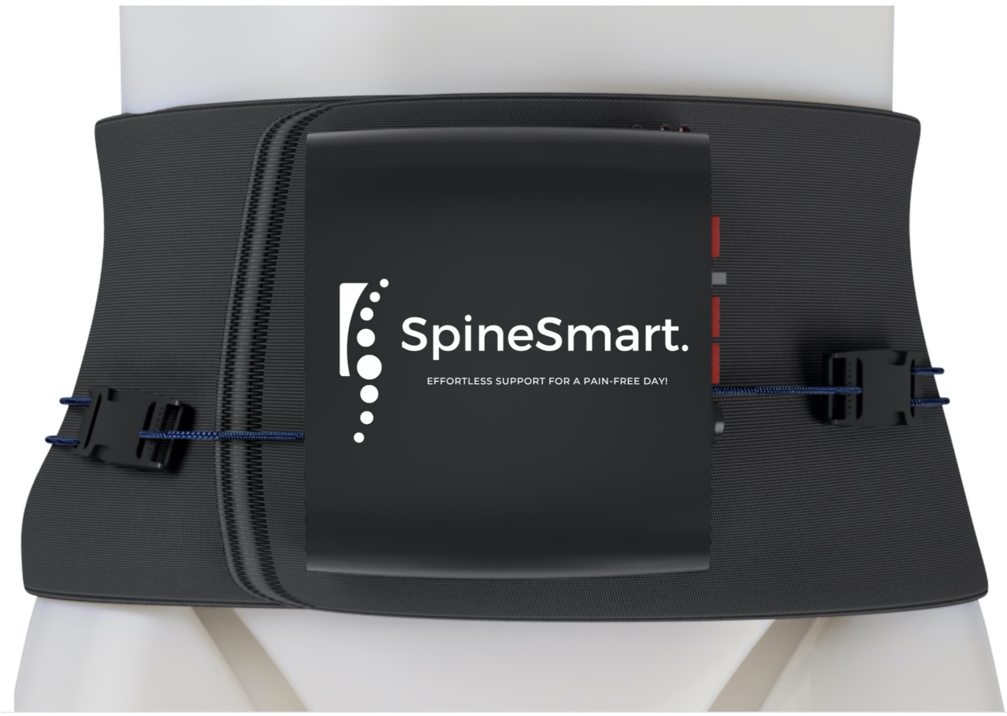
Products design & systems engineering
Microengineering Section, Master Phase
Fall 2022

SpineSmart
Active Lumbar Support Belt

Group 6

Skander Ben Messaoud (SCIPER: 302153)
Edwin Bertschy (SCIPER: 287811)
Ambrine Douhane (SCIPER: 282708)
Pierre-Jean Renaud (SCIPER: 297134)
Noemi Willmann (SCIPER: 301786)

Team coach: Samuel Youcef Benketaf
Professors: Yves Bellouard / Edoardo Charbon



Executive Summary

According to a study performed by the Swiss league against rheumatism, 50 % of the Swiss population experiences moderate back pain around the lumbar region. A common solution to address these pains is the usage of lumbar support belts which provide support to this region and alleviate stress on the lower back.

However, these belts come with many limitations, the main one being their lack of adaptability to the user whose back needs a different level of support throughout the day.

SpineSmart is an innovative product aimed to tackle this issue by providing an automated lumbar support belt which adapts to the users' needs and optimises the support throughout its usage. Using a set of sensors and an automated tightening system, the device is capable of responding to the user's activity and offering the optimal level of support accordingly, for a comfortable, pain-free experience.

The primary market for SpineSmart are workers who are at risk of experiencing back pain, which we estimate to be over one million people in Switzerland. In addition to being accessible to the mainstream public, the product also has great potential for usage in medical settings, particularly for the rehabilitation of back injuries. Caregivers could prescribe the use of SpineSmart for accurate diagnosis and support control.
If mass-produced, we estimate the production cost of the device to be approximately 80 CHF, which makes the product competitive in the premium lumbar belt market with a margin of 100% to 200%.

Figure 1: SpineSmart, effortless support for a pain-free day.

Contents

List of Figures

List of Tables

1 Product concept

1.1 General description

In a recent study made by the Swiss league against rheumatism [22], people of different age groups have been asked about the frequency at which they have been experiencing back pain: 88% of the respondents indicated that they have experienced back pain at least once during the previous 12 months. 50% of the respondents even indicated experiencing these pains several times a month.
Lumbar aches are a very common health issue, some specialists even consider back pain as the *"disease of the century"*. These pains often arise due to stressful solicitation of the lumbar region, typically following muscle sprains when lifting heavy objects with improper form, but they can also be caused by improper spine posture in the sitting position as well as weak dorsal musculature, among other reasons.

Lumbar support belts have been a common solution for relieving back pain. They exist in different sizes and shapes to address different levels of lumbar pain.
The functioning of these belts relies on providing additional support to the lumbar muscles and rectifying the spine's posture in a way to diminish the load on the intervertebral discs, which will relieve stress on the spine and the lumbar muscles, and therefore alleviate pain in the region. They also prevent stress on the back by limiting its range of motion.

However, there are some use cases where the traditional belt has some drawbacks:

- Comfort: These belts are often rather uncomfortable to wear as they keep one constantly tight.
- Prevention of back pain and posture correction: The lumbar belt is not recommended for exclusive preventive purposes, in fact by providing additional support for the back over a long period and limiting its range of motion, the stability of the back would decrease over time (proprioceptive loss) [23]. Some studies even indicate that overusing a belt can weaken the back muscles and atrophy them [20].
- For people experiencing chronic pain: Wearing support belts over long periods of time could similarly result in back stability issues after a certain period due to the constant support that the back is now getting used to.

It is in this context that we thought of enhancing a traditional belt to solve these issues, and come up with a solution to normalise the usage of lumbar belts for prevention purposes, as well as making it more effective for combatting back pain.

Our idea consists in making an augmented support belt. It relies on automating the tightening adjustment of the belt depending on the position of the user's back and activity. The belt would tighten when it detects a stressful position for the back of the user and loosen when it detects an activity for which the user does not need support. In addition to that, the belt could save data and provide diagnostic to the user or to a caregiver, who can prescribe a support curve throughout a certain period which accounts for the user's specific needs. This would not only overcome the problem of having prolonged unnecessary support to the back, it would also make the device more effective at relieving pain while providing a more comfortable experience to user since the belt would not apply constant pressure on the waist.

1.2 Brief market analysis

The lumbar belts' market has been rapidly growing as people are adopting a more sedentary lifestyle which could increase the risk of experiencing back pains, but they are also gaining more awareness on the impacts it could have on their health, which in return makes the demand for these products more important. While classic support belts are now common on the market, they do not provide the ideal level of support throughout the day and might end up being uncomfortable to use, lowering the solution's effectiveness. Other types of active belts whose key added value is the use of heat for back pain treating have also been brought to the market.
Our main target are the in-office workers or any person that would like to get spine support for relieving back pain.

1.3 Unique selling points

The SpineSmart support belt technology uses a versatile control algorithm that adjusts the tension of a lumbar belt's tightening cables using a motor, providing a customised and dynamic support experience that adapts to the activity of the user throughout the day. The system features adjustable control parameters that can be modified to suit the specific needs of each user, enabling a wide range of applications. Data collected from the belt is used to inform the control algorithm and is made accessible to healthcare professionals for monitoring and analysis. This data-driven approach, combined with the ability to provide dynamic support with a high level of granularity, allows for precise and effective treatment, making our lumbar support belt a highly innovative solution in the field of medical rehabilitation, and a reliable back pain relieving device for workers experiencing these pains.

2 Technical solution

2.1 Specifications

The problem definition process resulted in the establishment of a set of engineering specifications, which are organised in the table in Appendix 7.5.

These specifications are decomposed in four categories:

- Function: describes qualitatively a function that the system shall do.
- Constraint: describes quantitatively the minimum level of performance for the system to work.
- Performance: describes quantitatively a target level of performance of the system.
- Nice-to-have: describes functionalities that would be good to have if they do not bring addition constraints to the design.

2.2 Design phase: solution explored

During the design phase, a number of potential solutions were explored in order to find the most effective and efficient design for the product. The focus of these explorations was on three main areas: software and control development, structural and functional development, and ergonomic development. The following section will provide an overview of the various solutions that were considered and how they were evaluated.

2.2.1 Belt

The first requisite for the product would be to have a belt to work with. We could either try to build our own belt, or directly buy a certified one from the market and augment it. We obviously went with the latter option as the belt is not the main focus of the project and that we do not have the time and expertise to produce one. However, several types of belt exist on the market and we had many options. Two types of belts stood out : The first and most common type of belts used elastics for the tightening while the second type of belts used cables. We thought that a belt with cables would be easier to augment which is why we opted for the adjustable belt from Futuro 3M [8].

2.2.2 Tightness sensing

It is relevant to have a good feedback of the tightening for reasons such as safety and comfort. In fact, we would like the belt to provide a tightening level that is considered as high enough for a good support, but low enough for both the hardware and the user's safety (refer to 2.3 "Security concerns and integrity of the hardware" for more details.)

Two devices for sensing the tightening of the trunk have been explored and found inconclusive.

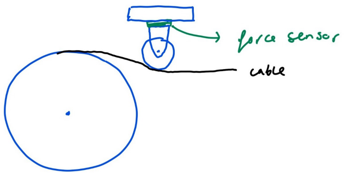


Figure 2: Pressure cell

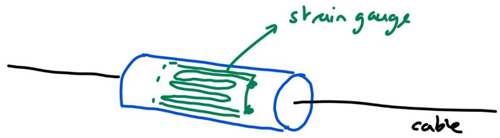


Figure 3: Strain gauge

A pressure cell has been tested to obtain the force applied by the belt on the trunk in the first hand, and the force between a pulley and the tightening cable on the other hand. It appeared that since many factors have an important impact on this force, in particular the user's position or the initial manual tightening set by the user, the values obtained from the pressure sensor were not useful. It has been concluded that determining the cables' tightening would be a more most efficient manner of quantifying the belt's support level.

Strain gauges have been tested with this idea in mind. They were put on a rigid material, and the deformation of this material would lead to a change of resistance, leading to voltage variation across the Wheatstone bridge which is measured on the Arduino after amplification. These gauges are however fragile and generally meant for higher end applications, in addition to being technical to implement which brought us to think about other alternatives of sensing the tightening.

A final idea was to take advantage of the relationship between the motor's torque and the current it draws to determine how much it pulls on the cables, thus allowing to quantify the belt tightening.

2.2.3 Movement sensing

Some tests have been performed with external accelerometers and gyroscopes (in particular GY-521 module), but bugs appeared rapidly, which is why we decided to focus on exploiting the internal IMU of our Arduino.

2.2.4 Structure

Two main ideas emerged when the structure had to be chosen. The first idea was to implement two compartments each with a motor and pulley to tighten on both sides of the belt simultaneously. The second idea was to have a single structure with a unique pulley and motor that would be placed on the frontal region of the belt. They are shown in Fig. 15 and 16. The differences between the two are summed up in Tab. 1.

Single pulley mechanism	Double pulley mechanism
+ No cables between the elements	+ up/down decoupling possible
+ single bloc	– 2 mechanisms to be produced
+ nothing under the arms	– belt shortening needed
– system for unplugging the cables has to be developped	– shear in the velcros

Table 1: Pros and cons of both structures

After considering both options, we chose the single pulley mechanism due to its simplicity and the fact that it does not require the production of multiple mechanisms or the modification of the belt. The only drawback was the need to develop a system for unplugging the cables. In contrast, the double pulley mechanism offered the advantage of decoupling tightness levels from the upper and lower parts of the belt, but it required the production of two mechanisms and the need for belt shortening. There was also the potential for shear in the Velcros securing the system to the belt. Overall, the single pulley mechanism was the more appealing choice for our application.
Another idea was to use the pulley in the opposite direction with a system of ratchet, but this one was abandoned for complexity stakes. A simple sketch of this idea is shown in Fig. 17 in Appendix.

2.2.5 Prototyping approach

As for the ideation phase, the prototyping and development strategy was adapted to each of the three fields, with the appropriate testing tools.

- For the software and control development, the electrical circuit had to be developed which was done using a breadboard and jumpers. The software and the circuit were being constantly updated as the tests were performed. After the electric circuit was set, two iterations of the veroboard were designed with the aim of optimising the size of the prototype.

- For the structural and functional development, a functional prototype was built, with the purpose of testing the resistance and the main functions of the system such as the worm gear system or the user interface, and allowed to rectify some design errors like the size of the pulley (Fig. 4). This prototype was also used to verify that the final product would meet all the functional and constraint requirements (Appendix 7.5).

- For the ergonomic development, an ergonomic prototype was built, embedding all the functionalities from the functional prototype, with a compact profile and minimal weight (Fig. 5). This prototype was used to test how the final product would perform in terms of performance and nice-to-have requirements (Appendix 7.5).

Figure 4: Functional prototype

Figure 5: Ergonomic prototype

2.3 Technical choices made

2.3.1 Selected design

Microcontroller:
We have chosen to use an Arduino Nano for its compact size and since it offers a sufficient amount of ports for our application. A higher end model of Arduino Nano was used, namely an RP2040, as it offers additional functionalities such as the machine learning core that we used for the movement detection, but also supplementary connectivity such as IoT cloud for synchronising data with other devices.

Actuation:
The actuation system is used for tightening the belt. As shown in Fig. 6, it consists in a DC motor with an integrated encoder, a worm screw, a pulley and cables.

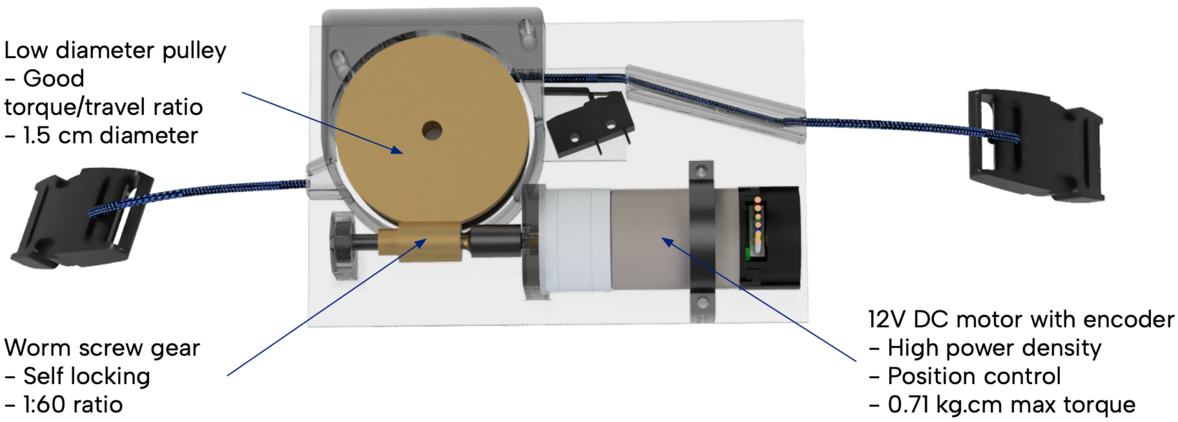


Figure 6: Actuation mechanism

We chose to use a 12 V DC motor rather than a stepper motor since it delivers a higher torque for a more compact size and weight. The motor has an integrated 48 CPR encoder allowing accurate position control. It delivers $T_{mot} = 0.71\,\text{kg} \cdot \text{cm}$ of maximum torque at its 1.8 A stall current.

To drive the motor, we use an L298N H bridge motor driver which uses a PWM from the arduino to drive the motor. We observed that this driver was quite inefficient as it resulted in a voltage drop of up to 4 V between its input voltage and the voltage that it delivers across the motor. We have therefore used four 18650 battery cells in order to deliver a voltage of around 16 V on the driver (12 V on the motor) and maximise the torque we can get out of our motor.

