Bachelor Thesis

Creative Technology

The development of an interactive mountain bike simulator to help injured mountain riders reduce the fear of re-injury while cornering

Jinze Wang

Supervisor: Max Slutter

Critical observer: Mannes Poel

University of Twente

Drienerlolaan 5

7522 NB Enschede

Netherlands

07-07-2023

Acknowledgments

I would like to express my deepest gratitude to Max and Mannes for their invaluable assistance throughout this project. Their guidance, expertise, and support have been instrumental in completing this research. Writing has been a personal challenge as a practical person focused on design and building. Their patience, encouragement, and help in refining my writing skills have been immensely valuable, and I have learned much from their expertise in this area.

I would also like to extend my heartfelt thanks to my friends from the mountain biking community, who generously contributed their time and support. Their participation in the user research and testing was crucial in gathering valuable insights and feedback. This project would not have been possible without their willingness to collaborate and share their experiences.

Thank you all for your dedication, assistance, and friendship. Your contributions have made a lasting impact, and I am sincerely grateful for your involvement in this project.

Abstract

The fear of re-injury can highly influence the athletes' performance and lead to quitting the sport. This project examines the effectiveness of a mountain bike simulator in reducing the fear of reinjury among mountain bike riders, specifically focusing on the fear while cornering.

This study aims to develop a mountain bike simulator that combines interactive technology with scene exposure and education to help injured mountain bikers improve their skills and reduce the fear of re-injury.

The study employed user testing and gathered feedback from injured mountain bike riders to assess the mountain bike simulator's training value and fear reduction effectiveness. Key findings reveal that participants valued the simulator's training capabilities, especially for cornering skills and speed control. While immediate fear reduction was not evident, the simulator showed promise in enhancing training motivation. This project contributes to the growing understanding of fear reduction in mountain biking and paves the way for further advancements in the field.

Keywords: Interactive technology, mountain bike simulator, fear reduction, cornering skills, training motivation, user testing.

Table of content

Chapter 1 Introduction

1.1 Background

Mountain biking is a popular sport involving specially designed bicycles on off-road terrains, it is enjoyable but challenging, and the varying terrain conditions pose risks to mountain bike riders. The high injury rate in mountain biking is close to 43 injuries per 1,000 hours of riding [1]. Moreover, many mountain bike riders experience the fear of re-injury after the accidents [2], which can heavily impact their confidence and rehabilitation results [3], and even lead to being unable to return to sports [4]. Thus, overcoming the fear of re-injury is essential for practical sports training and rehabilitation for injured athletes.

1.2 Problem and Goal

For mountain biking, cornering skills are widely recognized as critical but pose a significant risk for accidents. A study revealed that 19.4% of injuries occurred during cornering maneuvers [5]. Crashing while negotiating turns is a prevalent type of accident, and the severity of injuries depends on terrain and speed. Thus, mastering cornering skills through training is essential for safety and accident prevention.

In recent years, interactive technology and gamification have been increasingly used in sports training to enhance quality and effectiveness. Cases such as virtual coaching and virtual reality have shown that implementing those technologies can improve training motivation for athletes [6, 7, 8, 9, 10, 11,12].

However, there needs to be more dedicated training equipment specifically designed for mountain bike cornering skill training and fear reduction. Therefore, to mitigate risks and prevent reinjury, it is crucial to provide mountain bikers with a secure training environment to enhance their skills and mental strength. There is a need to develop a training system explicitly tailored for cornering skills and reduce the fear of re-injury.

1.3 Research Questions

The goal of this graduation project is outlined above. The following primary research question and accompanying sub-questions have been formulated below.

Primary research question

How can an interactive mountain bike simulator help injured mountain bike riders reduce the fear of re-injury while cornering?

The following questions will be answered

- What kind of injuries are mountain bikers going through?
- What kind of reason causes the injury?
- What kind of factors contribute to the fear of re-injury?
- Why does fear of re-injury exert such a substantial influence on athletes?
- What kind of solution is suitable for reducing the fear of getting re-injury?

Chapter 2- Background Research

2.1 Literature Review

The primary goal is to tackle the fear of reinjury that plagues mountain bike riders who have suffered injuries during cornering. It is essential to fully grasp the specific circumstances that led to their injuries and subsequent fear to accomplish this. By doing so, we can gain valuable insights into their fear and develop strategies to alleviate it effectively. Additionally, a thorough understanding of mountain bike cornering is critical to improving rider technique and reducing fear.

2.1.1 Injuries in mountain biking

Sports injuries are inevitable and can cause physical and psychological trauma. Physical injuries, such as back pain and knee pain, are the most common injuries in cycling, according to studies from Kotler [13] and Clarsen [14]. Both traumatic injuries (e.g., scrapes, lacerations, contusions, fractures, or dislocations) and nontraumatic injuries (e.g., overuse and degenerative injuries) can occur while mountain biking. Recreational mountain bike riders are more prone to non-traumatic injuries. In contrast, high-speed riders or those navigating technical terrain, traffic, or large groups may be more prone to traumatic injuries. Chronic injuries, such as overuse injuries, can cause severe functional impairment and require medical attention. Acute injuries are more likely to attract attention in accidents. For instance, a study by Chow [15] found that 90% of surveyed mountain bike riders who crashed on their bikes had a limb injury. Schwellnus and Derman [16] also reported similar findings that acute injuries account for most injuries in bike accidents.

According to a study carried out by the Enduro World Series [17], it was discovered that 40.7% of the 1,940 Enduro riders experienced a notable injury lasting more than a month. The report further revealed that shoulder/clavicle injuries accounted for 25.6% of all significant injuries, followed by wrist injuries (9.6%), knee injuries (7.1%), and head injuries (7.1%). Notably, concussions constituted the third most prevalent diagnosis, reported by 4% of all riders.

2.1.2 How did riders get injured?

With its inherent thrill and excitement, mountain biking often entices riders to test the boundaries of their skills and abilities. However, it is essential to acknowledge that mountain biking also involves inherent risks. Regardless of one's level of preparedness or cautious riding, hazards such as rocks, trees, boulders, bike selection, and even fellow riders pose significant dangers during the sport.

Mountain bike accidents can occur due to various factors. Rider errors, Poor trail conditions, and lack of protection gears cause mountain biking injuries. However, rider error is the most common cause of mountain bike accidents, including misjudging speed or distance, taking risky maneuvers, improper braking, losing control on descents, or not maintaining proper body position while riding.

A study from Ehn et al. [5] shows that between 2018 and 2020, injury events during mountain biking were analyzed, 52.1% (873 cases) of injuries happened during **downhill riding**, 31.8% (534 cases) on **flat terrain,** and 6.4% (108 cases) while **riding uphill,** on the **technical nature of the trail** (19.7% or 331 cases), **negotiating a turn** (19.4% or 325 cases). Regarding trail familiarity, 74.8% (1254 cases) of injuries occurred on trails riders need to become more **familiar with**. Several other factors commonly associated with injury events, including **inexperience** (22.5% or 378 cases), were reported. Notably, 27.1% (454 cases) of injury events had no particular cause.

2.1.3 Fear of reinjury

Injured athletes commonly experience fear of re-injury when returning to sport. Moreover, fear of re-injury is a common psychological factor influencing return to sport (RTS) outcomes [18]. Physical injuries can reduce an athlete's performance, and therefore an effective rehabilitation program should include physical and psychological interventions for most athletes to recover [19] entirely. Besides physical damage, psychological issues such as loss of confidence and fear of reinjury are also critical. During rehabilitation, athletes' anxiety levels will likely ebb and flow as they experience setbacks and achieve milestones. The greatest fear is expected in the acute stages and may reduce as rehabilitation progresses. However, as athletes move towards RTS, they will likely experience increased fear of re-injury, with the trend shown in Figure 2.1.

Figure 2.1: Fear over time across the rehabilitation stage[24].

2.1.4 Solutions to reduce the fear

As mentioned above in cyclists' challenges, the fear of injury significantly negatively affects rehabilitation outcomes, including preventing a triumphant return to sport [20] and poor performance [21]. Many treatments can be applied to the rehabilitation program. Almost every rehabilitation training will require Participants to have self-motivation, also known as self-efficacy, which is confidence in one's ability to perform a set of actions. Such self-motivation is essential because the greater a person's confidence, the more likely they will initiate and continue training activities that will result in a positive outcome in terms of recovery[22].

Table 2.3 presents a range of interventions aimed at reducing fear among individuals, and these interventions have shown promise in addressing the fear of re-injury. Common psychosocial interventions include education to provide information about injuries [23] and setting specific and measurable goals on proper rehabilitation techniques, exercises, and protocols that can empower individuals to participate in their recovery actively [24]. Exposure to fearful situations, taking into account the injury history, current abilities, and comfort levels of the patients, and social support which can offer emotional reassurance, empathy, and understanding to help regain motivation [25, 26].

Table 2.3: Common psychosocial interventions to reduce the fear

Intervention	Explanation
Education	Provides knowledge and understanding about the injury, treatment, and recovery process, dispelling misconceptions and promoting confidence.
Goal setting	Involves setting realistic and attainable goals that help individuals focus their efforts, track progress, and build a sense of accomplishment, reducing fear and enhancing self-belief.
Exposure	Gradual exposure to feared activities or movements helps individuals regain confidence, overcome apprehensions, and trust their bodies again, reducing fear and anxiety.
Social support	Having a solid support system, such as friends, family, or support groups, provides emotional support, reassurance, and motivation, contributing to fear reduction.

2.2 Mountain bike cornering

As one of the essential skills in mountain biking, it has dramatically affected the occurrence of accidents. Wrong curve alignment and speed control will significantly impact the degree that affects the driver's control over the subsequent bike behavior.

2.2.1 Cornering techniques

The attainment of a seamless turn during biking necessitates the precise coordination of bodily and bicycle movements. The optimal process for executing a turn comprises a series of prescribed actions, commencing with the reduction of velocity, followed by a slight counter-steering towards the intended direction, learning the bicycle into the turn, and ultimately aligning the front wheel with the desired trajectory [27].

- **Steering**

  The handlebar plays a crucial role as the primary control point for steering in a bicycle. It is essential to recognize that steering and leaning contribute to the bike's turning, which suggests the need to separate these two actions. By decomposing the steering and leaning components, we can better understand and manage the different aspects of bicycle maneuvering.

- **Leaning**

  Leaning refers to the rider pressing and rolling the bike and their body sideways towards the ground. The rider can navigate corners smoothly and safely by achieving an appropriate leaning angle while maintaining the desired speed. During cornering, riders must lean their bodies and bikes to match the degree of the turn. The tighter and faster the turn, the greater the amount of leaning required [27]. Figure 2.4 illustrates the relationship between riding speed and the corresponding leaning angle needed.

- **Speed control**

  The braking technique and speed control is another essential factor for corners. Proper braking techniques can maintain ideal tire traction. Applying brakes too abruptly or unevenly can upset the bike's balance, making it difficult to steer. It is essential to modulate braking power smoothly to maintain control and traction. Particularly when approaching corners or technical sections [27].

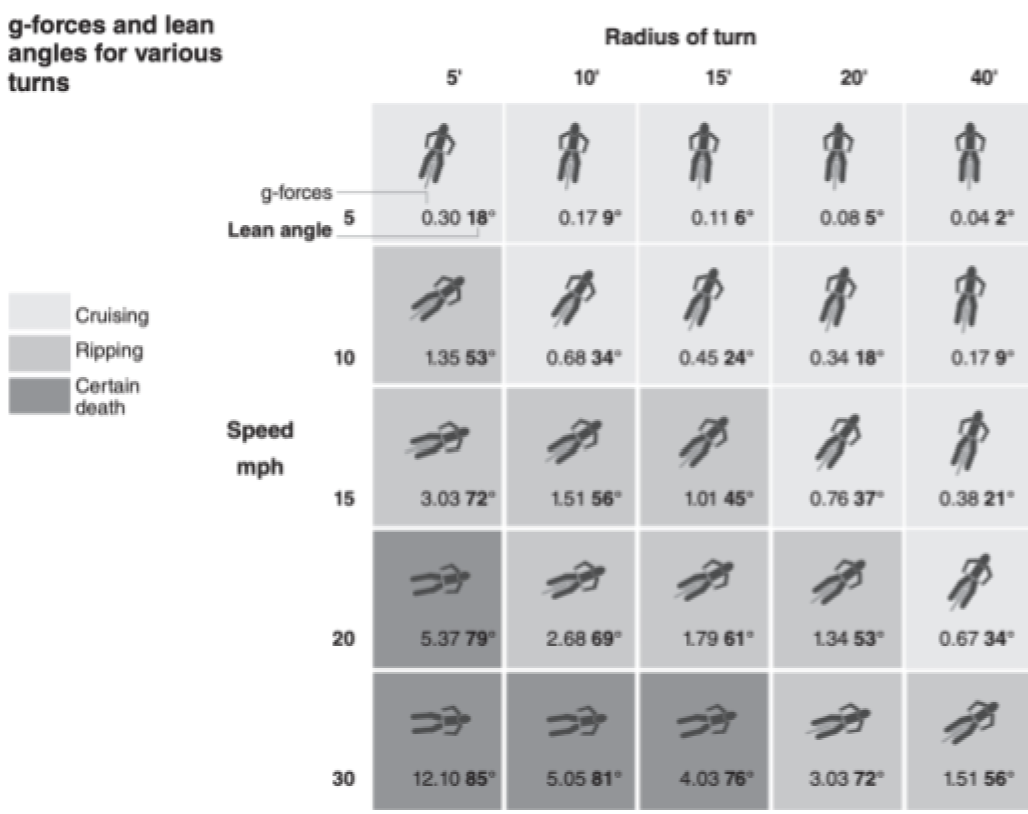


Figure 2.4 Graph of the radius of turn, lean bike angle, and speed relationship [27].

2.2.2 Kinematics of cornering

- **How does the bike turn**

  At very low speeds, such as speeds slower than walking, bike **steering** is achieved by turning the handlebars in the desired direction. However, **leaning** the bike at high speeds becomes more effective for changing direction. Leaning provides excellent stability and reliability compared to steering. When riders steer, the tire encounters bumps or irregularities on the ground, causing it to deflect in the desired turning direction (see Figure 2.5). On the other hand, when leaning the tire, it smoothly rolls around the turn, similar to a cone. Steered tires tend to slide, while leaned tires tend to maintain traction. (see Figure 2.6)

  The act of leaning the bike causes the handlebars to turn naturally. The bike will tend to travel in a straight line without leaning. Conversely, if the rider over-turns, the front wheel will tend to push forward and destabilize the bike.

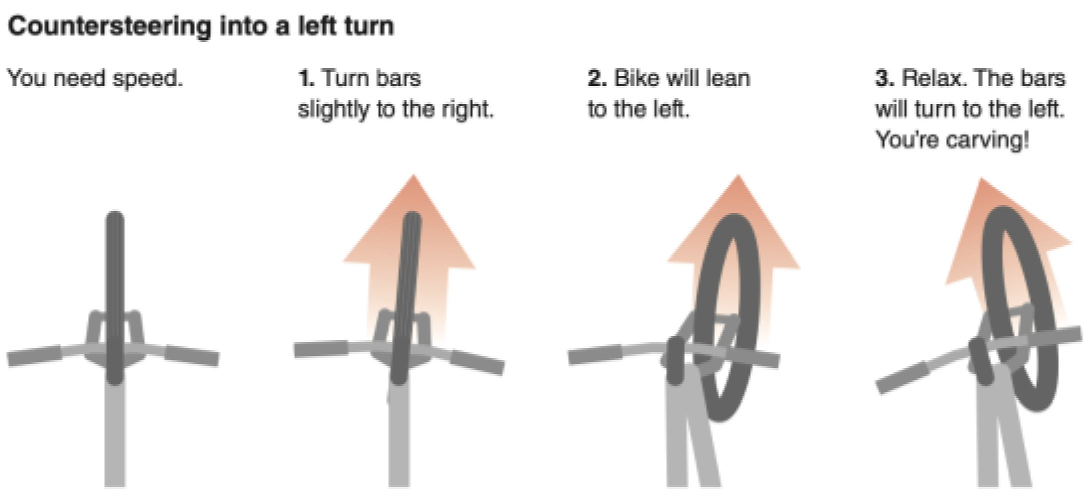


Figure 2.5 The steps of turning a bicycle [27].

