International Journal of Advanced Engineering and Management Research

Vol. 9, No. 02; 2024

ISSN: 2456-3676

SPLIGN: Smart-belt Posture Monitoring System Based on AI-algorithms for Sitting Persons

Ferdews Tlili[1], Rim Haddad[2], Ridha Bouallegue[1]

[1]Carthage University, Department of Applied Mathematics, Signals and Communication, Raoued Raoud, 2083 Ariana, Tunisia

[2]LAVAL University, Department of Genie Electrique et Genie Informatique, 2325 Rue de l'Universite, Quebec, Canada

doi.org/10.51505/ijaemr.2024.9207 URL: http://dx.doi.org/10.51505/ijaemr.2024.9207

Received: Feb 20, 2024 Accepted: Feb 29, 2024 Online Published: Mar 21, 2024

Abstract

The back pain is the main common health problems on this decade because of the sitting for a long time tilted on the computer for working, studying or playing. The back problems are widely spread for different ages from young to old people. Last medical research demonstrates that sitting with the posture align can prevent and remedy many spine problems.

In this paper, we propose 'SPLIGN' which a posture monitoring system designed to help maintaining the good posture during sitting. The SPLIGN is a smart belt equipped with inertial sensors. A mobile and web applications are developed for monitoring and remind the user to correct posture.

The proposed system is based on a detailed study of the machine learning algorithms in order to choose the best accurate algorithm for posture prediction. The main studied algorithms are Convolutional neural network (CNN), The K-nearest Neighbors (KNN), Support-vector machines (SVM), Decision tree classification, Random forest, Naive Bayes Classifier and Boosting algorithm. The test results demonstrate that The Random Forest algorithm has the best accuracy 99.67% compared to the other algorithms with appropriate processing time 67.7 ms for real time posture monitoring system.

Keywords: Machine learning, monitoring system, inertial sensors, mobile application, posture monitoring

1. Introduction

The humanity evolution has known a wide involving of the machine in different living fields such as agriculture, medical, learning, industry, etc. At the first time, the machines are designed to execute specific task. After that, the researchers are developed these machines in order to have an intelligent one which can be able to analyze data and aid for decision. The artificial intelligent is started in 1943with the proposal of the neural Network Model [1]. The main objective is to make the machine thinking and taking decision. The machine learning methodology is based on data learning or without data collected and classified data or unclassified ones. There are four machine learning methods in literature: supervised, unsupervised, semi-supervised and reinforcement learning methods [2]. The variety of the machine learning methods and the data analysis algorithms demonstrate the importance of the machine learning algorithms to making easiest and comfortable human being life by involving intelligent robots and machines for travelling, medical diagnostics, performing repetitive and complicated tasks, ...

In medical field, the machine learning applications help the doctor for diseases detection. In fact, the diseases diagnostics are based on set of symptoms that are defined by one or many physiological parameters. Machine learning algorithms can process quickly whatever the complexity and the variety of the data. In literature, the latest studies are focused on the efficiency measurements of machine learning applications for prediction diseases such as breast cancer, heart disease, liver disease, diabetes disease and kidney disease [3].

In this paper, we will study the application of machine learning algorithms for posture monitoring as the back pain become one of the main health issues in the decade because of remaining seating for long hours. Then, we will analyze and test machine learning algorithms efficiency in order to implement the best algorithm for our proposed solution.

2. Related Work

The spine problems become an important topic to study by many researchers to avoid and prevent the back pains because of the spread of this issues for different ages from young people to old ones. Recently, in literature, many studies propose solutions for posture monitoring systems. In this paper, we concentrate our work to a smart belt based on AI solutions.

Kulikajevas et al. [4] present their proposed posture monitoring system based on images recognition. A recurrent hierarchical Artificial Neural Network (ANN) is proposed to recognize the postures from the RGB-D frame sequences. The proposed system detects six postures (3 main postures,6 sub-postures): Sitting straight, Forward posture (Lightly hunched, hunched over, Extremely Hunched) and Backward posture (Partially lying, lying down). The dataset is collected by recording each position for 30s three times a day for each participant. Then the dataset is labeled manually. The system presents the accuracy 91.47% of posture recognition and the prediction time 94ms. The system is able to recognize the posture with only 30% body view in the frame. The system is limited to define three main postures that can be extended to more postures. The dataset is built with limited number of participants and healthy ones. In addition, the system is not portable as it is based on fixed camera.

Sinha VK et al. [5] propose a machine learning based posture monitoring system. The system is composed by as mart phone for data treatment and an inertial sensor stuck on the neck for data collection. The system predicts five positions: Left Movement, Right Movement, Front Movement, Back Movement and Straight Movement. In this paper, the authors study the accuracy of three machine learning algorithms (KNN, SVM, Naive Bayes) for each sitting position. The dataset was generated by six participants (four males and two females) with ages 26 ± 3years old. The test results demonstrate that the SVM algorithm has the best prediction accuracy 99,89% compared to KNN (k= 3, k=5, k=7 and k=11) and Naive Bayes algorithms. In addition, the right movement presents the best accuracy for all the machine learning algorithms applied compared to the other sitting positions. The system has limitations in number and variety of the participants to construct the dataset. The time of algorithm processing is not analyzed in this paper.

Katia Bourahmoune et al. [6] propose a sitting posture monitoring system for posture detection and feedback correction. The Life Chair system proposed is composed by a cushion equipped with nine pressure sensors for posture detection and four vibration motors for feedback and posture correction. The smart cushion is placed on the backrest of the Chair. The authors analyze the application and the performance of seven supervised machine learning algorithms: random forest (RF), naive Bayes (NB), decision trees (DT-CART), linear regression (LR) Linear discriminant analysis (LDA), k-nearest-neighbors (k-NN) and neural network multilayer perceptron (NN MLP). Fifteen sitting postures are classified by the proposed system: Upright, Slouching Forward, Extreme Slouching Forward, Leaning Back, Extreme Leaning Back, Left Shoulder Slouch, Right Shoulder Slouch, Left Side Slouch, Right Side Slouch, Left Lumbar Slouch, Right Lumbar Slouch, Rounded Shoulders, Forward Head Posture, Slight Correction Needed, and No User (i.e., no contact with the Life Chair). A main dataset was built for the machine learning algorithms processing. Eighteen healthy subjects (12 male and 6 female) participated to construct the dataset. The participants are organized to three groups belong BMI parameter (BMI is the ratio of the weight on the square of the height): high BMI, normal BMI and low BMI. The results of the system tests demonstrate a better performance of the machine learning algorithms when taking in account the BMI parameter with the sensors information. The Random Forest algorithm with 30 trees presents best global accuracy98.82% compared to the other machine learning algorithms. In addition, the authors teste the random forest algorithm in different chair types: Mid back, standard back, high back, wide back, small back. The results demonstrate that the chair with standard back has the best accuracy on the global dataset92%, and the chair with wide back has the best accuracy on the group dataset (dataset collected using this type of chair) 98.29%. The chair with mid back presents the worst accuracy 56%. Therefor the accuracy of the system depends on the type of the seat used that limits the portability of the system. The prediction time is not measured for system performance analyzes. However, it is an important performance parameter for the real time systems.

Firgan Feradov et al. [7] report an automated bad posture detection system using the motion capture sensors. The system is equipped with 52 sensors placed on: 8 on the hands, 6 on the back and 38 on the fingers. The proposed solution differentiates between three main postures: SK: regular working posture when using a standard keyboard; EK: regular working posture when using an ergonomic key board; EKC: correct working posture when using an ergonomic keyboard. The posture detection is based on the accelerometer information with computing of the Hjorth's parameters (Activity, Mobility and Complexity) for each sensor channel. Three datasets are created for posture classification: first dataset containing the features collected from the sensors placed on the arms, the second dataset containing the features calculated from the sensors placed on the back and the head and the third dataset containing the features from the sensors placed on the fingers. Four machine learning algorithms are applied for posture

ISSN: 2456-3676

classification: KNN, Decision Tree, SVM Linear and SVM Gaussian Kernels. The SVM linear algorithm presents the best classification accuracy for the different features' datasets. The classification accuracy for the head and body features is 87.7%. In this article, the authors created important datasets containing various information from the different parts of the body that influence the sitting postures. In fact, the huge number of the sensors used for postures detection made the system complex for use.

Qilong Wan et al. [8] propose a sitting posture monitoring system based on the pressure image information. The system is composed of the 32x32 pressure sensors placed on the seat of the chair. The pressure information is processed and translated on image information according to a pressure intensity. Three main pressure area are monitored: the left hip, the right hip and the caudal vertebra. The system differentiates four postures: Straight, Back leaning, left leaning and right leaning. A dataset is collected by requesting ten participants (9 males and 1 female) to perform the postures sequentially and record the pressure and image information for each posture. In this study, the Support Vector Machine (SVM) algorithm is applied for the posture classification. The performance of the SVM algorithm with different kernels is studied. The SVM with polynomial kernel has the best average accuracy 89.6%. The best accuracy is reached for the leaning back posture 91% classified by the SVM with polynomial kernel algorithm. The proposed solution is able to locate the hip position not only for small angle (0° -15°), but also the medium angle (15° - 30°) and large angle (30° - 45°) with certain adaptability. However, the portability of the system is limited as it is inserted in the chair. In addition, the dataset collected is based on healthy and young people that is lack of variety.

Roland Zemp et al. [9] propose an instrumented chair for sitting posture detection. The chair is equipped by a mention sensor placed on the backrest and ten pressure sensors placed on the seat. Seven machine learning algorithms are applied for posture prediction: Support Vector Machines (SVMs), Multinomial Regression (MNR), Boosting, Neural Networks (NN), Radial Basis NN, Random Forest (RF) and Combination: Boosting, NN, RF. And seven sitting postures are differentiated using this system: sitting postures: upright position, reclined position, forward inclined position, laterally tilted right/left position, crossed legs, the left leg over the right one/the right leg over the left one. Forty-one subjects (16 females and 25 males) participate to record the measurements for dataset collection. The random forest algorithm presents the best accuracy for posture detection 90.9%. Furthermore, the combination of the three machine learning algorithms (Boosting, NN, RF) reaches a good accuracy90.8%. However, the combination of the algorithms does not present the best accuracy compared to the algorithm applied separately. The main limitation of the system is the standup of the user of the chair for each measurement to perform there set of the pressure sensors.