The transmission between the torque and the pulley is done through a worm screw gear. This type of gear has been chosen for its self locking capabilities which ensures that the tightening on the belt can be maintained without the need of providing supplementary power from the motor. The gear that we selected has a 1:60 reduction ratio.

Concerning the pulley, we used a relatively low diameter of 1.5 cm in order to have a good compromise between the provided torque and the tightening speed. A plain bearing is used to allow rotation of the pulley.

Tightening sensing:
In order to sense the tightening, we ended up opting for a current sensor connected to the motor. The value of the current can be easily linked to the external torque on the motor (they are proportional for a DC motor) and thus to the tightening of the cables. The ACS712 sensor signal had to be processed to make it exploitable, namely by using a moving average filter to smoothen its signal. A final resolution of 50 mA could be achieved and was sufficient for our application.

Movement sensing:
The movement is sensed by the IMU of the Arduino Nano RP2040, an LSM6DSOX. We used a library [3] from the Sensors Software Solution Team of STMicroelectronics that implements a machine learning algorithm for movement recognition. The software collects 75 samples of data from the IMU on an interval of 3 seconds and uses it to identify and categorise different types of movement: biking, walking, driving, running, and idle (no movement).

User interface:
The user interface comprises of an on/off button, a + and a - button, a setup button and a slider for switching between the manual and automatic mode. On the upper side of the device, three LEDs have been implemented to communicate with the user : on the right part, a red LED is used to indicate low battery. On the left part, a pair of red and green LEDs are used to indicate the different modes and states of the system.

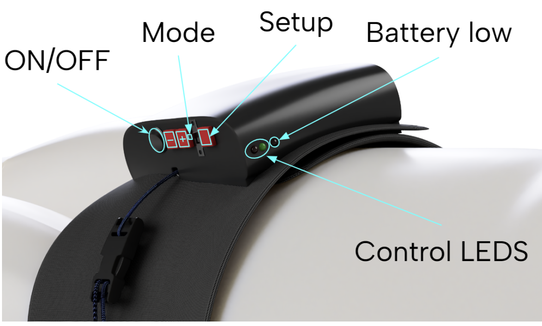
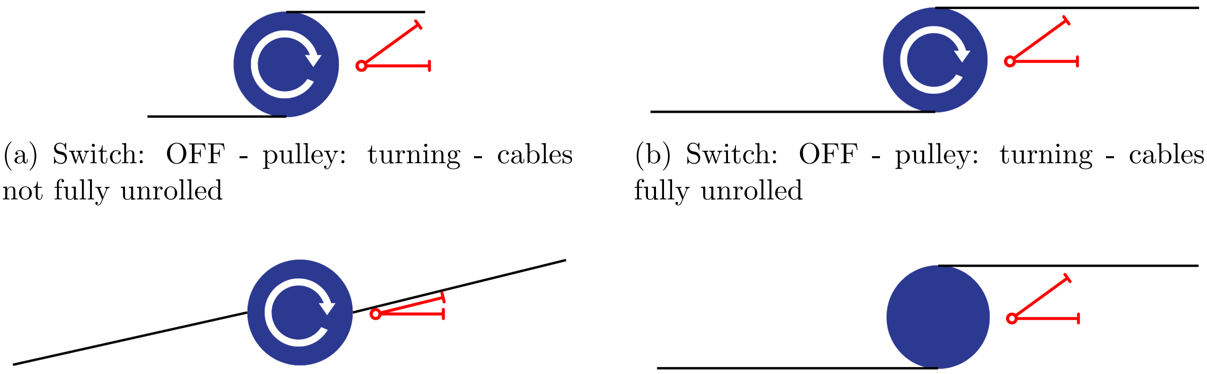


Figure 7: User interface

Control:
The software implementation of the device works as the following : when the system is turned on, an initial setup phase takes place. The setup will initially check the battery level and indicate on the low battery LED if there is need for recharging. If the battery level is sufficient, the system will perform the homing, which commands the motor to keep rotating and unrolling the cables from the pulley until reaching the 0 position which

is detected with the limit switch (Fig. 8).



(a) Switch: OFF - pulley: turning - cables not fully unrolled

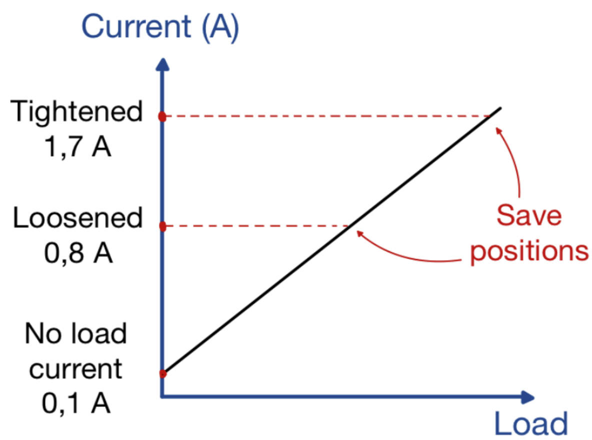
(b) Switch: OFF - pulley: turning - cables fully unrolled

(c) Switch: ON - pulley: turning - cables fully unrolled

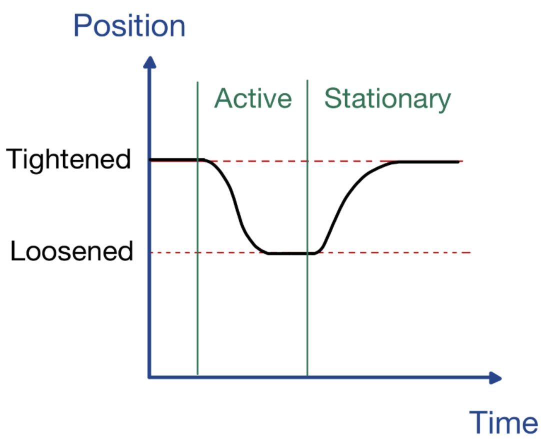
(d) Switch: OFF - pulley: stopped - cables fully unrolled

Figure 8: Principle of the limit switch homing

Then, once the homing is done, the system will await that the user presses the + button to start gradually tightening itself while monitoring the current. Two current thresholds are defined, one for the loosened position and the second for the tightened one. The system will save the positions at which these currents are reached in order to use them in the automatic mode. (Fig. 9a).



(a) Automatic setup



(b) Automatic mode

Figure 9: Both automatic mode and setup graphs

Once the setup is done, the loop() function is called as long as the system is on. This loop constantly reads the button inputs as well as the motor current and battery voltage, and it uses these inputs to commands the different modes, namely the low battery, manual, automatic, and setup modes:

- The low battery mode will turn on the low battery LED and interrupt the motor to indicate to the user that the system needs to be charged.

- The manual mode or automatic mode can be selected with the slider. In the manual mode, the control loop will use the + and - buttons inputs to command the motor to tighten or loosen the belt. The user can recognize that he is on the manual mode if he sees that the green and red control LEDs are turned off on the user interface.
- In the automatic mode, the system will enable the movement detection algorithm and change the tightening automatically based on the user's activity. As seen in Fig. 9b, it will alternate between the tightened and loosened positions initially determined in the setup, based on the user's physical activity. When he is active and that the belt is loosened, only the green LED will be on, and when he is inactive and that the belt is tightened, only the control red LED will be on. The transition between the different positions is performed by a PID controller which has been tuned to have a relatively damped response.
- Finally, if the user would like to manually change the tightening levels of the automatic mode, he can use the third setup button and adjust these positions with the + and - buttons. At the first press of the setup button, the red LED will start blinking, indicating he can adjust the tightened position. At a second press of the setup button, the green LED will be blinking, indicating he can adjust the loosened position. At the third click, the setup is finalized and the belt will come back to the manual or automatic mode depending on the position of the switch.

Security concerns and integrity of the hardware:
In our prototype, the motor was driven until its stall regime (around 1.8 A), at which the maximal tightening that could be achieved was not sufficient to harm the user's back and corresponded to a moderate amount of tightening. We imagined that a user could over sollicitate the motor by commanding it to increase the tightening more than its maximum for an extended period, which would cause overheating of the motor and damage it. We have therefore implemented a current limiting timer which will detect if the motor current is superior to 1.5 A for over two seconds, in which case it will disable further tightening for a few seconds to allow it to cool down. Whenever this stall regime is detected, the two left LEDs will blink.

However, if the system was optimised and that it provided more torque, it could potentially overtighten on the lumbar region and cause harm to the user before the motor stalls. In this case, the current limiting code would be used to ensure the user's safety and to prevent harming his back.
In addition to that, the user can at any moment open up the buckles and detach the systems' Velcros as a safety measure.

2.3.2 Concept of Operation (Con Ops)

An example of daily usage of the system can be seen in Fig. 10:

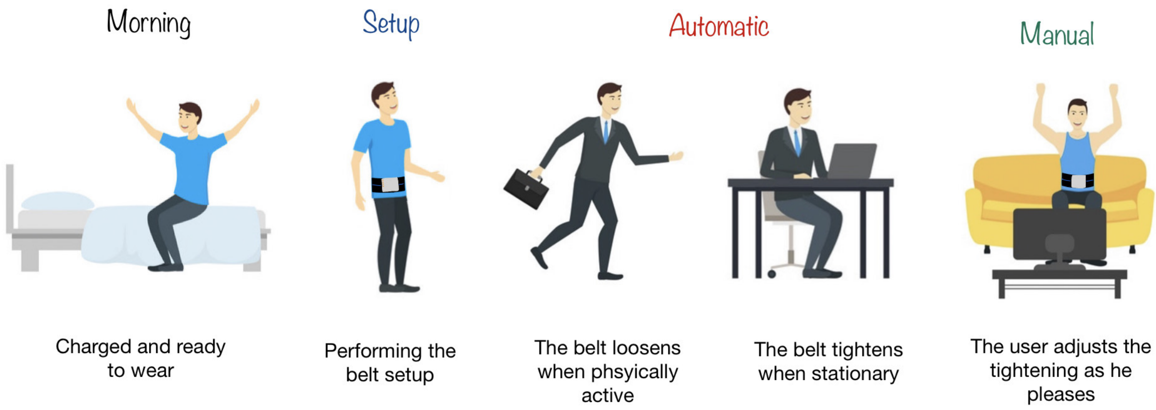


Figure 10: Example of daily usage

The device can be worn in the morning after being recharged from the previous days' usage during the night.

The user should start by wearing the belt. He can then attach the device using the velcro and turn it on with the power button. It will automatically start the setup by doing the homing. Once the homing is done, the user can attach the clips and press the + button which will finish the automatic setup by tightening the cables. The user can also change the setup of the automatic mode by pressing the setup button whenever he wants to during the day.

The belt is meant to be used in its automatic mode throughout the working day as it will optimise the level of support depending on the users' activity.

If the user would like to get more comfortable and to define manually the level of support, he can do that using the manual mode.

2.3.3 Analysis/modelling of the key functions related to the unique selling points

The main unique selling point of SpineSmart is its ability to provide the optimal level of support throughout its usage. This level of support is mainly dependant on the physical activity of the user, his posture, the time of the day, and his back condition if he is undergoing a rehabilitation treatment.
Movement detection is therefore a key functionality for this product. For our prototype, as a proof of concept, basic motion detection has been implemented to distinguish a physically active or inactive user.
The software basis is versatile and allows to offer different levels of support and tightening. The user is provided with an intuitive user interface allowing him to configure the device to maximize his comfort.
While the automatization of back support in reaction to the user's activity has been demonstrated, the system can still be upgraded to take into account the specific needs of

the user defined by his caregiver throughout a period of time.

2.3.4 Key elements in the design and how they could be further optimised

Efficiency:

The theoretical maximal load that can be pulled on both cables neglecting all friction is equal to the following :

$$F = T_{mot} \times i_{gear} / \frac{d_{pulley}}{2} = 0,71 \times 60/0,75 \approx 40\,\text{kg}$$

However in practice the system only delivered around 12 kg of total maximum load. We observed that a slight misalignment in the worm gear axis lead to an important increase in friction and a highly decreased performance. This is most likely due to the low tolerances of the 3D printed build as well as a relatively average fixation of the motor. Having higher tolerances for the more critical areas could lead to an important improvement in performance. The rotating pulley also presents some friction and its fixation could be optimized with the usage of a bearing.

Power consumption:
An improved efficiency will allow to improve the power consumption of the motor since the same output torque on the pulleys will be achieved with a lower current. Another way of improving the consumption would be to change the highly inefficient motor driver that we used with another more efficient one.

Movement and posture detection:
Our prototype's movement detection was limited to sensing whether the user was active or not. However, in practice, posture also plays an important role and is regarded as a important source of lumbar pain. A new iteration of the movement detection algorithms accounting for the bad postures would make the device more efficient for tackling back pain.

Data log and customized level of support:
The belt could become effective for rehabilitation by implementing a data log system that would indicate to a caregiver the support provided to the user's back as well as his physical activity. An additional functionality allowing the caregiver to prescribe different levels of support throughout a period of time would make the product an effective solution for a rehabilitation of the back following a lumbar injury.

2.4 Manufacturing choices and proposals for future production

2.4.1 Prototype fabrication

As said in section 2.2.5, two prototypes were developed for this project.
The functional prototype was created using a 3D printed frame for fast prototyping, and the mechanism machined from the aluminium, brass, and steel elements bought on the

market, as shown in Fig. 11. The motor was mounted to the frame using nuts and bolts and hot glue was used to secure the long cables with connectors in place for easy disassembly. The prototype was housed in a simple oversized laser-cut box for the test phase and the demo day.
The ergonomic prototype was based on the functional prototype with improvements including a new mapping of the veroboard, shortened cables, and the use of a CAD software to create a smooth shell design with parametric design functionalities. The cables were also compactly managed for a thinner profile, while still allowing for reversible assembly. Both prototypes utilised battery cells borrowed from the SKIL and soldered to form a 16 V rechargeable pack that could be charged using a laboratory power supply.

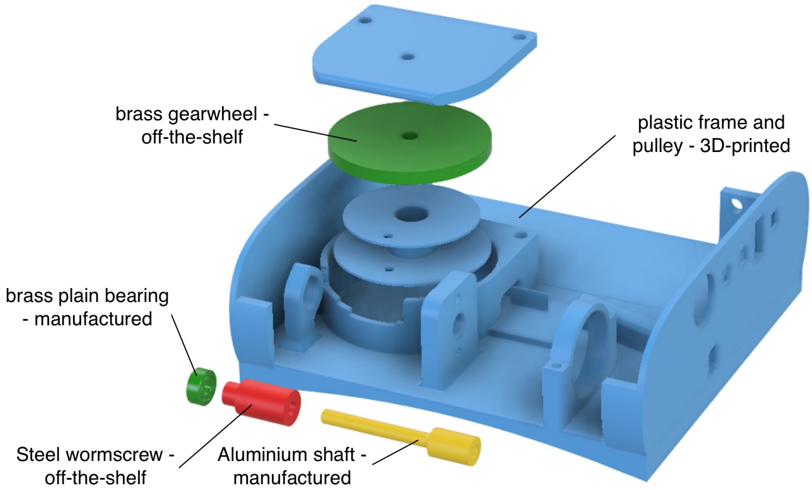


Figure 11: Prototypes manufacturing process - functional and ergonomic prototype

2.4.2 Final product (Estimated costs for manufacturing)

A bill of materials was drawn for the ergonomic prototype, amounting to a total of CHF 190.-, decomposed in table 7 in Appendix. It was used as a baseline for a cost estimation of a bigger scale product manufacturing.

The market analysis developed further in section 4.1.3 reveals a market of two million persons, a size of batch of 10k products was chosen accordingly. The batch size influences the means of development and production, which in turn impacts the cost breakdown.
For this scale, it becomes relevant to use plastic injection instead of 3D printing, and to develop a custom PCB instead of the prototyping board with independent components to reduce the costs related to the assembly and purchase of components.
The injection molding prices of a frame and shell similar to our product can be divided by a factor of ten [1].
By analogy, the price of the PCB and circuit components could be safely be divided by two, reducing the price by approximately CHF 35.- Applying an additional economy of scale of 10%, the manufacturing price of the system drops down to CHF 114,90.- as shown in table 2. This price does not include labor and shipping costs.

