeria de Computació
- Treball tutoritzat per: Fernando Vilariño Freire (Ciencia de visió de computació)
- Curs 2018/19

some research on stress and want to use the devices of the company Polar cite polar, in order to detect the heart rate of the people and calculate the value of stress, and recommend different exercises to be able to recover from stress.
The objective of this project is to collaborate with the teachers and students of Laurea University to develop an Android application that allows to detect heart rate through the device of the Polar company and calculate stress, with the aim of recommending different activities, mainly breathing exercises and sleeping tips to recover from stress.

2 Objectives

The project is called Harmony, and consists basically of five parts:

1. **The Android application [18]**: It is responsible for obtaining values about heart rate, calculating stress, and recommending breathing exercises.
For this part it is necessary to define some requirements for the device: implement for devices of 5.3 inches, and with a system superior to Android 4.4.
And as functional requirements of the application are: Implementation of multiuser, Internet connection, both to download or upload data, or to see videos inserted from YouTube and also calculate the values of stress level and quality of sleep in real time.
This last part due to the lack of Polar device, a heart rate simulator had to be implemented.

2. **Remote Server**: Allows multi-user functionality and connection from anywhere.

3. **MySQL Server [19]**: It is responsible for storing user data, stress values calculated by HVR, sleep quality, different breathing exercises and sleeping advice.

4. **Heart rate generator**: Due to that we can not get a real device from the Polar company to detect the heart rate, we have implemented an application that mapping a table with user's data and simulates the values of the heart rate in scene determined, all this data will be stored directly to the database.

5. **Heart rate model generator**: This generator will download data from the database, uninterested values will be filtered by an outlier detector, and finally the Bayesian rede algorithm will be applied to generate the model, which will be introduced to the application in order to simulate real data.

In Fig. 1 is shown the relationship diagram between different parts of the project.

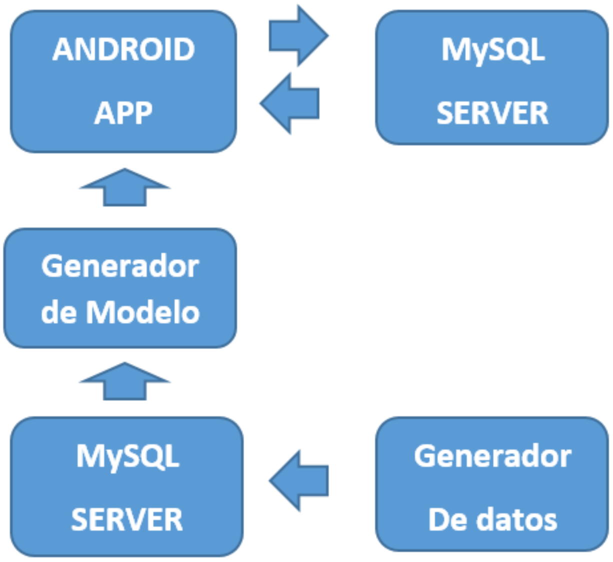


Fig. 1: Diagrama de la relación

3 State of the art

Currently there are many applications that manages stress: StressScan [5], Welltory [6], etc. Even the Polar company also offers this type of applications linked with their devices.
On the other hand there are also applications to calculate sleep quality: Sleep Cycle [7], Runtastic Sleep [8], etc.
Today stress is a very important issue for society, and for this reason there are many applications of this style.
On the other hand, in our project we want to combine all these functions in a single project to give a better way to recover from stress.

4 Methodology

4.1 Agile Methodology [9]

The project uses the agile methodology: Scrum, which basically consists in dividing the project into Sprint that lasts approximately one week. We have decided to use this method, since it consists of a project that involves many people, and in this way the development process is easy because it is iterative, which allows us to reorganize the tasks during the meeting of each Sprint. And also have more control over the development results.

Thanks to this methodology, it allows us to implement an iterative design, in which it is possible to design, implement and evaluate each defined task of the project in a more comfortable and more frequent way, in order to find development errors more quickly.

4.2 Service Design [10]

The project also applies the service design methodology, basically using the student team of Laurea University.
This methodology allows better application of the agile methodology, in which the relationship between people, infrastructure, communication and materials is increased to improve the quality of the project, the interaction between the provider and the users and the experience of the same.

In the Fig. 2 you can see the model of service design applied during the development of the project .

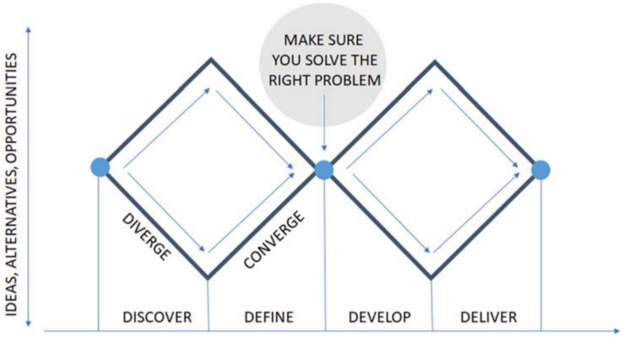

MAKE SURE YOU SOLVE THE RIGHT PROBLEM
IDEAS, ALTERNATIVES, OPPORTUNITIES
DIVERGE
CONVERGE
DISCOVER
DEFINE
DEVELOP
DELIVER

Fig. 2: Double diamond, (Stickdorn et al. 2018)

4.3 Project Planning and Management

The project officially begins on Monday, February 18, 2019, and will be completed on Thursday, July 11, 2019, during this period different meetings are held with the tutor of the TFG, meetings with Laurea students, research, designs and the implementations of the applications, delivery of the follow-up reports, the final report and the presentation.
The project was really started on Wednesday, April 3, 2019, where the first meeting with Laurea teachers was held.
The Fig. 12 of the section of appendices is the Gantt diagram of the project.

The development of the project basically consists of the Sprint and meetings. During each Spring it is dedicated to implement application. And in each meeting the tasks are defined.

In each meeting the following structure is followed:

1. Requirements assumed from the previous meeting.
2. Prototype generated.
3. Prototype Interface for the mobile app.
4. Database proposed.
5. Prototype Interface for the web site.
6. HRV-profile generator: Web interface.
7. Summary of Links to the project's outcomes: Slack, Trello, GitHub.
8. Proposed steps from UAB for next deliverables.
9. Questions from UAB.
10. Questions from Laurea.
11. Agreement on next deliverable.
12. Update of the agenda.

4.4 Tools

Due to being a large project and made up of different parts, we have decided to use the following list of software to be able to develop the project.

1. **Trello [11]:** Online software, used to implement the agile methodology, is a free software, has the basic functionalities to implement the agile methodology, define Sprint, tasks, calendars, etc.
2. **Slack [12]:** Free and online software, used to communicate between different members of the project and share resources and information about the project.
3. **Skype [13]:** Free software, used to make meetings online, during this project we have used the Skype for Business version.
4. **Github [14]:** Free software, used as the repository of the Android application, in the reference you can find the link of the repository.
   In order to use the functionality of storing code in private, the professional version with the student license had to be ordered.
5. **Android Studio [15]:** Free software, used to develop the Android application, works in java, it also has an Android device simulator. The codes implemented in the application can be found in the Github.
6. **Xampp [16]:** Free software, used to create a local server, which allows you to configure the PhpMyAdmin server, to be able to communicate between Android and the database.
7. **PhpMyAdmin:** Server that is part of the Xampp application, used to configure the database system in MySQL.
8. **Notepad++ [17]:** Free software, basically used to encode the .html and .php files, also used to take notes and ideas in the meeting.

5 STRESS LEVEL AND SLEEPING QUALITY MEASUREMENT

5.1 Heart Rate Variability (HRv)

5.1.1 General definition

A healthy heart beat contains healthy irregularities. Even if your heart rate is, say, 60 beats per minute, that doesn't mean that your heart beats once every second – or at one-second intervals like a clock Fig .3.
Rather, there is variation among the intervals between your heartbeats. The interval between your successive heartbeats can be, for example, 0.85 seconds between some two succeeding beats and 1.35 seconds between some other two.

5.1.2 RMSSD

Even though the difference is measured in parts of seconds, you can actually feel the difference.
Here's a tip for anyone who wants to experience it: place a finger gently on your neck or wrist and find your pulse.

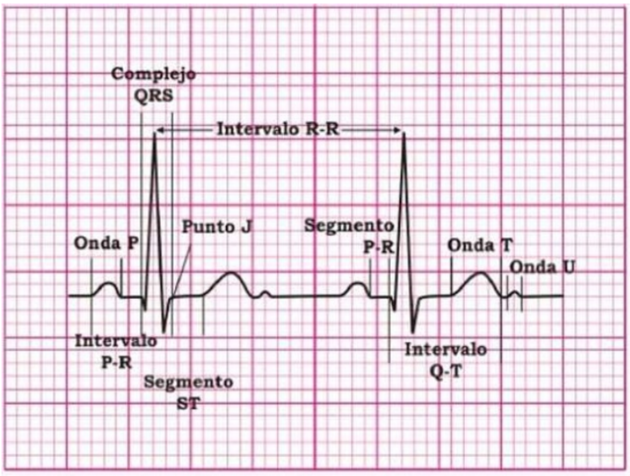


Fig. 3: Electrocardiogram

You should feel that the longest intervals take place when you exhale, and the shortest intervals when you inhale.