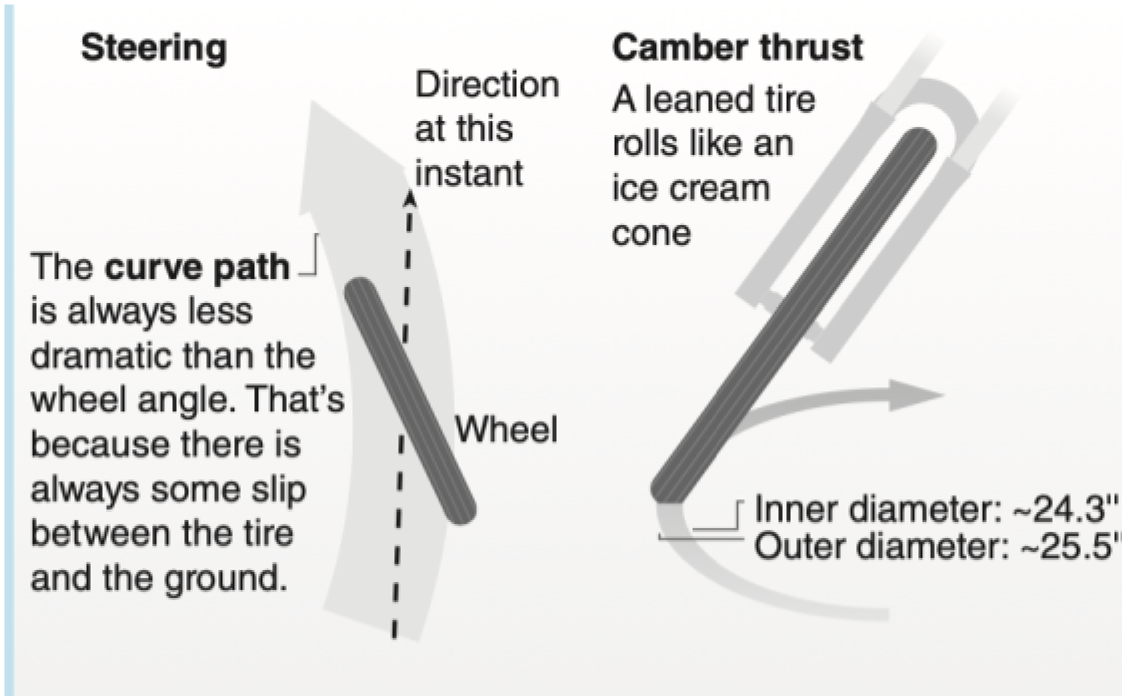


Figure 2.6 How does steering and leaning affect the cornering [27].

2.2.3 Cornering strategy

The decision for a line choice on making a turn is critical in mountain biking. Many corners are in series, like the cornering shown in Figure 2.7. For negotiating the series turns, the decision for the first corner will affect the next corner. The wrong line chosen can cause loss of speed or over speed which can cause unnecessary crashes or injury [27].

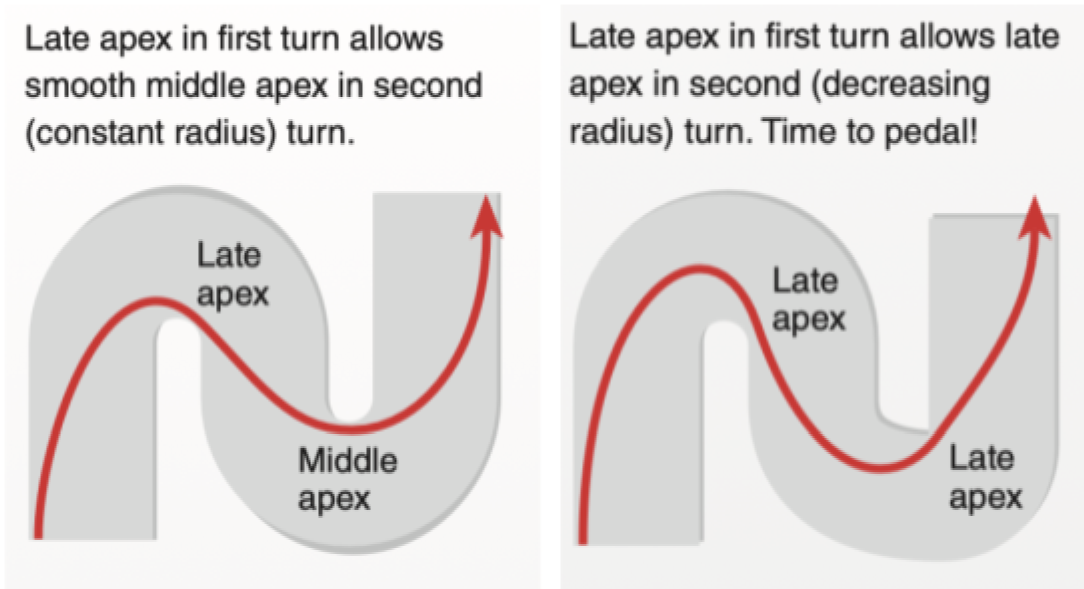


Figure 2.7: Series turns [27].

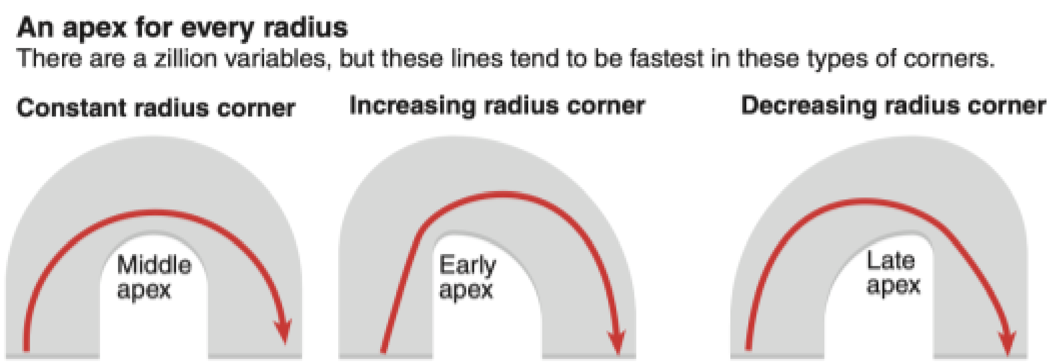


Figure 2.8: Cornering strategies apply on different type of corners [27].

The handling of different cornering strategies relies on the type of corners, but all corners can be attributed to the choice of apex when entering the corner; they are classified as middle apex, early apex, and late apex (see Figure 2.8).

- **Middle Apex**

  With a central apex, the riders enter from the outside with moderate speed, carve past the middle of the turn, and exit wide. In a consistent turn with good traction, a middle apex will exit faster than any other line.

Advantage: This is the mathematically perfect line. Riders carry the most speed through the entire turn.
Disadvantage: riders cannot see the exit very well. Poor traction and unseen obstacles can cause trouble.

- **Early Apex**
  With an “early apex” or “early entrance,” rivers can enter fast inside and dive right into the corner.

  Advantage
  can carry max speed into the turn. Riders can protect the inside line from some badger who is trying to pass over.

  Disadvantage
  riders must make a tight turn after the apex, which can lead to overshooting the corner or stalling on the exit.

- **Late Apex**
  With a late apex, riders must slow down, enter wide, initiate their turn, and then accelerate to the exit.

  Advantage
  riders can see farther through the turn before committing to a line. The line is relatively straight on the exit. Riders can start pedaling earlier and carry max speed out of the turn. \

  Disadvantage
  Nothing.

2.3 State-of-the-art

Technology has been widely applied to sports training and rehabilitation. This chapter will describe some latest achievements in mountain bike training and rehabilitation.

2.3.1 The current state of technology in bike training and fear reduction

Virtual Reality (VR) and virtual coaching has been widely applied to bike training by creating simulated environments that mirror real-world terrain and conditions. VR and Virtual coaching can provide the benefits of training without the potential risks and avoid getting injured.

2.3.1.1 Virtual coaching

Many current bike training devices are stationary bikes, those smart trainers that can simulate different types of terrain and resistance levels. Which is specifically designed for endurance and pedaling skill. It offers the function to connect smart devices, and users can change the training program and Synchronize training data. Meanwhile, the smart trainer also integrated a virtual coach since the trainer can use Bluetooth connection with users' mobile devices such as smartphones and watches. The data synchronization allows the smart training program to make a training plan based on user preferences and settings. Virtual training and coaching platforms like Zwift [28], TrainerRoad [29], and many others provide a range of training programs, virtual rides, and competitive racing experiences.

An example of a smart trainer is Wahoo KICKR (see Figure 2.9). It is an independent device that allows riders to exercise indoors without needing a real bike. It consists of a stand with a built-in resistance mechanism. Riders mount the device and pedal against the resistance provided by the trainer. The trainer can simulate the experience of riding a bike, providing a cardio workout and allowing riders to improve their fitness and endurance. Meanwhile, users can access various features and information from the built-in screen. The screen typically displays data such as speed, distance, time, and calories burned. It also shows real-time metrics like cadence (pedaling speed), heart rate, and power output if the user has additional sensors or a compatible device connected.

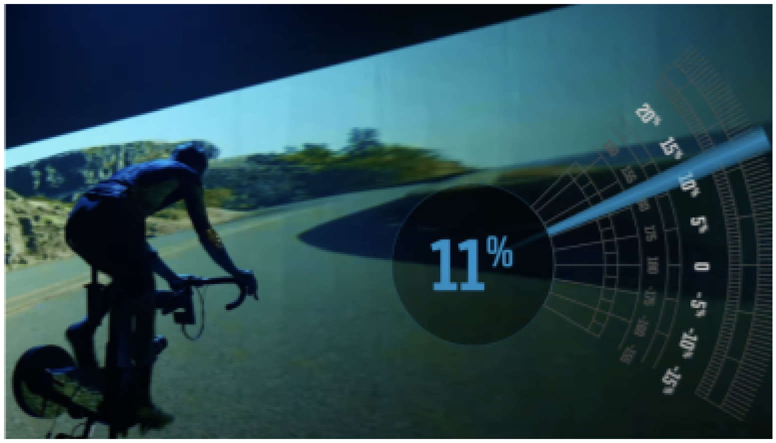


Figure 2.9 Wahoo smart trainer[39]

A stationary bike trainer itself does not directly provide fear reduction functions. However, it can be used within a broader fear reduction intervention program. The trainer offers a controlled and safe environment for injured riders to engage in cycling activities when they cannot ride outdoors, which can contribute to fear reduction through gradual exposure and increased confidence [30, 31].

Advantages

- Personalization: A virtual coach can provide personalized training plans based on individual fitness levels, goals, and preferences.
- Progress tracking: A virtual coach can track training progress over time, adjust the training plan accordingly, and provide valuable insights into rider performance.
- Realistic training environment: A competent trainer can simulate real-world riding conditions, making indoor training sessions more engaging and effective.
- Accountability: Working with a virtual coach can provide a sense of accountability and motivation, which can help stay on track with the training plan.
- Flexibility: With a smart trainer and virtual coach, riders can train on schedule. This can be incredibly convenient for cyclists with busy schedules or living in inclement weather areas.

Disadvantages

- Lack of social interaction: While virtual training platforms provide a realistic training environment, they may need more social interaction and camaraderie in group rides or outdoor training sessions. Which can make the training experience less enjoyable for some cyclists.

- Limited feedback: While virtual coaching services can provide personalized training plans and feedback, they may need to provide the same level of feedback and interaction as an in-person coach.
- Monotony: Indoor training on a smart trainer can become monotonous, leading to boredom and a lack of motivation, negatively impacting training progress.

2.3.1.2 Virtual Reality

VR technology has been widely applied in fear reduction treatments and has proven effective [32], particularly in treating phobias and anxieties. While VR is widely used in bike training, such as balance practice and skill development [33, 34, 35], its application in fear reduction, specifically for bike sports, is currently limited.

One VR project [36] focused on creating a motion platform for mountain biking (see Figure 2.11). It aimed to provide an immersive experience of riding in the highly technical, challenging, and scary terrains in the first person of view, and the project has been considered using exposure therapy to reproduce a frightening environment seen as a therapeutic means of overcoming fear.

The design involved securing the bike in an alloy bracket, with a speed sensor attached to the rear wheel and a handlebar with a sensor to indicate the steering angle. The entire bicycle was placed on a platform resembling a boxing ring, featuring a soft ground that allowed the bike to roll within a limited range without tipping over.

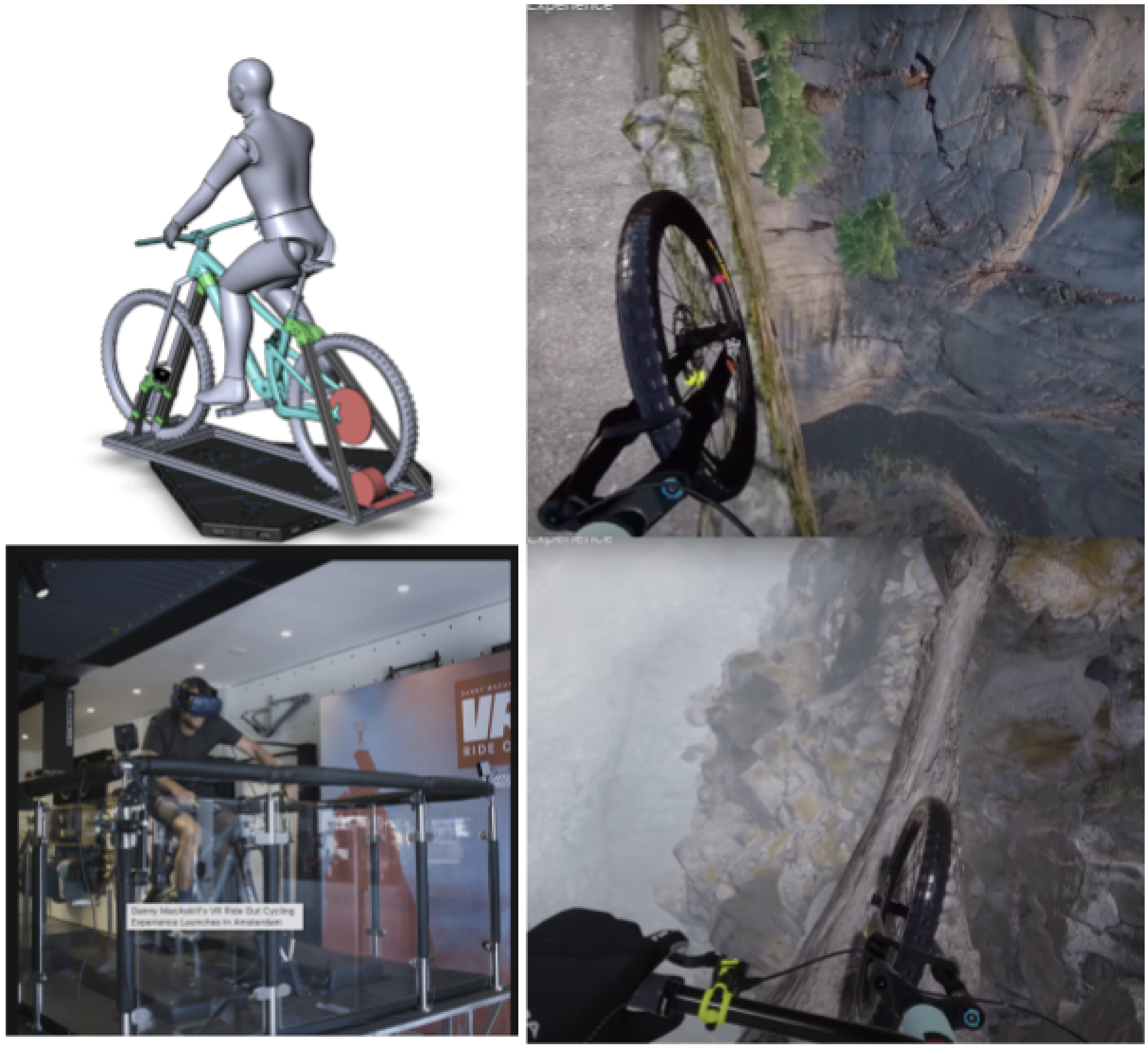


Figure 2.10: Danny MacAskill's VR Ride Experience project [36].