Item	Total price	Price reduction : injection molding	Price reduction : PCB	Price reduction : Economy of scale
Lumbar Belt 3M Futuro	CHF46,60			-CHF4,66
Wormscrew Gearwheel Z 1:60, module 0.75	CHF22,95			-CHF2,30
Platine SBC-MotoDriver	CHF7,30		-CHF3,65	-CHF0,37
Climbing rope 2 mm - 1m	CHF0,70			-CHF0,07
Set of 2 quick release buckles for backpack - 20mm	CHF4,00			-CHF0,40
4.4:1 Metal Gearmotor MP 12V with 48 CPR Encoder	CHF39,96			-CHF4,00
slide switch	CHF2,86		-CHF1,43	-CHF0,14
Arduino RP2040	CHF33,28		-CHF16,64	-CHF1,66
Rocker Switch	CHF0,11		-CHF0,06	-CHF0,01
18650 Battery cells, 3.7V	CHF18,00		-CHF9,00	-CHF0,90
LED 5mm (T13/4)	CHF0,42		-CHF0,21	-CHF0,02
10k resistors	CHF0,03		-CHF0,02	CHF0,00
push buttons	CHF0,88		-CHF0,44	-CHF0,04
Hall current sensor module ACS712	CHF1,19		-CHF0,60	-CHF0,06
Wires and connectors	CHF2,00		-CHF1,00	-CHF0,10
Vero board 80x40mm	CHF3,00		-CHF1,50	-CHF0,15
Hardware and fasteners	CHF1,50			-CHF0,15
PETG 3D printed parts	CHF6,00	-CHF5,40		-CHF0,06
TOTAL	**CHF190,78**	**-CHF5,40**	**-CHF34,54**	**-CHF15,08**
				CHF114,90

Table 2: New bill of materials - potential effective product

2.4.3 Quality analysis and control

The quality control should cover two main elements : the physical product, and the firmware operating it. Different strategies can be adopted for each of them. First, quality control can be performed on the manufacturing:

- By inspecting raw materials: Make sure that the materials being used to make the belt are of good quality and free from defects.
- By testing the belt during production: Conduct tests to ensure that the belt is functioning properly and meets all specifications.
- By checking finished products: Inspect the finished belts for any defects or problems.
- By performing functional testing: Test the belt to make sure it is functioning correctly and meeting all requirements.
- By following up with customers: Ask customers for feedback on the belt to see if it is meeting their needs and expectations.

Regarding the firmware, the quality can be tested before the launch of a new product, and improved with regular updates through the smartphone application. A quality control over this kind of firmware can take multiple forms:

- Testing the algorithm: Conduct tests to ensure that the algorithm is functioning properly and meeting all specifications. This might involve creating test cases that cover a range of different input conditions and verifying that the output is correct.
- Reviewing the code: Have a team of experienced developers review the code to ensure that it is well-written and follows best practices.
- Monitoring performance: Monitor the performance of the algorithm in real-world use to make sure it is functioning as expected.
- Making updates from customers feedback: Asking customers for feedback and data logs to verify and improve the control models.

3 Intellectual property IP analysis

3.1 Prior art search

As our product concept is a novelty, we only looked at devices with some features of our belt, which should be implemented.

Lumbar support belts:
Many non-active and manual lumbar support belts exist on the market. They exist with all kinds of different tightening systems. Some of the lumbar support belts have included heating pads, EMS or vibrating units in order to further relieve pain. (e.g. [9])

Posture correction:
Some smart posture correctors exist. Some products and papers could be found about this topic. However, those posture correctors do not help to improve your posture and are not pain relieving. They only detect a bad posture and remind the user to straighten the back. For example, the “Intelligent Posture Corrector” e.g. [10] or the paper. (e.g. [21])

Automatic tightening:
Automatic tightening for the lumbar support belt could not be found online. However, other automatic tightening systems for example for self-lacing shoes could be found. (e.g Nike [12])

3.2 Patent search

The first step in the IP analysis is to analyse the primary competitors. This will allow us to understand if the product can be patented. By operating a simple functional decomposition of the product, we extract four main concepts and mechanisms that could be patented: lumbar belts, support pads for lumbar belts, wire tightening systems for wearables, and automated tightening systems for wearables.
For each category, the most relevant families of patents have been gathered in the following table, including a list of the differences with the concept of the product. For each of them, the main differences between the current design and the patented design have been listed.

Table 3: Patent search

Ref	Title	Applicant	Link and differences with current design	Image
US2009292230A1 [14]	Lumbar support belt	GIBAUD [FR]	**Similarities:** Lumbar belt design Strip fasten together over the stomach **Differences:** Elastic textile straps, ours are not elastic No automation possibilities V-shape in the back for additional support, the support belt does not have that	Appendix 20a
US11464663B1 [18]	Lumbar therapy belt (with adaptive pads)	ANSON DAVID [US]	**Similarities:** Support system integrated into a lumbar belt **Differences:** different attachment system for the patches	Appendix 20b
US2004097857A1 [13]	Brace with integrated lumbar support system	AMEI TECHNOLOGIES INC [US]	**Similarities:** Support pads for lumbar belts **Differences:** Not the same support strategy: air pressure Instead of dynamic support	Appendix 20c
KR101492477B1 [16]	Closure system for braces, protective wear and similar articles	BOA TECHNOLOGY INC [US]	**Similarities:** Wire tightening system for wearables **Differences:** Manual actuation, metal wire, no adaptation to lumbar belt	Appendix 20d
US2014094728A1 [15]	Motorized tensioning system for medical braces and devices	BOA TECHNOLOGY INC [US]	**Similarities:** Motor tightening system for wearables **Differences:** smaller adjustment range, no automation i.e.tightness autonomously adjusted depending on the user's movement. Actuation system not removable from the shoes	Appendix 20e
CN106231942A [17]	Footwear having motorized adjustment system and removable midsole	NIKE INNOVATE CV	**Similarities:** Motor tightening system for wearables **Differences:** smaller adjustment range, adapted only to shoes.	Appendix 20f

All the mentioned patents in the table are still valid. Therefore, the lumbar support belt as used in the prototype is not a novelty. To be precise, the lumbar support belt was bought and only the cables were changed. The same is true for the adaptive pads, they came with the bought lumbar support belt and were also patented in 2019 by another company.

There already exists a brace with an integrated lumbar support system which regulates the pressure on the lumbar region with air pressure. However, this is not a dynamic system as the system is and is not automatic because the user still has to manipulate the air manually.
There already exists a patent for a wire tightening system for wearables, though the mechanism for the actuation is not specifically mentioned. Our mechanism does not leave enough freedom for a patent, especially because there exists already two patents for motorised tensioning systems. However, both have smaller adjustment ranges and one of them is only adapted to shoes.

3.3 Discussion on opportunities for IP and possible IP strategy

In order to patent and trademark SpineSmart in Europe EUIPO (European Union Intellectual Property Office) has to review the application and ensure that it meets the necessary requirement. This includes a review of the product as have smaller adjustment ranges, and one to be shown how the name is being used in commerce. If the name is accepted SpineSmart gets a trademark and exclusive rights to use the name SpineSmart for active lumbar support belts. We would be in class 10 - Medical Instrument Products.

The augmented support belt tackles issues such as imprecise support by automating the tightening adjustment of the belt depending on the position of the user's back and providing the optimal level of support throughout the day. The concept of the product represents its main innovation. In fact, no product, which provides this functionality, has been found on the market, nor has a patent. A viable product that fulfills this function would be a novelty in the market and could therefore be patented.

In terms of the technical implementation of the belt, the system relies on a 6 axis IMU to detect the different positions of the user and control the motor, which is not particularly innovative to be . The other components used in the device are standard such as the worm gear, the DC motor with encoder and the current sensor.

Lastly, the implemented code cannot apply for a patent but for a copyright. We used and existing library for the movement detection of our prototype, however, in future iterations, if a customised and highly performing model for posture and movement detection is developed then it could potentially be copyrighted.

3.3.1 Strategies to protect it

Having a patent on a product brings many advantages and opportunities. The main and most obvious advantage is an exclusive right. This also brings a stronger market position, because competitors are not allowed to use the invention and therefore the company may get a market leader. Furthermore, one may have the opportunity to sell the idea even if it is not the group who wants to explore the business. An important aspect is also that it can be useful in case of a lawsuit and one does not need to pay for the licences of competitors, if they invent something similar. The product would be in class 10 - Medical Instrument Products as shown in table underneath.

Serial N° (E)	**Indication of goods**	**Serial N° (F)**	**Basic N°**
B 0289	Belts, electric, for medical purposes	C 0361	100175
B 0292	Belts for medical purposes	C 0360	100045
B 0301	Belts (Orthopaedic [orthopedic] -)	C 0362	100047

Table 4: Elements of the list of goods of Class 10: Surgical, medical, dental and veterinary apparatus and instruments, artificial limbs, eyes and teeth; orthopedic articles; suture materials. [19]

To sum up the IP-analysis, we have little to no liberty to patent any specific parts of the lumbar support belts and we, therefore, do not focus on this part. However, the idea of an augmented active lumbar support has enough freedom to be patented, as it brings novelty because it is an invention that does not exist on the market yet. Secondly, an augmented lumbar support belt is an invention. Lastly, it is industrially applicable as we could show with the prototype.
We would first need to fill in the patent application. This includes e.g. the title and the abstract.

Title:	Augmented Active Lumbar Support Belt
Abstract:	An augmented active lumbar support belt can be adjusted to the lumbar support belt, with wires in the back, which gets actively controlled by sensors which detect the position of the user. Furthermore, a diagnostic feature is implemented.

The next part are the claims:

1. A lumbar support belt comprising:
    - The lumbar support belt must be made out of scratchable material.
    - A control unit connected to the wires
    - Sensors (IMU-6) disposed on the lumbar support belt and connected to the control unit
    - The control unit is configured to adjust the tension in the wires based on the sensors
    - The sensors can detect the user's movement and position
2. The lumbar support belt of claim 1, further comprising a wireless communication module for transmitting and receiving data to and from the control unit in order to have a diagnostic feature.

If we receive a trademark for SpineSmart in order to protect the brand and to prevent others from using the same or similar name for competing products, as well as a patent for the automatic lumbar support belt then there would be two options to consider. Firstly, we could start a company and follow the pre-business plan or secondly, or we could sell it to another company.

Lastly, there are other options than patents, for instance trade secrets. However, after careful consideration, we have decided not to apply for trade secret protection for SpineSmart. While trade secret protection can be a useful tool for protecting the product, it is not always the most appropriate option for every product or situation.

Firstly, trade secrets are only protected as long as they remain secret. If the information becomes known to others, we may lose the trade secret protection. This can be a risk because the novelty of the product is not in the engineering but in the idea of the product and hence it could be easily copied.
Secondly, trade secret protection is typically limited to the specific information that is kept secret. If we develop additional features or improvements for the product, we may not be able to protect these innovations through trade secret protection.
Finally, trade secret protection does not provide the same level of legal protection as other forms of IP, such as patents or trademarks. If someone else uses the trade secrets without permission, we may have difficulty enforcing rights and may need to rely on other legal remedies.
Overall, we have determined that trade secret protection is not the most suitable option for SpineSmart and we decided that patents and trademarks are more suitable.

4 Aspect of a Pre-Business plan

4.1 Market

In Fig. 21 in Appendix, we tried to further investigate the pains and gains and what kind of values we wish for our product.

4.1.1 Research

A study made by the Swiss league against rheumatism [22] indicates that muscle contractures were cited much more often as pain sources by people exercising a “most seated” activity (63.4%), explaining why 79% of people surveyed try to take care of their posture. Sitting or standing with bad posture (arched back, bending forward) is a common source of back pain among people with non physical or partially physical occupations as this can lead to contractions in the back which are reported by 57% of people who experience back pain. This gives a good overview of the extent of back pain issues resulting from in-office work in Switzerland.
The Swiss Confederation reports about Opportunities and Challenges of the Prevention and Healthcare Market state that even if “the plethoric supply of new products leads to uncertainty”, it is clear that we predict a “growing market potential due to the increasing social importance of health” [6]. Today, the healthcare industry’s global sales are estimated at over 4000 billions US dollars per year, where medical technology and diagnostics occupy 10% of the total market.

4.1.2 Main competitors

Traditional back belts are already on the market, but they still present some remaining issues. These belts often have a recommended short period of usage, and wearing them simply for prevention is not advised. In fact, standard belts provide a constant support, even though the actual support that is needed by the user is more intermittent and depends on his physical activity and posture. By determining and providing the ideal level of support to the back, oversupport can be avoided and a viable belt for pain relief and prevention can be made.
As our main competitors, we can identify: *Bota*, that only sells passive lumbar belts and *Urgo*, that sells heating pads and electro-therapy lumbar belts.

4.1.3 Market size

Considering that:

- in Switzerland, 5.1 millions people are workers [7]
- more than 2 out of 3 employees have had, have or will have low back pain [22]
- this pain is caused at 57% from bad posture and stressful movements [22]

Then, a good approximation of the number of workers in the need of a solution in Switzerland is $5.1 \cdot 10^6 \cdot \frac{2}{3} \cdot 0.57 = 1'920'000$.

Knowing the needs in terms of releasing back pain for workers and the growing market of health worldwide, we consider it is relevant to develop such a device. In addition to that, office workers who have trouble maintaining correct posture could also use the belt to prevent harming their backs.

4.2 Strategy towards commercialisation

Commercialisation can be subdivided into different categories. First, it has to get approved, then, clients have to be targeted and reached by communication and finally we need suppliers.

4.2.1 Approval

The Swiss regulation are the same as in the EU and are defined by swissmedic as: "Medical devices are any medical equipment, instruments or consumables that predominantly come into contact with the human body and/or that investigate the human body, as well as accessories for these devices." [24] Therefore, we interpret a lumbar support belt as a medical device in the risk class I as a wheelchair. In this class we have to verify safety and performance. However, a declaration of Conformity and a registration number is needed. [24]

4.2.2 Model and communication

First, discussions will be engaged tn conclude an agreement with Futuro (3M), the company that sells the belt for which we developed our system.
Then, the lumbar support belt is planned to be sold to big companies with a lot of employees with back pain as well as pharmacies. Nowadays, back pain is very common among office workers sitting all day. Therefore we plan to sell it B2B (Business to Business). This has been decided because in the Swiss Back report of 2020 [22] it is mentioned that employers should do something to prevent back pain. 49 % of Swiss employers do not have any preventive measures. Hence, many employees have to do something against it and we could sell them our product. The belt and the mechanism are designed such that they can be easily separated. This allows employer to buy and lend SpineSmart to its employees, and wash the belt between a change of users.
As we work with a B2B model we need to have a good website and a good search engine appearance. Another important factor is to get in contact with big companies to sell our product.

4.3 Organization and planning

Our company can only function with a clear structure. Above everyone there must be a Chef Executive Officer for planning the long-term strategy. Underneath the CEO, the Chief Technical Officer is responsible for the technology infrastructure. SpineSmart is only a prototype, therefore we need engineers who develop the product further. Software engineers are also needed as we wish to further develop the software part for diversifying our clients. About the management and business - non-technical solution - a Chief Financial Officer and Human Resources responsible have to be appointed.

We would need to produce an actual product from our prototype. In order to do so we need money. First of all, we need a better prototype to attract investors. In order to pay for this we could do that by starting crowdfunding or by participating in competitions (e.g. venture). Once a presentable prototype is available we would need more money for production. In order to get more money we could look out for investors, as it is easier to convince those with an actual prototype.
Once a product exists, one may think about launching the product, we plan to launch our product in Switzerland, we think Switzerland is a good country to start because our first product tackles back pain for office workers, 47% of workers in Switzerland are sitting at least 6 hours per day, and 29% of the workers are sitting between 3 and 5 hours. [22] In this first phase of selling our product, we could further develop the software in order to get more companies interested in our product. However, this product may still be more expensive because at this point a large production cannot be afforded.