Yong Min Kim et al. [10] present a sitting posture monitoring system for children. The system is a chair equipped with cushion containing 8 X 8 pressure sensors placed on the seat. The system is designed for detecting the main common postures for children when sitting on the chair: Sitting straight, Lean left, lean right, sitting at the front of the chair, sitting crossed-legged on the chair. To predict the posture, six machine learning are applied: Convolutional Neural Network

(CNN), Naive Bayes (NB), Multinomial Logistic Regression (MLR), Decision Tree (DT), Neural Networks (NN) and Support Vector Machine (SVM). Tn children with age between 7 years old and 11 years old participate to the experiments of this study. The Convolutional Neural Network (CNN) algorithm presents the best accuracy for posture detection 95.3%. According to study, the accuracy of the posture prediction is impacted by the body weight, the accuracy is better for the high weight. The weight is a subjective criterion for posture correction.

Tariku Adane Gelaw [11] propose a sitting posture monitoring system. The system consists of a chair equipped with 32 by 32 pressure sensors placed on the seat and the backrest. The machine learning algorithms are studied to predict six sitting postures: back, empty, left, right, front and still. The deep learning algorithm "Deep Neural Network (DNN)" demonstrate best accuracy for posture prediction93% compared to the machine learning algorithms: Random Forest (RF), Gaussian Naïve Bayes (GNB), Logistic Regression (LR) and Support Vector Machine (SVM). The dataset is constructed by 50 participants. The study demonstrate that the accuracy is better when reducing the class labels. The DNN accuracy is around 98% when the class label is reduced to four class. In this study, the dataset is collected in lab environment that is limited to specific situation. In addition, the process time is not presented in this study, however it is a real time system. According to the literature studies, many posture monitoring systems proposed are based on the pressure sensors or image recognition. However, the application of the machine learning algorithms for posture recognition is not widely studied for the inertial sensors-based system.

In fact, in this paper, we will study the applicability of the machine learning algorithms for posture monitoring system based on inertial sensors. In addition, we will detail the implementation of our proposed solution 'SPLIGN' (Spine Align) that is composed of smart belt equipped with inertial sensors.

3. Global system architecture

3.1 Overview

The developed IoT solution architecture can be brought down to three main components as shown in Figure 1 below: hardware (gadget), software (mobile/web application) and the server (database and broker). The communication between these components is assured by different communication protocols depending on the context: MQTT (Message Queuing Telemetry Transport) for sensor data transfer and HTTPS for database queries.

Vol. 9, No. 02; 2024
ISSN: 2456-3676

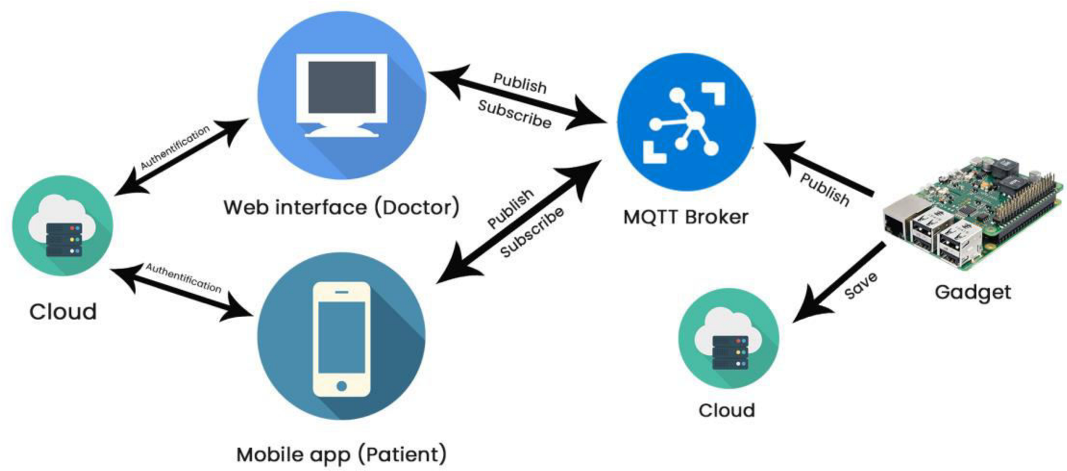

Figure 1: Solution architecture overview

3.1.2 Gadget design

3.2 Gadget design

The proposed wearable device "SPLIGN" is a smart belt equipped by inertial sensors. The sensing module is an IMU that allows to measure the evolution of an object in space. It can measure linear and angular accelerations in the three axis of space. It can be composed of an accelerometer, a gyroscope, and a magnetometer, or an accelerometer and a gyroscope. In this study, we used the MPU6050 composed of an accelerometer and gyroscope only as the magnetometer is sensitive to the interference and is not reliable for small slouching angles. In order to get a complete information about the sitting postures, we propose to place the inertial sensors: one on the middle of the back and one on one shoulder. The sensor on the vertebra spine eight and nine can measure the slouching angle variation of the spine. In some cases, the spine is straight, however the bad shoulders position causes uncomfortable sitting posture. According to previous study [12], we found that having an information from one shoulder is sufficient to complete the sitting posture information. The belt design chosen for implementing the system is characterized by a suitable coverage of the back and shoulders that allow the support of the materials used, simple to wear and comfortable for user as shown in Figure 2.

ISSN: 2456-3676

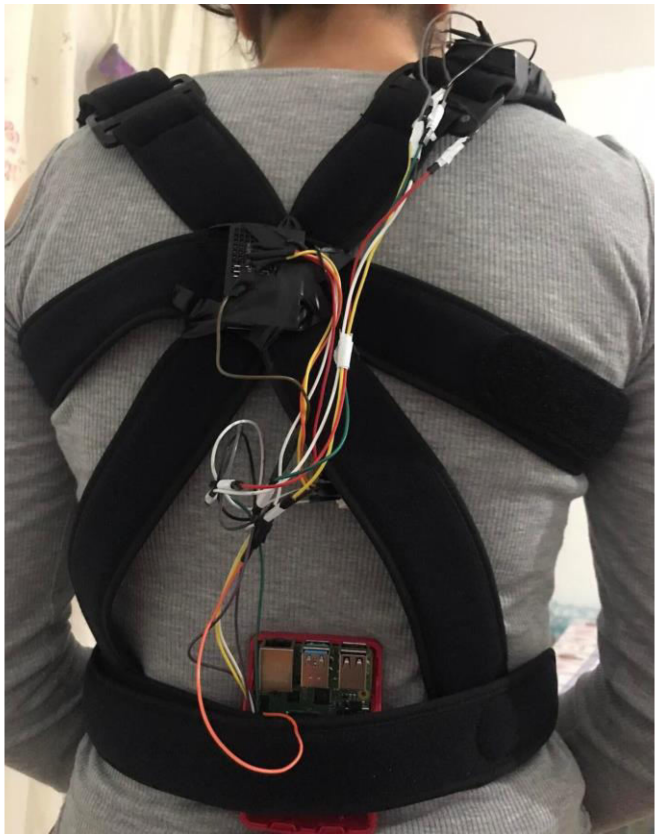

Figure 2: Smart Belt Architecture

3.3 Sensing information measurement

The software implemented in the gadget consists of reading the outputs of the sensors and converting them into readable angles that can be used later for the machine learning training. The IMU used for the proposed posture monitoring system is composed of three axis accelerometer and three axis gyroscope. The acceleration measurements (Ax, Ay and Az) are converted to inclination angles according to the three axis (x, y and z) via basic trigonometry rules as the equations (1), (2), and (3).

$$\theta = \frac{A_{x,OUT}}{\sqrt{A^2{}_{y,OUT} + A^2{}_{z,OUT}}} \tag{1}$$

$$\Psi = \frac{A_{y,OUT}}{\sqrt{A^2{}_{x,OUT} + A^2{}_{z,OUT}}} \tag{2}$$

$$\Phi = \frac{\sqrt{A^2{}_{y,OUT} + A^2{}_{z,OUT}}}{A_{x,OUT}} \quad (3)$$

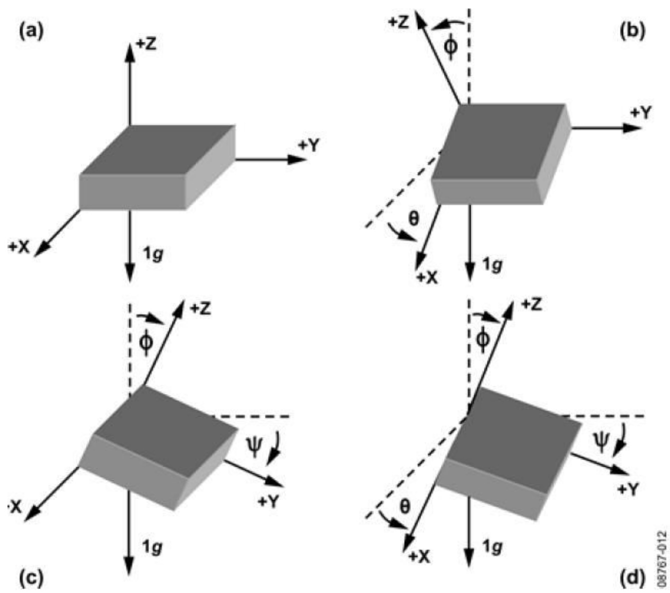

Fig. 3. Angles of the accelerometer.

In addition, the rotation angles are calculated from the angular velocity measured from the gyroscope according the three axis as shown in the equation 4.

$$\theta = \int \omega \, dt \quad (4)$$

To overcome the accelerometer and gyroscope errors, and limitations, the sensing information collected from accelerometer and gyroscope are combined using the complementary filter. This Filter is based on weighted values from the measurements of the accelerometer and gyroscope in order to calculate a global tilt angle as shown in equation (5).

$$\theta_{final} = 0.98 \times Angle_{gyroscope} + 0.02 \times Angle_{accelerometer} \quad (5)$$

4. Machine Learning algorithms study

In the literature, many machine learning algorithms are proposed for different intelligent applications. A study is performed in order to select the machine learning algorithms applicable for the architecture and the specificity of our proposed real time posture monitoring system. This model should be able to predict the most common sitting postures (good posture, slouching over, slouching right, and slouching left). In fact, we proposed to base on a labeled data for posture prediction. For that, the supervised machine learning algorithms are recommended to be applied for the proposed system as these supervised algorithms relies on labeled input data to learn a function that produces an appropriate output when given new unlabeled data. In this section, we

Vol. 9, No. 02; 2024

ISSN: 2456-3676

will detail the machine learning algorithms proposed to be applied and tested for our posture monitoring solution.