Advantage

- Realistic training environment: VR can provide a realistic training environment that simulates real-world cycling conditions. By using a VR headset, cyclists can immerse themselves in a virtual world miming outdoor terrain, weather conditions, and traffic.
- Variety of training options: With VR, cyclists can choose from various virtual courses and training programs, including virtual group rides, races, and structured workouts. This variety can make indoor training more engaging and effective.
- Improved motivation: By providing a more engaging and immersive training experience, VR can help cyclists stay motivated and focused on their training goals. The sense of

immersion and interaction with other virtual riders can also make training more fun and social.

Disadvantages

- Motion sickness: Some cyclists may experience motion sickness or discomfort while using VR, mainly if the virtual environment is too immersive or realistic. Which can limit the effectiveness of VR training for some individuals.
- Limited real-world training experience: While VR can provide a realistic training environment, it cannot fully replicate the outdoor cycling experience, including wind resistance, changes in terrain, and the sensory experience of being outside. Which can limit the transferability of skills and training to real-world cycling.

Discussion about VR in bike training

VR devices have shown effectiveness in various contexts. For example, Ranky [37] researched a virtual reality augmented cycling kit and reported promising outcomes in bike rehabilitation training. Additionally, Rimer [38] studies have shown that VR significantly helps reduce the fear of heights.

However, it is essential to consider the potential side effects of current VR technology. Researchers such as Schramka and Mittelstaedt [39, 40] have highlighted the issue of motion sickness experienced during bike riding simulation using VR. They conclude that the current VR technology may not be suitable for such simulations due to the intense motion sickness caused by the multiple body movements involved.

While side effects like motion sickness should be considered, VR in cycling training still holds potential and can benefit skill training and fear reduction.

2.3.2 Summary of the state-of-the-art

For now, there is a lack of integration between VR and virtual coaching in bicycle training. However, by combining the potential of VR in fear of reinjury treatment with the benefits of virtual coaching, injured mountain bikers could greatly benefit from a comprehensive approach that addresses both physical and psychological aspects of their recovery from the fear of reinjury and improve their biking skills.

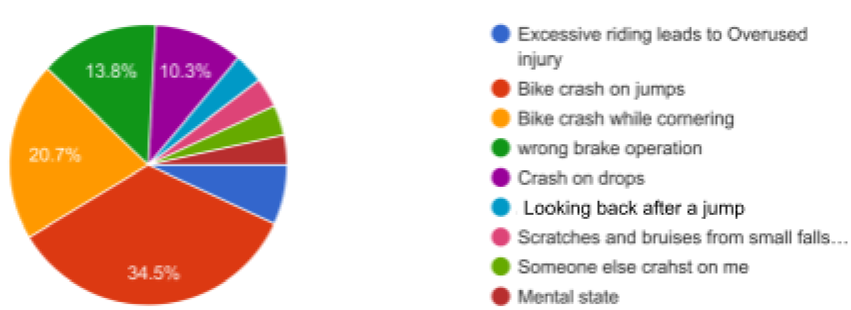
2.4 Questionnaire survey on mountain bike riders

A comprehensive survey conducted as part of the background research revealed that a significant percentage of riders above intermediate skill levels reported experiencing injuries, with half of them describing their injuries as severe. Cornering, failed jumps, and improper brake operations were identified as the main causes of these injuries. Moreover, many riders expressed fear of reinjury, with an average fear level. It revealed that mountain biking differs from other cycling activities due to its rough terrain. Additionally, it highlighted the need for digital biking training programs to focus on mountain biking skills. A questionnaire survey has been carefully designed to understand the needs of injured mountain bike riders. This survey comprised relevant questions to gather valuable insights from the riders and address their specific requirements.

2.4.1 User research results

The survey collected 29 responses from riders with diverse injury conditions and histories. These responses provided valuable insights into the specific needs of injured riders and the varying levels of fear they experience about reinjury. To analyze the data, rigorous filtering techniques have been applied to identify significant findings. The results of our analysis are presented below, showcasing the impact of our research.

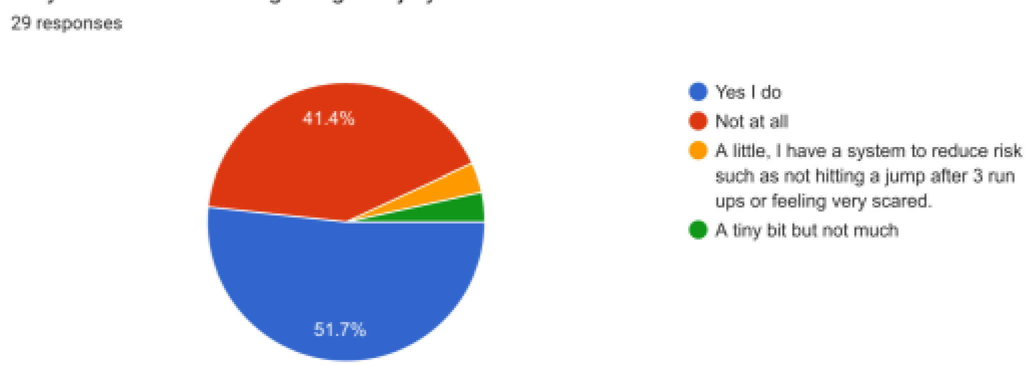
Chart 2.11: Which situation best describes the cause of your injury?
29 responses



As shown in Chart 2.11, the result shows jumps (32.1%), brake operations (14.3%), and cornering (21.4) are the majority of reasons a bike crashes. At the same time, other causes, such as fall crashes and overuse injuries, accounted for a smaller proportion, accounting for only 10.7% and 7.1%, respectively. Other causes of injury each accounted for less than 5%.

It is worth noting that the improper use of the brakes will lead to the loss of control of the bicycle, which will also affect the cornering of the mountain bike, so the accidents caused by the loss of brake control can be linked to the accidents of the curve and the jump, so the proportion of the two Should be higher than the actual data.

Chart 2.12: Do you have the fear of getting re-injury?



This question shows that more than half of the riders in the survey reported fear of re-injury (see Chart 2.12).
Furthermore, as shown in Table 2.11, riders with more than minor injuries reported fear of re-injury.

Table 2.11: Correspondence between injury history and fear of re-injury

Injured history:	Do you fear re-injury?
Bad injury	Yes
Bad injury	Yes
Bad injury	Yes
Bad injury	Yes
Bad injury	Yes
Bad injury	Yes
Small injury	Yes

Small injury	Yes
Small injury	Yes
Small injury	Yes
Small injury	Yes

Chart 2.13: Have you received any training or therapy for overcoming your fear?

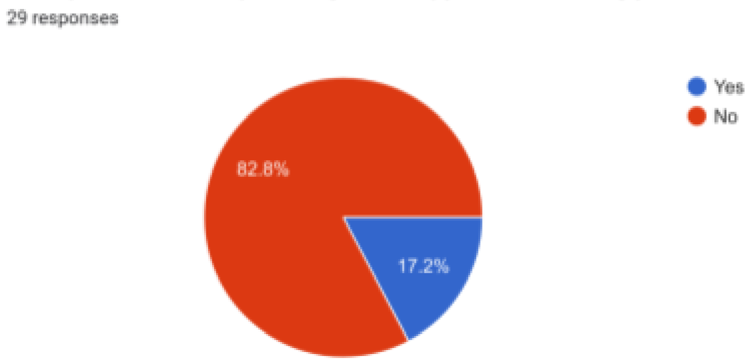
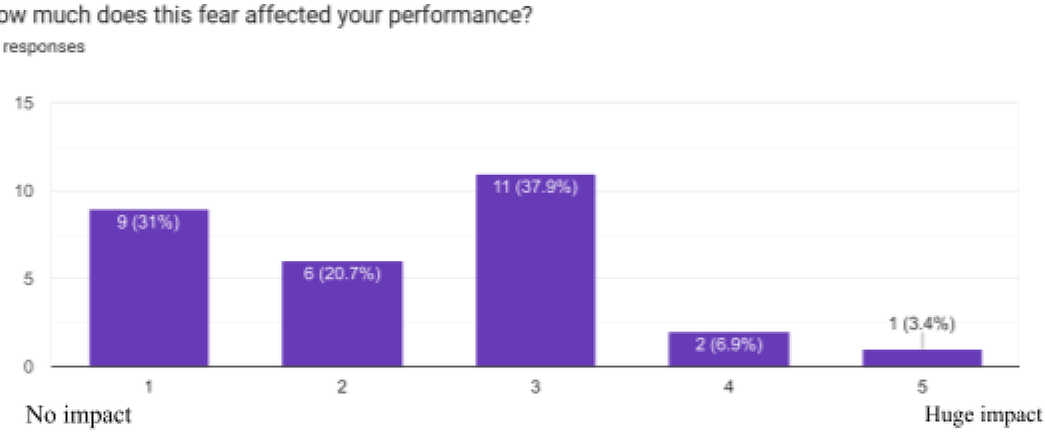


Chart 2.14: How much does this fear affected your performance?



Those two results (see Charts 2.13 and 2.14) show that most riders did not get treatment for fear of getting reinjured, and the fear affects their performance. 14 out of 29 riders reported that the fear has more than a middle impact on their performance.

Chart 2.15: When you get injured, do you want to train at home with a training device?

29 responses

As shown in chart 2.15, For riders who prefer to continue training once they have mobility back from injuries, the groups of riders who prefer to start with indoor training have a relative proportion with those who prefer to train outdoors. Others who prefer to rest until full recovery also took a considerable proportion. However, this condition can be variable due to the injury.

Chart 2.16: Do you think overcoming the fear of re-injury could benefit from physical activity?
28 responses

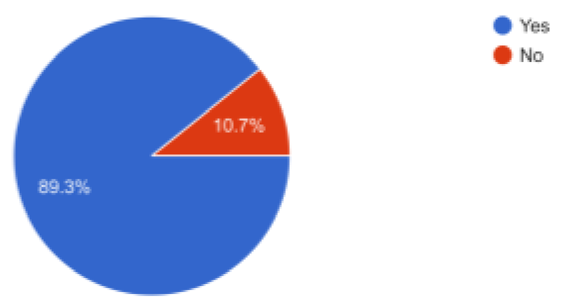
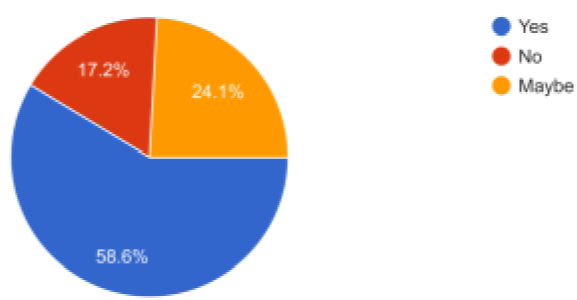
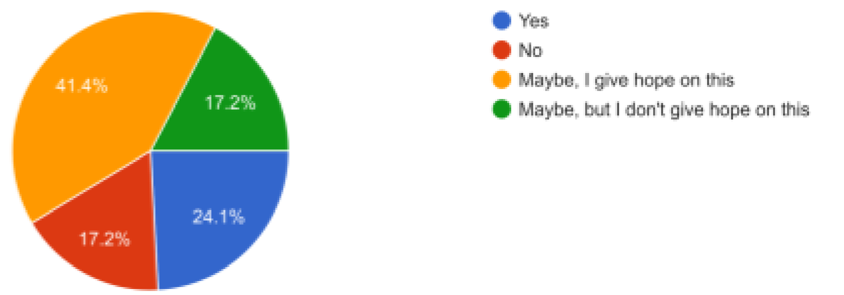


Chart 2.17: Do you think specific training for the mistakes you made will help you develop confidence and overcome fear?
29 responses



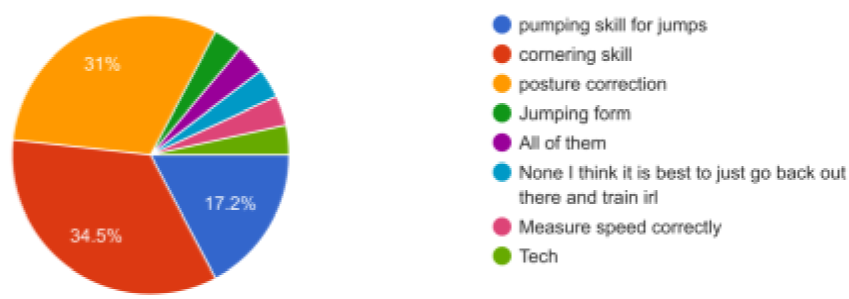
Charts 2.16 and 2.17 show that there is a large proportion of the interviewed riders believe that target training on the previous mistake can help them reduce anxiety and the fear of re-injury, and many of them believe physical training can be helpful, for the riders reported themselves had treatment for fear of reinjury, 80% of those riders share the agreement that physical training can help overcome the fear and anxiety.

Chart 2.18: Do you think a simulative bike trainer for mountain bike skills can overcome the fear of re-injury?
29 responses



The results in Chart 2.18 shows a quarter of the mountain bikers questioned thought mountain bike simulators could be used well to help overcome fears, with 41.4 percent skeptical but hopeful. 34.4% of people have a negative attitude towards alignment. But overall, mountain bike simulators have the potential to help overcome fear for most mountain bikers.

Chart 2.19: Which skill do you want the interactive bike trainer targeting on?
29 responses



Regarding which skill should be targeted by the trainer, the result shows (see Chart 2.19) that most riders prefer to practice their cornering skill and riding posture. Riding posture can highly influence bike control and agility and is essential for cornering. The third most significant request is the jump-pumping skill. It is essential for gaining speed and stabilization at jumps [27], but compared with the posture correction and cornering skill, it can be considered but needs to take priority.

Conclusion

In conclusion, this survey provides valuable insights that can guide the research and project design for finding a solution to help riders overcome their fear of re-injury. The survey indicates that jumps, brake operations, and cornering are the main reasons for bike crashes, and riders who have had injuries are more likely to fear reinjury. Additionally, physical training is considered a practical approach to overcoming fear, and most riders prefer to practice cornering and riding posture skills. The survey also shows a significant interest in developing a mountain bike simulator, which could be a valuable tool for training. These findings highlight the importance of addressing specific skills and providing practical physical training to help riders overcome fear and prevent future injuries.

Chapter 3- Methods and Techniques

This chapter will cover the methods and techniques used to design the
Interactive bike trainer. The design process will be presented. After this, the stakeholder identification and analysis method will be discussed. Lastly, it will be discussed how the requirements were set.

3.1 Method

The '*Design Process For Creative Technology design process*' (see Figure 3.1) has been followed to keep the process organized. The research process consists of three phases, ideation, specification, and realization. The Ideation phase has three starting points: **1.** search user needs/stakeholder requirements to understand their situation and problem, which is done using the MoSCoW method [42]. This method categorizes the requirements from most important to least important. The categories are: Must have, Should have, Could have, and Will not have. **2.** Find technology solutions, including the existing solution and possible solutions. **3.** Generate a creative idea or a brainstorm regarding the findings. Those three phases are connected and can be repeated to improve the ideation quality.