4.4 SWOT analysis

With the strengths, weakness, opportunities and threats (SWOT) model advantages and disadvantages can be identified (see Fig. 22 in Appendix 7.7).

4.5 Pricing analysis/competition

As shown in section 2.4.2, the unit cost of production is around 114 CHF. As the plan is to cooperate with 3M, the firm that produces the belt, it is realistic to consider that the price of the total system can be highly lowered at around 80 CHF, knowing that the belt itself constitutes one third of the production cost. It is also relevant to compare SpineSmart belt to other electrical lumbar support belts (e.g. heat to further reduce pain), those belts cost around 150-200 CHF.

It has been considered fair to set margins to 200% to cover all sorts of costs (HR, premises, hardware, patent, taxes...), this is why we plan to sell this augmented belt at 240 CHF to the buyer, and 100 CHF to central purchasing bodies.

5 Project Managment

5.1 General strategy

First, it was critical to define the project precisely. The project's objectives, deliverables, scope, and goals. Stakeholders had to be identified. Secondly, we created a project plan. This included schedules and tasks that needed to be done. A plan for the 500 CHF budget and the critical path should be identified as well. Next, roles had to be assigned for building the prototype. Furthermore, different groups were created following the work-breakdown structure (Fig. 12). Next, the Gantt chart should be followed by each member of the team as well as adapted when needed in order to deliver on time and within budget. The progress should always be discussed with the TA and the rest of the team members. In order to do so two meetings per week were scheduled. Those meetings were always divided into four parts. In the first part, the group members updated the team. In the second part, the group decided how to continue the project with the problems that occurred. In the third part, the group identified the different tasks and assigned them to the members of the team. And last was always a work session.

5.2 Work-breakdown structure

We created a work-breakdown structure based on five subpilers.

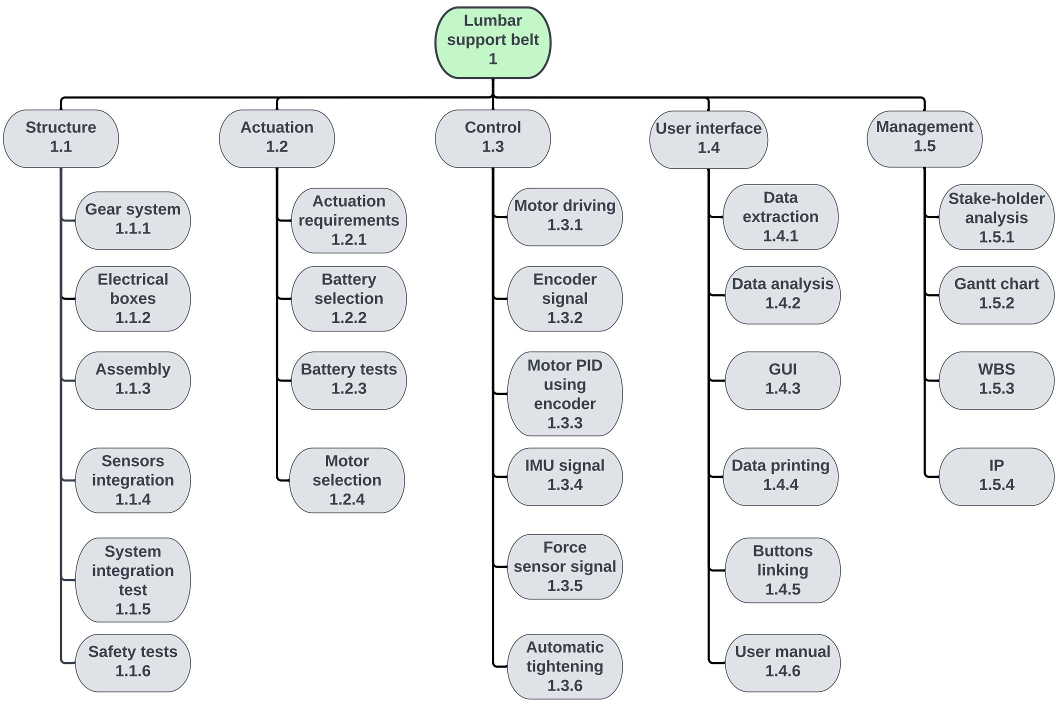


Figure 12: WBS of the lumbar support belt

5.3 Stake-holder analysis

The different stakeholders and their associated interests have been compiled in table 5.

Stakeholders	Needs
Patient	to benefit from more effective and comfortable devices to treat or relieve back pain
Employee	to take care of his/her own health/security during working hours in a comfortable manner
Doctor	to provide the best treatment to back pain
State	to levy taxes and ensure the product is safe
Supplier	to earn money from sales contracts
Pharmacy	to provide better health services to its clients
Employer	to take care of its employees carrying loads
Health insurance system	to save money by funding less expensive treatment for back pain and preventing injuries
Federal Office of Public Health (FOPH)	to avoid workplace injuries
EPFL	to train future engineers to design/develop systems
Teachers/TA	to make sure the objectives defined for the project are achieved

Table 5: Stake-holder analysis table

A stake-holder graph was created from the information gathered in table 5, using the MIT stake-holder value network system. The outcoming network is shown in Fig. 13.

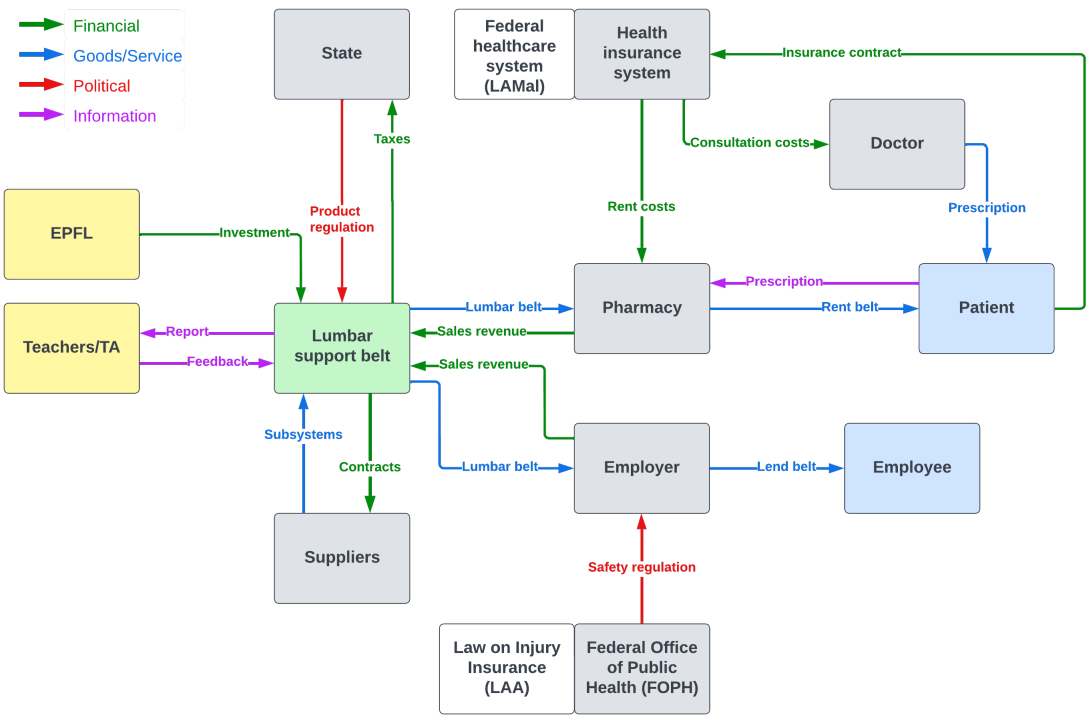


Figure 13: Stake-holder value network

5.4 Gantt chart

5.4.1 Initial plan

In our initial Gantt chart, the time was estimated with the 3-point estimate (PERT). However, this brought difficulties because the team members had only little experience with a lot of future tasks. The Gantt chart is structured in a topic selection, concept phase, design, implementation, prototyping and report phase. The work breakdown structure was the starting point to further analyse upcoming tasks. (See initial Gantt chart in Fig. 23 in Appendix)

5.4.2 Deviation from original planning

We actually had many problems with the force of the motor as well as programming the control unit. We did not plan enough time and got behind schedule such that we had only very little time for the sensor and user interface as we first had to solve the prior problems in order to get further. (See updated Gantt chart in Fig. 24 in Appendix)

5.4.3 Reflection on how the project went: difficulties encountered, remedies, etc

One of the biggest problems was clearly that many tasks took more time than planned and that as we had for a long time only one Arduino, only one group could work on it and we got stuck on the actuation and control and could not explore the other parts such as the user interface, diagnostics and especially the posture detection as much as we intended to.

One of our first problems was that we initially planned to 3D print our whole structure including the axis of the motor. However, we had to learn that this is not strong enough and too flexible. We, therefore, changed the axis and later also the bottom of the structure such that it is less flexible.

Another problem that came up was changing from one Arduino to another. We started with a cheap Arduino Nano from SPOT and as the Arduino Nano RP 2040 arrived, the control team continued working on the Nano, such that sensor control and userinterface teams could advance. However, once we wanted to combine the code we found out that for instance the button library does not work on the new Arduino Nano and new solutions for already solved problems had to be found.
Almost at the end of the project we had some electric short circuits and burnt two Arduinos. Firstly, this had the consequence that we had to find another Arduino on very short notice. Secondly, we burnt the Arduino of the ergonomic prototype and had to present our first (functional) prototype at the presentation in a less fancy casing.

5.5 Who did what?

We divided our project into different pillers Structure, Actuation, Control, User interface and Management as discussed in the work-breakdown structure.

Skander: During the first period of the prototyping, I explored the different motor options for our system. I borrowed a stepper motor from SKIL to perform tests on it and determine if it would be suitable for our application. We then decided to order a DC motor which I performed the initial tests on and implemented some of its control functions such as the PI controller. I then focused, jointly with Edwin, on developing the control software for the manual and automatic modes and other functionalities such as the current limiting. I then helped in the wiring and assembly of the ergonomic prototype and tuned some of the software's parameters as the tests were performed.

Edwin: At the beginning of the project, I was responsible for the structure and actuation design. I designed and 3D printed the first version of the single pulley mechanism for the design review. Along with Skander, I helped design the software and control architecture. Together, we wrote code for manual and automatic modes, enabling the user to interact with the belt using buttons and visualizing the system state through the use of LEDs. In addition to these tasks, I contributed to the design of the electric circuit and the optimized design of the veroboard for the ergonomic prototype. On the management side, I played a role in the creation of the WBS and stakeholder analysis diagrams.

Ambrine: I was in charge with Noemi of the sensors for sensing the accelerations in translation and rotation. I bought at the Spot-DLL the components (in particular sensors, Arduino, wires) and made tests with it. I began developing a strategy for linking the data sensed to a special type of movement using machine learning, involving the creation of a decision tree. It appeared at some point that this type of code already exists and is available open-source online, so we decided to use it directly. At the end I made the electrical circuit of the first prototype and I made the soldering on the vero board of it. When Edwin made an optimized version of it, I also participated to the fabrication of the second protoype.

Pierre-Jean: At the beginning of the project, I was responsible for the conception of the frame with Edwin. I also worked on the development of the mechanism and the optimization of the transmission chain while Edwin and Skander were optimizing the electrical losses. In addition to these tasks, I was in charge of the integration and management of physical interfaces. I also contributed to the 3D modeling and casing for both functional and ergonomic prototypes. Finally, I played a role in the assembly and cable management of the ergonomic prototype.

Noemi: First, Ambrine and I explored the different sensors and tried out how they work and how they can be used efficiently, we had to redo this task once we got the new Arduino. Then I made some research on the machine learning core of the Arduino RP and made tests with it. Furthermore, I looked up different methods for a user interface and implemented one of them at the very end. I also tried to implement an algorithm to detect whether someone is standing or sitting, but as the other team needed the Arduino more urgently we decided that it is better to not continue this part and I worked on the report.

6 Conclusion

To summarize, the hands-on experience was highly rewarding from both technical and managerial viewpoints. What has been achieved within time constraints is something to be proud of. If the project were to be repeated, there are certainly some areas for improvement. Firstly, including more spare parts in the shopping list would have been beneficial from a management perspective. A significant amount of time was spent debugging the software when the problem actually layed within the hardware. Therefore, having two microcontrollers available from the start could have saved valuable time, and also allowed to work on the software in parallel. One important learned lesson was that it is not necessary to reinvent the wheel, especially given the limited time and resources. This was particularly the case with the movement detection algorithm, which would not have been feasible to implement from scratch within the short time frame of a few weeks. From a mechanical standpoint, sourcing a worm gearbox directly from the market could have been a viable alternative to our custom solution. Naturally, there will always be room for further improvements as we strive towards a pain-free future.

The work completed on this project has established a strong foundation for future development. Some potential areas for improvement include reducing the size of the system to make it more discrete under clothing, increasing its humidity-resistant capabilities, creating a mobile application to control the belt, expanding the features of the automatic control to accommodate a wider range of cases, and implementing machine learning to adapt to individual users.

As we look towards the future development of SpineSmart, it is essential to consider sustainability throughout the product life cycle. It is our duty to minimise its impact on the environment by using sustainable manufacturing practices.

7 Appendix

7.1 Initial project idea

The idea was to create a system that would allow people to share selected objects with their neighbours. This object library would be integrated into an accommodation and was meant to enhance the sense of community, provide convenience, and promote sustainability. The system would be used by individuals living in collective spaces such as student housing, housing cooperatives, or open plan communities. To use the system, users would download an app, locate the desired object, and use their phone to virtually "handshake" with the owner and unlock the object for use. The design of the system had to consider challenges such as security, ease of use, flexibility, and insurance. The system would be implemented in the form of modular kits and would be evaluated based on metrics such as the number of rents and the number of products let for rent. The first prototype would be demonstrated in an FMEL residence.