4.1 Convolutional Neural Network (CNN)

One of the main parts of Neural Networks is Convolutional neural network [13] (CNN). CNNs use image recognition and classification in order to detect objects, recognize faces, etc. They are made up of neurons with learnable weights and biases. Each specific neuron receives numerous inputs and then takes a weighted sum over them, where it passes it through an activation function and responds back with an output.

The main advantage of this classification algorithm is that detects automatically the important features without any human supervision. The CNN algorithm characterized by very High accuracy in image recognition problems. A distinguishing feature of CNNs is that many neurons can share the same filter. This reduces the memory footprint because a single bias and a single vector of weights are used across all receptive fields that share that filter, as opposed to each receptive field having its own bias and vector weighting. However, the CNN algorithm needs a large training data. It is unable to be spatially invariant to the input data. It does not encode the position and orientation of the object.

4.2 K-Nearest Neighbors (KNN)

The K-nearest neighbors (KNN) [14] algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. The KNN algorithm hinges on this assumption being true enough for the algorithm to be useful.

The main advantage of KNN over other algorithms is that KNN can be used for multi-class classification. Therefor if the data consists of more than two labels or in simple words if you are required to classify the data in more than two categories then KNN can be a suitable algorithm. The algorithm is simple and easy to implement. There is no need to build a model, tune several parameters, or make additional assumptions. The algorithm is versatile. It can be used for classification, regression, and search. The main limitation of this algorithm is that the KNN gets significantly slower as the number of examples and/or predictors/independent variables increase.

4.3 Support-Vector Machines (SVM)

Support-vector machines (SVM) [15] are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. SVM is used as a linear or non-linear classifier based on the kernel used. If we use a linear kernel, then the classifier and thus the prediction limit are linear. In order to separate two classes, we need to draw a line. The line is such that there is a maximum margin. This line is drawn at equal distance from the two sets. We draw two other lines on each side, called support vectors. The SVM algorithm is effective in high dimensional spaces. It has a regularization parameter, which encourages the user to avoid over-fitting. Nevertheless, the SVM algorithm is limited to linear problems. It is not suitable for non-linear problems and not the best choice for many features. If the number of features is much greater than the number of samples, avoid over- fitting in choosing Kernel functions and regularization term is crucial.

4.4 Decision Tree Classification

Decision tree classification [16] is the most powerful classifier. A decision tree is a flowchart like tree structure, where each internal node designates a test on an attribute (a condition), each branch represents a test result (true or false), and each leaf node (terminal node) contains a class label. Based on this tree, splits are performed to differentiate classes in the given original data set. The classifier predicts which of the classes a new data point belongs to based on the decision tree. The algorithm is simple to be interpretable with no need for feature scaling. Also, it works on linear / nonlinear problems. However, the decision tree algorithm gives poor results on very small datasets. In addition, the overfitting can easily occur.

4.5 Random Forest

Random forest [17], like its name implies, consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction. So, the prerequisites for random forest to perform well are that there needs to be some actual signal in our features so that models built using those features do better than random guessing. And, the predictions (and therefore the errors) made by the individual trees need to have low correlations with each other. Random forest algorithm can be used for both classifications and regression tasks. As a decision tree algorithm, Random Forests are less influenced by outliers than other algorithms. It provides higher accuracy through cross-validation. Random forest classifier will handle the missing values and maintain the accuracy of a large proportion of data. The main limitations of the random forest algorithms are the high complexity and the long-time training compared to decision tree classification.

4.6 Naive Bayes

Naive Bayes Classifier [18] works on the basis of Bayes' theorem. The fundamental assumptions made are that all features are independent of each other and contribute equally to the result; all are of equal importance. It is a probabilistic classification model whose node is Bayes Theorem. The $Nave$ Bayes algorithm is efficient and unbiased by outliers. It works on non-linear problems. It is probabilistic approach. The fact that all features are independent of each other and contribute equally to the result is not always valid in real life. It is based on the assumption that the characteristics have the same statistical relevance. Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. This limits the applicability of this algorithm in $real - world$ use cases. This algorithm faces the '$zero - frequency$ problem' where it assigns zero probability to a categorical variable whose category in the test data set wasn't available in the training dataset. It would be best if you used a smoothing technique to overcome this issue. Its estimations can be wrong in some cases, so you shouldn't take its probability outputs very seriously.

Vol. 9, No. 02; 2024

ISSN: 2456-3676

4.7 Boosting

Boosting algorithms [19], unlike many ML models, aim to improve prediction power by training a series of weak models, each one correcting the limitations of its predecessors. Boosting, as an ensemble model, has an easy-to-read and interpret algorithm, making prediction interpretations simple. Through the utilization of clone methods like bagging,
random forest, and decision trees, the prediction capability is effective. Boosting is a sustainable approach for reducing over-fitting. Because every classifier is required to rectify the inaccuracies in the predecessors, boosting is sensitive to outliers. As a result, the technique is overly reliant on outliers. Another downside is that scaling up the process is nearly impossible. This is due to the fact that each estimator is based on the accuracy of preceding predictors, making the operation difficult to simplify.

5. Dataset implementation

5.1 Sitting Postures selection

A good posture, as shown in Figure 4, is a good alignment of the spine [20]. It is important to stand up straight and that the vertebrae are aligned. This posture must be maintained at every moment of the day, while sitting and doing an activity. The spine must be straight. Sitting should be at an angle range between 80° and 120° compared to the vertical. If not, your body will have to make extra efforts to adapt to an unnatural situation. This is how back pain develops. [21]

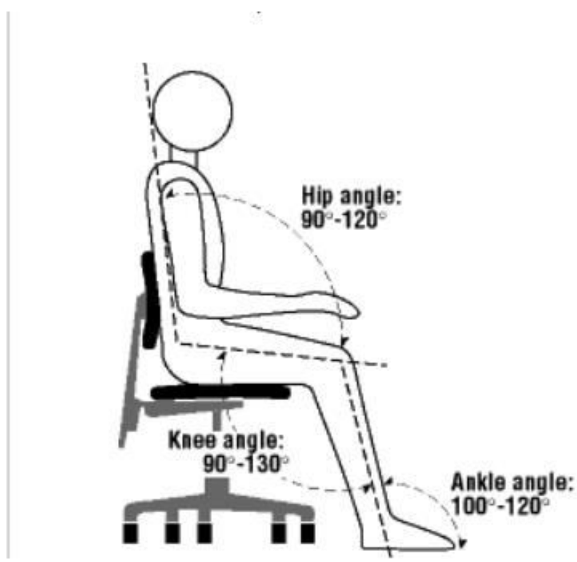

Fig. 4. Angle measurements of the correct posture

In this study, five usual postures were considered to be classified. These positions cover the most common habits when sitting. Tab.1 and Figure 5 represent how the position were performed.

Vol. 9, No. 02; 2024

ISSN: 2456-3676

Table 1. Different postures

Posture	Classification	Description
Posture1	*Good*	Straight (GP)
Posture2	*Bad*	slouched over (SO)
Posture3	*Bad*	slouched right (SR)
Posture4	*Bad*	slouched left (SL)

Fig. 5. Possible Sitting postures: (a)Good posture, (b) Slouching over, (c) Slouching right, (d)Slouching left.

5.2 Dataset collection

In this section, we detail the collection and the labelling data for the dataset construction. The collected data is the angles measured for each position as mentioned in the previous section. The two IMU placed one on the middle of the back and one on the right shoulder measure the signals from the accelerometer and the gyroscope. Then the measurements are converted to readable angles ($\theta 1$, $\theta 2$, $\phi 1$, $\phi 2$, $\Psi 1$, $\Psi 2$) and (gx1, gx2, gy1, gy2, gz1, gz2). Participants for the data collection are nine persons (6 females, 3 males) the age between 21 and 23 years old. And they are in good health. The participants are asked to perform the four positions (good posture, slouching over, Slouching right, Slouching left). Each participant is asked to maintain the good posture for 2.5min (150s). The samples collected for each measurement are 300 samples for good posture. Then the participants perform the bad postures respectively: slouching over, slouching right, slouching left during 3min (180s). In total, 360 samples are collected for bad posture as each position is maintained for 50s. After the data collection, we labelled all the data manually 0 and 1: 1 for good posture,0 for bad postures.

ISSN: 2456-3676

6. Discussion

In this section, we discuss the applicability of the detailed machine learning algorithms according to the architecture and the sensing information. In the proposed solution, the prediction algorithm will be based on the tilt angles detected from the sensors according to the three axis. The collected Dataset is composed of 300 samples of the good posture and 360 samples of slouching right, left and over. The features collected from the sensing unit as detailed in section 4.2.3 are the three angles of the accelerometer and the rotation angle of the gyroscope from the 2 IMU sensors: one placed on the middle of the back and the second placed on one of the shoulders. For first tests, we proposed a training dataset with the minimum features in order to study the efficiency of our solution. According to the dataset collected, the CNN algorithm is not suitable for our proposal as this algorithm requires a large dataset and a large number of attributes (features). The SVM algorithm is a supervised algorithm based on the calculation of the data threshold. The score is calculated depending on tested data and compared to the threshold in order to classify into the defined category. As the proposed training dataset is categorized on four classes according to the positions, the SVM algorithm can be applied on our solution.

According to the description of the Naïve Bayes algorithm in the previous section, the algorithm is simple to implement and suitable for real time applications. It is applicable for

a short training data. For that, we will implement and test the Naïve Bayes algorithm for our proposed system for sitting posture monitoring. The KNN algorithm is based on the calculation of the data proximity. In fact, our dataset is composed of a discreet data composed of the measurements of the slouching angles. It is easy to implement. And the

KNN algorithm does not perform training before data prediction.
For that, the training period is zero. That’s why, it is a suitable algorithm for our real time application. The decision Tree is easy to understand and simple to apply. In addition, the algorithm is working correctly in spite the lack of the training data. It does not require a large dataset. The Decision Tree algorithm is suitable for discrete values and numerical data as the collected data of our dataset. It is fast and efficient. So, it is suitable for our real time application. The random Forest algorithm is composed of decision trees. In fact, the characteristics of decision tree algorithm are available for the random forest algorithm. In addition, the random Forest is the aggregation of the decision tree algorithm results. For that it is characterized by a high accuracy. And, it is efficient to predict with a large missing data. The Boosting algorithm is composed of the combination of multiple machine learning algorithms with the same data set. It is based on the averaging techniques for data predictions. For that the Boosting algorithm is able to cover the limitation of the data missing in training data and the used algorithms to combine the Boosting algorithm. As a summary, we apply in our proposed system the following algorithms: SVM, Naïve Bayes, KNN, Random Forest, Decision Tree and Boosting.