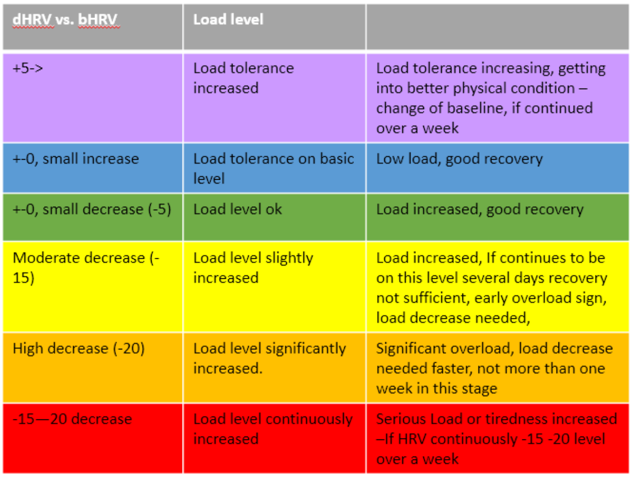
There are many ways to calculate HRV, in this project the RMSSD algorithm is implemented, since it is considered more stable. In the following equation eq:1 you can see the algorithm of RMSSD. In this project it was decided to calculate in an interval of 10 minutes.

$$\sqrt{\frac{1}{N-1}\left(\sum_{i-1}^{N-1}((R-R)_{i+1}-(R-R)_i)^2\right)}$$

eq:1 RMSSD Algoritm

5.2 Stress level

The level of stress indicates that this stress is the user, in this project we have defined that there are 6 levels of stress. To determine which level the user is located, it is necessary to compare the average daily value of HRV with HRV baseline of own user. In the following table Fig. 4 you can see which range of values belongs to each level.

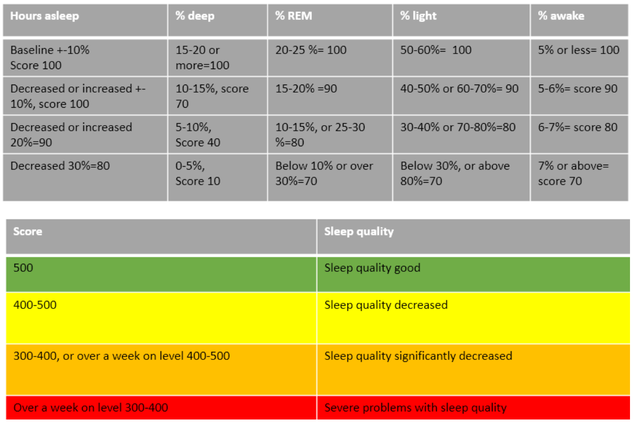


dHRV vs. bHRV	Load level	
+5->	Load tolerance increased	Load tolerance increasing, getting into better physical condition – change of baseline, if continued over a week
+-0, small increase	Load tolerance on basic level	Low load, good recovery
+-0, small decrease (-5)	Load level ok	Load increased, good recovery
Moderate decrease (-15)	Load level slightly increased	Load increased, If continues to be on this level several days recovery not sufficient, early overload sign, load decrease needed,
High decrease (-20)	Load level significantly increased.	Significant overload, load decrease needed faster, not more than one week in this stage
-15—20 decrease	Load level continuously increased	Serious Load or tiredness increased –If HRV continuously -15 -20 level over a week

Fig. 4: Stress level table

5.3 Sleep quality

The quality of sleep is a value between 0 and 500, where 500 is the best quality, and is divided into 4 levels. Apart from the levels, it is also classified into 4 states, light sleep, deep sleep, REM sleep, and awake. To calculate the quality of sleep you need the value of HRV detected in every 10 minutes and the pulsation speed that is basically low or high, from these two values mapping in the state table you can determine in which sleep state it is locate the user. And finally it depends on the percentage of time each state can be mapped in the sleep quality table. In the following table Fig.5 represents the table of sleep quality.



Hours asleep	% deep	% REM	% light	% awake
Baseline +-10% Score 100	15-20 or more=100	20-25 %= 100	50-60%= 100	5% or less= 100
Decreased or increased +-10%, score 100	10-15%, score 70	15-20% =90	40-50% or 60-70%= 90	5-6%= score 90
Decreased or increased 20%=90	5-10%, Score 40	10-15%, or 25-30 %=80	30-40% or 70-80%=80	6-7%= score 80
Decreased 30%=80	0-5%, Score 10	Below 10% or over 30%=70	Below 30%, or above 80%=70	7% or above= score 70

Score	Sleep quality
500	Sleep quality good
400-500	Sleep quality decreased
300-400, or over a week on level 400-500	Sleep quality significantly decreased
Over a week on level 300-400	Severe problems with sleep quality

Fig. 5: Sleep quality table

6 HRV GENERADOR

6.1 Justification for a HRv generator

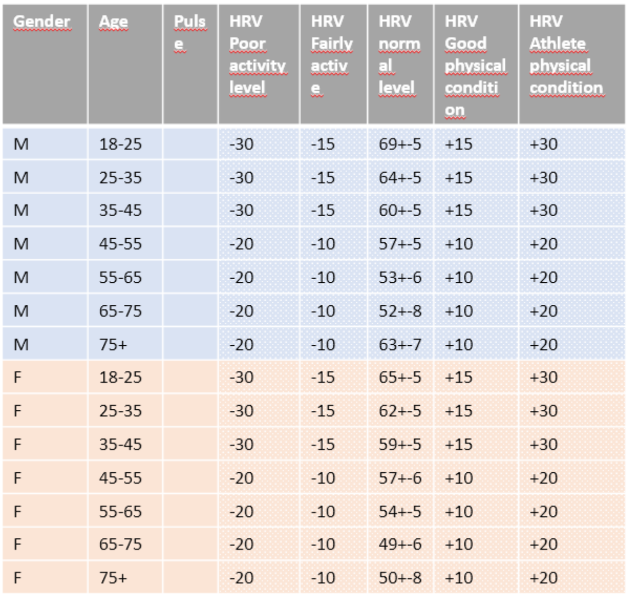
The application must be connected with an external device in order to detect the heart rate to be able to complete all the functionalities, but at the moment we have not been able to obtain any device, for this reason we have decided to implement a data generator application of HRV value and a model generator of HRV value.

Basically the data generator generates HRV data by mapping a HRV Baseline table, and it will be stored directly to the data base. And the model generator will use this data to train and generate a JSON file model, this model can be transformed into a new HRV Baseline table and used in the data generator and is also introduced to the application in order to simulate real data of HRV replacing the physical devices that detect the heart rate.

6.2 Aplication

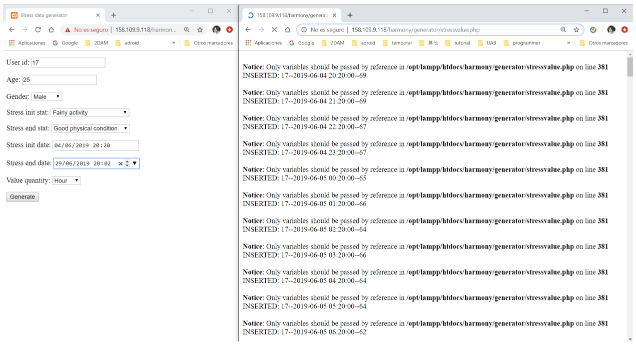
6.2.1 HRV Generador: Lookup table mapping

The data is generated from the conditional algorithm, which is extracted by the table Fig. ref fig: tabla$_HRV_Baseline_bHRV, basicallyc$



Gender	Age	Pulse	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25		-30	-15	69+-5	+15	+30
M	25-35		-30	-15	64+-5	+15	+30
M	35-45		-30	-15	60+-5	+15	+30
M	45-55		-20	-10	57+-5	+10	+20
M	55-65		-20	-10	53+-6	+10	+20
M	65-75		-20	-10	52+-8	+10	+20
M	75+		-20	-10	63+-7	+10	+20
F	18-25		-30	-15	65+-5	+15	+30
F	25-35		-30	-15	62+-5	+15	+30
F	35-45		-30	-15	59+-5	+15	+30
F	45-55		-20	-10	57+-6	+10	+20
F	55-65		-20	-10	54+-5	+10	+20
F	65-75		-20	-10	49+-6	+10	+20
F	75+		-20	-10	50+-8	+10	+20

This data will be used to train a new model of HRV in the model generator, and it can also be used to show it in the application, since several sections of the application communicate with the database in order to generate user reports. Using this data generator it is very simple to generate this data, since it allows the user to configure the scene and simulate all HRV data over a long period. In the following image Fig. 7 you can see the interface and the result of the data generator.


Stress data generator
No es seguro | 158.109.9.118/harmon...
Aplicaciones
Google
2DAM
adroid
Otros marcadores
158.109.9.118/harmony/generato
No es seguro | 158.109.9.118/harmony/generator/stressvalue.php
temporal
其他
tutorial
UAB
programmer

User id: 17

Age: 25

Gender: Male

Stress init stat: Fairly activity

Stress end stat: Good physical condition

Stress init date: 04/06/2019 20:20

Stress end date: 29/06/2019 20:02

Value quantity: Hour

Generate

```
Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-04 20:20:00--69

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-04 21:20:00--69

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-04 22:20:00--67

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-04 23:20:00--67

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 00:20:00--65

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 01:20:00--66

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 02:20:00--64

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 03:20:00--66

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 04:20:00--64

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 05:20:00--64

Notice: Only variables should be passed by reference in /opt/lampp/htdocs/harmony/generator/stressvalue.php on line 381
INSERTED: 17--2019-06-05 06:20:00--62
```

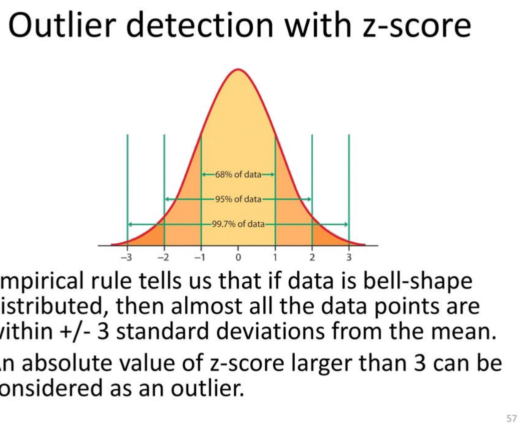
Fig. 7: HRV generador

6.2.2 Model generador with bayesian networks and outlier filtering

The model generator is used to generate the HRV Baseline table, basically many data is entered, and the outlier detector is used to eliminate the anomalies, and the Bayesian network is applied to generate the model, export it in the JSON file, and this file it can be converted into the new HRV Baseline table, which will be used in the data generator or to simulate the cardiac frequency detection device in the Android application.