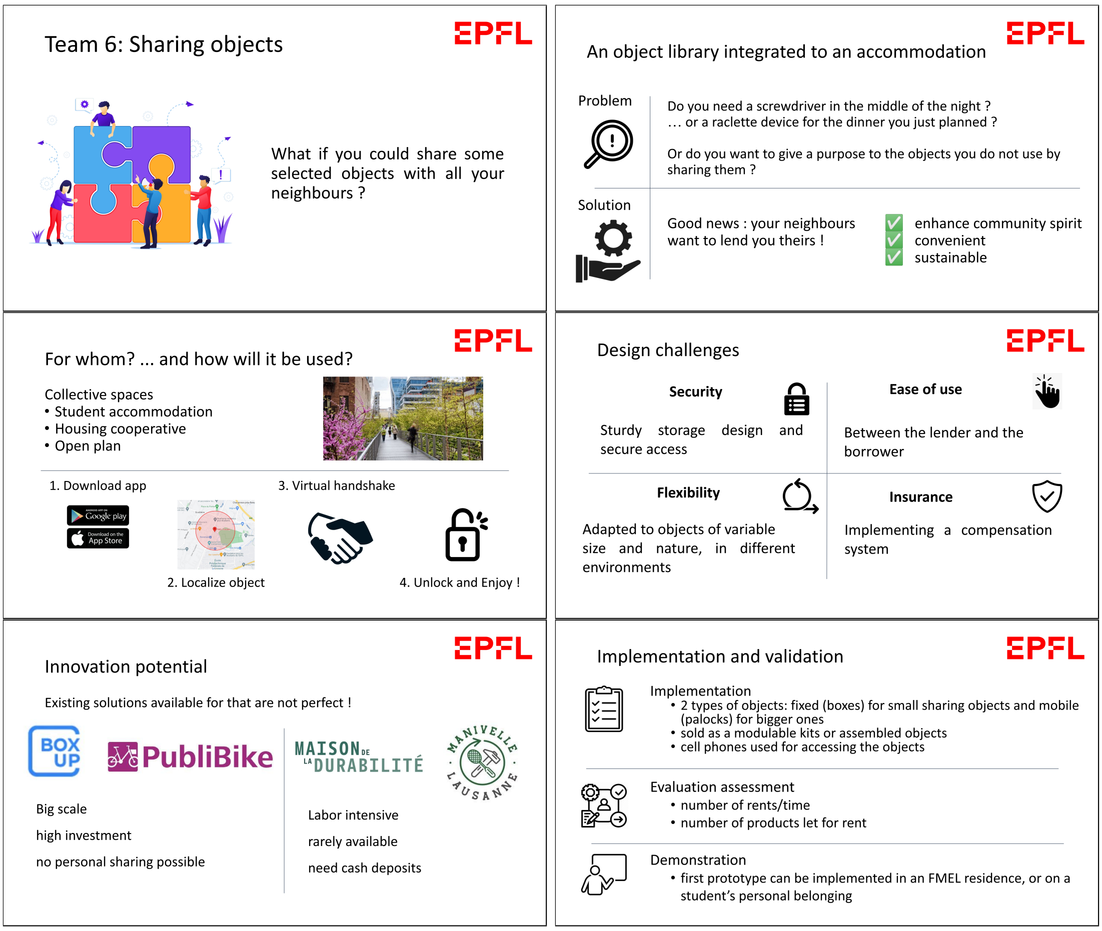


Figure 14: Slides of initial project idea

7.2 Technical drawings

Figure 15: 2 pulleys mechanism sketch

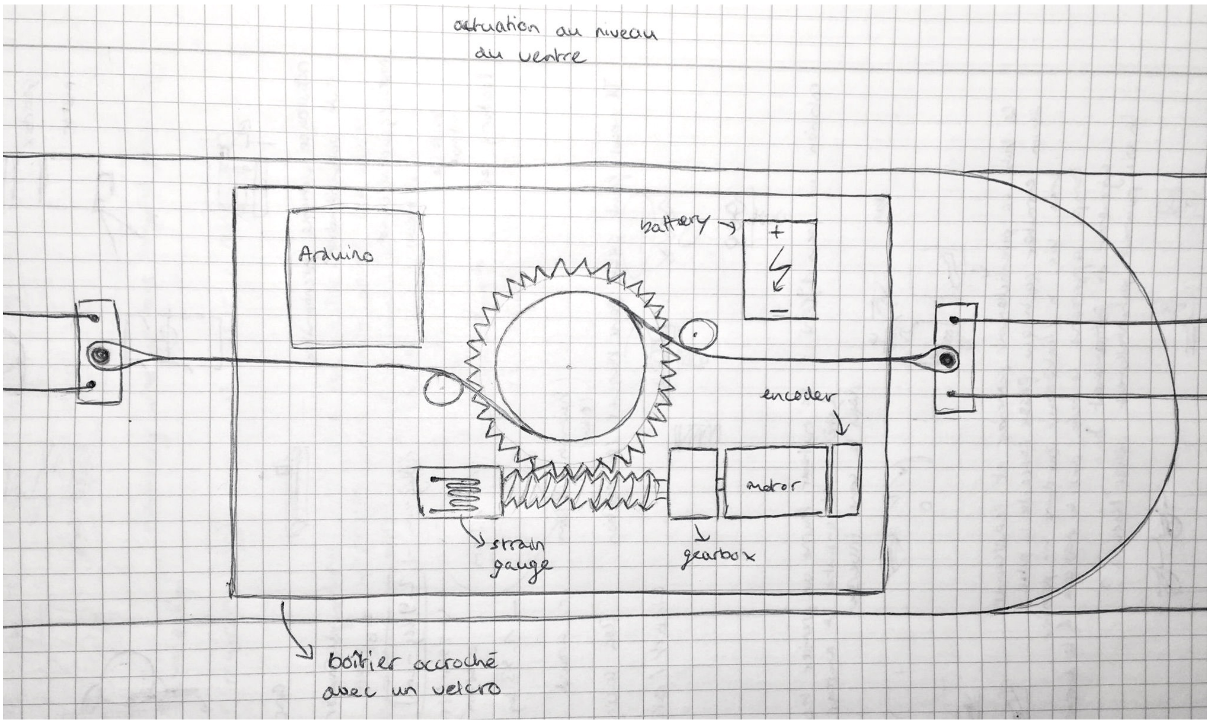


Figure 16: Alternative single pulley mechanism sketch

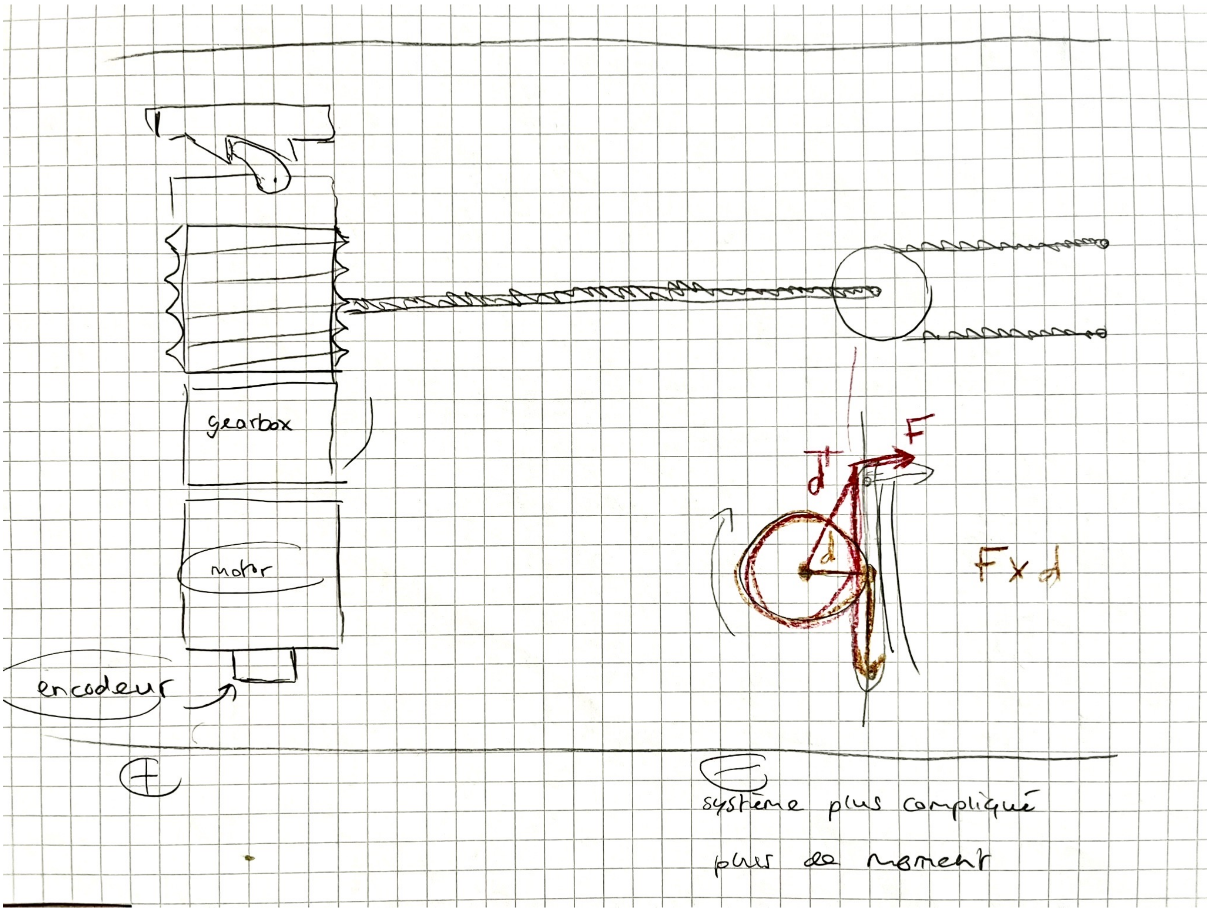


Figure 17: Alternative single pulley with pawl mechanism sketch

7.3 Circuit diagram

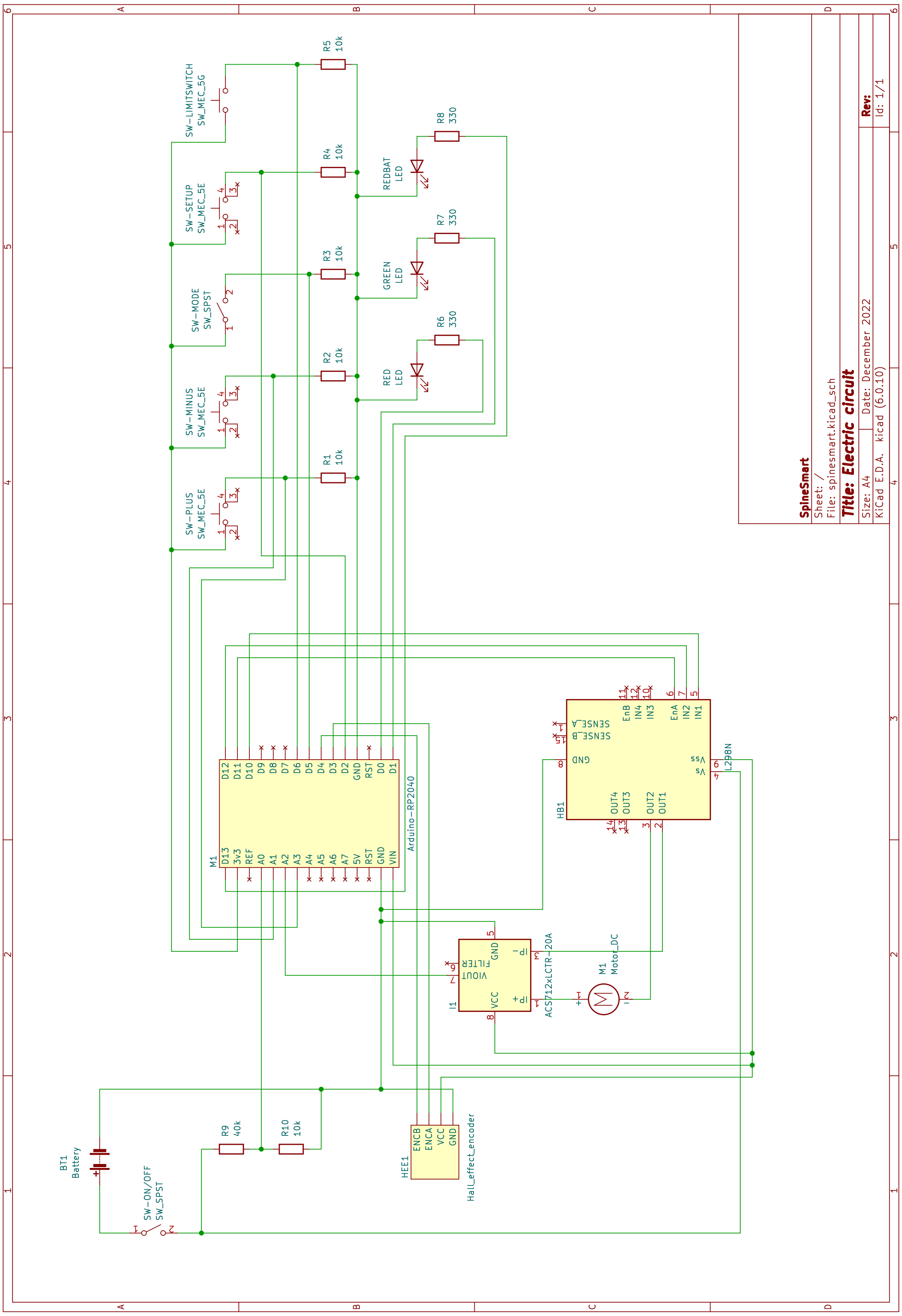


Figure 18: Circuit diagram

7.4 Block diagram

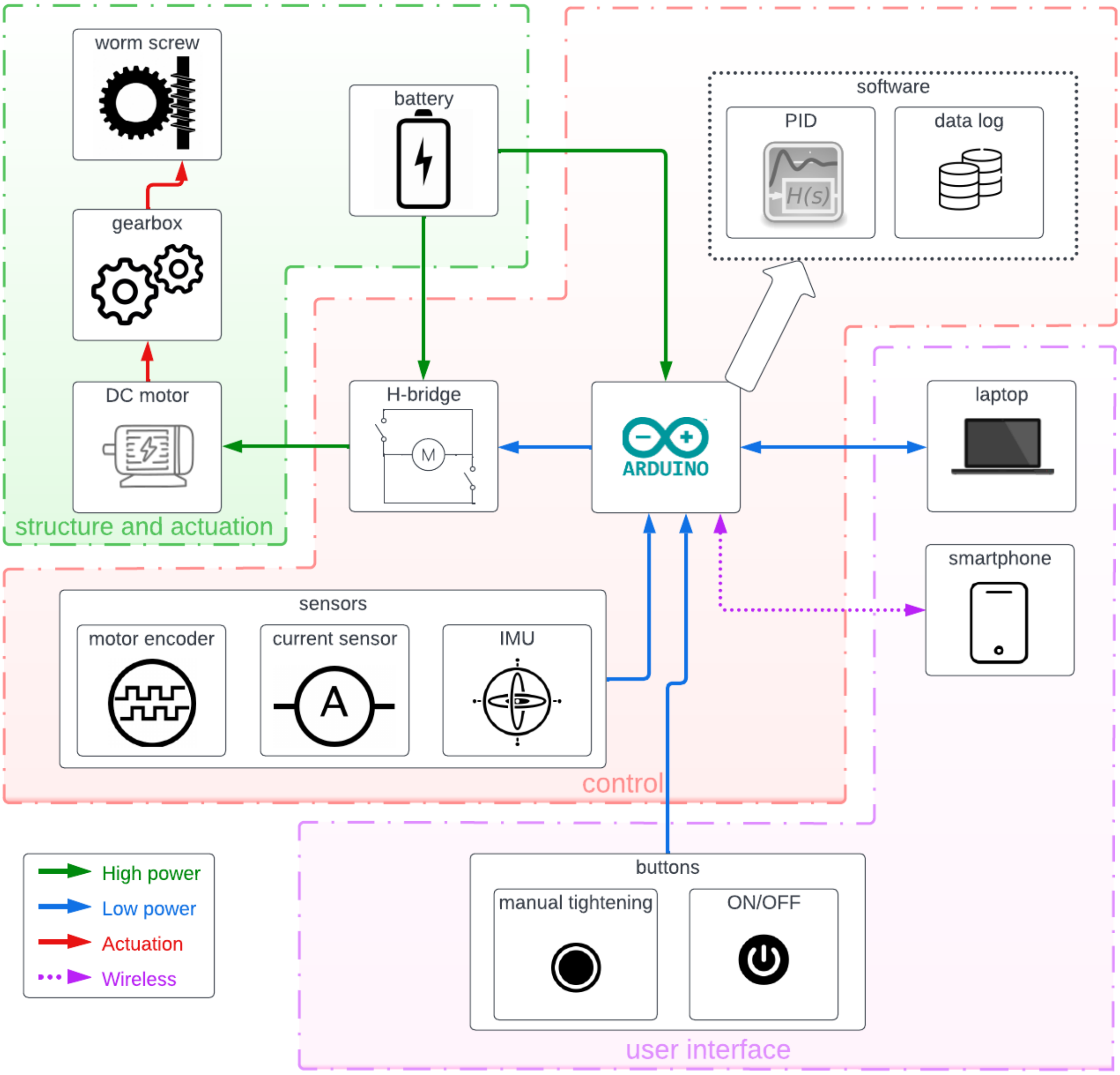


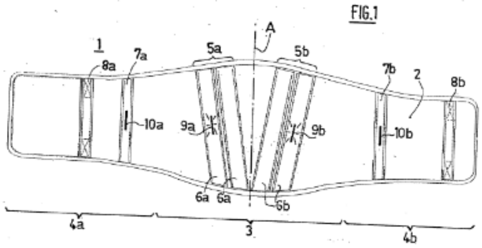
Figure 19: Block Diagram

7.5 Design requirements table

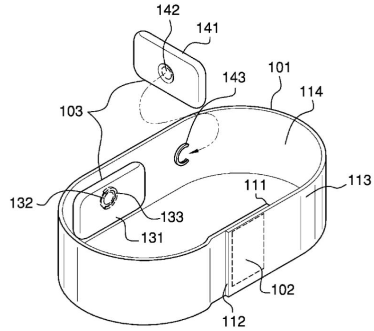
Ref	Requirement	Type	Metric	Details
1	The belt shall rebalance the pressures and loads on the intervertebral discs.	functional	depends a lot on different patients	achieved by providing an increase in intra-abdominal pressure, mainly needed for patients who have pain already while standing. "depends on the corpulence (young modulus of i.e. abdominal etc decrease),furthermore, it depends on single or double fabric thickness but the applied pressure is in the range of 1.06 kPa or 2.16 kPa respectively[4] ->depends a lot on fabrics [11]". Surprisingly, until now, no work has investigated the pressure distribution around the patient's trunk, even though this distribution is closely related to the belt design. For more detail, a renewed instrumentation and experimental protocol is necessary. This instrumentation should give detailed information on the basic physical or geometrical parameters, namely the pressure applied on the trunk, the body shape and the strain on the belt. ->only case studies [5]
1.1	The belt shall increase the lumbar spine rigidity	functional	depends a lot on different patients	"there is lumbar and joint instability, which means that one vertebra moves in relation to another enough to cause an injury to a ligament or other structures in the lumbar spine. The study showed an increase in lumbar spine rigidity in subjects who wore a belt, which is associated with better lumbar stability. " [2]
1.2	the belt has to stay in place without recurent intervention of the user	constraint	less than 2 user's intervention per working day, i.e. 6 hours of operation without moving	
1.3	The rest of the system shall be compatible with a chosen existing model of lumbar belt	functional		Choice of the team : 3M Futuro ceinture lombaire ajustable
2.	The belt shall dynamically readapt to provide good support at all time for varying situations during an average user's day	functional		
2.1	The actuation system shall have a adjustment range adapted to the user	constraint		
2.1.1	The actuation system shall apply a force on both tightening cables	constraint	0-8kg force on both sides	
2.1.2	The actuation system shall apply the required force over the total tightening travel	constraint	20 cm travel on both sides	
2.2	The actuation system shall stay in the safe range for the user	constraint		
2.2.1	the tightness of the wire shall stay bellow the acceptable maximum tightness level	constraint	max 8kg	
2.3	The belt shall not provide over-support to the back when not needed to keep the muscles and nerves stimulated	functional		