7. User Interfaces Design

We proposed a sitting posture monitoring system for healthy user to prevent back pains and to correct bad posture. For that, we develop a mobile application for the user in order to monitor his posture anywhere in office, at home, etc. In the other side, we develop a web application for doctors in order to monitor the patient posture via statistics during a selected period of time. In this section, we detail the user and doctor interfaces design.

7.1 Mobile Application

The mobile application will integrate interfaces to authenticate the user and to detect and correct the posture. The user can sign in, sign up and log out after authentication. He can view his posture in the Home Page in real time, view his posture evolution per chosen period, view percentage of good posture and check his previous statistics in the Stats Page. He is able to view his profile and to access to a Support Page where he can find some tips to maintain good posture. We developed an Android and iOS applications which support the different detailed views.

7.2 Web Application

The web application is developed for the doctor in order to monitor the patient behavior. A registration and login interfaces are implemented for doctor authentication. The web application offers to the doctor an interface to consult the patient invitation for posture monitoring. He can accept or refuse the invitation. The doctor can analyze the posture behavior of the patient via the tracking interface. He can also consult and edit his profile.

8. System Implementation

8.1 Setting up of the Smart Belt

Our proposed solution SPLIGN (Spine + Align) is composed of a hardware and software parts. In fact, the choice of the sensing module and the software for the collected information treatment is detailed in this section.

8.1.1 Hardware Choice

The Sensing module of SPLIGN system is composed of the sensors for the posture information detection and the microcontroller for data collection and treatment. Since the sensors measurements of an accelerometer and a gyroscope are sufficient for the four posture positions to detect and the magnetometer is sensitive to magnetic fields, a magnetometer to give you an exact angle, but it takes longer to get an accurate result, and it’s not good at fine changes in angle which can reduce the performance of our system, we have chosen to use two IMU MPU6050 composed of an accelerometer and a gyroscope (MPU6050) [22]. The Raspberry Pi 4 [23] was chosen as a Micro-controller because it is powerful, it has a lot of inputs/outputs, it uses many protocols to connect to other hardware devices and it responds perfectly to the solution needs. A prediction of the posture can be made using one sensor placed in the middle of the back, but in this case, the solution won’t be precise or accurate. Because having a bad posture may not depend only on the inclination angle but also on the position of the shoulder, we have chosen to place two IMU, one in the middle of the back and the other on the top right shoulder. The picture below demonstrates the hardware set up of the proposed solution.

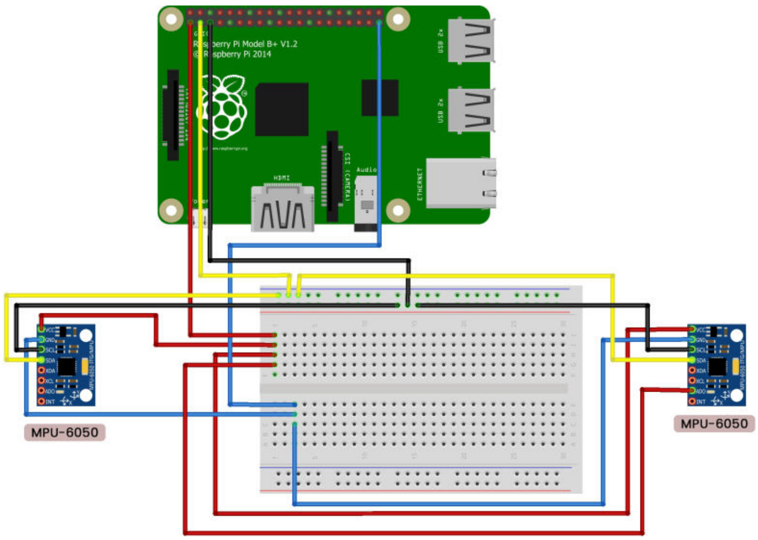

Fig. 6. Gadget architecture.

8.1.2 Software Choice

The software used in the gadget consists of reading the outputs of sensors and converting them into readable angles that can be used later for the machine learning training. For processed data transfer, the gadget uses The Message Queuing Telemetry Transport (MQTT) [24]. MQTT is a lightweight open messaging protocol used to transport data between devices. It is commonly used in IoT solutions as it is designed for small size messages. It does not consume high amount of energy and it uses minimal network bandwidth. To send data using MQTT protocol, it is necessary to set up an MQTT server. To do so, the Mosquitto MQTT server will be installed. Mosquitto [25] is an open source message broker that implements several versions of the MQTT protocol, it is relatively a lightweight software, making Mosquitto the perfect choice for dealing with the MQTT protocol on the Raspberry Pi. An MQTT broker is a server that receives all messages published by the clients and then routes them to the corresponding subscribed clients. For the project, a set of brokers were tested. To choose a broker, it must support Web Sockets, and it must be secure. Among the tested brokers is mentioned: Mosquito, Hive MQ, EMQX. Finally, the mosquito broker was configured on a virtual machine hosted on EMQX. MQTT is a publish/subscribe protocol. The gadget publishes the data to a topic on which the user's device (MQTT client) will subscribe. Once the data is published, the subscriber receives it.

For the frontend and backend implementation, we choose the Flutter framework for the mobile and web application with Firebase database. The user interface is made using the Flutter framework. Flutter is an open-source UI SDK developed by Google. It enables developers to write code for Android, iOS and web responsive applications simultaneously, which justifies the choice to adopt it for the solution.

Vol. 9, No. 02; 2024

ISSN: 2456-3676

Flutter has nested architecture that offers a more understandable code called widget tree and supports many packages and functionalities. Also, Flutter relies on Dart language, which is an easy and flexible programming language and facilitates writing code [26]. Firebase is an open source Backend-as-a service (BaaS), categorized as a NoSQL database that stores data in JSON-like documents. Firebase offers authentication services, database, storage, and hosting service that the application will benefit from all of them. Firebase is simple, lightweight, friendly, and industrially recognized. Each user has multiple attributes in the database such as id, name, weight, email, age, height, gadget id, and a role (patient or doctor). Firebase supports Flutter, which justifies its choice as a backend to facilitate users management.

8.2 Machine Learning Algorithms Evaluation

8.2.1 Dataset Collection

As detailed in section 4.3, the participants for dataset construction are invited to perform the four sitting positions (Good sitting posture, Slouching Left, Slouching Right, Slouching Over). The measurements are collected during the slouching period periodically each 2s. The data collection periodicity 2s is appropriate for the proposed real time posture monitoring system. The collected measurements from the two sensors are converted to significant angles ($\theta 1$, $\theta 2$, $\phi 1$, $\phi 2$, $\Psi 1$, $\Psi 2$) and (gx1, gx2, gy1, gy2, gz1, gz2). Then the data is categorized according the four categories (Good posture, SO, SL, SR). And the collected data is labelled by the value 1 and 0, good posture and bad posture respectively.

8.2.2 Machine Learning Algorithms Testing

Before implementing our proposed sitting posture monitoring system, we tested and analyzed the machine learning algorithms using the collected dataset in order to choose

the best suitable algorithm for the system implementation. The machine learning algorithms efficiency is depended on the functional parameters. For that, many testing and study phase are performed in order to choose the appropriate parameters for the machine learning algorithms execution. In fact, for the random forest algorithm, we remark that the number of trees impact the accuracy and the processing time of the algorithm. We performed tests of the random forest algorithm with changing the number of the trees and checking the accuracy and the processing time variation. The goal of these tests is to find the optimum number of trees with good accuracy and short processing duration. As shown in Figure 7, the accuracy of the random forest algorithm is in continued evolution when the number of the trees increased. The algorithm accuracy is stabilized from the number of the trees 30. In the other side, the processing time is increasing with the number of the trees evolution as shown in Figure 8. The number of trees 30 is the best-balanced value for the random forest algorithms to have best accuracy with optimum processing time. In the Table 2, we detail the functional parameters chosen for the machine learning treatment.

Vol. 9, No. 02; 2024

ISSN: 2456-3676

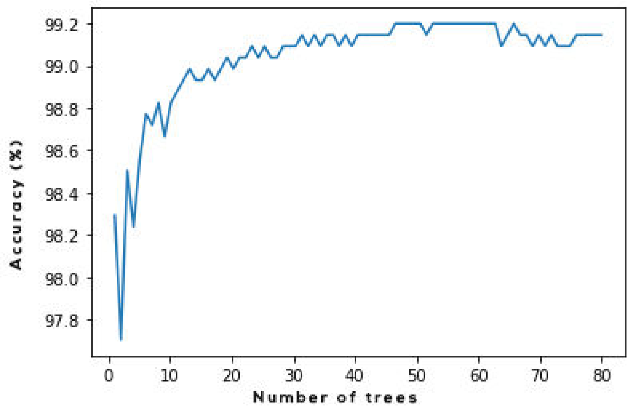

Fig. 7. Accuracy variation (Random forest) depending on number of trees.

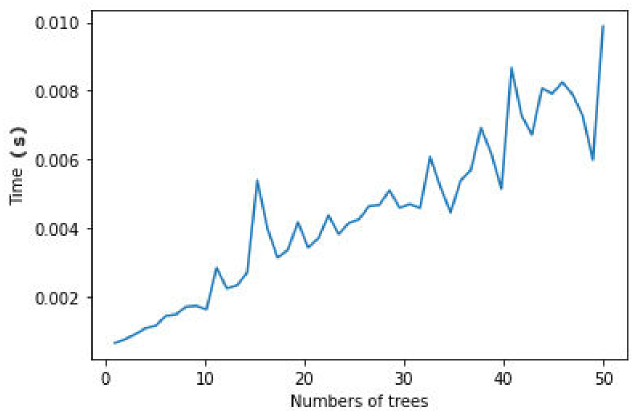

Fig. 8. Time response (Random forest) variation depending on number of trees.

Table 2. Machine learning algorithms functional parameters.

Algorithm	Parameters
KNN	*N neighbors=40*
SVM	*Type: SVC, Default parameters*
Decision Tree	*Min sample leaf=1*
Random Forest	Number of trees=30 trees
Boosting	*Type: AdaBoost, n estimators=50, random state=0*
K-means (unsupervised)	*N clusters=2*

In this section, we study the results of the machine learning algorithms testing in order to decide the best algorithm for our sitting posture monitoring system. The efficiency of the machine learning algorithms is analyzed via the main following performance metric the accuracy, the precision and the processing time.