6.2.3 Detection (8)

Outlier detection is the process of detecting and subsequently excluding outliers from a given set of data.
An outlier may be defined as a piece of data or observation that deviates drastically from the given norm or average of the data set. An outlier may be caused simply by chance, but it may also indicate measurement error or that the given data set has a heavy-tailed distribution.
In the project is used in the implementation of the generator, basically is to filter all the anomalies of the dataset, to be used later in the training of the generator model.



Outlier detection with z-score

- Empirical rule tells us that if data is bell-shape distributed, then almost all the data points are within +/- 3 standard deviations from the mean.
- An absolute value of z-score larger than 3 can be considered as an outlier.

57

Fig. 8: Outlier Detection

6.2.4 Bayesian network

Bayesian networks (BNs), also known as belief networks (or Bayes nets for short), belong to the family of probabilistic graphical models (GMs). These graphical structures are used to represent knowledge about an uncertain domain. In particular, each node in the graph represents a random variable, while the edges between the nodes represent probabilistic dependencies among the corresponding random variables. These conditional dependencies in the graph are often estimated by using known statistical and computational methods. Hence, BNs combine principles from graph theory, probability theory, computer science, and statistics.
In the project, this algorithm is implemented to generate the model of HRV value generator, due to its node structure, where statistics are applied with probability, it is possible to simulate the value of HRV through user parameters: age, gender , the level of health and physical activity performed.

7 Proposed prototype system

En the Fig. 9 you can see the relation between each part of prototype system.

7.1 Remote server

To get the application working anywhere, we have implemented a remote server using the CVC computer. Basically it consists of an ordered linux, where it has the web port (port 80) and the SSL port (port 22345) open to the public and a fixed ip. With these configurations and the XAMPP application that provides the Apache and the MySQL server, it allows us to create a remote server that allows the connection functionality of the application.
Currently the way to work on this server is through an SSL connection, using the Putty application to perform SSL

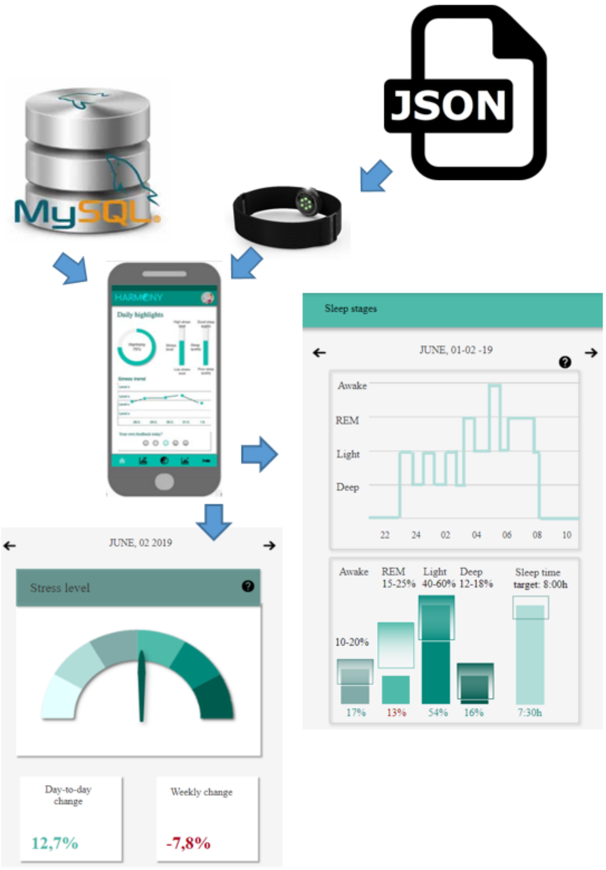
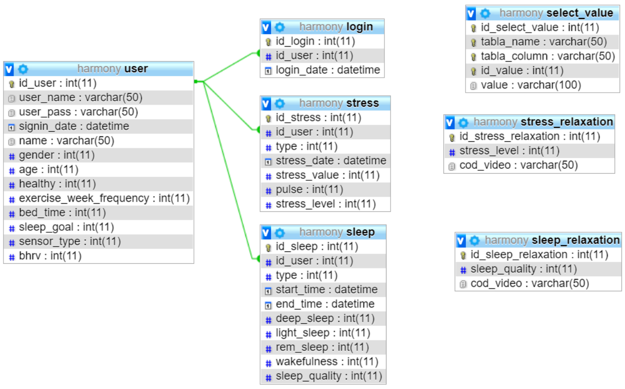


Fig. 9: Project Structure

Linux commands, and to upload and download files using the Filezilla application.

7.2 Database

For this project they have decided to use MySQL, instead of SQL, because this is free software, and its simplicity to implement and communication with other platforms.
It is currently configured on PhpMyAdmin server of Xampp.
The name of the database is called Harmony, just like the project. And in the Fig. (1) you can see the relational diagram of the tables .



harmony **user**
id_user : int(11)
user_name : varchar(50)
user_pass : varchar(50)
signin_date : datetime
name : varchar(50)
gender : int(11)
age : int(11)
healthy : int(11)
exercise_week_frequency : int(11)
bed_time : int(11)
sleep_goal : int(11)
sensor_type : int(11)
bhrv : int(11)

harmony **login**
id_login : int(11)
id_user : int(11)
login_date : datetime

harmony **stress**
id_stress : int(11)
id_user : int(11)
type : int(11)
stress_date : datetime
stress_value : int(11)
pulse : int(11)
stress_level : int(11)

harmony **sleep**
id_sleep : int(11)
id_user : int(11)
type : int(11)
start_time : datetime
end_time : datetime
deep_sleep : int(11)
light_sleep : int(11)
rem_sleep : int(11)
wakefulness : int(11)
sleep_quality : int(11)

harmony **select_value**
id_select_value : int(11)
tabla_name : varchar(50)
tabla_column : varchar(50)
id_value : int(11)
value : varchar(100)

harmony **stress_relaxation**
id_stress_relaxation : int(11)
stress_level : int(11)
cod_video : varchar(50)

harmony **sleep_relaxation**
id_sleep_relaxation : int(11)
sleep_quality : int(11)
cod_video : varchar(50)

Fig. 10: Harmony relational diagram

1. **Tabla: select_value:** In this table all the constant values of type text that can appear in other tables are stored, for example the gender field of the user table, or the description field of the stress_exercise table, in this way if we change the text value by theme of translation will not affect the information stored.
2. **Tabla: user:** This table stores all the user's values, basically the features that you can configure in the Android application. That is to say the values of personal profile and profile of the activities, for example the age, level of health, time of sleep, the type of sensor that has, etc.
   A part of this data is also stored the username and password.
   Basically with this data it is possible to calculate the HRV values of each specific user.
3. **Tabla: login:** This table stores the user's connection to the application. This way you can verify when a user has connected to the application.
4. **Tabla: stress:** In this table, all HRV values and the pulse taken from each user are stored. Basically it saves the values of HRV, the pulse, the time that had taken these values and the type. The type field indicates whether this value is calculated in 10 minutes, or is an average in hour, day or month.
5. **Tabla: stress_relaxation:** This table stores different YouTube video codes, related to your stress levels.
6. **Tabla: sleep:** In this table, all the values related to the quality of the user's sleep are stored. Basically it stores the value of sleep quality, the start date of sleep and the date of the end of sleep, the percentage of time of deep sleep, light sleep, rapid eye movement sleep and awake, there is also a field of type , which indicates whether all these values are taken in hour, or is an average of day or month.
7. **Tabla: sleep_relaxation:** In this table different video codes of YouTube are stored, related to the quality of the dream.

7.3 Android Application

The Android application is implemented in the IDE Android Studio, and will be ready for the Android system with the SDK version higher than 16, and with the 5.4-inch screen. The application basically consists of connecting the user, detect heart rate, show the results of stress in graphical form, calculate the quality of sleep and also show it in graphic form, apart from a breathing exercise module to recover stress and sleeping tips to improve the quality of sleep.
The application to implement the design of model-view-controller, although in Android Studio, the views are the .xml files, and the drivers use these views, in the following image Fig. 11 you can see the class diagram of the application. And in Fig. 13 of the section of appendices you can see the class diagram of the mobile aplication.
And in the appendix section the wireframe of the application and the prototype of the main screens of the application are available: Fig. 14 and Fig.15.