2.3.1	The belt shall untighten if no support is needed	functional		
3.	The belt shall sense some parameters of the body for actuating properly corresponding to the good situation	functional		
3.1	The belt shall recognize a certain type of movement (standing-up, wrong posture while sitting down, walking...)	functional		threshold are set based on data analysis on lots of movements done by different people
3.1.1	The belt shall sense the accelerations and rotations of trunk to categorize the movement	functional		
3.1.2	The belt shall sense the tightening on the user	functional		
4.1	The belt shall be financially accessible	constraint	300% of the price of a normal medical belt	standard price range for a medical belt in switzerland : between CHF 50.- and CHF 100.
4.2	The belt shall be electrically autonomous	constraint	8.5h of autonomy (one charge per average working day in switzerland)	
4.3	The belt shall adapt to all morphologies (can have have multiple sizes)	constraint		
5.1	The belt shall be cheap	performance	Target : 150% of the price of a normal belt	standard price range for a medical belt in switzerland : between CHF 50.- and CHF 100.-
5.2	The belt shall be comfortable to wear	performance		
5.2.1	The belt shall be lightweight	performance	Target : 1.5kg	(existing belts with battery and sensors 1.5-2kg)
5.2.2	The belt shall not tighten abruptly	performance	Response time target : 3s	
5.3	The belt shall easy to put on/off	performance		
5.3.1	The belt shall be put on in a short amount of time	performance	Target : 40s to put on the body and turned on	
5.3.2	The belt shall be put off in a short amount of time	performance	Target : 40s to turn off and put off the body	
5.4	The system shall be removable in case of malfunction even if the the belt is tightened	function		
5.5	The belt shall not be visible underneath the user's top	performance	Target : not visible under a simple sweater	
5.6	The belt shall be reusable	function		
5.6.1	The belt shall be washable	function		
5.6.2	The active system shall be separable from the belt.	function		
6	Nice-to-have functionalities	nice-to-have		
6.1	The belt shall provide data log to the caregiver or the user for diagnosis and tracability of improvements	nice-to-have		
6.2	The belt shall have additionnal pain relief modules using different technologies (heat/massaging)	nice-to-have		
6.3	The belt shall help the patient to develop the habit of maintaining a correct posture	nice-to-have		

Table 6: Engineering specifications - Spinesmart lumbar support belt

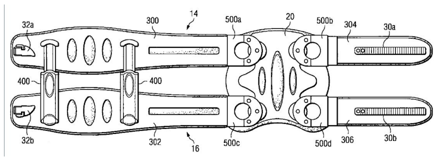
7.6 IP-analysis



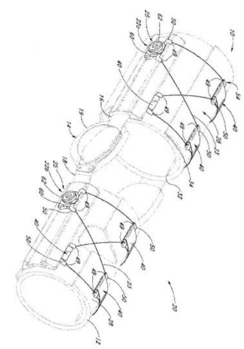
(a) US2009292230A1



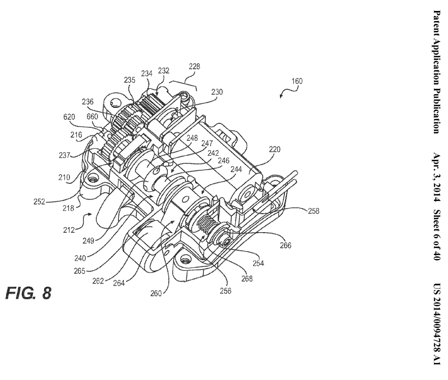
(b) US11464663B1



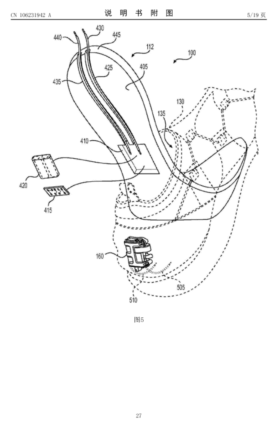
(c) US2009292230A1



(d) KR101492477B1



(e) US2009292230A1



(f) CN106231942A

Figure 20: Schematics of deposited patents

7.7 Project Management

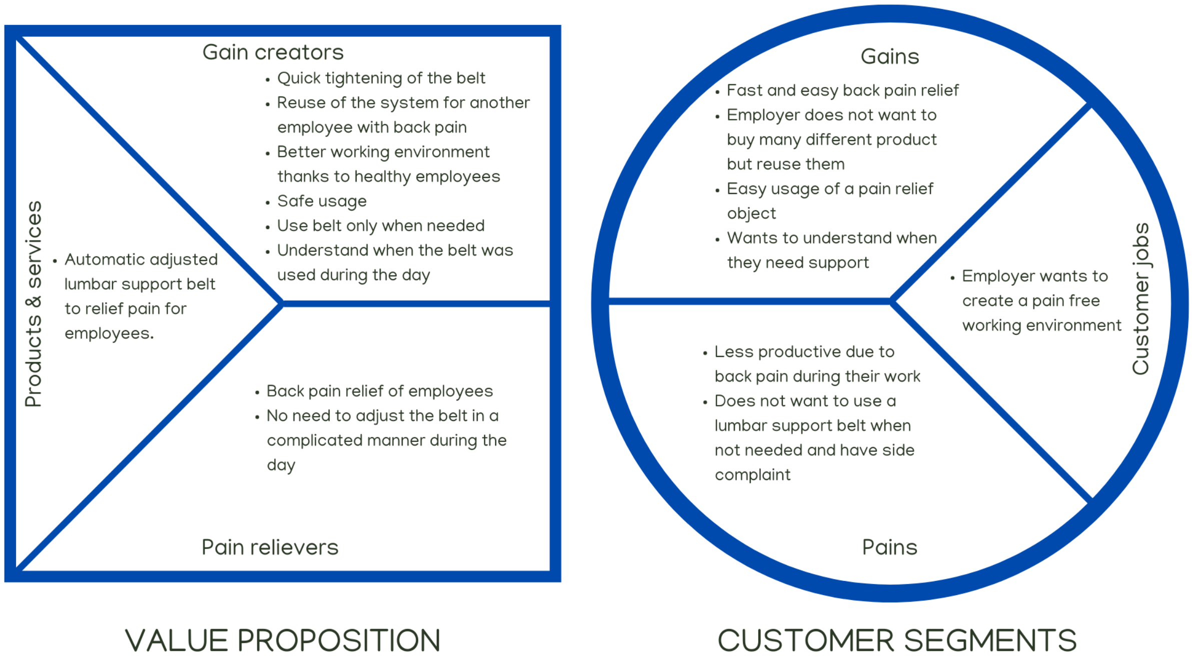


Figure 21: Value proposition

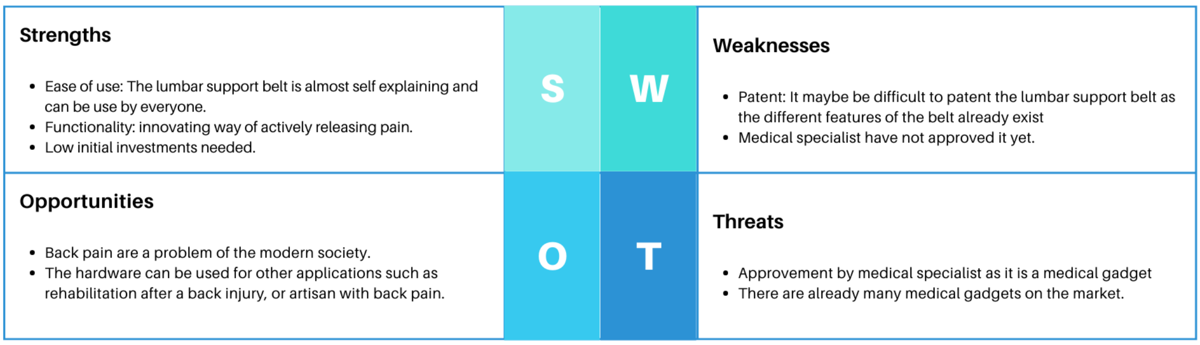


Figure 22: SWOT analysis

Figure 23: Initial Gantt chart

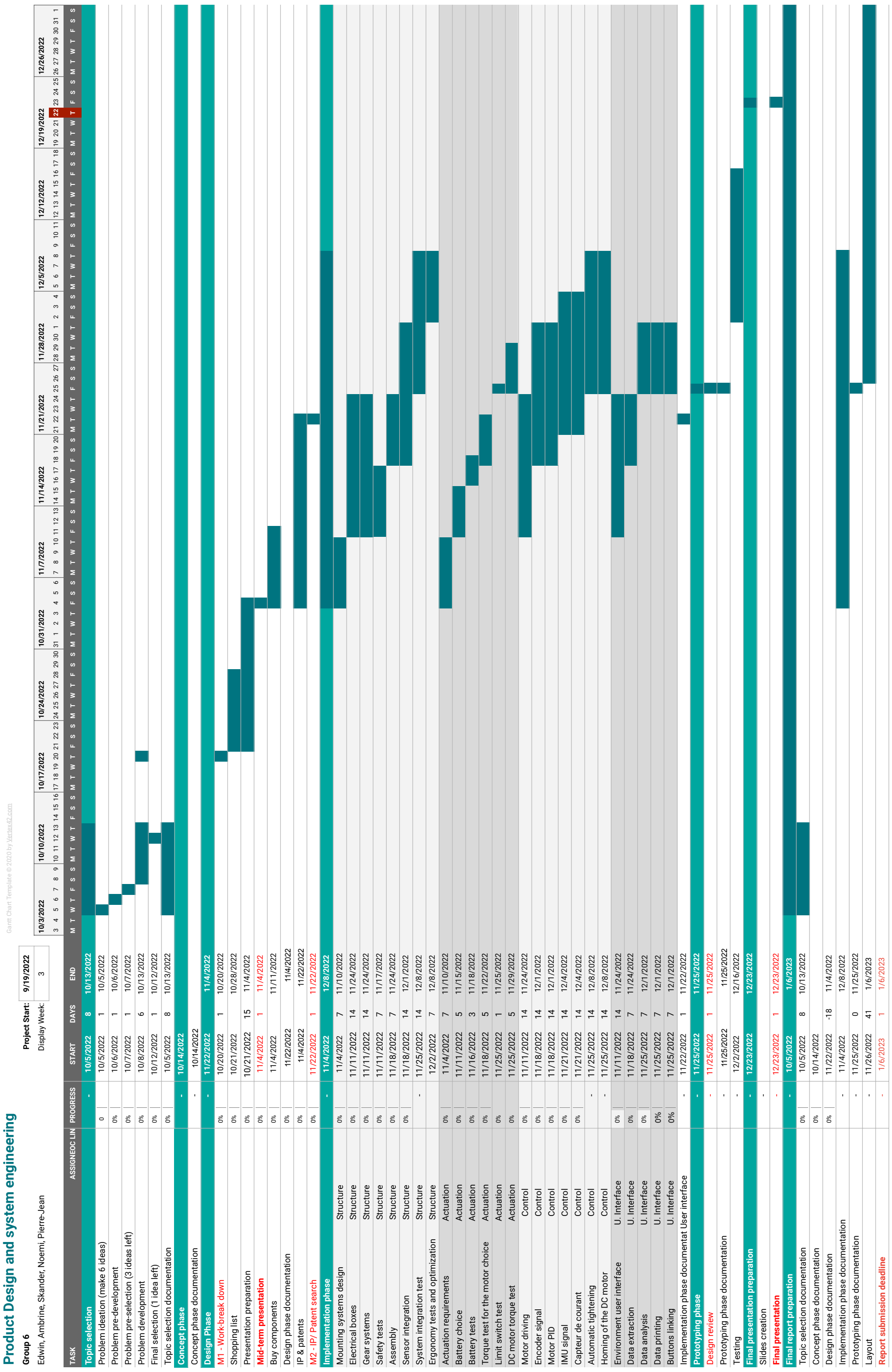
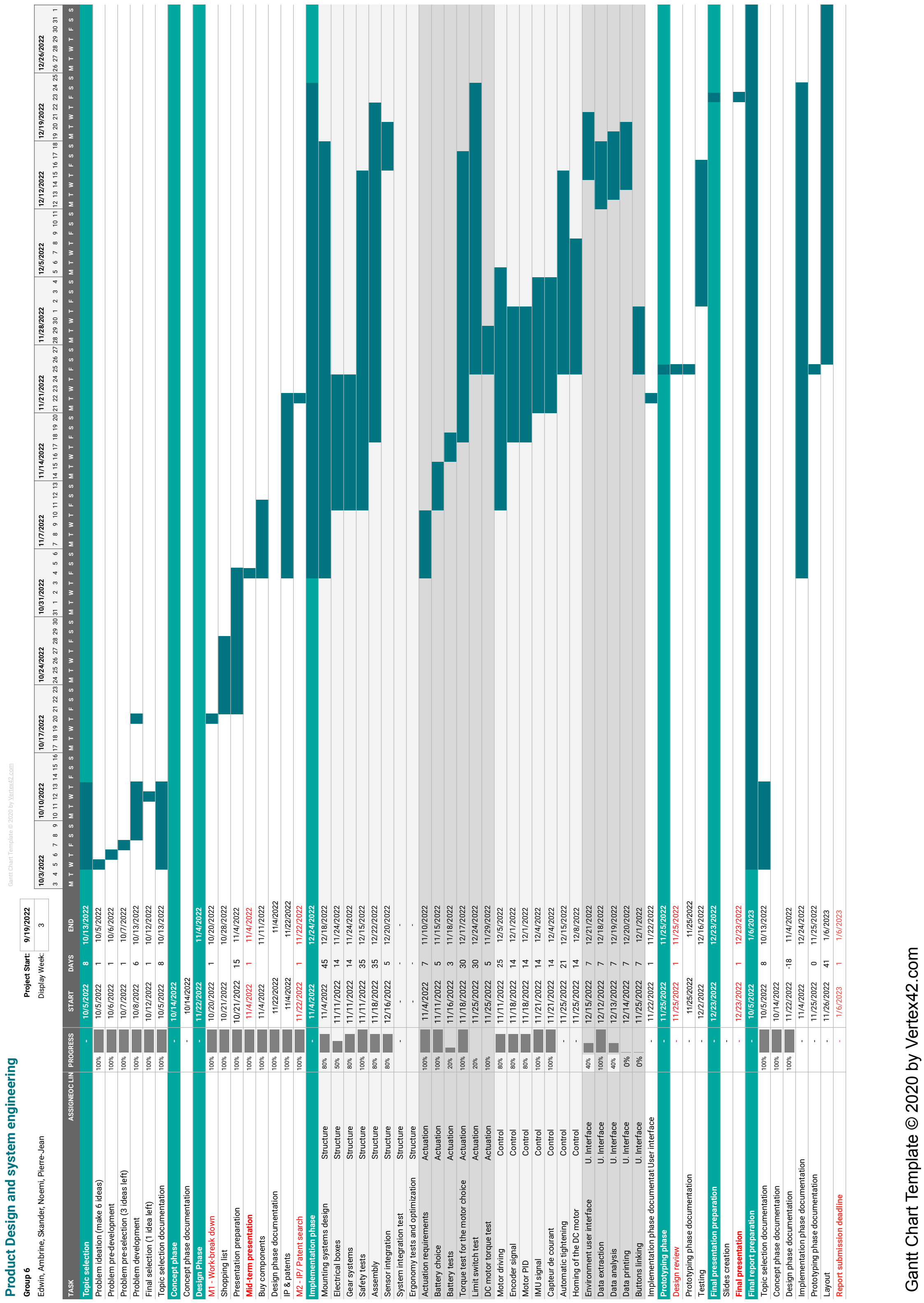