ISSN: 2456-3676

$$Accuracy = \frac{\#\text{Correct Predictions}}{\#\text{Records}} \quad (6)$$

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

TP: True Positives
FP: False Positives

Two test phases are performed in order to evaluate the machine learning algorithm. The first test phase consists of dividing the collected dataset in 2 parts: the 30% of the dataset is used for the test and 70% of the dataset is remained as a training data. The Table 3 presents the tests results of the first test phase for the machine learning algorithms. In global, the accuracy of the most trained algorithm is good as it is more than 97%. Only the K-means algorithm has the worst accuracy 57.79% compared to the other algorithms. The Random Forest algorithm with 30 trees has the best accuracy 99.198%. Also, the Boosting algorithm presents a good accuracy 99.014%. The precision evaluation is same as the accuracy. The random Forest and the Boosting algorithms have the best precision 98.990% and 98.70% respectively. As the proposed solution is a real time solution, the processing time is an important metric for the choice of the machine learning to be used. In fact, the SVM algorithm demonstrate the less time processing 0.00064s. The K-means algorithm presents the long-time processing 0.357s. In addition, the algorithms with the best accuracy Random Forest and Boosting algorithms presents an appropriate processing time for a real time solution 0.0677s and 0.0102s respectively. As the K-means present a weak accuracy and long processing time, we drop it from the evaluation in the next test phase.

Table 3. Algorithms results (First Test).

Algorithms	Accuracy (%)	Precision (%)	Time (s)
KNN	*97.91*	*98.67*	*0.114*
SVM	*98.56*	*98.247*	*0.00064*
Decision Tree	*98.717*	*98.583*	*0.00896*
Random Forest (30 Trees)	**99.198**	**98.70**	**0.0677**
Naives Bayes	*89.532*	*91.05*	*0.002785*
Boosting	*90.014*	*98.990*	*0.0102*
K-means (unsupervised)	*57.79*	*56.5*	*0.357*

A second test phase is proposed in order to ensure the machine learning evaluation. So new data was collected from persons that don't participate to the dataset collection. Six persons (3 males and 3 females) participate to evaluate the proposed system. Their ages are between 15 and 26 years old. The participants perform the different postures (Good posture, Slouching Over, Slouching Left, Slouching Right) and maintain each posture for 25s. In this evaluation phase, we studied the global accuracy as the first test phase and the accuracy for each posture separately.

The global accuracy for each algorithm is good as it is more than 83%. The Random Forest algorithm with 30 trees demonstrates the best global accuracy 99.67%. It is better than the Boosting algorithm 97.12% and Decision tree algorithm 97.64%. For the good posture position, the Random Forest and the SVM algorithm present the best accuracy with 100% value for the two algorithms. The Naive Bayes algorithm has the best accuracy 100% for the slouching over position detection, with good accuracy for the Decision tree 99.85% and Random Forest 99.84% algorithms. The Random Forest algorithm present the best accuracy 99.83% for the slouching left posture. The Naive Bayes algorithm has the best accuracy 100% for the slouching right posture. We remark that the 100% accuracy is found in case of the good posture, slouching over and slouching right. It may due to the positions chosen for the sensors. The sensor placed in the middle of the back detects with good accuracy the position related as the good posture with aligned back and the slouching over with tilt back. In addition, the sensor placed on the right shoulder is able to get better data for detection of the slouching right. Globally, the processing time for the six studied algorithms is appropriate for the real time solution. In fact, SVM algorithm has the short processing time 0.001237s. In addition, the random forest has acceptable processing time 0.013415s.

Table 4. Algorithms results test phase (Second Test).

Algorithms	GS Acc	GP Acc	SO Acc	SL Acc	SR Acc	Time
KNN	*90.96*	*75.58*	*98.04*	*99.52*	*85.1*	*0.114*
SVM	*88.88*	*100*	*88.6*	*96.97*	*68.59*	*0.001237*
Decision Tree	*97.64*	*97.65*	*99.85*	*92.31*	*98.4*	*0.00896*
Random Forest (30 Trees)	**99.67**	**100**	**099.84**	**99.83**	**98.56**	**0.013415**
Naives Bayes	*83.86*	*26.75*	*100*	*99.52*	*100*	*0.002775*
Boosting	*97.12*	*99.72*	*99.77*	*85.74*	*99.84*	*0.009081*

As summary, according the results detailed previously, the Random Forest with 30 trees algorithm presents the best accuracy results for global accuracy calculation and the separately accuracy calculation by postures. Also, the time processing of the random forest algorithm is appropriate for the real time characteristic of the proposed solution. So, the Random Forest algorithm with 30 trees used for data processing, we will choose.

8.2.3 Complete system Testing

In this section, we present the final tests for the complete proposed solution for sitting posture monitoring system. As detailed before, our system consists of a smart belt equipped by 2 IMU sensors for posture detection as shown in Figure below, mobile application is developed for the user posture monitoring. In addition, a doctor service is added for the patient posture monitoring via a web application.

ISSN: 2456-3676

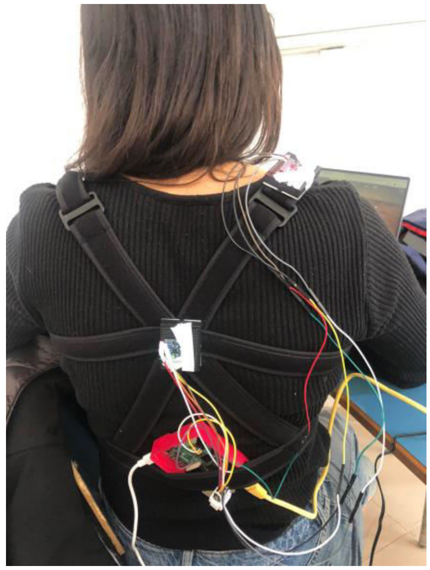

Fig. 9. Real Picture of the system testing.

The mobile application consists of different interfaces which help the user to easily correct his posture. The welcome interface appears at the first time the user opens the app. He chooses either to Login or to register.

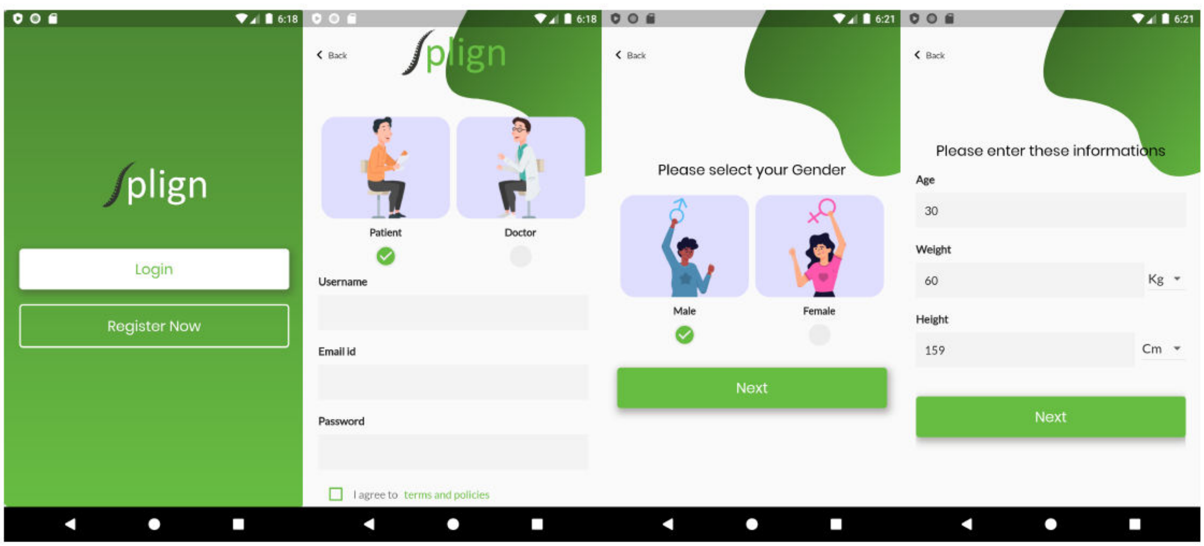

Fig. 10. Four interfaces of the mobile application: (a) Welcome

Interface, (b) Register Interface, (c) Gender Selection,
(d) Fill a Form.

ISSN: 2456-3676

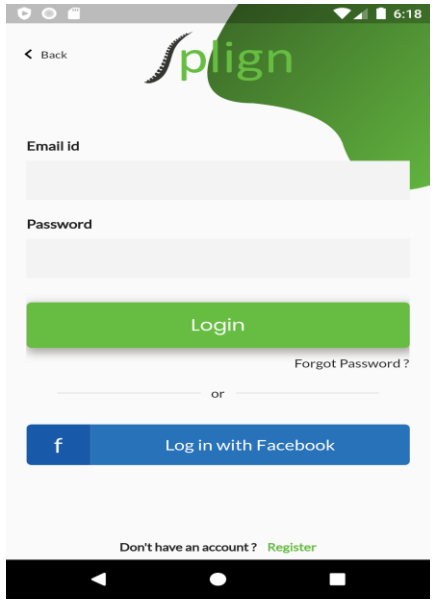

Fig. 11. Log in Interface.

The mobile App will provide feedback to the user by a sound and visual effects as shown in the pictures below.

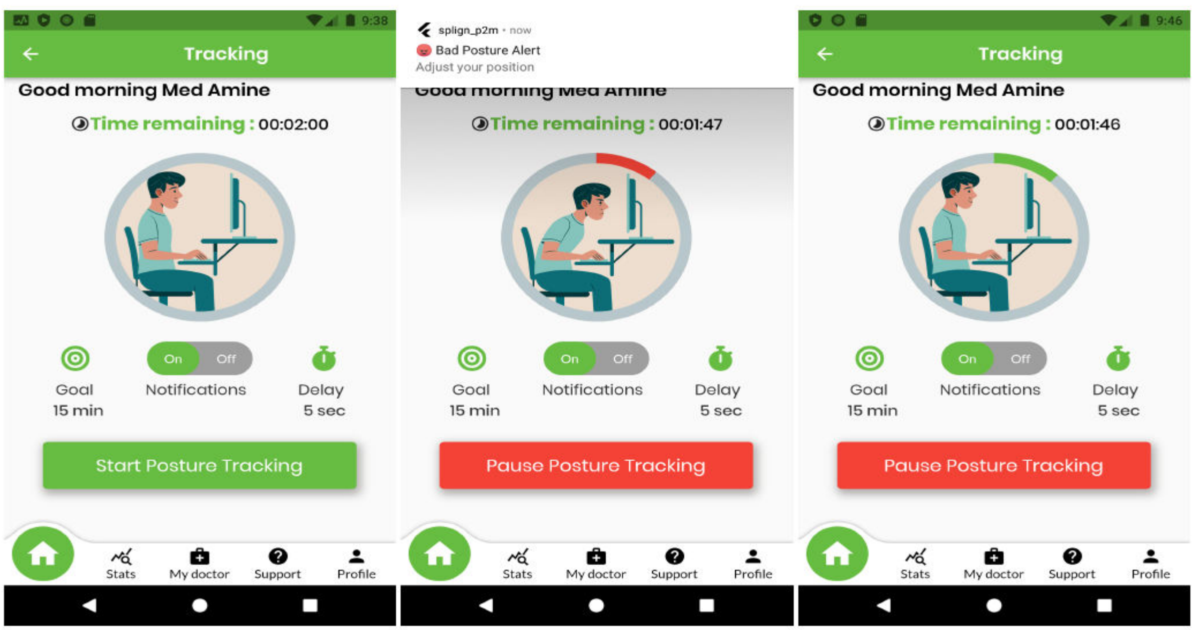

Fig. 12. Three interfaces of different postures: (a) Home

Interface, (b) Bad Posture, (c) Good Posture.