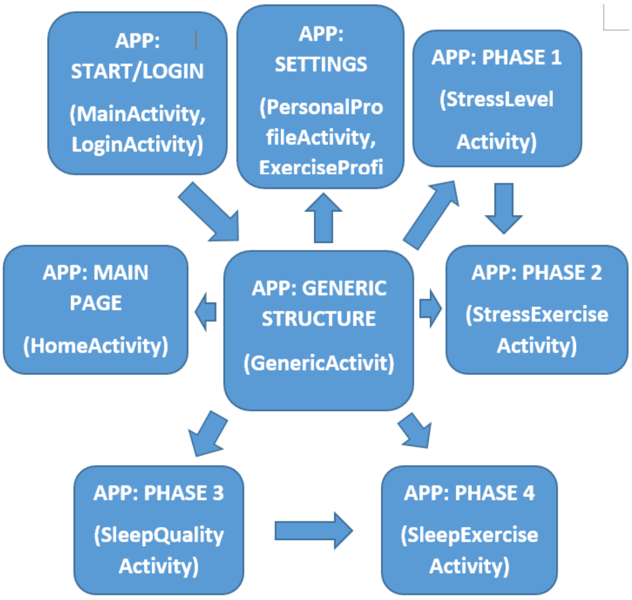


Fig. 11: Sequence diagram of the application screens

7.3.1 Model

In the model there are classes that store information for the application, for example user values, stress data, PHP links, etc.

1. **HearSensorService class:** This is a kind of service, where it runs in the background, basically to take the value of the user's pulse and stress, and upload to the server.

2. **ConstantValue class:** In this class the constant values are stored, for example the file paths and the default breathing time.

3. **DB_Key class:** In this class, the names of the tables and fields of the data base are stored.

4. **DB_Url class:** In this class the PHP links are stored in order to communicate with the database.

5. **HttpJsonParser class:** This is the class that makes the communication between the application and the data base, receives as a parameter the PHP link, and through the post method, sends requests to the server, and the server responds with the necessary information.
   All the information that circulates during the communication is encoded in JSON structure, since it is a quite powerful and simple structure for the transmission of data.

6. **Util class:** This is the class that has simple functions, but they are called in many parts of the application without needing to instantiate in an object, for example convert text to date of certain format, etc.

7. **VariableValue class:** This class stores data that are shared throughout the application, for which the Singleton pattern is implemented, so you do not need to create an object each time you make the reference.

7.3.2 Dialogue

In this package, basically, the view classes of the dialogues. El HarmonyDialog, MessageDialog, DownloadStressDataDialog, DownloadSleepDataDialog. In these views it is to be able to show alert messages, or to be able to download data.

7.4 Controller - View

The controller module consists of different kinds of type Activity, and each of these classes is linked to one of the different screens of the application, where it is linked by one more .xml files.

The application sequence diagram is available in the appendix section (11).
The view module consists basically of different .xml files, where each is a screen of the application, except some that is made up of several .xml files.

1. **MainActivity:** The application's home screen, where it connects to the data base to obtain constant values, for example sensor types, video codes, etc.

2. **LoginAcitivity:** The connection screen of the user, before performing the action to connect, is checked if the user file exists, if this file exists, the application will be automatically connected, in this way the user will now enter the email and password.
   Once the user is connected, he connects again with the data base to obtain the user's stress values.

3. **PersonalProfileActivity:** The personal profile screen, this screen is to register the user in case it is the first time you connect, and to modify the parameters of the profile.

4. **ExerciseProfileActivity:** The configuration screen is to modify the user's activity parameters.

5. **HomeActivity:** The home screen is to show the value of stress and quality of sleep in real time, with the detection of the heart rate and the state of the day according to the average value between the level of stress and the quality of sleep.
   There also appears warning dialogs when stress values exceed a limit.

6. **StressLevelActivity:** In this screen it is basically to show the result of value of the stress in average of hour, day and month and to show in graphic form.

7. **StressExerciseActivity:** This is the screen to perform breathing exercises, simulates the real situation, and provides certain exercises to recover from stress. It also provides youtube that helps to improve the level of the user's stress.

8. **SleepQualityActivity:** This screen is also to show the results of the values, but about sleep quality, also in the form of graphics.

9. **SleepExerciseAcitivity:** This screen simply shows some tips on how you can improve the quality of sleep.

It also provides youtube that helps to improve the level of the user's stress.

10. **GenericScreen:** This screen is basically a generic screen, which appears on almost all other screens, provides buttons to change the screen.

7.4.1 External library

During the development of the application, I have had to include several external libraries to facilitate the implementation of the project.

1. **MPAndroidChart [26]:** A graphic drawing library for Android Studio, provides many types of graphics, and is very simple to use.

2. **Cardiomood [27]:** a library that contains several indicators tools, for this project we have used the tool of SpeedometerGauge to indicate the value of stress.

3. **Youtube Api [28]:** the youtube library, which allows you to insert YouTube videos within the application. To use this library, it is necessary to register the project in the Google API reference, and also download the .jar file.

8 Results

This section shows the result of the two tests carried out to check whether the two generators: HRV generator and model generator for HRV work correctly.

In the Fig.16, Fig.17 and Fig.**??** of the appendix section you can see the display of the HRV data generated of stress level of a single user in a year. All these data is generated by the HRV generator, modified the paramenter of time, day and month.

In all these tests I have extracted the value of the level of health of the user, since the tests, tables and results will not be very extensive.

8.1 Test 1

This test is to assess whether the HRV generator is working correctly. Basically it is to introduce 3 HRV tables, and on each one generate several data between different user profiles.

The 3 tables introduced are: the first table with standard values of HRV, the second table with wider values to standard of HRV and the last table with short to standard values.

During the test, 4 users will be created, two men and two women with a large age variation, on each user one hour of data will be generated in which there are 6 values in each level of stress. In other words, at the end we will obtain 360 HRV values.

In the Fig.19 and the Fig.20 of the appendix section you can see the three tables entered and the generator results using the same user profile.

From these results it will be possible to observe that the data generator works correctly since it generates the values marked by the tables.

8.2 Test 2

In this test is to evaluate the model generator, we basically use the HRV generator of previous test with 3 type of table, these datas will be exported from CVS data base and will be introduced in the model generator, and 3 model files will be generated in format JSON.

In this case we have created more users with different profiles for general complex models, .

In the Fig.21, Fig.22 and Fig.23 of the appendix section you can see the 3 models generated in the form of tables to be able to compare with the tables that are introduced to the HRV generator.

From these results it will be possible to observe that the models generated are very similar to the tables introduced, although there are some small differences in values, this is due to the lack of more data to train, since this generator in the first step is filtered a part of the data using the outlier detector, because it can be considered anomalies, basically a part of the data of the two ends is eliminated. Even so, depending on the result, we can consider that this generator of the model works correctly.

9 Conclusions

The project is based on different sections: The Android application that is the core of the project in which it has the functionality to calculate the user's level of stress and sleep quality, show these results to users and provide tips to improve these values. The remote server that allows connection functionality with MySQL server, and with this a multiuser application is allowed. The MySQL server, which allows the application to upload and download data. The application of generating data, which allows to train data and generate a Json model, and with this model it is possible to simulate real data.

Apart from these, the data generated by the application can also be used to perform statistical studies between different user profiles, for example the ability to stress between ages or between genders.

10 Future developments

This project is a quite big project, due to the lack of time, many extra features have not been able to be implemented.

One of the functionalities that can be implemented for the future is to include the hardware device designed with Arduino, in this way it is possible to get real HRV values from the user. And another very interesting feature is to implement a Youtube video recommender, the application calculates the level of stress or quality of sleep, and search on Youtube for an appropriate video that improves the situation, but for this functionality it would take time and data to be able to train the model through Machine Learning.

11 ACKNOWLEDGEMENTS

This project is a research project between the UAB university and the Laurea University, it is a quite complex project. For this reason I would like to thank my tutor Fernando Variño, who gave me this opportunity to participate in this

project where I took advantage of many things that I acquired in the four years of the degree, and especially the new knowledge I acquired during the project research.
On the other hand, I also have to thank Master's student from Laurea University, who helped me a lot in the development of the project.

References

[1] Autonomous University of Barcelona. https://www.uab.cat/ [2019/03/09]

[2] Laurea University. https://www.laurea.fi/en/ [2019/03/09]

[3] Sari Kivine, Taru Satuli, Heidi Strang. Design Management - a collaboration project with UAB [2019/06/26]

[4] Polar. https://www.polar.com/es [2019/03/09]

[5] StressScan. https://apps.apple.com/us/app/stressscan-check-your-stress/id673503525 [2019/03/09]

[6] Welltory. https://welltory.com/ [2019/03/09]

[7] SleepCycle. https://www.sleepcycle.com/ [2019/03/09]

[8] RuntasticSleep. https://www.runtastic.com/ [2019/03/09]

[9] Agile methodology.https://linchpinseo.com/the-agile-method/ [2019/03/09]

[10] Marc Stickdorn, Markus Horness, Adam Lawrence, Jakob Schneider. 2018 This Is Service Design Methods: A Companion to This Is Service Design Doing.