Figure 24: Updated Gantt chart



7.8 Code

Main code:

```
#include "LSM6DSOXSensor.h"
#include "lsm6dsox_activity_recognition_for_mobile.h"
#include "thingProperties.h"

// IMU DEFINITIONS
#ifdef ARDUINO_SAM_DUE
#define DEV_I2C Wire1
#elif defined(ARDUINO_ARCH_STM32)
#define DEV_I2C Wire
#elif defined(ARDUINO_ARCH_AVR)
#define DEV_I2C Wire
#else
#define DEV_I2C Wire
#endif
#define SerialPort Serial
#define INT_1 INT_IMU

// MOTOR DEFINITIONS
#define ENCA 3        // YELLOW
#define ENCB 4        // WHITE
#define MOTPOS_IN1 10 // digital
#define MOTNEG_IN2 12 // digital
#define MOTPWM 11
#define MOTCURRENT A2

// BUTTONS DEFINITIONS
#define BUTTONPLUS A3
#define BUTTONMINUS A1
#define BUTTONSETUP 2
#define BUTTONMODE 5
#define LIMITSWITCH 6

// LED DEFINITIONS
#define LEDRED 0
#define LEDGREEN 1
#define LEDBATTERY 13

// BATTERY DEFINITION
#define BATVOLTAGE A0

// Boolean to describe if the user is physically active or inactive
bool automatic_setup = 0;
bool homing = 0;
int tightening_auto_max = 0;
int tightening_auto_min = 0;

// Parameters of the motor and PID
volatile long int posi = 0;
int range_limit = -7000;
long prevT = 0;
float eprev = 0;
float eintegral = 0;
```

```
int target = 0;
float u = 0;

// LED
unsigned long last_toggle = 0;
unsigned long toggle_period = 500;
unsigned long last_change = 0;
unsigned long change_period = 1000;
bool active = 0;

// Buttons
int but_plus_state = 0;
int but_minus_state = 0;
int but_mode_state = 0;
int last_state_mode = 0;

unsigned long last_debounce_time = 0;
unsigned long debounce_delay = 50;
int limitswitch_state = 0;

// Mode
bool automatic = 0;

// Current sensor and battery voltage parameters
float VperAmp = 0.185;
float curr_Offset = 2.50;
bool current_limit = 0;
bool voltage_trigger = 0;
bool low_battery = 0;
unsigned long voltage_trigger_time = 0;

const int numReadings = 10;
int readings[numReadings]; // the readings from the analog input
int readIndex = 0;          // the index of the current reading
int total = 0;              // the running total
int average = 0;            // the average

bool stall_trigger = 0;
unsigned long stall_trigger_time = 0;
bool stall = 0;
unsigned long stall_timer = 0;

// Interrupts
volatile int mems_event = 0;

// Components
LSM6DSOXSensor AccGyr(&DEV_I2C, LSM6DSOX_I2C_ADD_L);

// MLC
ucf_line_t *ProgramPointer;
int32_t LineCounter;
int32_t TotalNumberOfLine;

void INT1Event_cb();
```

```
//-----------------SETUP-----------------//
void setup()
{

  // Initialize serial for output
  SerialPort.begin(115200);

  pinMode(LEDRED, OUTPUT);
  digitalWrite(LEDRED, HIGH);
  pinMode(LEDGREEN, OUTPUT);
  digitalWrite(LEDGREEN, HIGH);

  bool bat_volt = check_battery_low();
  while (bat_volt)
  {
    digitalWrite(LEDBATTERY, HIGH);
    bat_volt = check_battery_low();
  }

  digitalWrite(LEDBATTERY, LOW);

  // Motor and encoder initialisation
  pinMode(ENCA, INPUT);
  pinMode(ENCB, INPUT);
  attachInterrupt(digitalPinToInterrupt(ENCA), readEncoder, RISING);
  pinMode(MOTPWM, OUTPUT);
  pinMode(MOTPOS_IN1, OUTPUT);
  pinMode(MOTNEG_IN2, OUTPUT);

  // Buttons setup
  pinMode(BUTTONPLUS, INPUT);
  pinMode(BUTTONMINUS, INPUT);
  pinMode(BUTTONMODE, INPUT);
  pinMode(LIMITSWITCH, INPUT);

  // IMU initialisation
  MLCsetup();

  // Initialize current array
  for (int thisReading = 0; thisReading < numReadings; thisReading++)
  {
    readings[thisReading] = 0;
  }

  for (int i = 0; i < 10; i++)
    motCurrent();

  limitswitchSetup();

  int start_counter = 0; // press on buttonplus to continue the setup
  while (start_counter < 10000)
  {
    but_plus_state = digitalRead(BUTTONPLUS);
    if (but_plus_state == HIGH)
    {
```

```
    start_counter = start_counter + 1;
  }
  digitalWrite(LEDRED, HIGH);
  digitalWrite(LEDGREEN, HIGH);
}

delay(1000);
automatic_setting();

while (!(tightening_auto_max && tightening_auto_min))
{
  loosenFully();
  delay(5000);
  automatic_setting();
}

target = tightening_auto_max;
active = 0;
initProperties();
ArduinoCloud.begin(ArduinoIoTPreferredConnection);
setDebugMessageLevel(2);
ArduinoCloud.printDebugInfo();
led_green = digitalRead(LEDGREEN);
led_red = digitalRead(LEDRED);
delay(2000);
ArduinoCloud.update();
delay(2000);
ArduinoCloud.update();
delay(1000);
}

//-----------------LOOP------------------//
void loop()
{
  ArduinoCloud.update();

  static bool but_plus = 0;
  static bool but_minus = 0;
  static bool switch_mode = 0;

  static bool but_setup = 0;
  static int last_setup = 0;
  static unsigned long last_debounce_time = 0;
  static int setup_state = 0;

  static bool reached_pos = false;

  check_setup_button(last_setup, last_debounce_time, but_setup,
                     setup_state, reached_pos);

  check_buttons(but_plus, but_minus);
  check_mode(switch_mode);

  float curr_val = motCurrent();
  check_stall(curr_val);
```

```
check_battery();

enum state
{
  Low_battery_mode,
  Automatic_mode,
  Manual_mode,
  Tightened_setup,
  Loosened_setup
};
state control_state;

if (low_battery)
  control_state = Low_battery_mode;
else
{
  if (setup_state)
  {
    if (setup_state == 1)
    {
      control_state = Tightened_setup;
    }
    else
    {
      control_state = Loosened_setup;
    }
  }

  else
  {
    if (switch_mode)
      control_state = Automatic_mode;
    else
      control_state = Manual_mode;
  }
}

int pos = 0;
bool reached = 0;
static unsigned long ledtimestamp = millis();

if (stall)
{
  if (millis() > (ledtimestamp + 500))
  {
    if (digitalRead(LEDRED) != digitalRead(LEDGREEN))
    {
      digitalWrite(LEDRED, LOW);
      digitalWrite(LEDGREEN, LOW);
    }
    digitalWrite(LEDRED, !(digitalRead(LEDRED)));
    digitalWrite(LEDGREEN, !(digitalRead(LEDGREEN)));
    ledtimestamp = millis();
  }
```

```
}

switch (control_state)
{
case Low_battery_mode:
  digitalWrite(LEDRED, LOW);
  digitalWrite(LEDGREEN, LOW);
  digitalWrite(LEDBATTERY, HIGH);
  setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
  break;

case Automatic_mode:
  if (mems_event)
  {
    mems_event = 0;
    LSM6DSOX_MLC_Status_t status;
    AccGyr.Get_MLC_Status(&status);
    if (status.is_mlc1)
    {
      uint8_t mlc_out[8];
      AccGyr.Get_MLC_Output(mlc_out);
      differentstate(mlc_out[0]);
    }
  }

  if ((abs(target - tightening_auto_max) < 10) && !stall)
  {
    digitalWrite(LEDRED, HIGH);
    digitalWrite(LEDGREEN, LOW);
  }
  else if ((abs(target - tightening_auto_min) < 10) && !stall)
  {
    digitalWrite(LEDRED, LOW);
    digitalWrite(LEDGREEN, HIGH);
  }
  digitalWrite(LEDBATTERY, LOW);

  pos = posi;
  reached = PID(target, pos);
  if (!reached & stall)
  {
    tightening_auto_max = pos;
    target = pos;
  }
  break;

case Manual_mode:
  adjustTightening(but_plus, but_minus);

  if (!stall)
  {
    digitalWrite(LEDRED, LOW);
    digitalWrite(LEDGREEN, LOW);
  }
  digitalWrite(LEDBATTERY, LOW);
```

```
    break;

  case Tightened_setup:

    digitalWrite(LEDGREEN, LOW);
    if (millis() > (ledtimestamp + 500))
    {
      digitalWrite(LEDRED, !(digitalRead(LEDRED)));
      ledtimestamp = millis();
    }
    digitalWrite(LEDBATTERY, LOW);

    if (!reached_pos)
    {
      int pos = posi;
      reached_pos = PID(tightening_auto_max, pos);
    }
    else
    {
      adjustTighteningSetup(but_plus, but_minus, 1);
    }
    if (!active)
    {
      target = posi;
    }

    tightening_auto_max = posi;
    break;

  case Loosened_setup:

    digitalWrite(LEDRED, LOW);
    if (millis() > (ledtimestamp + 500))
    {
      digitalWrite(LEDGREEN, !(digitalRead(LEDGREEN)));
      ledtimestamp = millis();
    }
    digitalWrite(LEDBATTERY, LOW);

    if (!reached_pos)
    {
      int pos = posi;
      reached_pos = PID(tightening_auto_min, pos);
    }
    else
    {
      adjustTighteningSetup(but_plus, but_minus, 0);
    }
    if (active)
    {
      target = posi;
    }
    tightening_auto_min = posi;
    break;
```

```
  default:
    digitalWrite(LEDBATTERY, LOW);
  }

  led_green = digitalRead(LEDGREEN);
  led_red = digitalRead(LEDRED);
  tightening = float(posi) / float(range_limit) * 100;
  x = int(active);
}

//---------------FUNCTIONS---------------//
void adjustTightening(bool plus, bool minus)
{
  if (plus != minus)
  {
    if (plus && posi > -8200 && !stall)
    {
      setMotor(1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else if (minus && posi < 0)
    {
      setMotor(-1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else
      setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
  }
  else
  {
    setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
  }
}

void adjustTighteningSetup(bool plus, bool minus, bool tighten)
{
  if (tighten)
  {
    if (plus && posi > -8200 && !stall)
    {
      setMotor(1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else if (minus && posi < tightening_auto_min)
    {
      setMotor(-1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else
      setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
  }
  else
  {
    if (plus && posi > tightening_auto_max && !stall)
    {
      setMotor(1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else if (minus && posi < 0)
    {
```

```
      setMotor(-1, 255, MOTPOS_IN1, MOTNEG_IN2);
    }
    else
      setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
  }
}

void accSmooth(int dir, int pwm_target, int ms)
{
  int acc = 0;
  bool done = false;
  unsigned long timestamp = millis();
  while (!done)
  {
    if (!(millis() < timestamp + ms))
    {
      timestamp = millis();
      acc = acc + 3;
      if (acc > 255)
        acc = 255;
      setMotor(dir, acc, MOTPOS_IN1, MOTNEG_IN2);
      if (acc >= pwm_target)
        done = true;
    }
  }
}

void loosenFully()
{
  float curr = 0;
  accSmooth(-1, 150, 5);

  bool reached = 0;
  long int pos = posi;

  while (!reached)
  {
    reached = PID(0, pos);
    pos = posi;
  }
}

void INT1Event_cb()
{
  mems_event = 1;
}

void MLCsetup()
{
  uint8_t mlc_out[8];
  // Led
  pinMode(LED_BUILTIN, OUTPUT);

  // Force INT1 of LSM6DSOX low in order to enable I2C
  pinMode(INT_1, OUTPUT);
```

```
  digitalWrite(INT_1, LOW);

  delay(200);

  // Initialize I2C bus
  DEV_I2C.begin();

  AccGyr.begin();
  AccGyr.Enable_X();
  AccGyr.Enable_G();

  /* Feed the program to Machine Learning Core */
  /* Activity Recognition Default program */
  ProgramPointer = (ucf_line_t *)lsm6dsox_activity_recognition_for_mobile;
  TotalNumberOfLine = sizeof(lsm6dsox_activity_recognition_for_mobile) /
                      sizeof(ucf_line_t);
  SerialPort.println("Activity Recognition for LSM6DSOX MLC");
  SerialPort.print("UCF Number Line=");
  SerialPort.println(TotalNumberOfLine);

  for (LineCounter = 0; LineCounter < TotalNumberOfLine; LineCounter++)
  {
    if (AccGyr.Write_Reg(ProgramPointer[LineCounter].address,
                         ProgramPointer[LineCounter].data))
    {
      SerialPort.print("Error loading the Program to LSM6DSOX at line:");
      SerialPort.println(LineCounter);
      while (1)
      {
        // Led blinking
        digitalWrite(LED_BUILTIN, HIGH);
        delay(250);
        digitalWrite(LED_BUILTIN, LOW);
        delay(250);
      }
    }
  }

  SerialPort.println("Program loaded inside the LSM6DSOX MLC");

  // Interrupt
  pinMode(INT_1, INPUT);
  attachInterrupt(INT_1, INT1Event_cb, RISING);

  // Need to wait for a time window before having the first MLC status
  delay(3000);

  AccGyr.Get_MLC_Output(mlc_out);
  differentstate(mlc_out[0]);
}

void differentstate(uint8_t status)
{
  switch (status)
```

```
  {
  case 0:
    active = false;
    target = tightening_auto_max;
    control_state_gui = "stationary";
    break;

  case 1:
    active = true;
    target = tightening_auto_min;
    control_state_gui = "walking";
    break;

  case 4:
    // jogging
    active = true;
    target = tightening_auto_min;
    control_state_gui = "running";
    break;

  case 12:
    active = false;
    target = tightening_auto_max;
    control_state_gui = "stationary";
    break;
  }
}

void readEncoder()
{
  int b = digitalRead(ENCB);
  if (b > 0)
  {
    posi++;
  }
  else
  {
    posi--;
  }
}

void setMotor(int dir, int pwr, int IN1, int IN2)
{
  analogWrite(MOTPWM, pwr);
  if (dir == 1)
  {
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
  }
  else if (dir == -1)
  {
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
  }
```

```
}

bool PID(int target, int pos)
{
  // PID constants
  float kp = 3;
  float kd = 0.8;
  float ki = 0.0;

  // Error
  int e = pos - target;

  // Time
  long currT = micros();
  float deltaT = ((float)(currT - prevT)) / (1.0e6);
  prevT = currT;

  // Derivative
  float dedt = (e - eprev) / (deltaT);

  // Integral
  eintegral = eintegral + e * deltaT;

  // Control signal
  float u = kp * e + kd * dedt + ki * eintegral;

  // Motor power
  float pwr = fabs(u);
  if (pwr > 255)
  {
    pwr = 255;
  }

  // Motor direction
  int dir = 1;
  if (u < 0)
  {
    dir = -1;
  }

  if (abs(e) > 10)
  {
    setMotor(dir, pwr, MOTPOS_IN1, MOTNEG_IN2);
    return 0;
  }
  else
  {
    setMotor(dir, 0, MOTPOS_IN1, MOTNEG_IN2);
    return 1;
  }

  // Store previous error
  eprev = e;
}
```

```
float motCurrent()
{
  delay(1);
  total = total - readings[readIndex];
  analogReadResolution(12);
  readings[readIndex] = analogRead(MOTCURRENT);
  total = total + readings[readIndex];
  readIndex = readIndex + 1;
  if (readIndex >= numReadings)
  {
    readIndex = 0;
  }
  average = total / numReadings;
  return (5 / 3.3) * (((average / 4095.0) * 3.3) - curr_Offset) / VperAmp;
  // 3.3V analog => max 6.4A
}

void limitswitchSetup()
{
  int pwm_setup = 150;
  int high_counter = 0;
  int low_counter = 0;

  digitalWrite(MOTPOS_IN1, LOW);
  digitalWrite(MOTNEG_IN2, HIGH);
  analogWrite(MOTPWM, pwm_setup);
  while (high_counter < 1000)
  {
    limitswitch_state = digitalRead(LIMITSWITCH);
    if (limitswitch_state == HIGH)
    {
      high_counter = high_counter + 1;
    }
  }

  digitalWrite(MOTPOS_IN1, HIGH);
  digitalWrite(MOTNEG_IN2, LOW);
  while (low_counter < 1000)
  {
    limitswitch_state = digitalRead(LIMITSWITCH);
    if (limitswitch_state == LOW)
    {
      low_counter = low_counter + 1;
    }
  }

  delay(1000);
  analogWrite(MOTPWM, 0);
  delay(200);
  posi = 0;

  if (digitalRead(LEDRED))
  {
    digitalWrite(LEDRED, !digitalRead(LEDRED));
  }
```

```
  if (digitalRead(LEDGREEN))
  {
    digitalWrite(LEDGREEN, !digitalRead(LEDGREEN));
  }
}

void automatic_setting()
{
  bool reached_max = 0;
  bool reached_min = 0;
  float curr = 0;

  accSmooth(1, 255, 5);

  while ((!reached_max && posi > range_limit))
  {
    curr = motCurrent();
    if (curr > 0.8 && !(reached_min))
    {
      tightening_auto_min = posi;
      reached_min = 1;
    }
    if (curr > 1.7)
    {
      reached_max = 1;
      tightening_auto_max = posi;
    }
  }
  setMotor(1, 0, MOTPOS_IN1, MOTNEG_IN2);
}

void check_battery()
{
  float average_volt = 0;
  float voltage_value = 0;
  for (int i = 0; i < 10; i++)
  {
    analogReadResolution(12);
    average_volt = average_volt + analogRead(BATVOLTAGE);
  }
  average_volt = average_volt / 10;
  voltage_value = (average_volt / 4095.0) * 3.3 * 5 + 0.1;
  //[V]+0.1V voltage divider with five 10k (3.3V analog => max 16.6V)

  float voltage_threshold = 15;
  if (voltage_value > voltage_threshold)
  {
    if (voltage_trigger == 1)
    {
      voltage_trigger = 0;
      voltage_trigger_time = 0;
    }
    return;
  }
  else
```

```
  {
    if (voltage_trigger == 0)
    {
      voltage_trigger = 1;
      voltage_trigger_time = millis();
      return;
    }
    else
    {
      unsigned long actual_time = millis();
      unsigned long delta_trigger = actual_time - voltage_trigger_time;
      if (delta_trigger > 1000)
      {
        low_battery = 1;
        digitalWrite(LED_BUILTIN, HIGH);
        voltage_trigger = 0;
        return;
      }
      else
      {
        return;
      }
    }
  }
}

bool check_battery_low()
{
  float average_volt = 0;
  float voltage_value = 0;
  for (int i = 0; i < 10; i++)
  {
    analogReadResolution(12);
    average_volt = average_volt + analogRead(BATVOLTAGE);
  }
  average_volt = average_volt / 10;
  voltage_value = (average_volt / 4095.0) * 3.3 * 5 + 0.1;
  //+0.1V voltage divider with five 10k (3.3V analog => max 16.6V)

  if (voltage_value < 15)
  {
    return 1;
  }
  else
  {
    return 0;
  }
}

void check_mode(bool &but_mode_state)
{
  but_mode_state = digitalRead(BUTTONMODE);
}

void check_buttons(bool &but1, bool &but2)
```

```
{
  bool but_plus_state = digitalRead(BUTTONPLUS);
  bool but_minus_state = digitalRead(BUTTONMINUS);
  if (but_plus_state == 1 and but_minus_state == 0)
  {
    but1 = 1;
    but2 = 0;
  }
  else if (but_plus_state == 0 and but_minus_state == 1)
  {
    but1 = 0;
    but2 = 1;
  }
  else
  {
    but1 = 0;
    but2 = 0;
  }
}

void check_stall(float current)
{
  float stall_current = 1.5;
  if (stall == 0)
  {
    if (current < stall_current)
    {
      if (stall_trigger == 1)
      {
        stall_trigger = 0;
        stall_trigger_time = 0;
      }
      return;
    }
    else
    {
      if (stall_trigger == 0)
      {
        stall_trigger = 1;
        stall_trigger_time = millis();
        return;
      }
      else
      {
        unsigned long actual_time = millis();
        unsigned long delta_trigger = actual_time - stall_trigger_time;

        if (delta_trigger > 1000)
        {
          stall = 1;
          digitalWrite(LED_BUILTIN, HIGH);
          stall_trigger = 0;
          stall_timer = millis();
          return;
        }
```

```
        else
        {
          return;
        }
      }
    }
  }
  else
  {
    unsigned long actual_time = millis();
    unsigned long time_diff = actual_time - stall_timer;
    if (time_diff < 5000)
    {
      return;
    }
    else
    {
      stall = 0;
      digitalWrite(LED_BUILTIN, LOW);
      return;
    }
  }
}

void check_setup_button(int &last_setup,
                        unsigned long &last_debounce_time,
                        bool &but_setup, int &setup_state,
                        bool &reached_pos)
{
  static unsigned long debounce_delay = 50;
  int reading = digitalRead(BUTTONSETUP);
  if (reading != last_setup)
  {
    last_debounce_time = millis();
  }

  if ((millis() - last_debounce_time) > debounce_delay)
  {
    if (reading != but_setup)
    {
      but_setup = reading;
      if (but_setup == HIGH)
      {
        setup_state = setup_state + 1;
        if (setup_state == 3)
          setup_state = 0;
        reached_pos = false;
      }
    }
  }
  last_setup = reading;
}
```