In this interface, is displayed the percentage of good posture per day and a diagram presenting the posture variation over time.

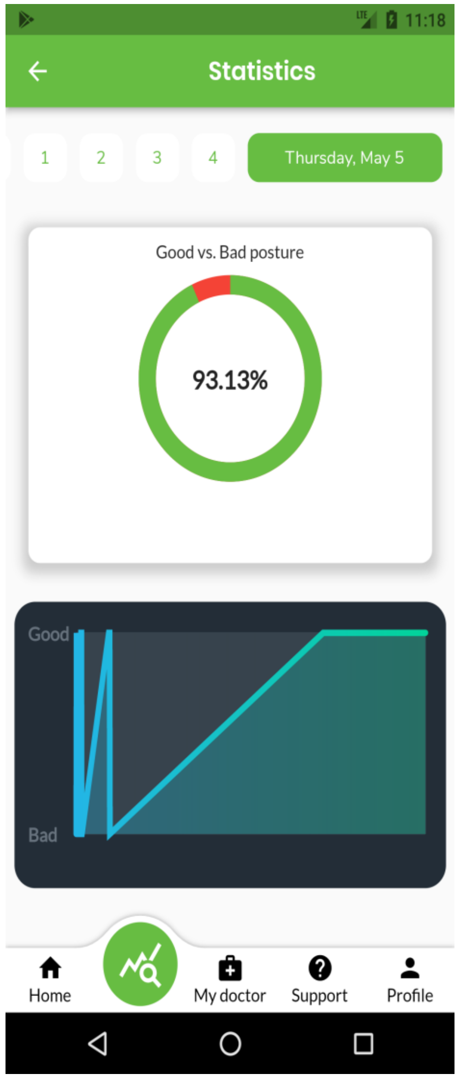

Fig. 13. Statistics Interface.

When the user opens the Doctor interface, he can search for a specific doctor, he can view his profile and he is able to send him an invitation.

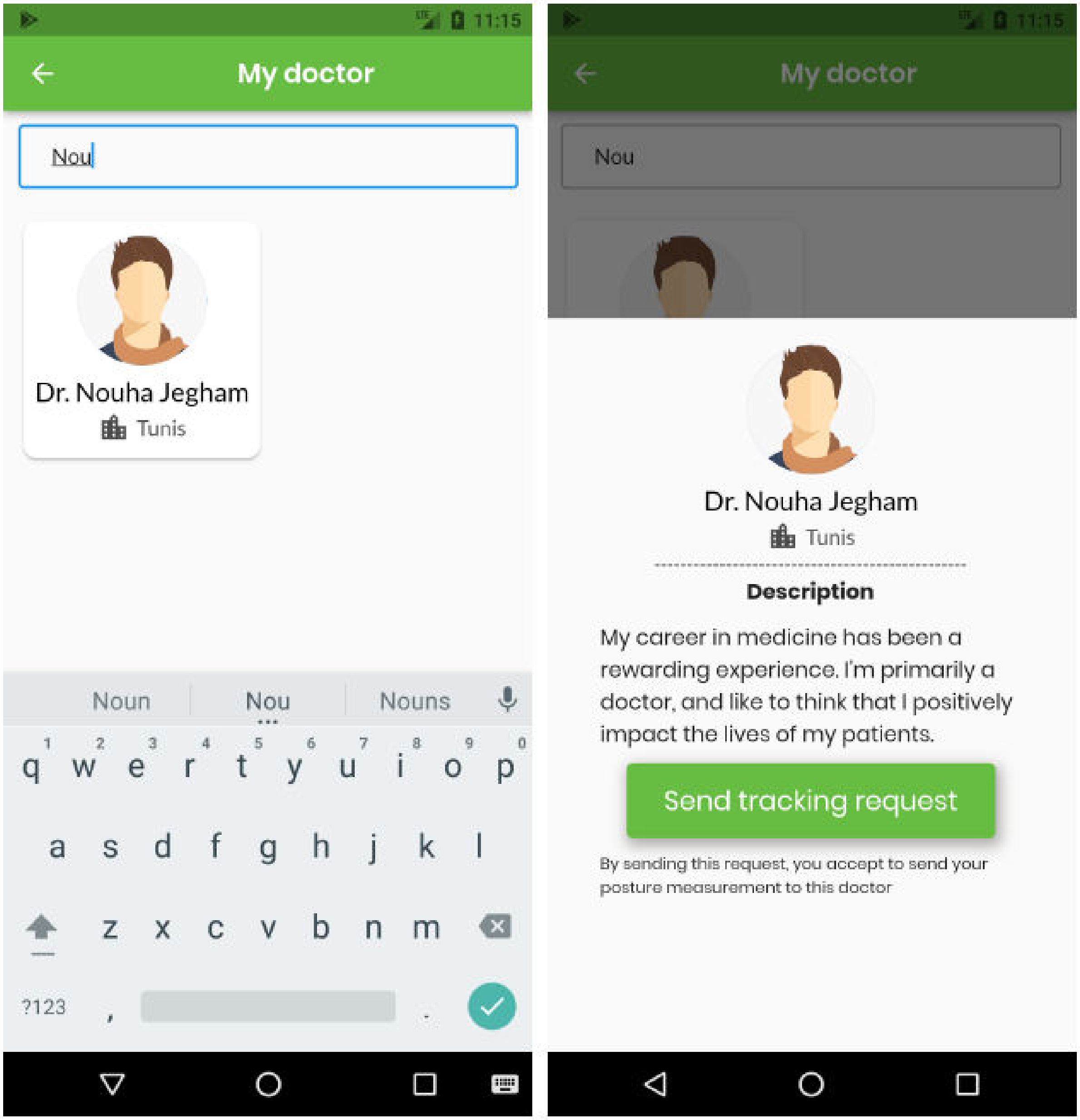

Fig. 14. Interfaces for doctor search: (a) My doctor search, (b)

Doctor Profile.

The user can find advice and answers to improve his posture in the following interfaces.

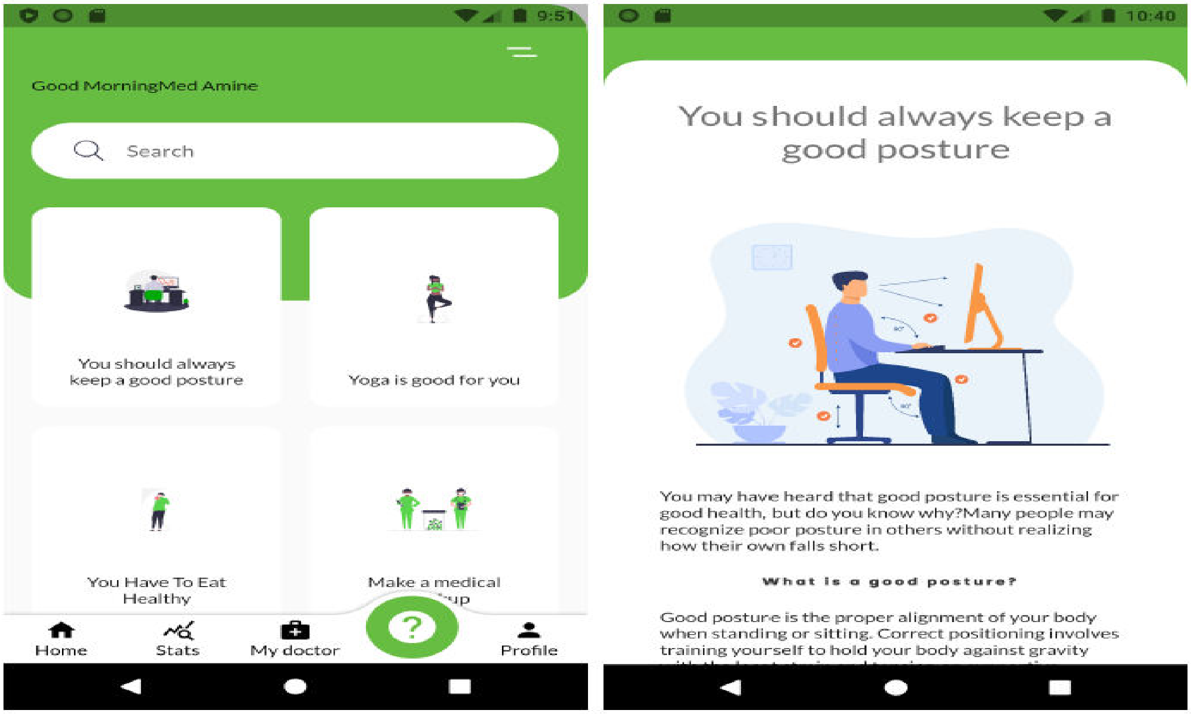

Fig. 15. Support and Information interfaces: (a) Support Interface, (b) Article.

The profile interface contains information used to identify the user, such as his name, age and portrait photo. The user can edit his profile by clicking on "My Account" button.