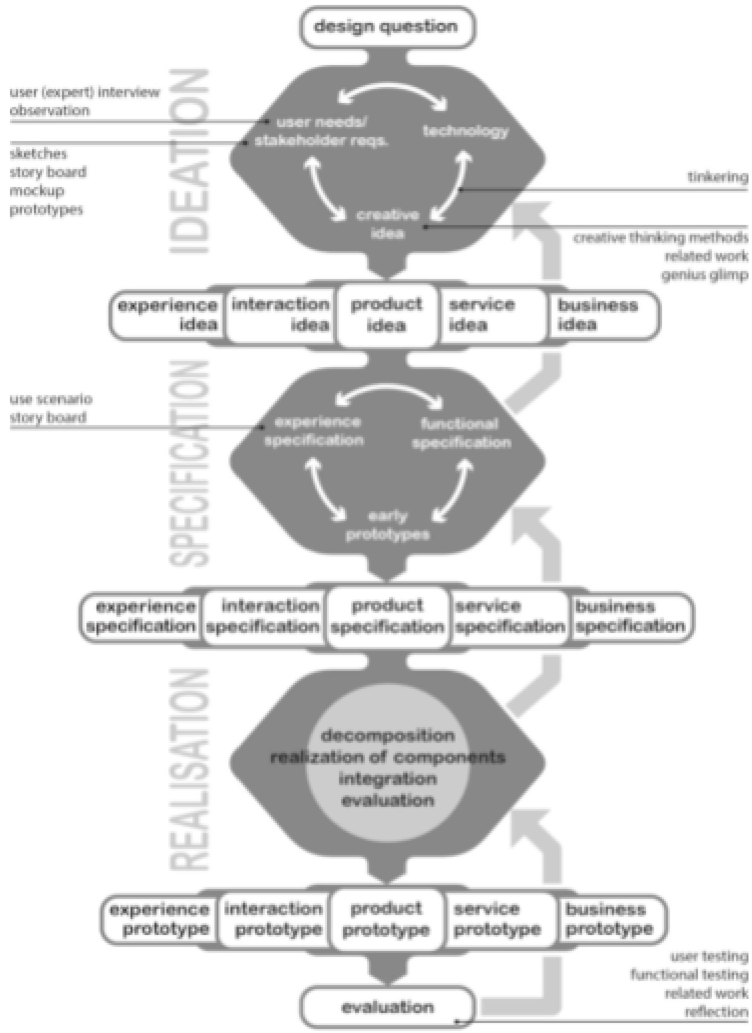


Figure 3.1– CreaTe Design Process Phases [41].

Considering the design question with those three aspects above, the design will be further considered under other factors. This project aims to introduce a solution to improve the training quality and reduce the fear of getting reinjured. Thus, the project focused on the sections of experience, interaction, and product/design ideations in the ideation stage before moving to project specification. Service and business ideas can be considered when the project is finished and approved effectively, but they can be considered later.

3.2 Techniques

3.2.1 Ideation

The ideation phase of this project was completed by the three-step process, beginning with user needs/stakeholder requirements, then creative ideas, and concluding with technology considerations. The initial step focuses on understanding and addressing the needs and requirements of the stakeholders, which involves identifying and analyzing stakeholders and delivering their specific requirements.

The ideation process transitions to generating creative ideas to explore initial design options for the technology being developed in the project. Once a range of creative ideas has been generated, the focus shifts to technology. Previous steps inform the evaluation and selection of technology options, leading to the finalization of the design choice.

3.2.1.1 Stakeholder Identification and Analysis

To make a thoughtful design, stakeholders should be identified. According to Mendelow [43], stakeholders can be analyzed based on two factors: their influence on the project and their level of interest. The stakeholders have their interests and roles. Some stakeholders have more influence and power than others. In order to identify the stakeholders and their interests, a stakeholder analysis will need to be implemented.

Figure 3. 2 shows the grid to help identify the role and how the stakeholders should treat them based on their power and interests. By doing so, stakeholders can be identified into four groups: keep satisfied, manage closely, monitor, and keep informed. In order to help the researcher prioritize the needs of different stakeholders to make decisions about research and design.

Figure 3.2– Power versus interest grid [49]

3.2.2 Specification

To determine stakeholder requirements, this project will utilize the MoSCoW priority checklist, which represents MO-must, S- should, Co-could, and W-won't. This method ensures thorough consideration of the primary users' needs. The MoSCoW method categorizes system requirements into four distinct categories. The "Must offer" category comprises non-negotiable needs, while the "Should offer" category includes essential but not vital requirements. Non-essential features that would be nice to have fallen under the "Could offer" category. Lastly, the "Won't offer" category pertains to requirements not deemed significant for the current development stage or a specific release.

To find those requirements, a semi-structured interview has been conducted with clients and potential users, enabling stakeholders to evaluate each requirement's value and prevent the oversight of critical ones.

3.2.3 Evaluation

The final phase of the project is the evaluation phase, during which the prototypes will undergo comprehensive testing. User testing will be conducted to assess the effectiveness and usability of the prototypes, as well as to identify design flaws and gather user feedback and suggestions for future improvements. The evaluation process will involve the evaluation of both low-fidelity and high-fidelity prototypes. The feedback gathered from the low-fidelity prototype evaluation will inform the improvements made to the high-fidelity prototype, which will then undergo the final evaluation.

Chapter 4- ideation

4.1 Stakeholder Identification and Analysis

4.1.1 Stakeholder Identification

Stakeholders have power, interest, and role. This information can be found in Table 4.1.

Table 4.1: Stakeholder Identification

Stakeholder(s)	Role
Mountain bike riders	Participants and clients
Coaches	Observer
Supervisor	Avidor and topic initiator
Project developer	Developer and researcher

The project aims to find a solution for injured mountain bikers to help them reduce the fear of re-injury. Thus they will be the primary stakeholders as well as the major research group and clients since they will be the end users of the project.

Coaches are considered observers and clients. The responsibility of coaches is mainly to provide training guidance for mountain bikers. In this project, they are regarded as observers and clients because coaches can provide guidance in training planning and help improve the design.

As the initiator, the project supervisor will serve as an observer and mentor, guiding each stage of the project. The project developers, on the other hand, take ownership of the project and are responsible for engaging with stakeholders and gathering information for research and design purposes.

4.1.2 Stakeholder analysis

Table 4.2: The power and interest level of different stakeholders.

Stakeholder(s)	Role	Interest	Power

Mountain bike riders	Participants	High	High
Coaches	Observer	Middle	Middle to low
Supervisor	Avidor and topic initiator	High	High
Project developer	Developer and researcher	High	High

Based on the stakeholder identification, the power and interest for the stakeholders can be filled (see Table 4.2). This project is for injured mountain bikers. Thus, they are the direct interest group and have limited power since they are not directly involved in developing this project. The training providers are also the stakeholders. They have little impact on the project's development and will only make connections when the actual device is developed. Supervisors and developers work very closely with each other in this project. They have regular meetings for progress checking and topic discussion. Therefore they involved the most power and interest.

Figure 4.3 Stakeholders analysis in the power/interest grid

4.2 Preliminary requirements

Preliminary requirements in Table 4.3 include all the design goals and expected functions.

Table 4.4: Prioritization of controller requirements

Requirements for game controller	**Moscow category**
Training content related to actual riding	Must
Safety	Must
Speed feedback	Should
Easy to use	Should
Steering and leaning feedback	Should

Table 4.5: Prioritization of game requirements

Requirements for the game	**Moscow category**
Provide clear feedback and helpful information	Must
Sync with a physical training device	Should
Sound	Could
VR headset	Could

4.3 Preliminary Concepts

The idea is to create a mountain biking simulator that faithfully replicates the real mountain biking experience, aiming to assist injured mountain bike riders in reducing their fear of reinjury specifically while cornering. This idea aims to merge exposure and education techniques with interactive technology to achieve a desired outcome where riders can gain training motivation and skills. Ultimately, the primary goal is to decrease the fear level among riders (see Figure 4.6).

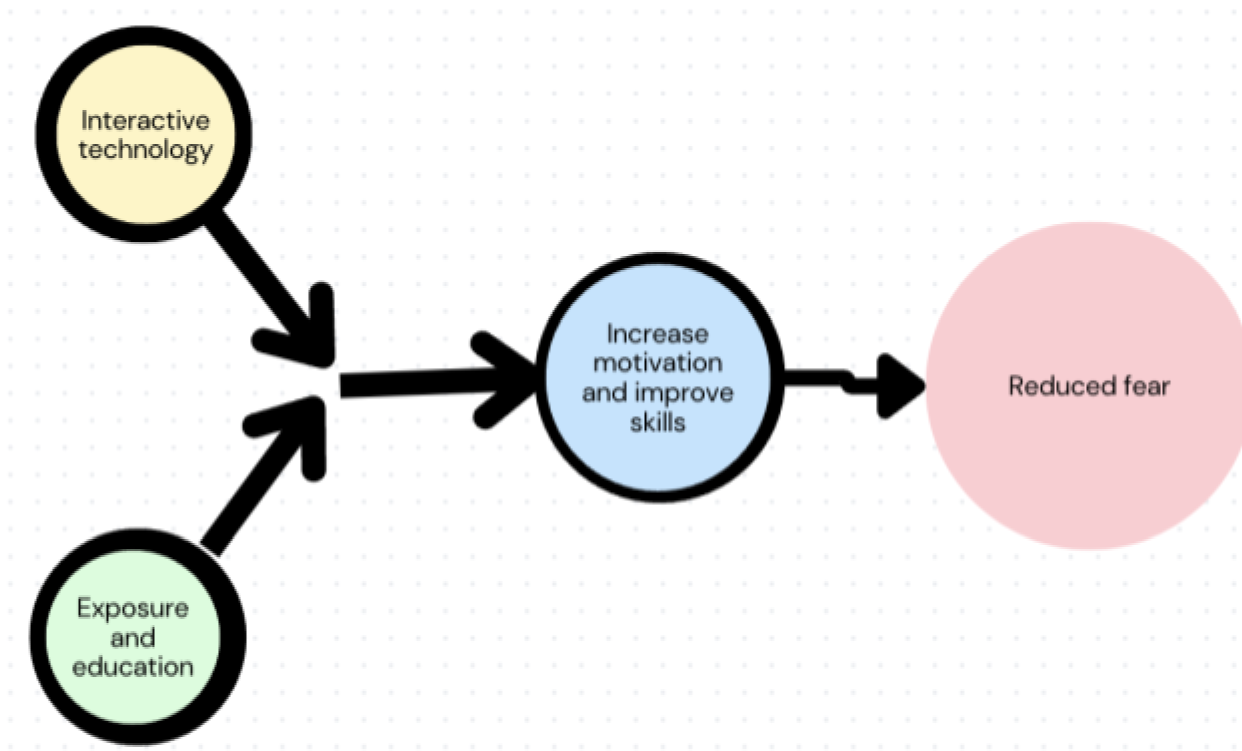


Figure 4.6: The development concept.

The core concept of this project revolves around introducing the exposure method, wherein the system recreates the daunting scenes experienced by injured riders, exposing them to the associated danger and fear within controlled conditions. Simultaneously, the project aims to provide a safe and constructive means for individuals to confront and overcome their fears. The integration of interactive technology serves as a facilitator in achieving this objective. The project adopts a construction concept centered around a game platform, enabling users to engage in a virtual mountain biking experience through a game controller. Users can interact with the mountain bike displayed on the screen and within the game environment (see Figure 4.7). This interactive experience allows users to immerse themselves in the real-life sensations of mountain biking while ensuring their safety. Additionally, the system incorporates cornering skills training to further enhance the user's proficiency in maneuvering through corners.

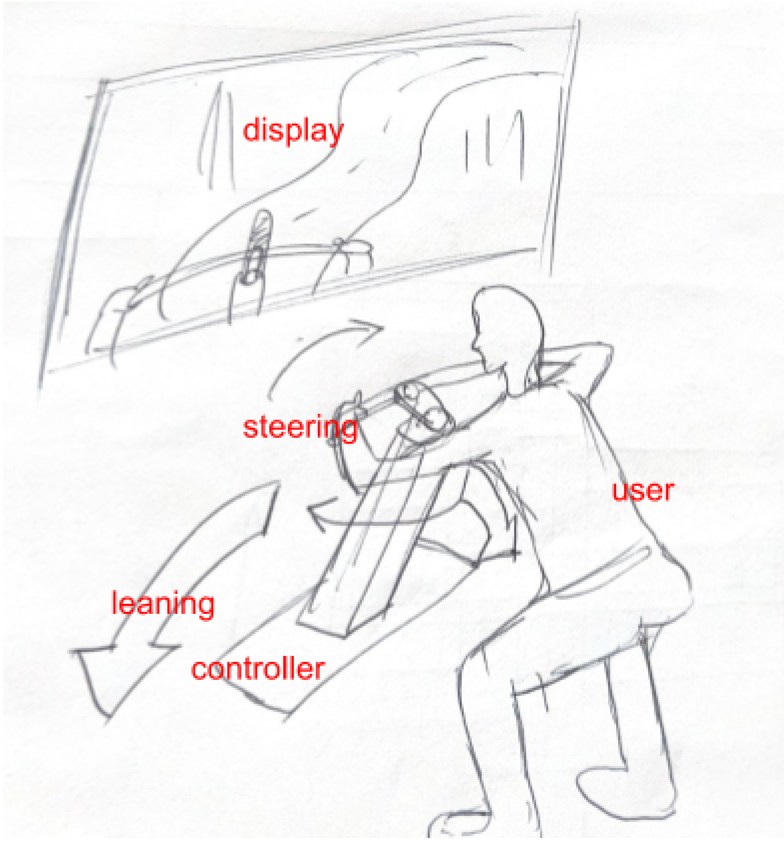


Figure 4.7: A mountain biking simulator game setup with a controller in the shape of the bike handlebar

4.3.1 Idea about the handlebar controller features design

Table 4.6 describes the functions that the controller should have as envisioned. Based on the real mountain bike functions, the controller should have the functions of steering, leaning, and brakes.

Table 4.6: Controller functions

Functions	Description
Steering function	Allows the user to rotate the handle with the pivot of the stem (see Figure 4.8).

Leaning function	Leaning a bike refers to the action of tilting the bicycle and shifting its center of gravity towards the inside of a turn with the contact point between bike tires and the ground (see Figure 4.9).
Brake function	Users should be able to control the speed by using a brake on the controller.

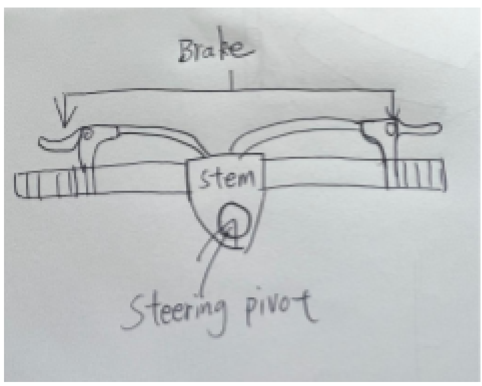


Figure 4.8: Bike handlebar

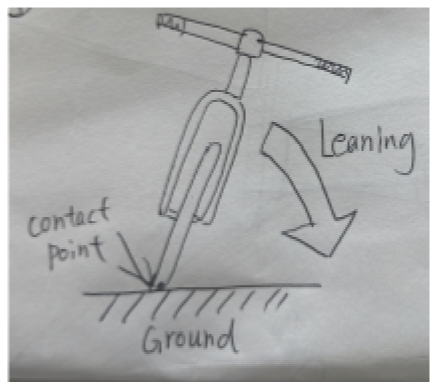


Figure 4.9: Leaning the bike

4.3.2 Game idea

In the game design aspect, the primary aim is to recreate the real-life riding experience from a first-person perspective, providing users with a visual and immersive environment that closely resembles actual mountain biking terrain. This design approach serves the purpose of exposure, allowing users to confront and overcome their fears in a controlled virtual setting.

The game will be synchronized with the game controller. When the user makes steering or leaning input using the controller, the bicycle model in the game will also respond accordingly. The bike will interact with the game environment and objects depending on the user inputs.

The game incorporates points and timing systems to encourage users to strive for better performance. Given that the project's objective is to address the fear associated with cornering, particular attention is given to the design of terrains featuring various cornering scenarios. By focusing on corners, the game provides users with a targeted and relevant experience that aligns with the project's intention of fear reduction during mountain biking.

Chapter 5 – Specification and low-fi prototype Development

Building upon the preliminary concept introduced in Chapter 4, this chapter aims to delve into greater detail and provide a comprehensive understanding of the prototype. Various aspects, such as data and parameters, tools, and materials, will be thoroughly discussed and clearly defined.

5.1 Specification of the controller

The first step in building the controller is clearly defining the size specifications. One possible approach is to measure the size and dimensions of real bikes. By doing so, the tilt angle and stack length can be determined accurately, ensuring that the controller closely matches the dimensions of actual bikes. This approach helps to enhance the authenticity and functionality of the system.

Head tube angle

For the development of this prototype, a head tube angle of 65 degrees has been selected. The head tube angle refers to the angle formed between the head tube of the bicycle and the ground. This particular angle was chosen because it aligns with the head tube angle commonly found in bikes ridden by the majority of riders [27, 44].

Handlebar Height

For the bike geometry, a handlebar height will influence the riding position, which is a crucial factor in reproducing a real riding feeling. The total length from the handlebar to the center of the wheel is the fork length added to the headtube length of the frame. In Figure 5.1, the fork length for this bike is 557 millimeters, and the head tube length is 99 millimeters for the medium size frame. Therefore the total length is 656 millimeters from the handlebar to the center of the wheel, which is the number we would like to use for the controller design.