[11] Trello. https://trello.com/ [2019/05/22]

[12] Slack. https://slack.com/ [2019/05/22]

[13] Skype. https://www.skype.com/ [2019/05/22]

[14] Github: Harmony. https://github.com/XYxXT/Laurea_UAB_Project [2019/05/22]

[15] Android Studio. https://developer.android.com/studio [2019/05/22]

[16] Xampp. https://www.apachefriends.org/es/index.html [2019/05/22]

[17] Notepad ++. https://notepad-plus-plus.org [2019/05/22]

[18] Programación android. https://developer.android.com/about [2019/03/09]

[19] MySQL. https://www.mysql.com/ [2019/03/09]

[20] Programación PHP. https://www.php.net/ [2019/04/08]

[21] Formato JSON. https://www.w3schools.com/js/js_json_intro.asp [2019/04/08]

[22] Recomender of movie. https://gallery.azure.ai/Experiment/Recommender-Movie-recommendation-3 [2019/04/08]

[23] Fred Shaffer1, J. P. Ginsberg. An Overview of Heart Rate Variability Metrics and Norms, Front Public Health, PMC [2019/04/08]

[24] Kim HG, Cheon EJ, Bai DS, Lee YH, Koo BH. Stress and Heart Rate Variability: A Meta-Analysis and Review of the Literature. Front Public Health, PMC [2019/04/08]

[25] Api de Polar. https://github.com/polarofficial/polar-ble-sdk [2019/04/08]

[26] MPAndroidChart. https://github.com/PhilJay/MPAndroidChart [2019/04/08]

[27] Cardiomood. https://github.com/ntoskrnl/AndroidWidgets [2019/04/08]

[28] Youtube api. https://developers.google.com/youtube/v3/ [2019/04/08]

APÈNDIX

A.1 Diagrama de Gantt

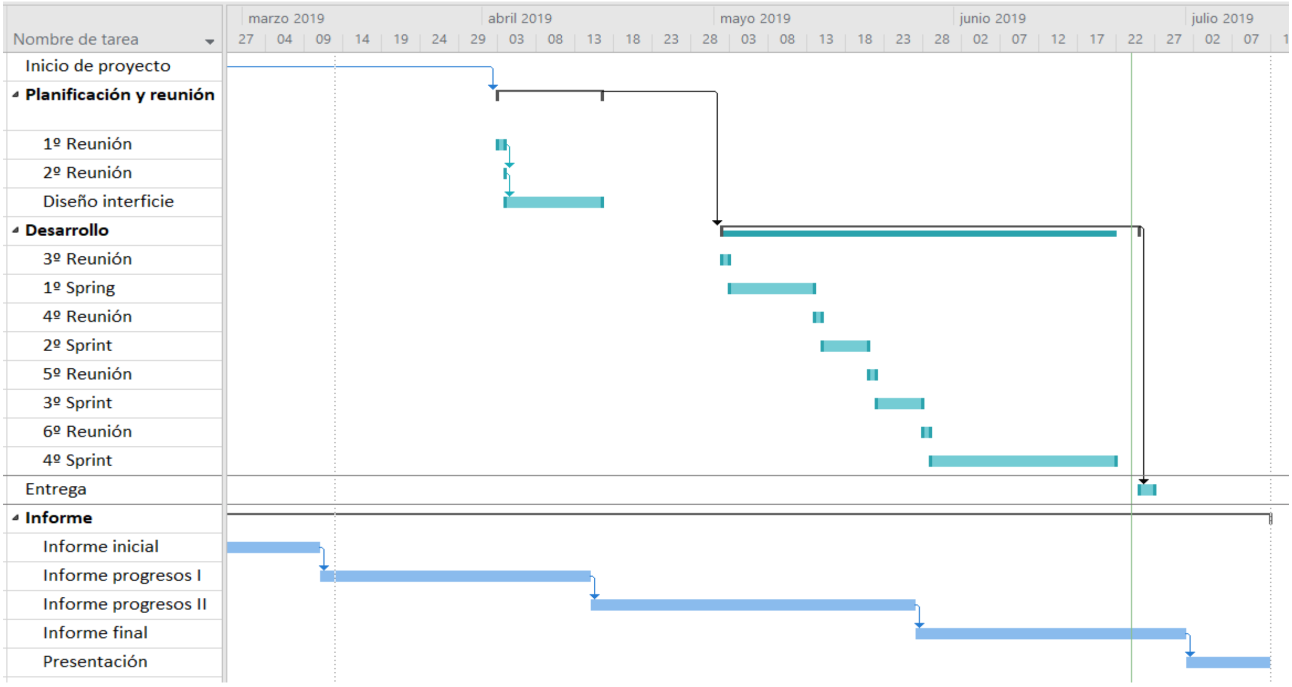


Fig. 12: Diagrama de gantt

A.2 Diagrama de clase

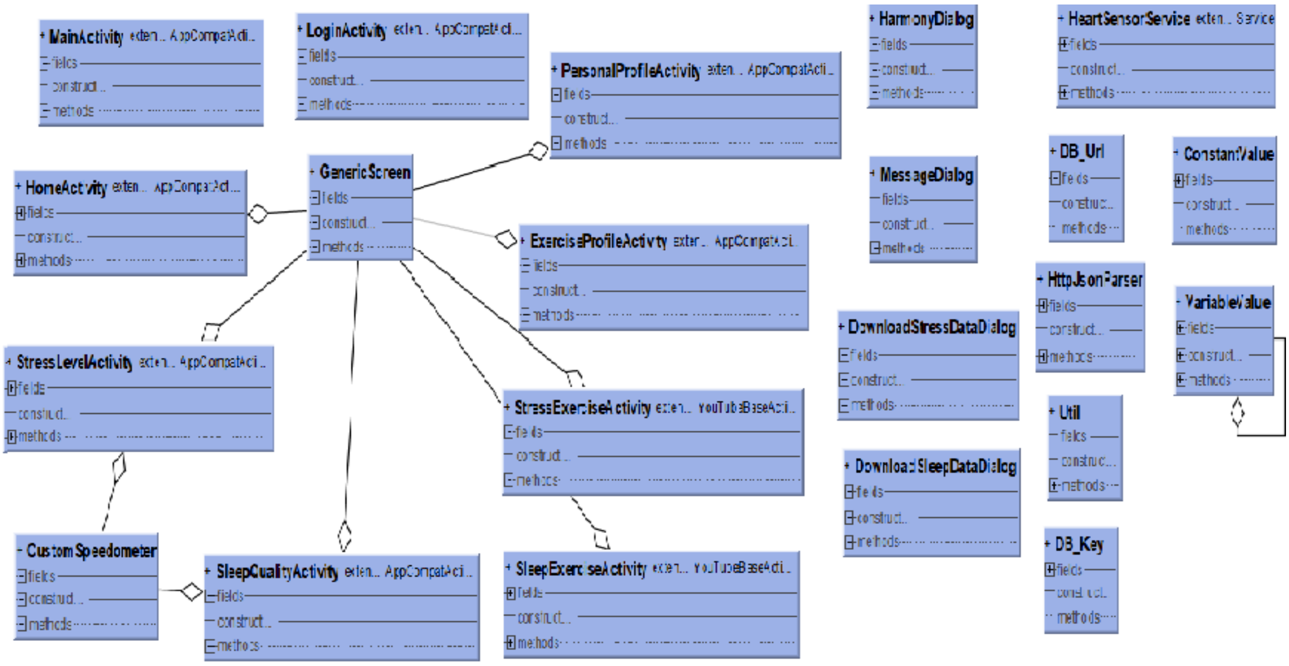

MainActivity extends AppCompatActi...
LoginActivity extends AppCompatActi...
PersonalProfileActivity extends AppCompatActi...
HarmonyDialog
HeartSensorService extends Service
HomeActivity extends AppCompatActi...
GenericScreen
ExerciseProfileActivity extends AppCompatActi...
MessageDialog
DB_Url
ConstantValue
HttpJsonParser
VariableValue
StressLevelActivity extends AppCompatActi...
StressExerciseActivity extends YouTubeBaseActi...
DownloadStressDataDialog
Util
DownloadSleepDataDialog
CustomSpeedometer
SleepQualityActivity extends AppCompatActi...
SleepExerciseActivity extends YouTubeBaseActi...
DB_Key
fields
constructors
methods