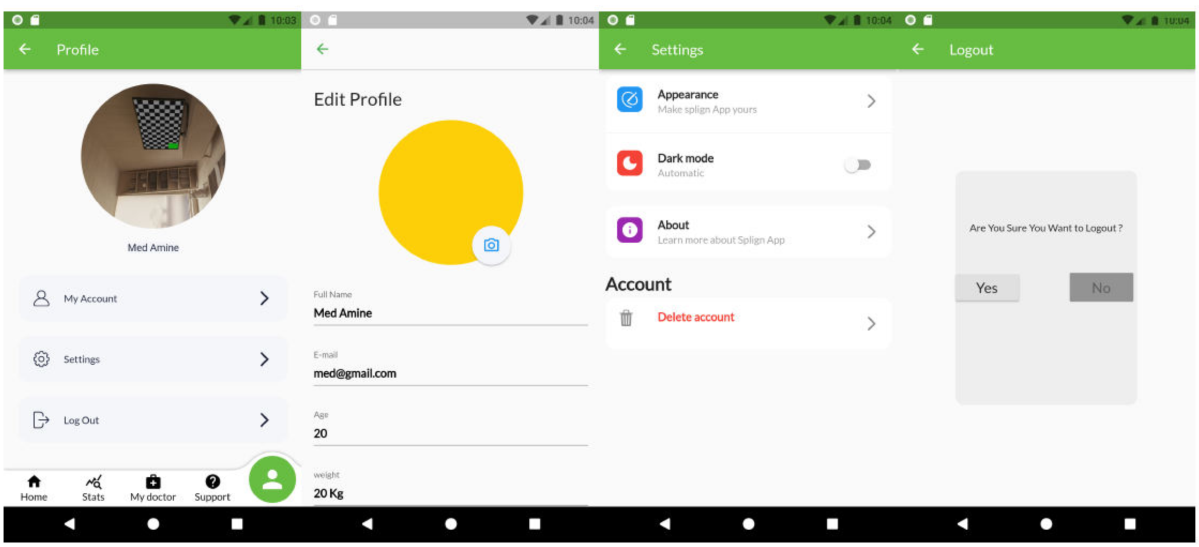

Fig. 16. Setting interfaces: (a) Profile Interface, (b) Edit

Profile, (c) Settings, (d) Log Out

Vol. 9, No. 02; 2024

ISSN: 2456-3676

The web application is developed especially for the doctor who can monitor the patient postures in a specific time. This application provides different views for the doctor in order to analyze the patient behavior during a period of time.

The doctor can log in or register if he doesn't already have an account.

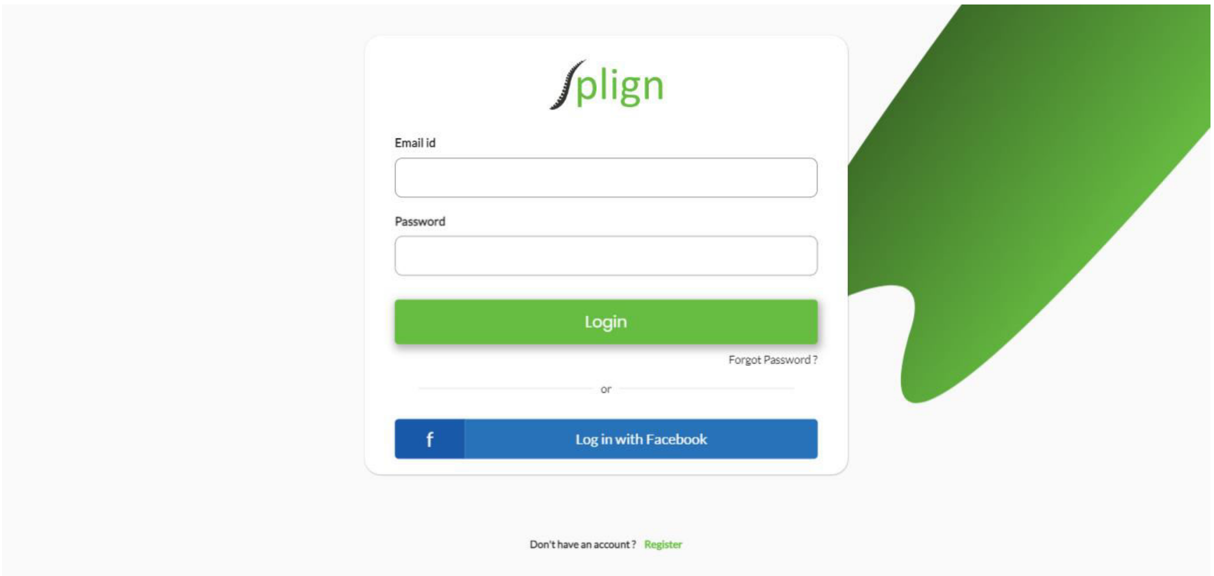

Fig. 17. Web application: Login Interface.

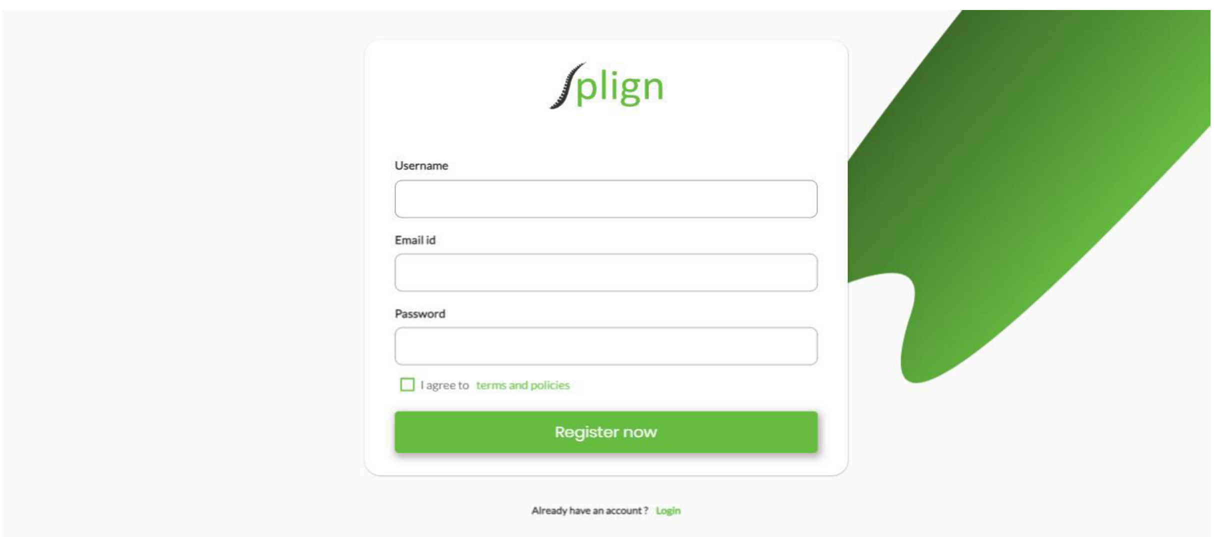

Fig. 18. Web application: Register Interface.

The doctor is able to see the invitations sent from the patients and he can choose which patient to accept and track.

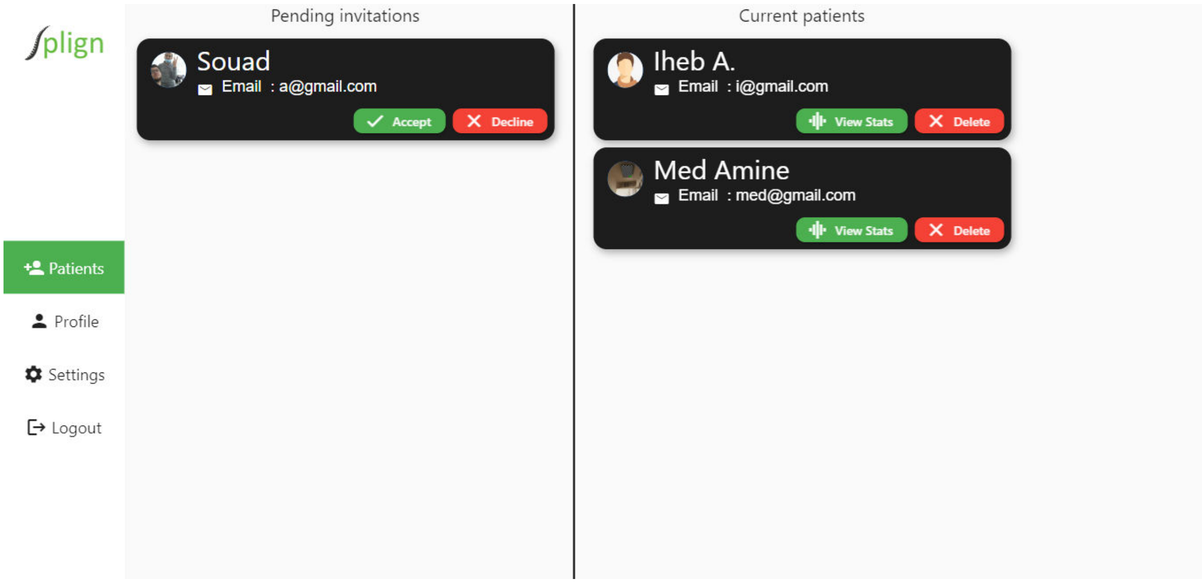

Fig. 19. Web application: Pending and Current invitations.

The doctor can track his data patient in this interface.

Fig. 20. Web application: Patient recordings.

The doctor can see the invitations sent from the patients and he can choose which patient to accept and track.

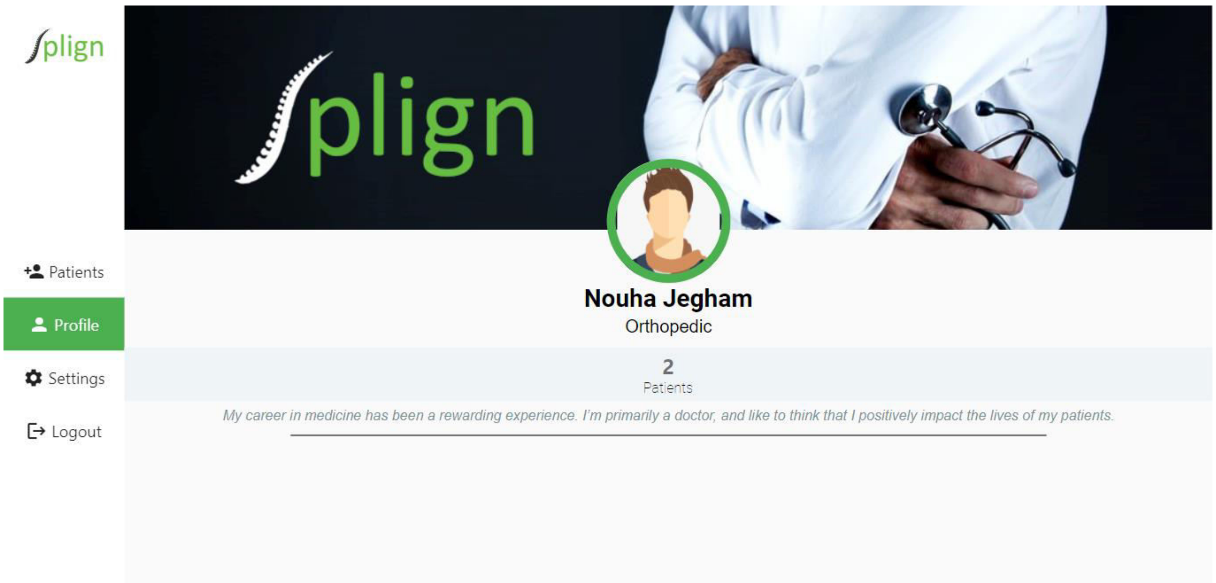

Fig. 21. Web application: Profile Page.