Figure 5.1: Mountain bike fork length example [45].

Leaning Angle: To accommodate the flexibility of injured athletes and ensure a comfortable range of motion, the leaning angle of the controller will be adjustable. This can be achieved by incorporating a limitation screw, and setting the leaning angle between 30 and 45 degrees.

Handlebar: A real bike handlebar will be integrated into the controller to provide a realistic and immersive experience. Additionally, using a real handlebar allows users to easily customize their setup by switching to different lengths and rises according to their preferences. The handlebar will be securely attached to the mounting cylinder on the controller using a stem clamp.

Brake: Brakes play a crucial role in the biking experience, and they will be an essential part of the controller. The braking force will be linearly proportional to the pressure that the rider applies on the brake lever.

Sensor Implementation: Parameters were used to measure the angles (see Figure 5.2). A practical solution is to utilize potentiometers as position/angle sensors. By rotating the knob of the potentiometer, the wiper inside the device will change its position on the resistive track. This change in position will lead to a corresponding resistance variation between pin 1 (Vcc) and pin 2 (Signal). Potentiometers typically have a turning range of 0 to 270 degrees, which is sufficient for this project as no design elements require rotation beyond 180 degrees.

- Sensor on steering: Using a potentiometer to measure the steering angle.
- Sensor on leaning: Using a potentiometer to measure the leaning angle.
- Sensor on braking: Using a potentiometer to measure the brake lever rotation angle, a larger value represents a harder pull on the brake lever, increasing braking power.

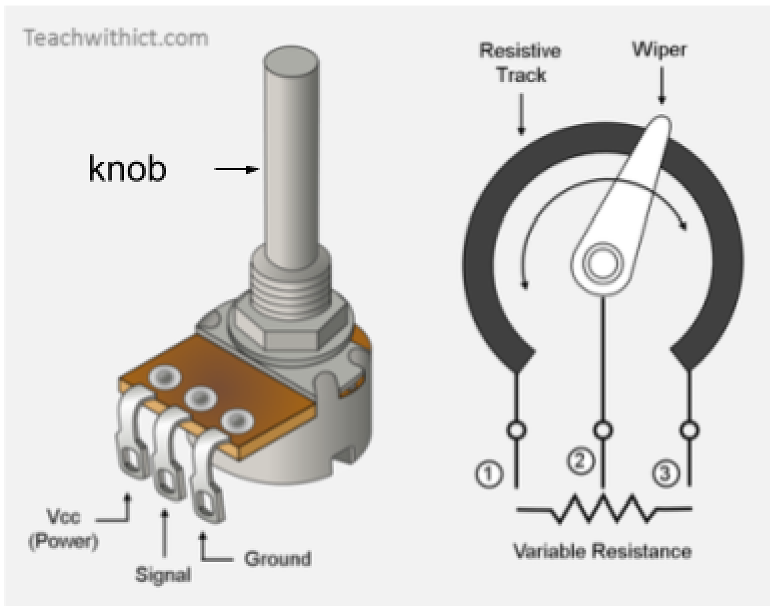


Figure 5.2: the structure of a potentiometer [46].

5.2 Specification of the game

To accurately simulate the steering and leaning actions of the bike in the game environment, a comprehensive programming approach is required. It is essential to consider the basic steering and leaning actions and the realistic physics governing the bike's behavior. Factors such as inertia influence The bike's performance, making steering more challenging at higher speeds.

To achieve this level of realism, the project will involve programming both Arduino and Unity. The overall plan for the programming implementation is outlined below:

1. **Arduino Programming:**
   - Configure Arduino to receive input from the bike controller, including steering angle and speed.
   - Develop code to process the input data, including implementing the relationship between maximum steering angle and speed.
   - Use Arduino's serial communication capabilities to send the processed data to Unity.

2. **Unity Programming:**
   - Set up the game environment in Unity to create a realistic biking experience.
   - Receive the data from Arduino via serial communication and interpret it within the Unity environment.
   - Implement the physics-based behavior of the bike, considering factors like speed, inertia, and the relationship between steering angle and speed.
   - Program the game controls to respond to the input from the bike controller, accurately reflecting the user's actions.
   - Visualize the bike's movements in the game environment, including the steering angle and leaning motion.

By combining the programming capabilities of Arduino and Unity, the project aims to create an immersive and realistic biking experience that accurately simulates the steering and leaning actions, taking into account the physics principles governing bike behavior.

5.2.1 Arduino

Arduino is a widely used open-source electronics microcontroller platform with Integrated Development Environment (IDE) software. It is equipped with input/output (I/O) pins, allowing developers to connect various sensors (as shown in Figure 5.3).

Integration with Unity: The proposed approach uses Arduino to establish communication between the game controller and the Unity game engine. Arduino will translate the measured values from the potential meters on the game controller to Unity. The sensors will be connected to the digital pins of the Arduino board.

Arduino Programming: Arduino IDE will be utilized for programming the Arduino board. The code will include instructions to read the real-time potential meter values from the plugin attached to the Arduino board.

Sensor Reading: The Arduino code will read the raw sensor values from the digital ports, capturing the data provided by the sensors.

Raw Value Modification: The raw values obtained from the sensors will be processed and modified to convert them into meaningful and useful data.

Serial Communication: Arduino will utilize its serial communication capabilities to facilitate data exchange between Arduino and Unity. The Serial library will be used to send data to the computer through the USB connection.

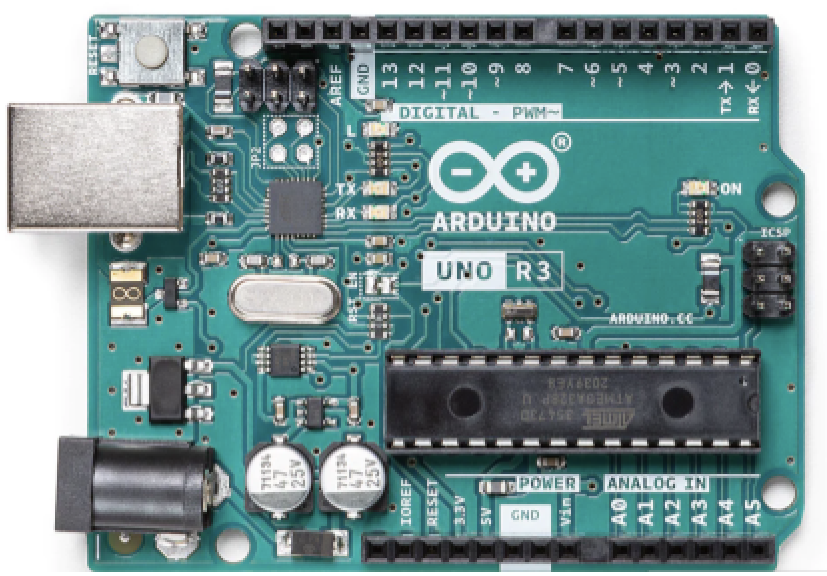


Figure 5.3 Arduino UNO [47].

5.2.2 Unity Game Engine

Unity is a game engine that allows developers to create games and interactive experiences across various platforms, and it can connect and exchange data with Arduino.

Figure 5.4: Unity Game Engine [48].

- **Unity setup**:
  - Set up a Unity game project and make sure it has the right setting to communicate with Arduino.
- **3D modeling**:
  - Bike model: A bike model should be made using Maya [49], a 3D modeling software.
  - Terrain design: suitable terrain should be made for the game, with possible features like trees and rocks.

- **Unity programming**:
  - **Data exchange**: Define a protocol for data exchange between Arduino and Unity by using C# script in Unity to establish communication with the Arduino board. Use the System.IO.Ports namespace to handle the serial communication.

  - **Bike behavior**: Write a script for the bike object to make sure it has the right bike behavior and can move, break steer, and lean based on the sensor data provided by Arduino.

- **Testing and iteration**:

- Test the setup by running both the Arduino code and the Unity project simultaneously. Ensure that the communication is working correctly and that the objects in Unity are responding as expected to the Arduino inputs. Finally, debug any issues that arise and improve the code.

5.2.3 SystemArchitecture

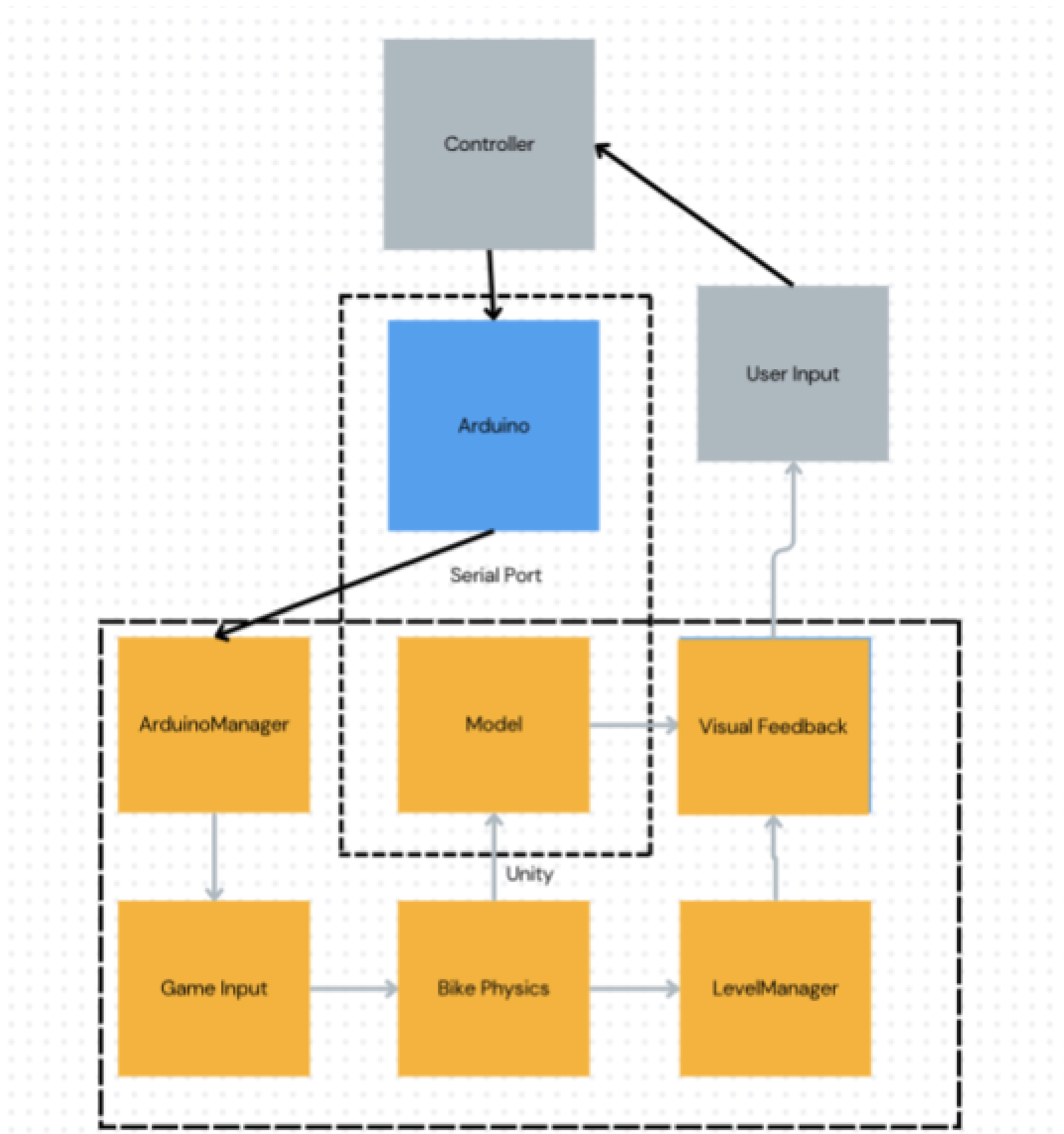


Figure 5.5: General system architecture

In this system architecture (see Figure 5.5), the user acts as the input method end, making input instructions to the game by operating the control handle. When the user uses the controller, the sensor installed on the controller will read the user's behavior and transmit the data to the Arduino. The Arduino will preprocess the raw data and then send it to the Serial port as a string. Subsequently, the Unity game engine will call the data in the string and assign them to different game variables in sequence. This behavior is performed by the **ArduinoManager** code. These assigned variables will change the game in the form of user input. The behavior of the object (bicycle) to interact with other objects in the game, such as the calculation of friction and the statistics of scores and time, and these two functions are handled by **Levelmanager**. All these changes will be presented on the screen with visual output, and the user will perform the following operation based on the visual feedback, thus completing the whole game logic.

5.3 Low-fi Prototype Development

The project development begins with the design and development of the bike controller. Based on the previously mentioned requirements and specifications, a detailed sketch has been created to clearly represent the controller's initial design. This sketching phase aims to refine the design and contribute to the subsequent modeling process. Visualizing the controller's layout and components through sketching makes it easier to make necessary adjustments and improvements before moving on to the actual modeling stage.