Fig. 13: Diagrama de clase de la aplicación

A.3 Wireframes and prototype of mobile application

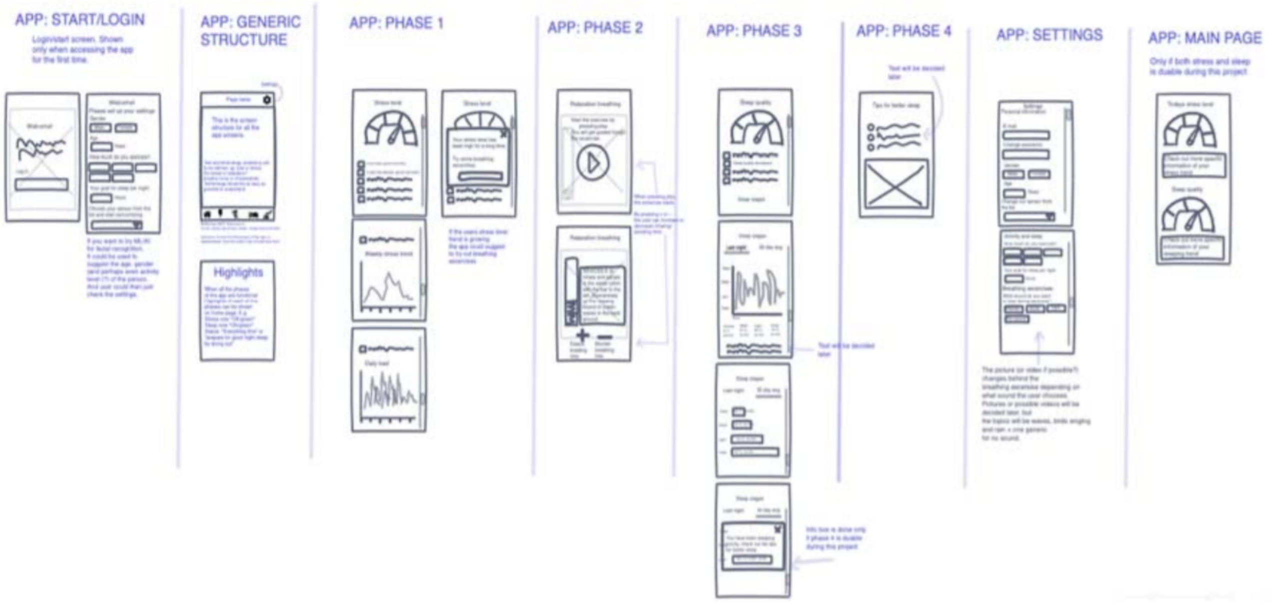
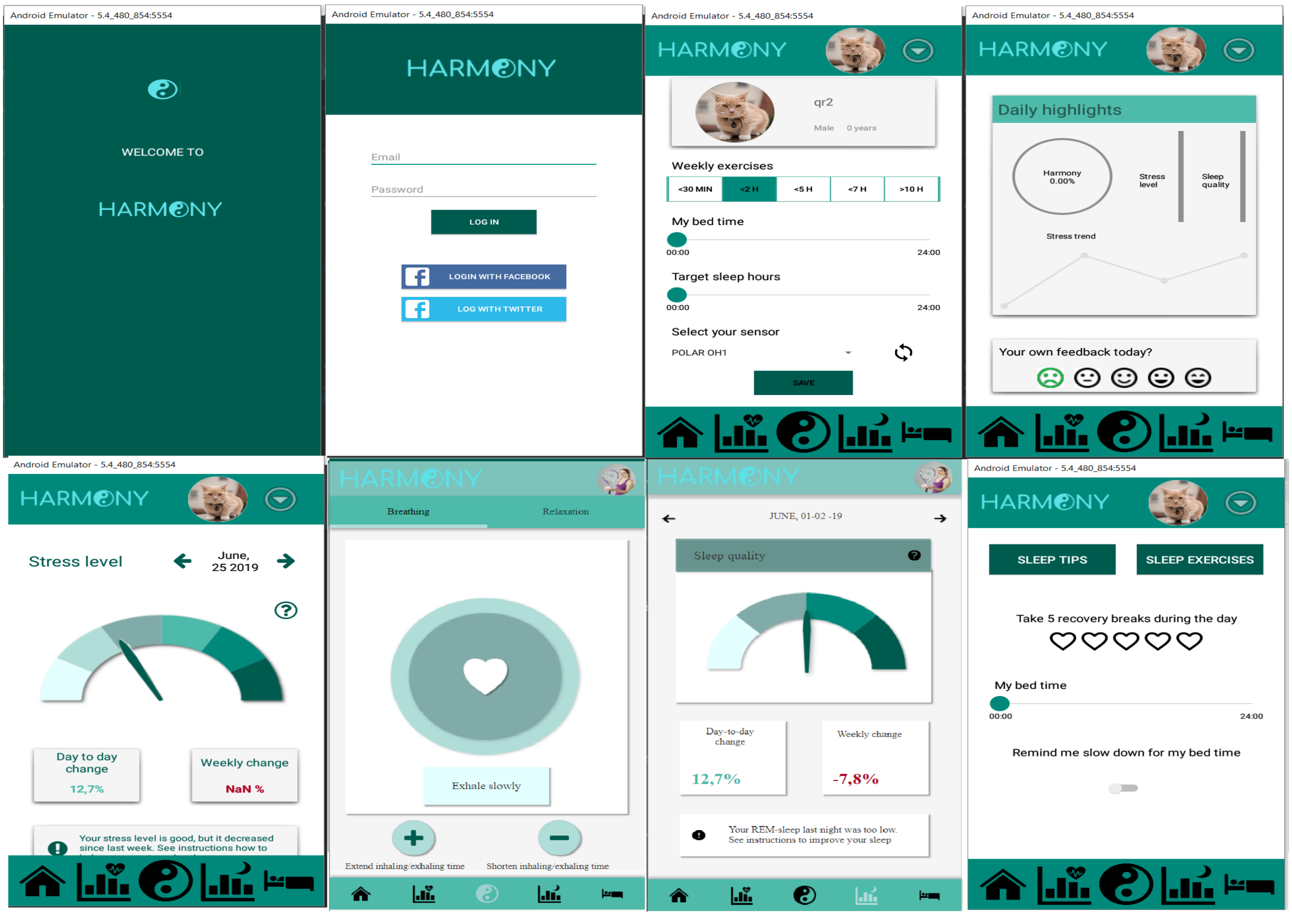


Fig. 14: Wireframes of the application, done with Invision



Android Emulator - 5.4_480_854:5554

WELCOME TO

HARMONY

Android Emulator - 5.4_480_854:5554

HARMONY

Email

Password

LOG IN

LOGIN WITH FACEBOOK

LOG WITH TWITTER

Android Emulator - 5.4_480_854:5554

HARMONY

qr2

Male 0 years

Weekly exercises

<30 MIN	<2 H	<5 H	<7 H	>10 H

My bed time

00:00 24:00

Target sleep hours

00:00 24:00

Select your sensor

POLAR OH1

SAVE

Android Emulator - 5.4_480_854:5554

HARMONY

Daily highlights

Harmony 0.00%

Stress level

Sleep quality

Stress trend

Your own feedback today?

Android Emulator - 5.4_480_854:5554

HARMONY

Stress level

June, 25 2019

Day to day change 12,7%

Weekly change NaN %

Your stress level is good, but it decreased since last week. See instructions how to

HARMONY

Breathing

Relaxation

Exhale slowly

Extend inhaling/exhaling time

Shorten inhaling/exhaling time

HARMONY

JUNE, 01-02 -19

Sleep quality

Day-to-day change 12,7%

Weekly change -7,8%

Your REM-sleep last night was too low. See instructions to improve your sleep

Android Emulator - 5.4_480_854:5554

HARMONY

SLEEP TIPS

SLEEP EXERCISES

Take 5 recovery breaks during the day

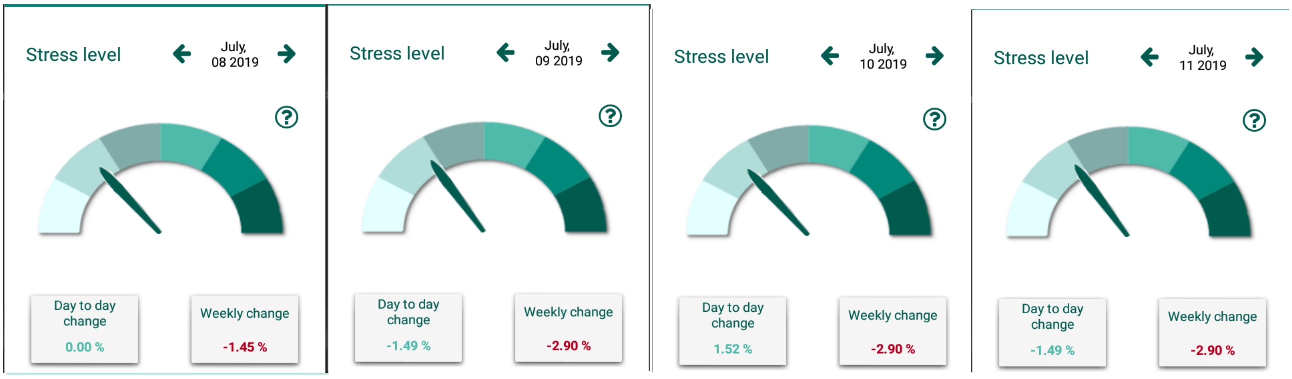
My bed time

00:00 24:00

Remind me slow down for my bed time

Fig. 15: Prototype of the application

A.4 Result: App



Stress level July, 08 2019

Day to day change 0.00 %

Weekly change -1.45 %

Stress level July, 09 2019

Day to day change -1.49 %

Weekly change -2.90 %

Stress level July, 10 2019

Day to day change 1.52 %

Weekly change -2.90 %

Stress level July, 11 2019

Day to day change -1.49 %

Weekly change -2.90 %

Fig. 16: Average daily value of stress level of 4 day

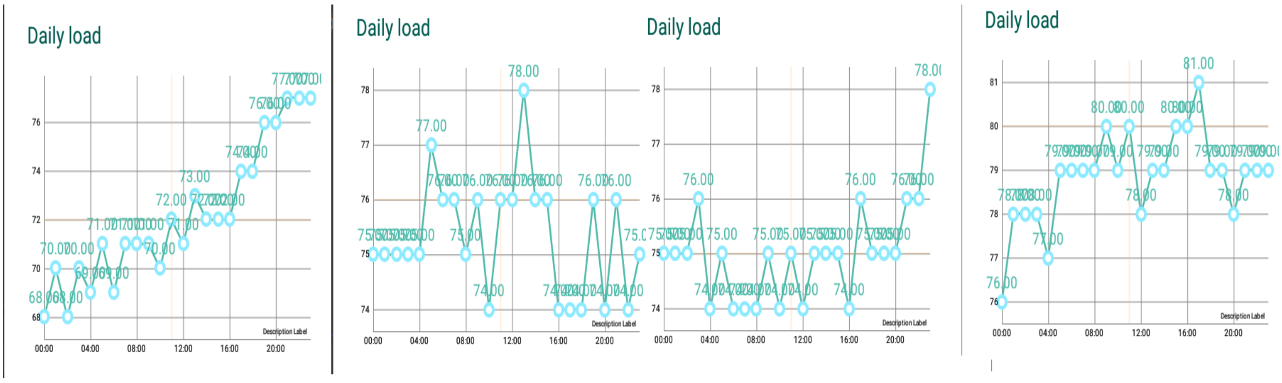
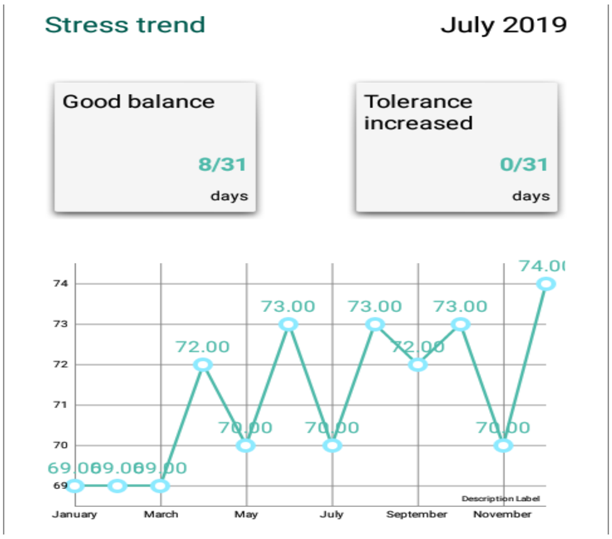