The doctor can edit his profile in this page.

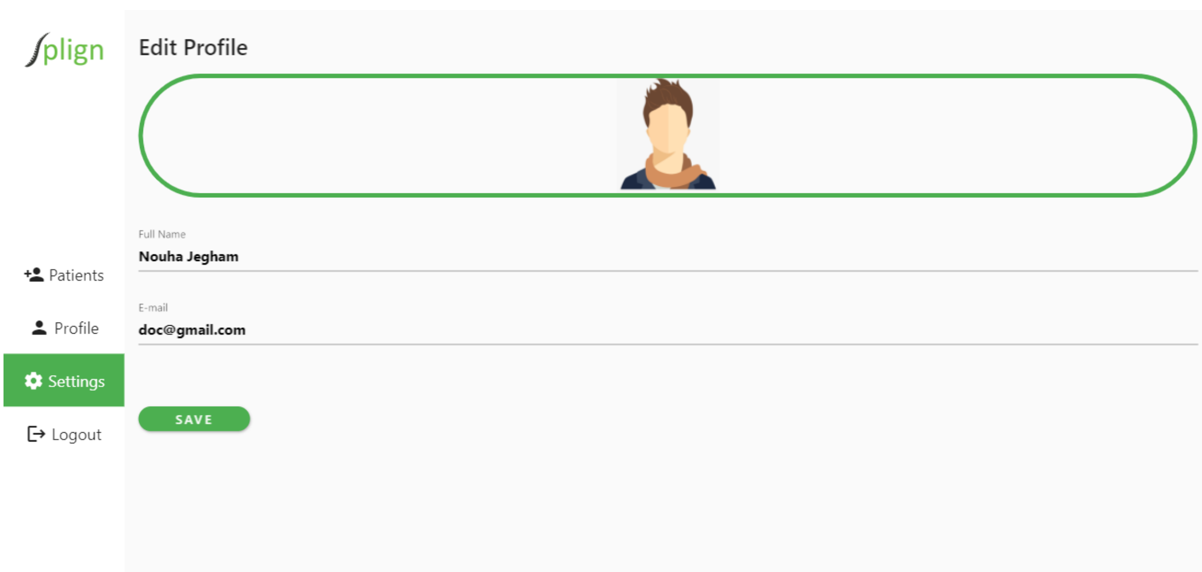

Fig. 22. Web application: Edit Profile.

9. Conclusion

In this paper, we detailed our proposed system for sitting posture monitoring. We studied and analyzed the applicability and the efficiency of the machine learning algorithms specially for the inertial sensors that is not previously treated in literature. The accuracy and time processing are analyzed for the following machine learning algorithm: Convolutional neural network (CNN),

The K-nearest neighbors (KNN), Support-vector machines (SVM), Decision tree classification, Random forest, Naive Bayes Classifier and Boosting algorithm. The strategy of the database implementation is detailed. And two test phases are explained. The Random Forest algorithms presents the best accuracy compared to
the other algorithms with 99.67% and time processing 67.7ms suitable for real time application. We finalized the implementation of complete system which is composed of a smart belt equipped with inertial sensors, mobile application for user posture correction and web application for doctor monitoring. The posture prediction is based on Random Forest algorithm to define the posture.

As a future work, we will test and adapt our proposed system to correct posture for healthy adult population (male and female) during working hours. We will study the accuracy improvement of the machine learning accuracy by proposing new methods of machine learning application. In addition, we can enhance the applicability and user experience of the proposed system by focusing our future study on expanding testing demographics, addressing potential discomfort from prolonged wear, increasing user engagement, evaluating long-term impacts, and exploring hardware adaptability.

Acknowledgments
We would like to express our gratitude to Higher School of Communication of Tunis students: Nouha Jegham and Mohamed Amine Toumi for their contribution to reach these results.

References

[1] Mohammed, Mohssen and Khan, Muhammad and Bashier, Eihab, Machine Learning: Algorithms and Applications., Publisher: CRCPress, 2016, ISBN: 9781498705387

[2] Carbonell, Jaime and Michalski, Ryszard and Mitchell, Tom and Carbonell, Jaime, Machine Learning: A Historical and MethodologicalAnalysis, Artificial Intelligence Magazine, 2002, vol. 4

[3] Patel, Yash, Machine Learning in HealthCare, Machine Learning, May 2021, DOI: 10.13140/RG.2.2.30835.71207

[4] Kulikajevas A, Maskeliunas R, DamaĹˇeviĂŤius R., Detection of sitting posture using hierarchical image composition and deep learning, PeerJ Computer Science 7: e442, 2021, DOI: 10.7717/peerj-cs.442

[5] Sinha, V.K., Patro, K.K., PĹ‚awiak, P., Prakash, A.J., Smartphone-Based Human Sitting Behaviors Recognition Using Inertial Sensor, Sensors, 2021, DOI: 10.3390/s21196652

[6] Bourahmoune, K., Ishac, K., Amagasa, T., Intelligent Posture Training: Machine-Learning-Powered Human Sitting Posture Recognition Based on a Pressure-Sensing IoT Cushion, Sensors, 2022, DOI: 10.3390/s22145337

ISSN: 2456-3676

[7] Feradov, F.; Markova, V.; Ganchev, T. Automated Detection of Improper Sitting Postures in Computer Users Based on Motion Capture Sensors, Computers, 2022, DOI: 10.3390/computers11070116

[8] Wan Q, Zhao H, Li J, Xu P., Hip Positioning and Sitting Posture Recognition Based on Human Sitting Pressure Image., Sensors (Basel), 2021 Jan, DOI: 10.3390/s21020426.

[9] Zemp R, Tanadini M, Plüss S, Schnüriger K, Singh NB, Taylor WR, Lorenzetti S., Applicationof Machine Learning Approaches for Classifying Sitting Posture Based on Force and Acceleration Sensors., BioMed Research International,2016, DOI: 10.1155/2016/5978489.

[10] Kim, Yong and Son, Youngdoo and Kim, Wonjoon and Jin, Byungkiand Yun, Myung, Classification of Children's Sitting Postures Using Machine Learning Algorithms., Applied Sciences, 2018, DOI: 10.3390/app8081280.

[11] Gelaw, T.A., Hagos, M.T., Posture Prediction for Healthy Sitting Using a Smart Chair., International Conference on Advances of Science and Technology ICAST 2021, Bahir Dar Ethiopia, November 2021, vol 411. Springer, DOI: $10.1007/978 - 3 - 030 - 93709 -$ 626

[12] Ferdews Tlili, Rim Haddad, Ridha Bouallegue, Raed M. Shubair,

Shoulder and Trunk Posture Monitoring System Over Time for Seating Persons., International Conference on Advanced Information Networking and Applications: AINA 2021, Toronto, ON, Canada, May 2021

[13] java T point, Introduction of Convolutional Neural Network in TensorFlow, [Online] Cited 2022-07-21. Available at: https://www.javatpoint.com/convolutional-neural-network-intensorflow

[14] Onel Harrison, Machine Learning Basics with the K-Nearest Neighbors Algorithm, [Online] Cited 2018-10-10. Availableat: https://towardsdatascience.com/machine-learning-basics-withthe- k-nearest-neighbors-algorithm-6a6e71d01761

[15] Data Analytics Post, SVM, [Online] Cited 2017. Available at: https://dataanalyticspost.com/Lexique/svm/: text=SVM2020(Support2020Vector2020Machine2020ou,ou2020de2020d20C320A9tection2020d27anomalie

[16] Geeks for Geeks, Decision Tree, [Online] Cited 2023-08-20. Available at: https://www.geeksforgeeks.org/decision-tree/

[17] Sruthi E R, Understanding Random Forest, [Online] Cited 2023-10-26. Available at:

https://www.analyticsvidhya.com/blog/2021/06/understandingrandom-forest/: text=Random%2020forest%2020is%2020a%2020Supervised,average%2020in%2020case%2020of%2020regression

[18] Geeks for Geeks, Naive Bayes Classifiers, [Online] Cited 2023-09-13. Available at: HTTPs://www.geeksforgeeks.org/naive-bayes-classifiers/

[19] Mrinal Singh Walia, Best Boosting Algorithm in Machine Learning In 2021, [Online] Cited 2021-04-27. Availableat: https://www.analyticsvidhya.com/blog/2021/04/best-boostingalgorithm- in-machine-learning-in-2021/

[20] Physiomed, Correct Sitting Posture: Office, [Online] Cited 2016-07-20. Available at:

https://www.physiomed.co.uk/uploads/guide/file/20/ Physiomed-Sitting-Guide-Correct-Sitting-Posture-Digital.pdf

[21] Canadian Center for Occupational Health and Safety (CCOHS), Working in a Sitting Position, [Online] Cited 2022-09-28. Availableat: https://www.ccohs.ca/oshanswers/ergonomics/sitting/sittingposition.html

[22] InvenSense, MPU-6000 and MPU-6050 Product Specification Revision 3.4, [Online] Cited 2013-08-19. Availableat: https://www.rlocman.ru/i/File/2017/01/28/MPU-6000-Datasheet1.pdf

[23] Rasberry Pi, Raspberry Pi 400 Tech Specs, [Online] Cited 2012. Available at:

https://www.raspberrypi.com/products/raspberrypi400/specifications/

[24] Corinne Bernstein, Kate Brush, Alexander S. Gillis, MQTT (MQ Telemetry Transport), [Online] Cited 2021. Available at: https://www.techtarget.com/iotagenda/definition/MQTT-MQTelemetry-Transport

[25] Eclipse, Eclipse Mosquitto: An open source MQTT broker, [Online] Available at: HTTPs://mosquitto.org/

[26] Flutter, Flutter documentation, [Online] Available at: https://docs.flutter.dev/