5.3.1 Description of the controller development

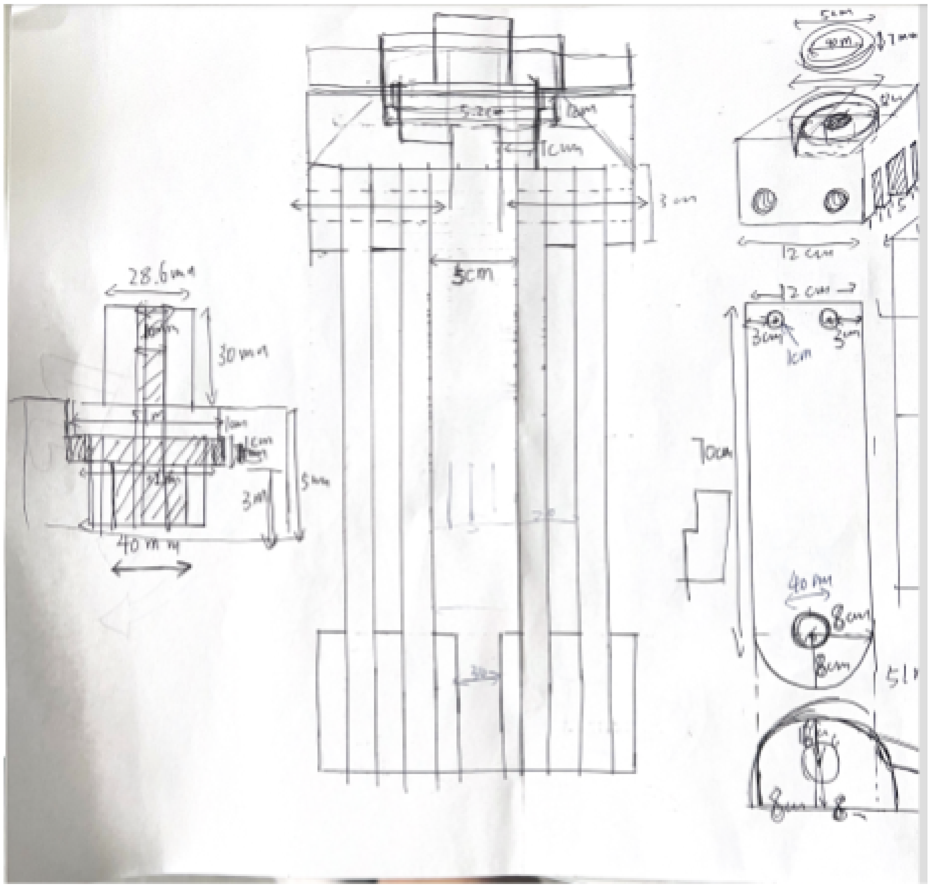


Figure 5.6: sketching of the controller design

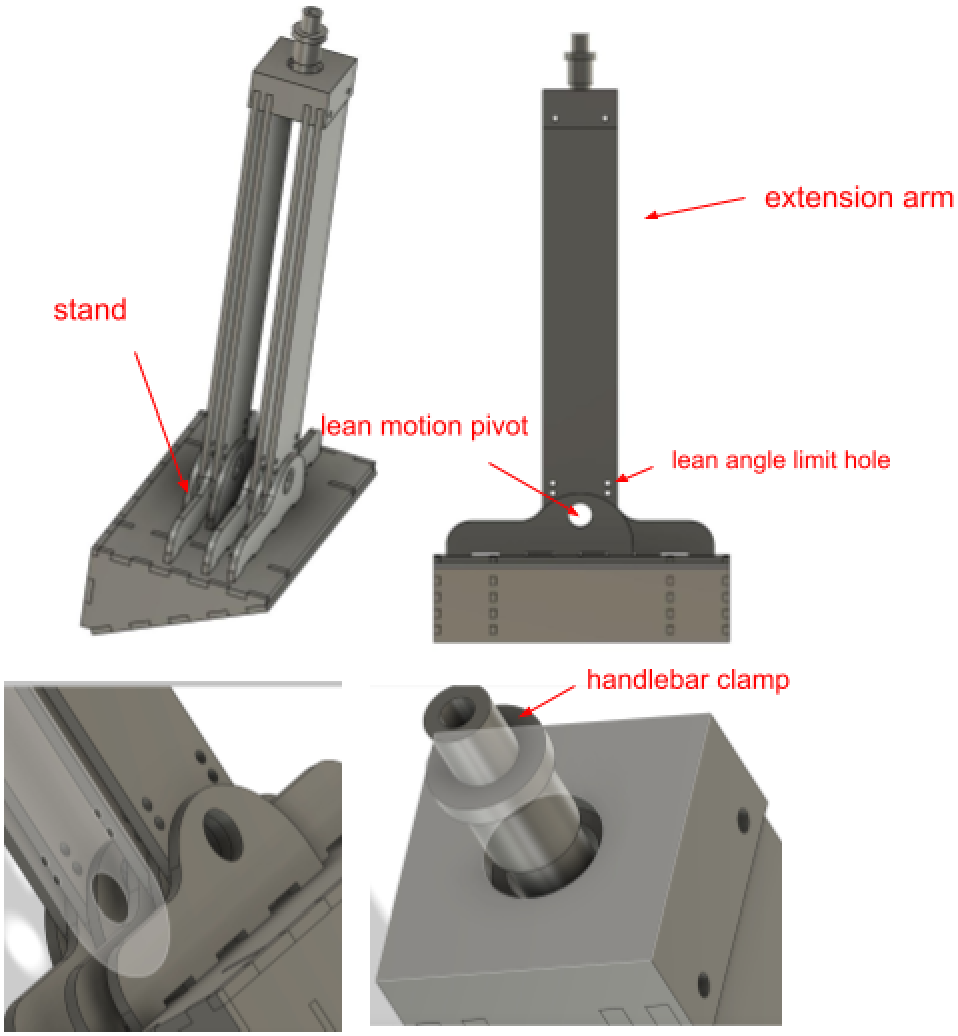


Figure 5.7 : Bike controller in 3D model

As shown in Figures 5.5 and 5.6, all parts are first drawn and designed before 3D modeling, and all connection joints have processed assembly simulation to ensure an expected fitting. As well as the moving parts like steering and leaning parts, the rotating functions test has also been done to ensure no design problem.

The controller is designed to be placed on the floor, and the part in contact with the ground is a pedestal, which can stabilize the whole controller and store the Arduino controller. The base was designed as a triangle. This design provides an inclination angle for the control lever above the base. This angle is mentioned in the above chapter. An inclination angle of 65 degrees was chosen to give users an experience close to the real mountain bike fork angle.

On the bottom area of the extension arm, a stand was made out of a four-layer structure. The extension arm will be connected to the base through a pivot point. The reason for choosing four-layer wood is to improve the sturdiness of the part To the extent that it breaks during use. The turning point is designed to meet the controller's function to realize the roll. The turning point is also provided by four supporting feet. A circular pipe connects the supporting feet and the extension arm above it. The center of the circular pipe is considered the turning pivot of the lean function.

The length of the extension arm has been discussed above. Based on the fork length and handlebar height of a real mountain bike, the length of the extension arm is set to 656mm. A square conversion base connects the uppermost openings of the four extension arms with holes through two long screws. Four slots at the bottom of the conversion base allow the four extension arms to be connected at intervals of 1 centimeter and cut to ensure that they can Overlap with no offset in the exact position. The conversion seat is square, except for the slot below for installing the extension arm. A hole in the center is designed to install an additional cylindrical part in the middle to connect the bicycle handlebar.

The underside of the cylindrical part is designed to have a diameter of 48 mm to fit the bearing. Above the bearing assembly position, a ring slightly larger than the outer diameter of the bearing is used to limit the bearing position and cover the bearing for aesthetic purposes. On the top of the cylinder part, a cylinder with a height of 3cm and a diameter of 28.6mm is designed to match the installation settings of the bicycle stem.

For the control handlebar, it does not need to be specially designed. Thanks to the design of the cylindrical conversion seat, the handlebar of an ordinary mountain bike can be directly installed and fastened on the cylindrical converter and connected with the entire controller.

Figure 5.8: Controller with bike handlebar attached

5.3.1.1 Sensor attachments

The sensor of the device is divided into three parts, the lean part, the steering part, and the brake part, and the movement mode of these three parts is rotation. Since the potentiometer is also an optional component, its installation and adaptation require their centers to be aligned with each other. For this reason, the sensors are respectively installed at the center of the circle at their corresponding positions by different small brackets (see Figures 5.9 and 5.10).

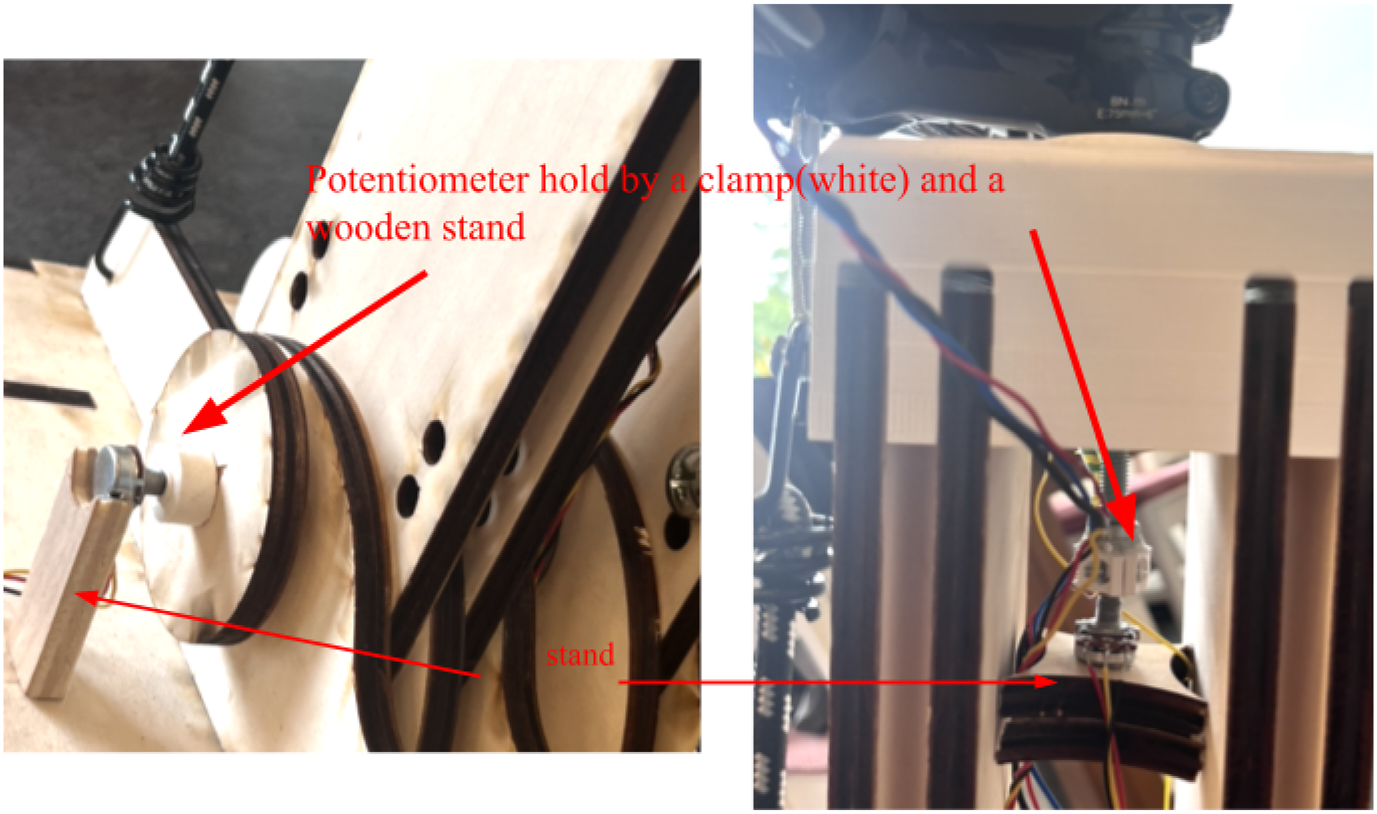


Figure 5.9: Sensor attachment at the leaning(left) and steering pivot (right)

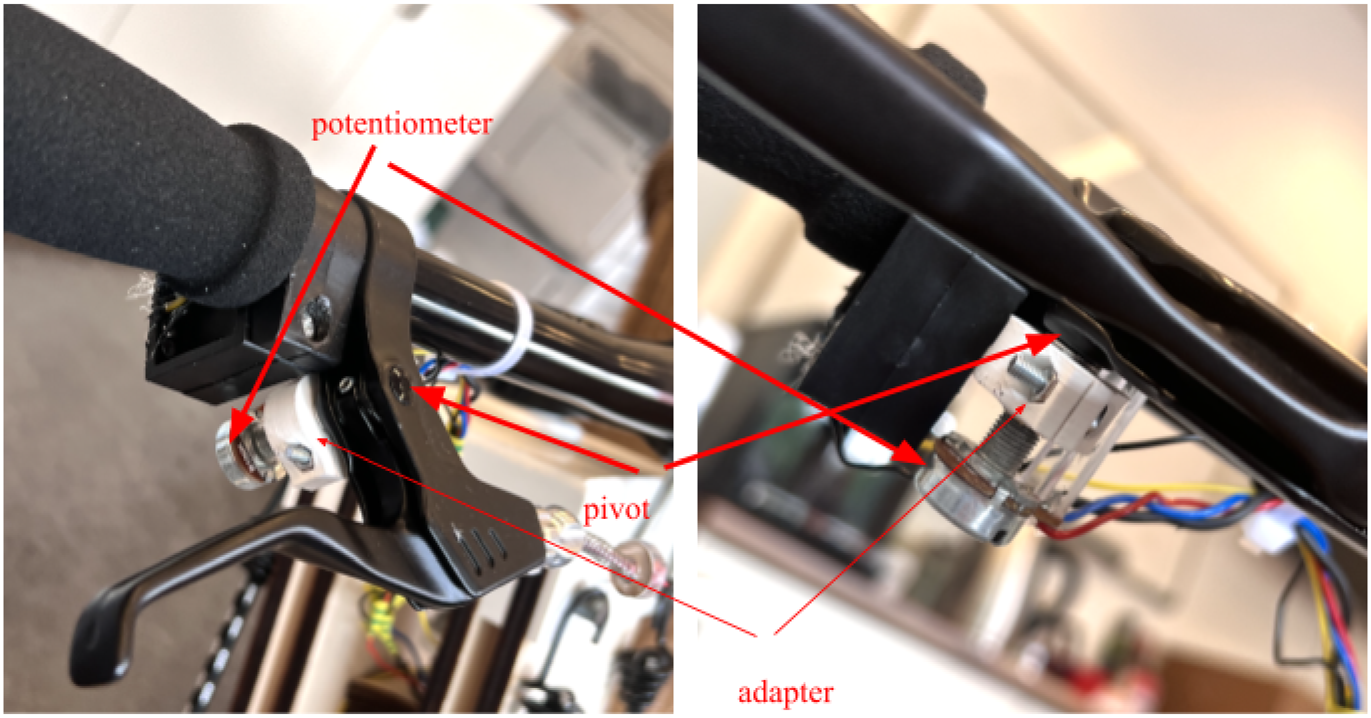


Figure 5.10: Sensor attachment on the brake lever.

All sensors have been placed at the center position of the rotating part, allowing the potentiometer and controller parts to rotate in sync for accurate data measurement. A 3D-printed plastic adapter was made to firmly lock the knobs of the potentiometer and the rotational shafts.

All the sensors are wired to the Arduino, and the Arduino and the breadboard required for the wiring are placed in a hidden storage space at the bottom of the controller (see Figure 5.11).

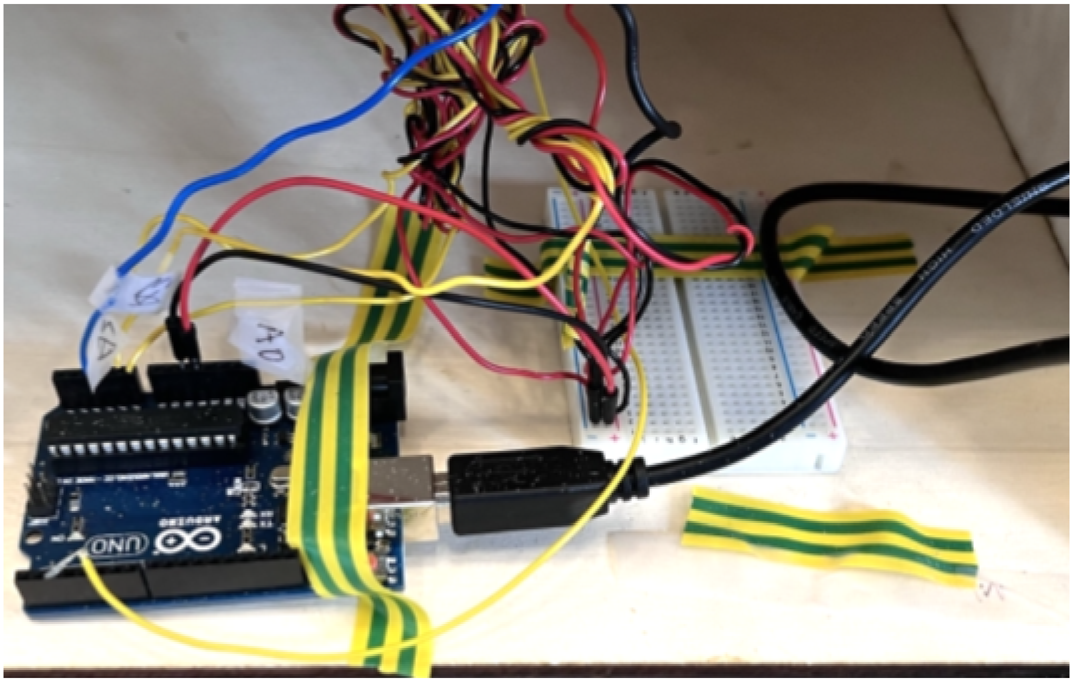


Figure 5.11: Arduino placed in the storage of the controller.

5.3.1.2 The Schematic of the Sensors and Circuit

The circuit below (Figure 5.11) shows the circuit diagram of how the potentiometers are connected to the Arduino board using *'Circuit Diagram Maker'* [50]. In the diagram, four potentiometers are connected to the Arduino. Two of them are used for measuring the rotation angle of the steering and leaning input. The other two potentiometers are connected to the brakes to measure the brake inputs. Since potentiometers require analogy reading, the four potentiometers are connected to analog pins A0 to A3. These pins can read voltages within a specified range and convert them into a corresponding digital value. The range is typically from 0 to 5 volts, corresponding to a digital value between 0 and 1023.

A digital pin (specifically D1) is utilized to connect a switch to the Arduino. Digital pins on Arduino are versatile and can function as either inputs or outputs. Switches, with their binary states of on and off, align perfectly with the digital pins' behavior. In the Arduino context, when the switch is turned on, it corresponds to a logic level of 0, while when turned off, it corresponds to a logic level of 1, which makes digital pins well-suited for connecting switches, allowing for easy integration and interaction within Arduino projects.