Fig. 17: Average hour value of stress level of 4 day



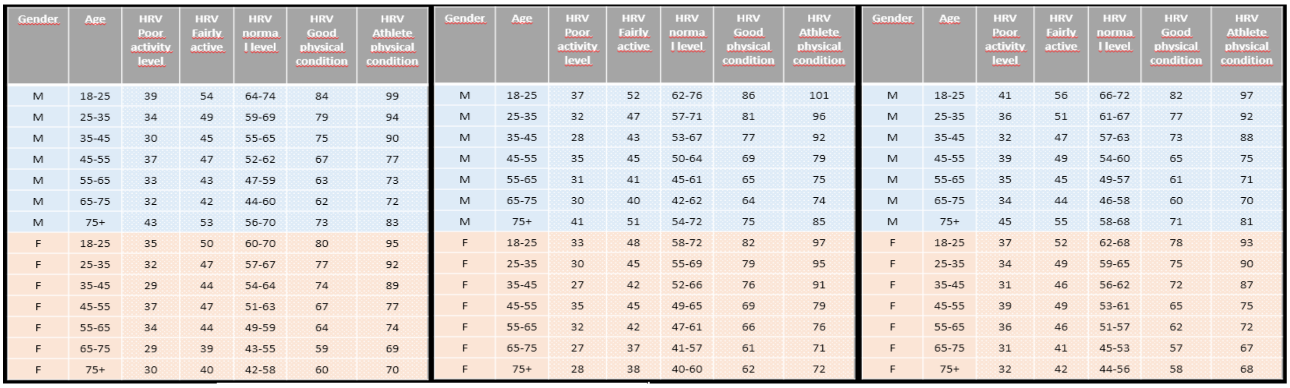
Stress trend July 2019

Good balance 8/31 days

Tolerance increased 0/31 days

Fig. 18: Average month value of stress level of year

A.5 Result: Test 1



Gender	Age	HRV Poor activity level	HRV Fairly active	HRV norma l level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	39	54	64-74	84	99
M	25-35	34	49	59-69	79	94
M	35-45	30	45	55-65	75	90
M	45-55	37	47	52-62	67	77
M	55-65	33	43	47-59	63	73
M	65-75	32	42	44-60	62	72
M	75+	43	53	56-70	73	83
F	18-25	35	50	60-70	80	95
F	25-35	32	47	57-67	77	92
F	35-45	29	44	54-64	74	89
F	45-55	37	47	51-63	67	77
F	55-65	34	44	49-59	64	74
F	65-75	29	39	43-55	59	69
F	75+	30	40	42-58	60	70

Gender	Age	HRV Poor activity level	HRV Fairly active	HRV norma l level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	37	52	62-76	86	101
M	25-35	32	47	57-71	81	96
M	35-45	28	43	53-67	77	92
M	45-55	35	45	50-64	69	79
M	55-65	31	41	45-61	65	75
M	65-75	30	40	42-62	64	74
M	75+	41	51	54-72	75	85
F	18-25	33	48	58-72	82	97
F	25-35	30	45	55-69	79	95
F	35-45	27	42	52-66	76	91
F	45-55	35	45	49-65	69	79
F	55-65	32	42	47-61	66	76
F	65-75	27	37	41-57	61	71
F	75+	28	38	40-60	62	72

Gender	Age	HRV Poor activity level	HRV Fairly active	HRV norma l level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	41	56	66-72	82	97
M	25-35	36	51	61-67	77	92
M	35-45	32	47	57-63	73	88
M	45-55	39	49	54-60	65	75
M	55-65	35	45	49-57	61	71
M	65-75	34	44	46-58	60	70
M	75+	45	55	58-68	71	81
F	18-25	37	52	62-68	78	93
F	25-35	34	49	59-65	75	90
F	35-45	31	46	56-62	72	87
F	45-55	39	49	53-61	65	75
F	55-65	36	46	51-57	62	72
F	65-75	31	41	45-53	57	67
F	75+	32	42	44-56	58	68

Fig. 19: Tables of test 1: Left-Standard, middle-wide and right-short

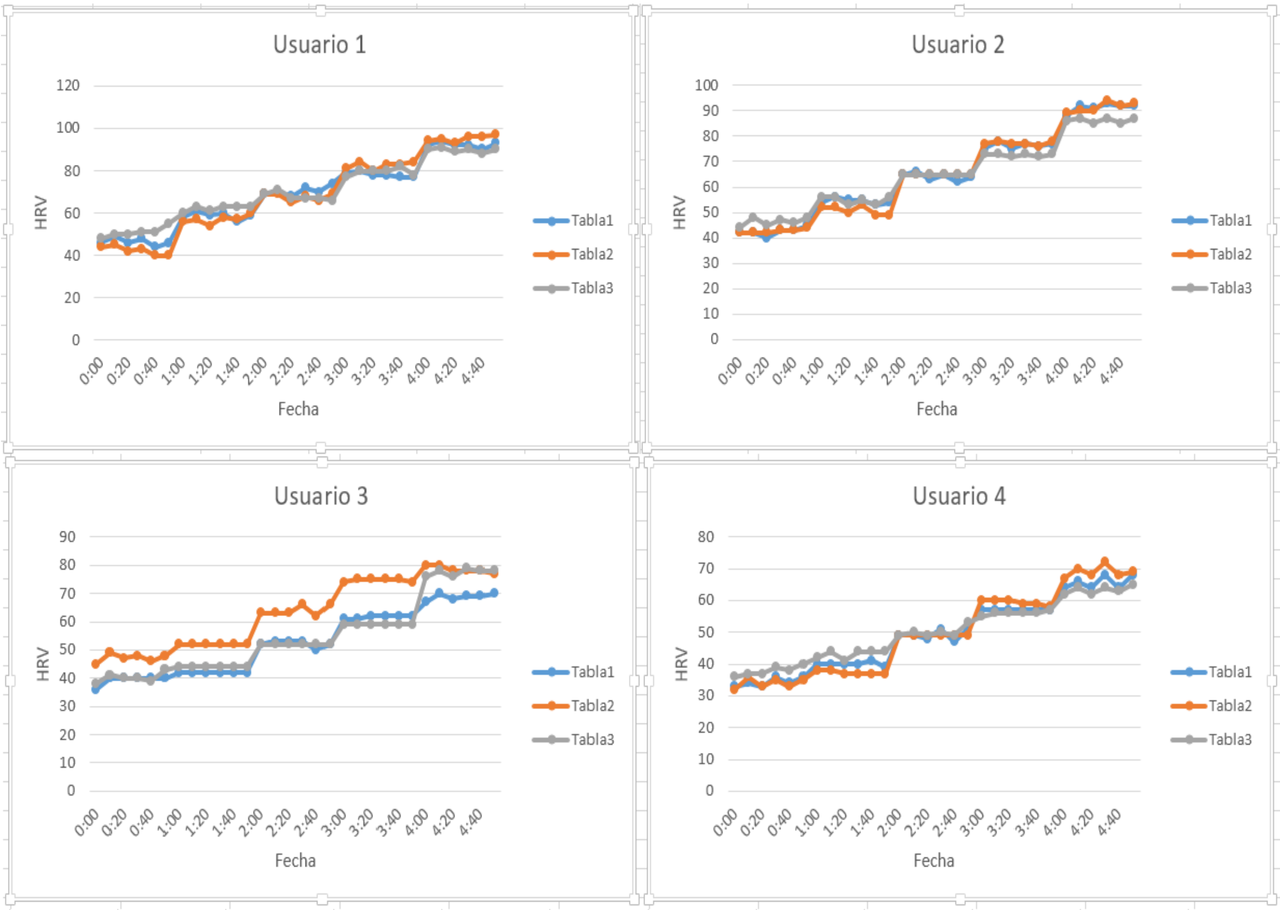
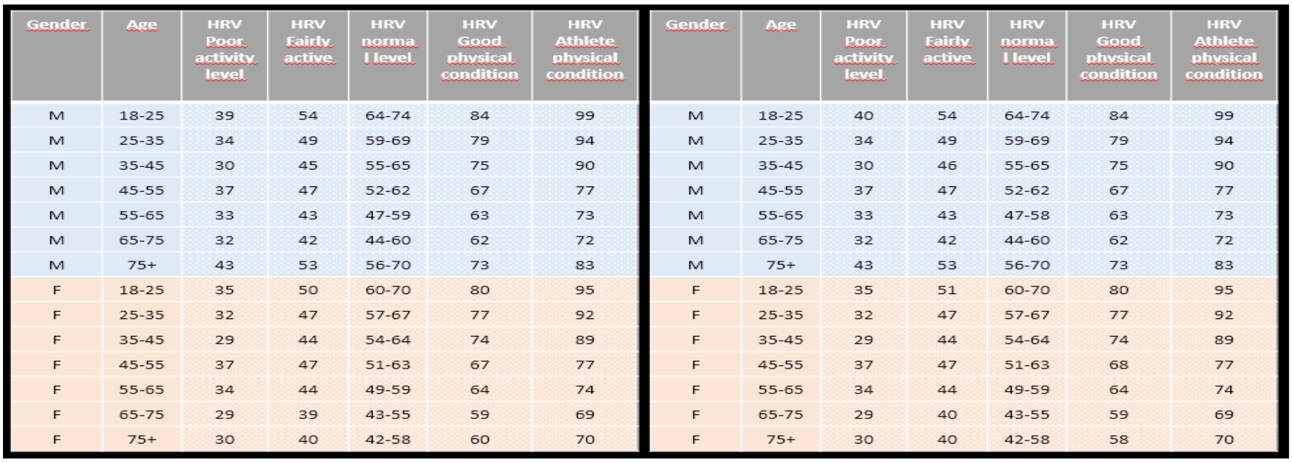