7.9 Bill of materials

Item	Supplier	Quantity	Total price
Lumbar Belt 3M Futuro	Digitech	1	CHF46,60
Wormscrew Gearwheel Z 1:60, module 0.75	Conrad	1	CHF22,95
Platine SBC-MotoDriver	Conrad	1	CHF7,30
Climbing rope 2 mm - 1m	Decathlon	1	CHF0,70
Set of 2 quick release buckles for backpack - 20mm	Decathlon	1	CHF4,00
4.4:1 Metal Gearmotor MP 12V with 48 CPR Encoder	Pololu	1	CHF39,96
slide switch	Distrelec	1	CHF2,86
Arduino RP2040	Distrelec	1	CHF33,28
Rocker Switch	SPOT DLL	1	CHF0,11
18650 Battery cells, 3.7V	SPOT DLL	4	CHF18,00
LED 5mm (T13/4)	SPOT DLL	3	CHF0,42
10k resistors	SPOT DLL	10	CHF0,03
push buttons	SPOT DLL	3	CHF0,88
Hall current sensor module ACS712	SPOT DLL	1	CHF1,19
Wires and connectors	SPOT DLL	1	CHF2,00
Vero board 80x40mm	SPOT DLL	1	CHF3,00
Hardware and fasteners	SPOT DLL	30g	CHF1,50
PETG 3D printed parts	SPOT DLL	200g	CHF6,00
TOTAL			**CHF190,78**

Table 7: Bill of materials - ergonomic prototype

References

[1] What is the 3D Printing vs Injection Molding Cost-per-Unit Breakeven? URL https://www.xometry.com/resources/injection-molding/injection-molding-vs-3d-printing/.

[2] To Wear or Not to Wear a Lumbar Support Belt? Researchers Investigate the Psychological and Biomechanical Effects. URL https://www.irsst.qc.ca/en/headlines/id/682/to-wear-or-not-to-wear-a-lumbar-support-belt-researchers-investigate-.

[3] X-NUCLEO-IKS01A3, July 2022. URL https://github.com/stm32duino/X-NUCLEO-IKS01A3. original-date: 2019-02-26T16:28:26Z.

[4] R. Bonnaire, W.-S. Han, P. Calmels, R. Convert, and J. Molimard. Parametric Study of Lumbar Belts in the Case of Low Back Pain: Effect of Patients' Specific Characteristics. In M. P. Nash, P. M. Nielsen, A. Wittek, K. Miller, and G. R. Joldes, editors, *Computational Biomechanics for Medicine*, pages 43–59, Cham, 2020. Springer International Publishing. ISBN 9783030159238. doi: 10.1007/978-3-030-15923-8_4.

[5] R. Bonnaire, W. S. Han, R. Convert, P. Calmels, and J. Molimard. Feasibility of a Full-Field Measurements-Based Protocol for the Biomechanical Study of a Lumbar Belt: A Case Study. *Biomechanics*, 2(2):174–188, June 2022. ISSN 2673-7078. doi: 10.3390/biomechanics2020015. URL https://www.mdpi.com/2673-7078/2/2/15.

[6] S. Eidgenossenschaft. L'avenir du marché de la santé, facteur de coûts et opportunité de croissance - confédération suisse (département fédéral de l'intérieur). https://www.newsd.admin.ch/newsd/message/attachments/7313.pdf", 2022.

[7] S. Eidgenossenschaft. Enquête suisse sur la population active au 1er trimestre 2022: l'offre de travail. https://www.bfs.admin.ch/bfs/fr/home/statistiques/travail-remuneration.assetdetail.22604462.html", 2022.

[8] Futuro. Ceinture lombaire ajustable. https://www.3msuisse.ch/3M/fr_CH/futuro-eu/products/~/FUTURO-Ceinture-Lombaire-46820-Ajustable/?N=4318+8711744+3289888909+3294529188&rt=rud.

[9] healthy livin' solution. LED Heated Back Brace — Belt Support Lumbar Traction Relief. https://www.healthylivin.solutions/products/led-heated-massager-back-brace-belt-support-lumbar-traction-relief?_pos=3&_sid=2275a8ff2&_ss=r, 2022. [Online; accessed 20-November-2022].

[10] H. livin' solution. Intelligent posture correction — Belt Support Lumbar Traction Relief. https://www.healthylivin.solutions/products/intelligent-posture-corrector-smart-posture-trainer?, 2022. [Online; accessed 20-November-2022].

[11] J. Molimard, R. Bonnaire, W. S. Han, R. Convert, and P. Calmels. In-silico pre-clinical trials are made possible by a new simple and comprehensive lumbar

belt mechanical model based on the Law of Laplace including support deformation and adhesion effects. *PLOS ONE*, 14(3):e0212681, Mar. 2019. ISSN 1932-6203. doi: 10.1371/journal.pone.0212681. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0212681.

[12] Nike. Nike adapt. https://www.nike.com/adapt, 2022. [Online; accessed 20-November-2022].

[13] E. P. Office. Us2004097857a1 brace with integrated lumbar support system. https://worldwide.espacenet.com/patent/search/family/032303481/publication/US2004097857A1?q=pn$%$3DUS2004097857A1, 2004.

[14] E. P. Office. Us2009292230a1 lumber support belt. https://worldwide.espacenet.com/patent/search/family/036933459/publication/US2009292230A1?q=pn$%$3DUS2009292230A1", 2009.

[15] E. P. Office. Us2014094728a1 motorized tensioning system for medical braces and devices. https://worldwide.espacenet.com/patent/search/family/049488641/publication/US2014094728A1?q=pn%3DUS2014094728A1, 2014.

[16] E. P. Office. Kr101492477b1 closure system for braces, protective wear and similar articles. https://worldwide.espacenet.com/patent/search/family/038948674/publication/KR101492477B1?q=pn$%$3DKR101492477B1, 2015.

[17] E. P. Office. Cn106231942a footwear having motorized adjustment system and removable midsole. https://worldwide.espacenet.com/patent/search/family/053002816/publication/CN106231942A?q=pn<$%$3DCN106231942A, 2016.

[18] E. P. Office. Us11464663b1 lumbar therapy belt. https://worldwide.espacenet.com/patent/search/family/036933459/publication/US2009292230A1?q=pn$%$3DUS2009292230A1", 2019.

[19] W. I. portal. Class 10: Surgical, medical, dental and veterinary apparatus and instruments; artificial limbs, ... https://www.wipo.int/classifications/nice/nclpub/en/fr/pdf-download.pdf?lang=en&tab=&viewMode=flat&dateInForce=20200101&classNumber=10, 2020.

[20] A. M. Products. Effect of lumbar bracing on spinal function. https://clinicaltrials.gov/ct2/show/NCT02251106, 2017.

[21] A. Rodriguez, J. R. Rabuñal, A. Pazos, A. Rodríguez Sotillo, and N. Ezquerra. Wearable postural control system for low back pain therapy. *IEEE Transactions on Instrumentation and Measurement*, 70:1–10, 2021. doi: 10.1109/TIM.2021.3057935.

[22] R. Schweiz. Rapport sur le dos suisse 2020. https://www.rheumaliga.ch/assets/doc/CH_Dokumente/medien/grosser-rueckenreport-2020/Rapport-sur-le-dos-2020.pdf", 2020.

[23] A. Seegers. Impact du port de ceinture de soutien lombaire de série sur la proprioception du rachis. https://www.google.com/url?q=https://aurore.unilim.fr/theses/nxfile/default/85a57bf1-090d-4bd5-90e5-6a5c090fbfe1/blobholder:0/P20173317.pdf&sa=D&source=docs&ust=1670593407575563&usg=AOvVaw2QqFKTMA_DBIA5R8fQAIP8, 2017.

[24] swissmedic. Regulation of medical devices. https://www.swissmedic.ch/swissmedic/en/home/medical-devices/regulation-of-medical-devices.html, 2022.