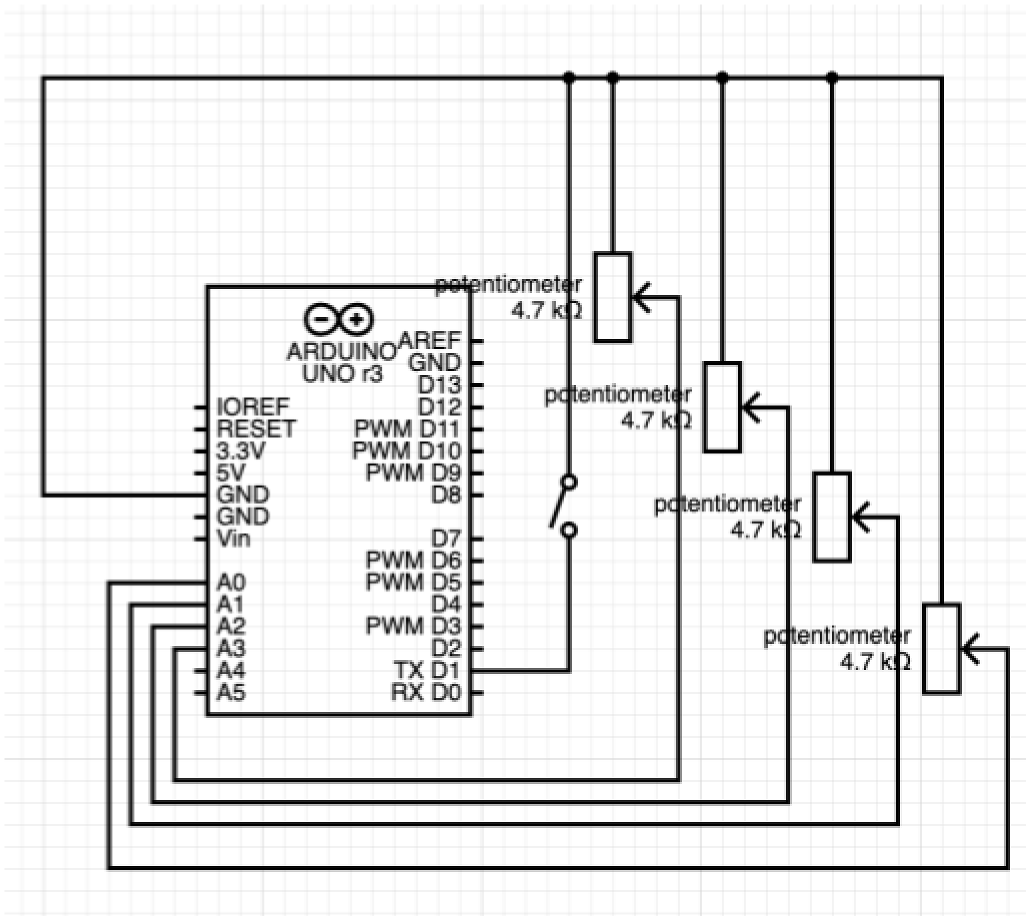


Figure 5.12: Circuit diagram of the Arduino setup.

5.3.2 Game development

5.3.2.1 Unity game setup

The development of this project begins with an extensive search for existing works that serve as valuable resources for research and study purposes. During this search, an insightful work titled *'Bikes and Bicycles | Unity'* [51] created by Ryan has been identified. Notably, Ryan's work evolved from an earlier motorcycle project called 'BikeBalance' [52], initiated by Vladimir Kudryashov. Both projects (see Figure 5.13) offer valuable insights into implementing bicycle kinematics within the Unity engine. These works serve as pivotal references, aiding the comprehension and integration of bike mechanics into the game system currently under development.

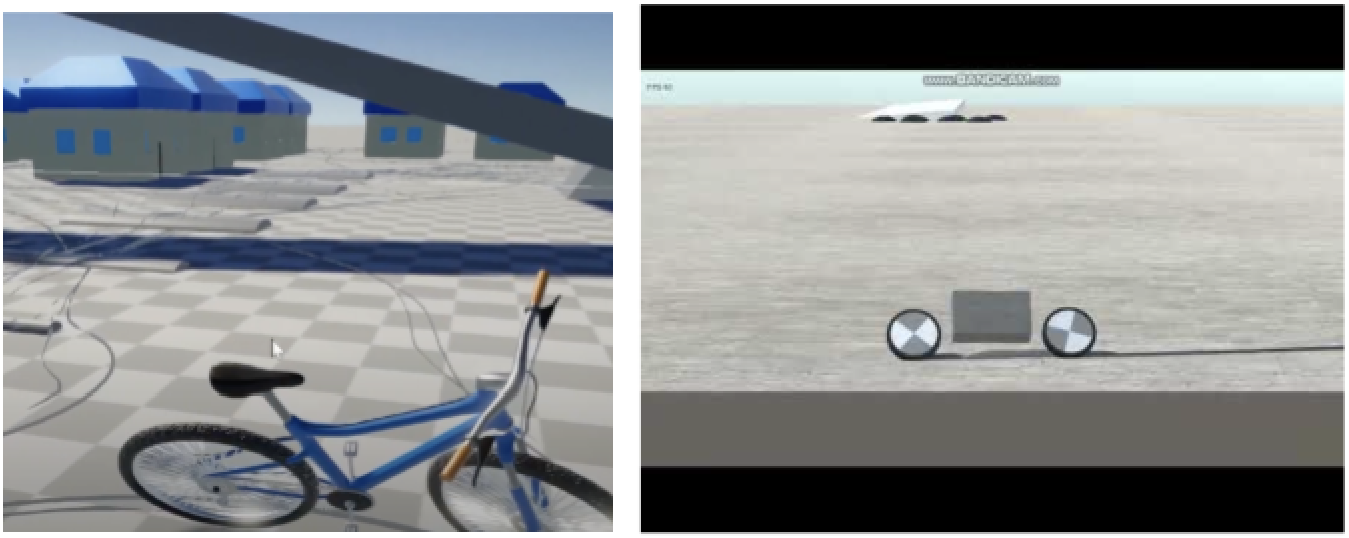


Figure 5.13: Bike physics in Unity developed by Ryan(left) [51], and Kudryashov(right) [52].

5.3.2.2 Bike Physics

5.3.2.2.1 Wheels

In the Unity engine, a physical system called ‘Wheel Collier’ was used in both projects. The ‘Wheel Collider’ is a special collider for grounded vehicles. It has built-in collision detection, wheel physics, and a slip-based tire friction model. The basic functions of wheel rotation, collision, and friction can be achieved by applying the Wheel Collider to the bike object. The description of the ‘Wheel Collier’ properties can be found in Table 5.14.

Table 5.14: Description of the properties in the WheelCollider menu [53].

Properties

Property:	Function:
Mass	The Mass of the wheel.
Radius	Radius of the wheel.
Wheel Damping Rate	This is a value of damping applied to a wheel.
Suspension Distance	Maximum extension distance of wheel suspension, measured in local space. Suspension always extends downwards through the local Y-axis.
Force App Point Distance	This parameter defines the point where the wheel forces will applied. This is expected to be in metres from the base of the wheel at rest position along the suspension travel direction. When `forceAppPointDistance = 0` the forces will be applied at the wheel base at rest. A better vehicle would have forces applied slightly below the vehicle centre of mass.
Center	Center of the wheel in object local space.
Suspension Spring	The suspension attempts to reach a **Target Position** by adding spring and damping forces.
Spring	Spring force attempts to reach the **Target Position**. A larger value makes the suspension reach the **Target Position** faster.
Damper	Dampens the suspension velocity. A larger value makes the **Suspension Spring** move slower.
Target Position	The suspension's rest distance along Suspension Distance. 1 maps to fully extended suspension, and 0 maps to fully compressed suspension. Default value is 0.5, which matches the behavior of a regular car's suspension.
Forward/Sideways Friction	Properties of tire friction when the wheel is rolling forward and sideways. See the *Wheel Friction Curves* section below.

As shown in Figure 5.15, it shows the friction should apply to the wheels. In Unity, the friction has been separated into forward and sideways. Combining these friction inconsistencies and employing a single contact point can result in unpredictable behaviors during bike operations. For instance, the bike may rotate or drift when executing fast turns, resulting in losing grip on the rear wheel and an airborne slide.

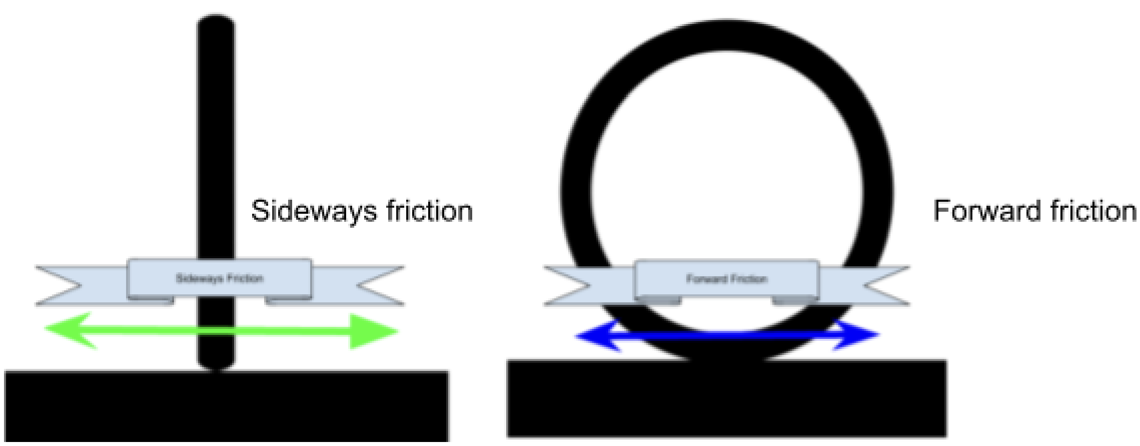


Figure 5.15: The friction apply to the wheels [53].

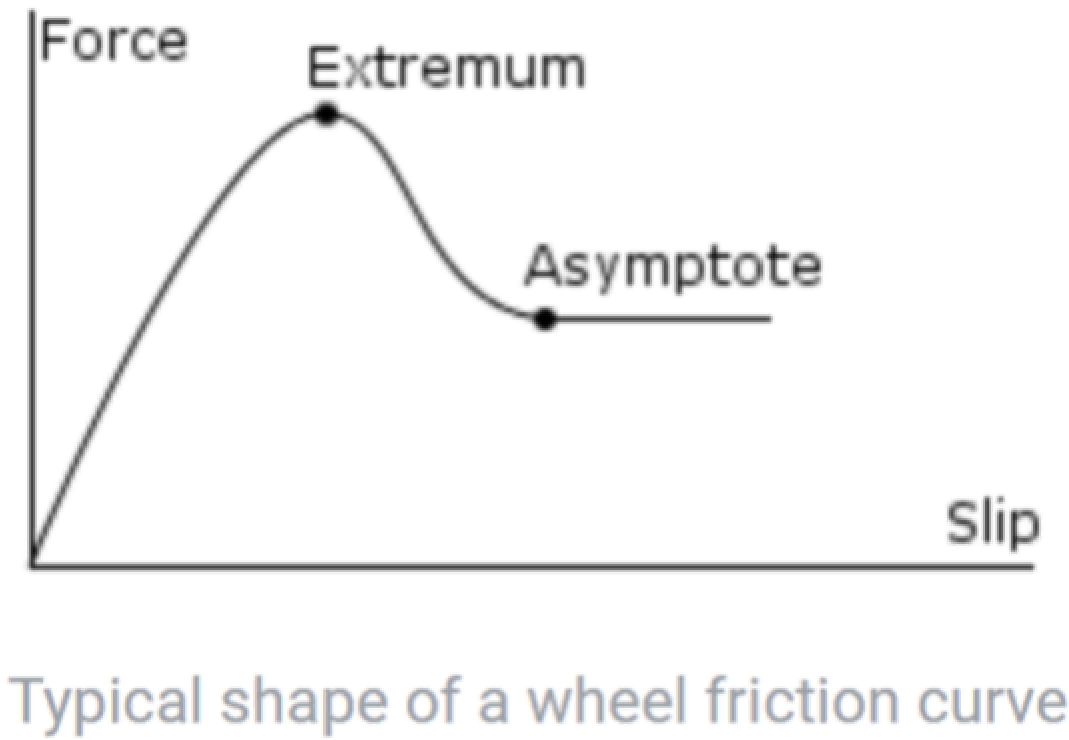


Figure 5.16: wheel friction curve for tries [53].

To accurately configure the grip levels, it is necessary to comprehend the grip curve of a tire and replicate its behavior. Manual from Unity [53] provided a 'Wheel Friction Curve' (see figure 5.16), representing the relationship between tire slip and the corresponding force exerted. A two-piece spline approximates it. The first section of the curve extends from (0, 0) to (ExtremumSlip, ExtremumValue), reaching a point where the curve's tangent is zero. The second section extends from (ExtremumSlip, ExtremumValue) to (AsymptoteSlip, AsymptoteValue), where the curve's tangent is once again zero.
The property of real tires is that they can exert high forces for low slip since the rubber compensates for the slip by stretching. Later, when the slip gets high, the forces are reduced as the tire slides or spins. Therefore, setting the extremum and asymptote value in the wheel collider setting is essential, and it requires many tuning and tests.

5.3.2.2.2 Bike setting

In the last section, the setting and modifying of the wheel objects have been explained. The properties assigned to the bike object can be found in **Table 5.17**.

Table 5.17: the property of the bike object in-game.

Property	Description
Handle	The handle unit is responsible for locating the transform of the bike handlebar model. The game object 'HandlePivot' should be assigned to enable the front wheel to spin with respect to the Y-axis on the steering tube.
COG	The COG (Center of Gravity) represents the center of gravity of the bike body. It is assigned to account for the interaction between the human rider and the bike. In this project, the bike and rider are treated as a single object, and the center of gravity of this combined object is positioned between the bike and the rider's body. A -0.1 rise on the Y-axis denotes the correct position in the game environment.
Pedal force	Although this project does not include a pedaling function, the bike still requires input for acceleration to counteract gravity. Pedal force represents the forward driving force applied to the rear wheel of the bike object, enabling the bike to accelerate when necessary.
Current steering angle/steering input/max steering angle	These parameters govern the steering behavior of the bike. When the bike controller is attached to the game and sends a steering input value to the Unity engine, it does not directly result in a physical steering angle. The steering input represents the desired amount of steering by the rider, but factors such as the bike's current speed and steering delay influence the actual steering behavior. The '**speed steering reductor**' section demonstrates how the steering angle is limited based on the bike's speed. With this function integrated into the bike, even if the user applies a significant steering angle through the controller, the Unity engine ensures that the bike stays within a safe steering range relative to its moving speed.

5.3.2.2.3 Leaning and steering control

User inputs do not apply to the wheel directly. The steering inputs are affected on the fork and then on the wheels. Considering this, To control the wheels in the game engine, the control of the handlebars needs to be transmitted to the front fork and then transmitted to the wheels. In the game set, the handlebar and fork objects have been filled with a collider and grouped and parented by the handlebar pivot object. The pivot is centered at the bike steering tube. By doing so, the whole front part, including the handlebar, fork, and wheel, can be able to rotate about an axis where the handlebar pivot is (see Figure 5.18).