Fig. 20: HRV Values generate of test 1

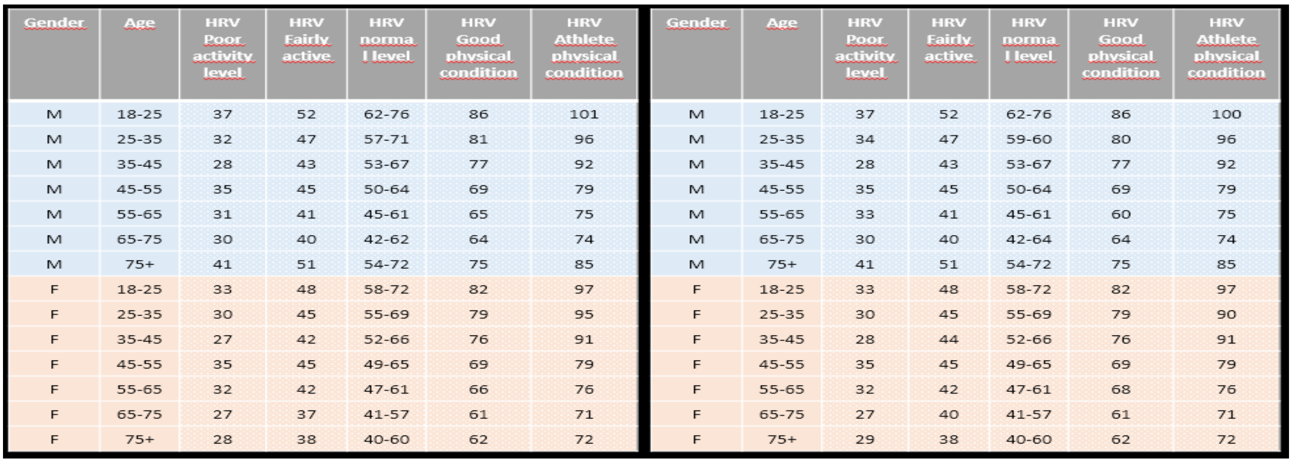
A.6 Result: Test 2



Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	39	54	64-74	84	99
M	25-35	34	49	59-69	79	94
M	35-45	30	45	55-65	75	90
M	45-55	37	47	52-62	67	77
M	55-65	33	43	47-59	63	73
M	65-75	32	42	44-60	62	72
M	75+	43	53	56-70	73	83
F	18-25	35	50	60-70	80	95
F	25-35	32	47	57-67	77	92
F	35-45	29	44	54-64	74	89
F	45-55	37	47	51-63	67	77
F	55-65	34	44	49-59	64	74
F	65-75	29	39	43-55	59	69
F	75+	30	40	42-58	60	70

Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	40	54	64-74	84	99
M	25-35	34	49	59-69	79	94
M	35-45	30	46	55-65	75	90
M	45-55	37	47	52-62	67	77
M	55-65	33	43	47-58	63	73
M	65-75	32	42	44-60	62	72
M	75+	43	53	56-70	73	83
F	18-25	35	51	60-70	80	95
F	25-35	32	47	57-67	77	92
F	35-45	29	44	54-64	74	89
F	45-55	37	47	51-63	68	77
F	55-65	34	44	49-59	64	74
F	65-75	29	40	43-55	59	69
F	75+	30	40	42-58	58	70

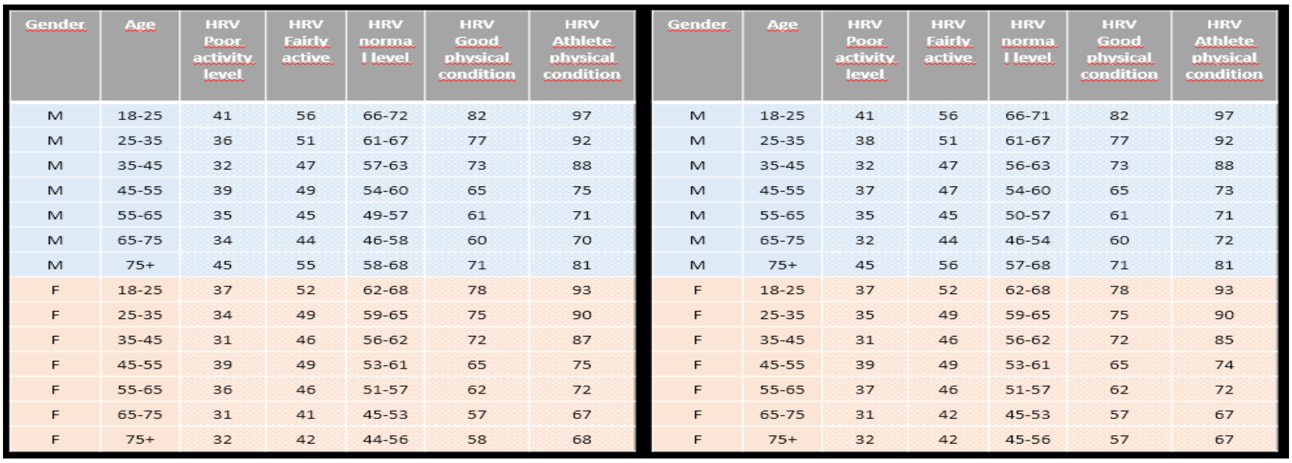
Fig. 21: Stantard: Left correct table - right generate table



Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	37	52	62-76	86	101
M	25-35	32	47	57-71	81	96
M	35-45	28	43	53-67	77	92
M	45-55	35	45	50-64	69	79
M	55-65	31	41	45-61	65	75
M	65-75	30	40	42-62	64	74
M	75+	41	51	54-72	75	85
F	18-25	33	48	58-72	82	97
F	25-35	30	45	55-69	79	95
F	35-45	27	42	52-66	76	91
F	45-55	35	45	49-65	69	79
F	55-65	32	42	47-61	66	76
F	65-75	27	37	41-57	61	71
F	75+	28	38	40-60	62	72

Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	37	52	62-76	86	100
M	25-35	34	47	59-60	80	96
M	35-45	28	43	53-67	77	92
M	45-55	35	45	50-64	69	79
M	55-65	33	41	45-61	60	75
M	65-75	30	40	42-64	64	74
M	75+	41	51	54-72	75	85
F	18-25	33	48	58-72	82	97
F	25-35	30	45	55-69	79	90
F	35-45	28	44	52-66	76	91
F	45-55	35	45	49-65	69	79
F	55-65	32	42	47-61	68	76
F	65-75	27	40	41-57	61	71
F	75+	29	38	40-60	62	72

Fig. 22: Wide: Left correct table - right generate table



Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	41	56	66-72	82	97
M	25-35	36	51	61-67	77	92
M	35-45	32	47	57-63	73	88
M	45-55	39	49	54-60	65	75
M	55-65	35	45	49-57	61	71
M	65-75	34	44	46-58	60	70
M	75+	45	55	58-68	71	81
F	18-25	37	52	62-68	78	93
F	25-35	34	49	59-65	75	90
F	35-45	31	46	56-62	72	87
F	45-55	39	49	53-61	65	75
F	55-65	36	46	51-57	62	72
F	65-75	31	41	45-53	57	67
F	75+	32	42	44-56	58	68

Gender	Age	HRV Poor activity level	HRV Fairly active	HRV normal level	HRV Good physical condition	HRV Athlete physical condition
M	18-25	41	56	66-71	82	97
M	25-35	38	51	61-67	77	92
M	35-45	32	47	56-63	73	88
M	45-55	37	47	54-60	65	73
M	55-65	35	45	50-57	61	71
M	65-75	32	44	46-54	60	72
M	75+	45	56	57-68	71	81
F	18-25	37	52	62-68	78	93
F	25-35	35	49	59-65	75	90
F	35-45	31	46	56-62	72	85
F	45-55	39	49	53-61	65	74
F	55-65	37	46	51-57	62	72
F	65-75	31	42	45-53	57	67
F	75+	32	42	45-56	57	67

Fig. 23: Short: Left correct table - right generate table