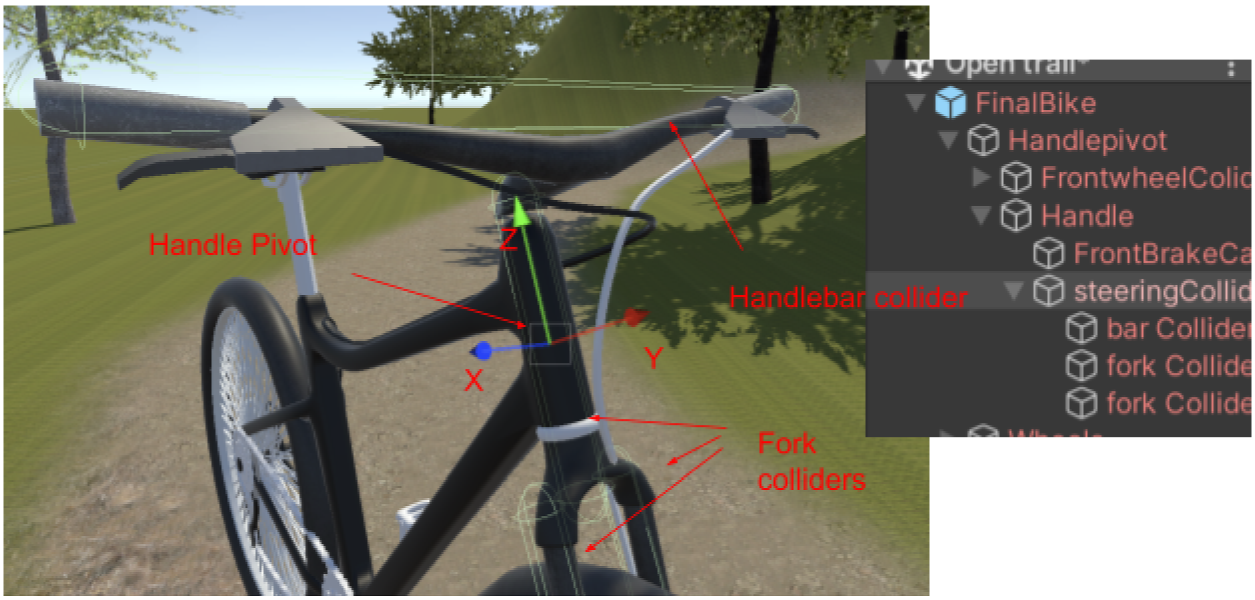


Figure 5.18: Bike steering control collider setting.

Like the physical steering logic, the bike frame leaning movement is also transmitted from the handlebar to the frame. When the handlebar forces the front wheel to lean to the side, the front fork collider cannot move because it is restricted in the frame model, so the frame and the rear wheels will follow this leaning trend and fall together with the front wheel to complete the leaning movement.

5.3.2.3 Terrain design in Unity

To facilitate terrain exposure and enhance cornering skill education for mountain bikers, the terrain design will incorporate specific cornering training elements to reduce the fear of reinjury during cornering. Drawing inspiration from the book *'Mastering Mountain Bike Skills'* and the discussions in Chapter Two, various cornering strategies have been explored.

The environment will be enriched by adding elements such as trees, grass, mountains, and obstacles. Some of these objects, including trees and material textures, will be sourced from the Unity assets stores [54] to enhance the visual appeal and realism of the terrain.

The open terrain (see Figure 5.19) allows players to ride freely within the game environment.

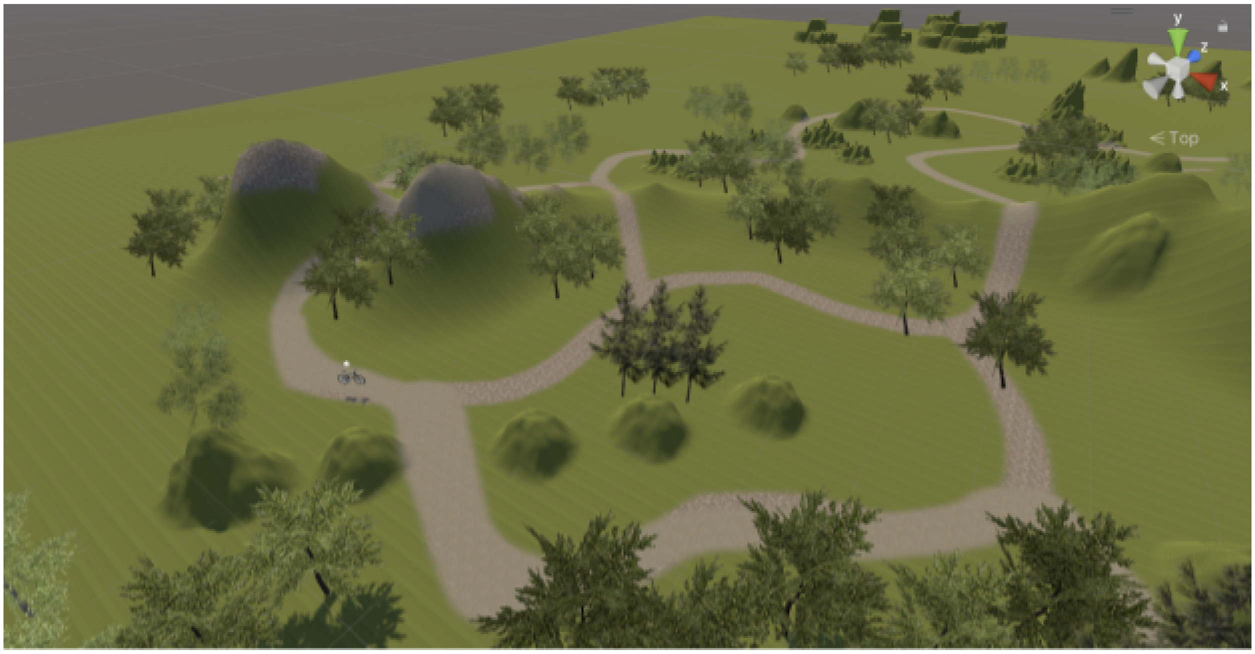


Figure 5.19: Open terrain layout

Chapter 6 – Low-fi Prototype Testing and Improvement Plan for Hi-fi Prototype

This project is currently at the stage of early reflection and the first user and functional testing as following the *'Create design Cycle'* (see Figure 6.1). This section presents the evaluation of this mountain biking simulator. This evaluation aims to gain feedback to improve the low-fi prototypes and examine the simulator's effectiveness and potential impact in improving bike and cornering skills while reducing the fear of reinjury among riders.

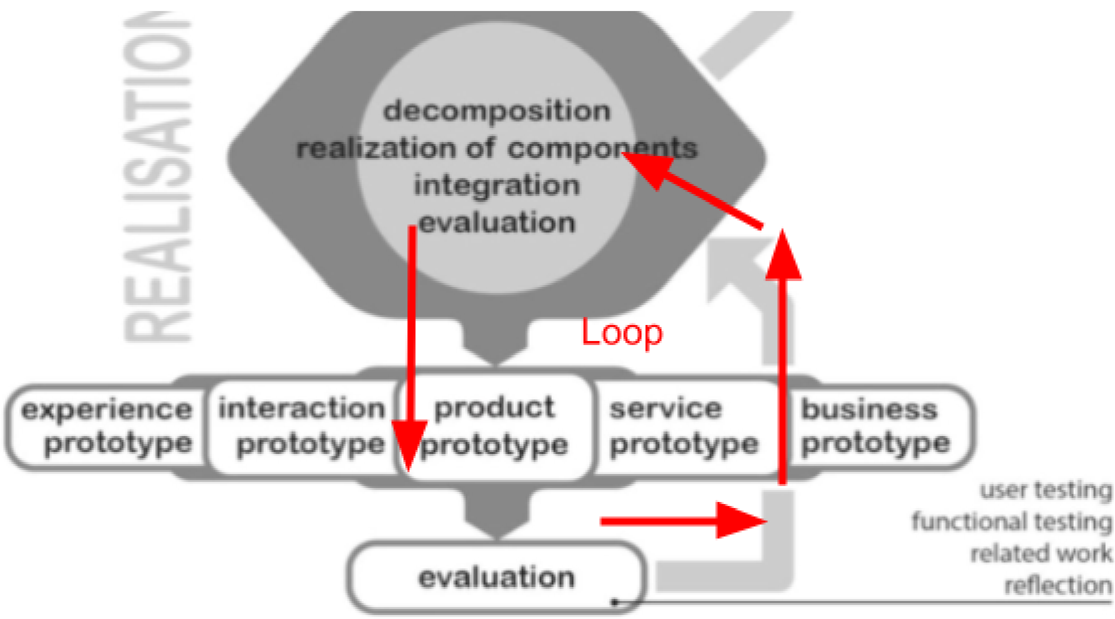


Figure 6.1 Evaluation and prototyping loop [41].

6.1 Low-fi prototype Evaluation

The evaluation focused on several key aspects, including participants' willingness to use the simulator to enhance their biking skills and reduce the fear of reinjury. Understanding the perceived value and potential adoption of the simulator among riders was crucial. Participants were also asked to share their experiences with the simulator, providing insights into any observed changes in skill levels, confidence, and suggestions for improvements for the hi-fi prototype development.

6.1.1 Methodology

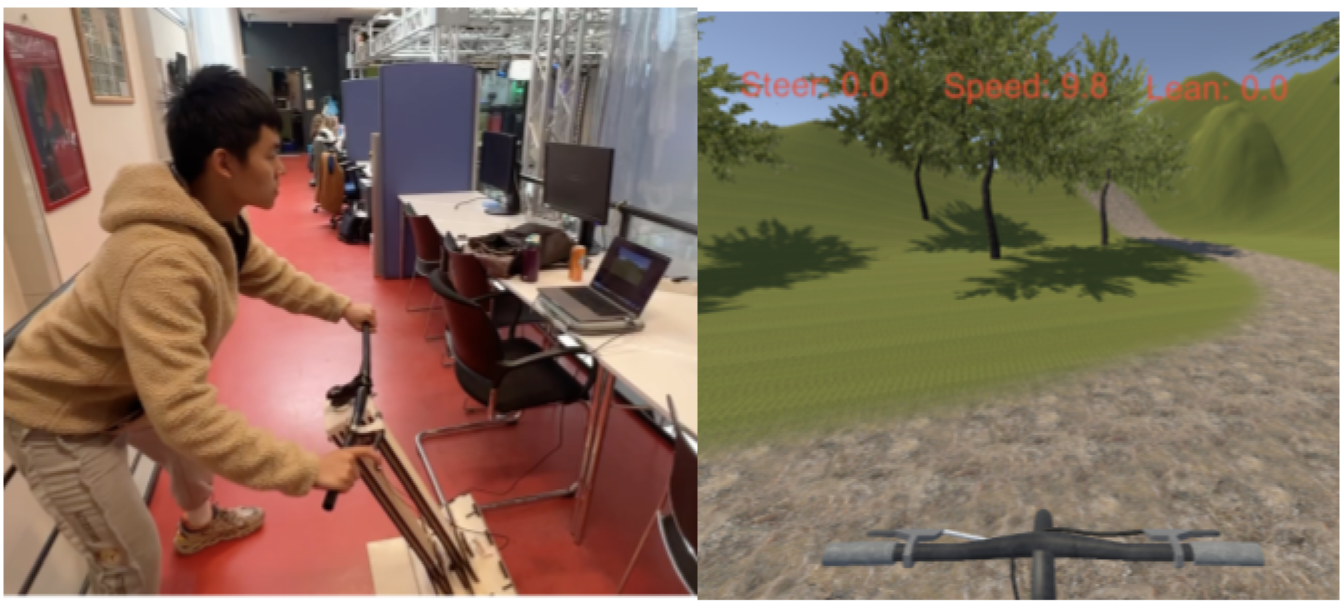


Figure 6.2 Low-fi game setup

During the evaluation, a user test was adopted. Three mountain bike riders with more than three years of riding experience who had been injured while cornering were invited to test the low-fi prototype. Participants were invited to engage with the simulator by trying the game using the specially designed controller (see Figure 6.2). Through qualitative open-ended questions, participants were encouraged to provide feedback on various aspects of the simulator, including its design, usability, and overall appeal.

Open questions will be asked from the participants, their feedback and suggestions will be carefully noted, and the process of using their answers will be followed by those aspects:

Data analysis

- Feedback and suggestions will be noted and listed for making an improvement plan.

Present the findings derived from the data analysis

- Address each research question and provide corresponding results.
- Summarize the key findings and outcomes of the evaluation.

Summarize the main findings and conclusions drawn from the evaluation

- Provide a clear improvement plan for the hi-fi prototype development.

6.1.2 Questions for low-fi prototype evaluation

Impression and performance

- What is your impression of this prototype? Do you like it?
- Provide feedback about specific aspects of the bike's behavior that you found realistic or unrealistic. (e.g., steering, leaning)
- Do you have any suggestions for improving the realism of the bike's behavior?

Training and rehabilitation

- Do you believe this mountain biking simulator can provide effective corner training and rehabilitation to reduce fear?
- Please provide feedback on specific aspects of the prototype that they think would improve cornering skills and overcome the fear of re-injury.

User interface and Controls

- Do you find the UI intuitive and easy to navigate?
- Do you think the map and terrain are well-designed? What is good, and what can be improved?
- What can be added and improved?

General questions

- Do you see the potential? Do you see yourself using what and why? And what can be improved?
- Do you feel more confident negotiating the corner after using this simulator?

6.2 Result from the lo-fi prototype evaluation

The suggestions to improve the low-fi prototype from the tested mountain bike riders are listed in Table 6.3.

Table 6.3: Test results from the lo-fi prototype evaluation.

Participants	Suggestion for controller	Suggestion for the terrain/UI	Suggestion for the game	Do you think it's helpful for cornering training?
Rider 1	A mark on the floor should indicate how far the user should step away from the controller. Steering is too flexible. It can have more resistance.	The map is quite small, and many corners are very tight and hard to pass. The UI is too simple and less informative.	The bike accelerates too quickly.	I think so. If the game can simulate traction under different conditions, like wet and dry, the system can be very helpful.
Rider 2	Brake lever feels too light, and it's better to have both brake levers. There should be a mark on the floor that shows how far the user should step away from the controller.	The map can be a single loop for starting the game. There can be many other levels with different terrain designs you can choose from.	System response that is a bit slow.	I see the potential of the system.

Rider 3	There should be a limitation for the steering angle instead of free spinning. There is no interaction with the lower body.	There should be more features to ride, like walls. Suggestion steering and the leaning angle should show on the screen.	The bike should not fall on the ground when you make a mistake, which can make the game frustrating.	I think it's very helpful because you will understand how the bike reacts to the steering and leaning input from riders.

6.2.1 Positive reflection from the test

During the testing phase, the game performed smoothly without any crashes, and the bike exhibited normal behavior by accurately responding to the users' steering and leaning inputs. The controller functioned effectively, allowing users to interact and engage with the game naturally. All participants showed active reactions throughout the testing process and expressed great excitement about the game setup. Despite the lo-fi prototype not offering a perfectly smooth bike control experience, all participants quickly adapted and learned how to use the controller within a short period of time effectively.

6.2.2 Limitations in low-fi prototype

During the testing phase, several issues and limitations were identified in addition to the expected normal behavior. Firstly, only one brake was attached to the handlebar, which led to abrupt deceleration and poor bike control when users applied the brake. Moreover, the potentiometer attached to the brake lever exhibited instability and inconsistency, resulting in unpredictable braking power and inadequate control in the game.

Participants also noted that the controller had some instability, occasionally moving on the floor when not applying downward pressure on the handlebar. Additionally, it was observed that participants stood too close to the controller initially, while the ideal distance was one meter away from their chest. They were also advised to lower their upper body for better control.

Another recurring issue was that participants often needed to remember to steer the handlebar back to the center position, leading to poor cornering performance during the tests. The layout of the terrain also posed challenges, with participants needing help to control the bike and navigate back to the main road.

Furthermore, due to the limited field of view, participants sometimes accelerated too quickly without noticing their speed, making them unable to brake in time before approaching a corner.

6.2.3 Summary of the low-fi prototype testing results

During the low-fi prototype testing phase, three mountain bike riders participated and provided feedback on the controller, terrain/UI, and overall gameplay. The participants had various suggestions for improvement in the controller, terrain, UI, and game setup.

Despite the identified areas for improvement, the testing phase went well overall. The game performed smoothly without any crashes, and participants were able to interact effectively with the controller. They showed enthusiasm and quickly adapted to using the controller.

6.3 Improve Plan and new requirements

- **controller:** The controller has been considered the highest priority in hi-fi prototype development. During the low-fi test, participants agree that the controller should have the highest impact on the game experience and feel as close to a real mountain bike as possible. Therefore, the following aspects need to be implemented.

    - There should be a rider position guide on the floor to help users understand where to place themselves to hold the handlebar at the right distance from the controller.
    - The steering range should be limited to -90 degrees to 90 degrees to prevent oversteering from the user. Meanwhile, steering resistance and self-return function should be added to the handlebar.
    - Two brakes should be added.
    - The brake lever should provide resistance for a better brake lever feeling.

- **Terrain:** The improvement of the terrain design takes the second priority. In the low-fi test, participants report that the terrain directly impacts the game experience since the riders' control over the bike depends on the terrain, a terrain similar to the real-life bike trails can enhance the user experience. Participants suggested that the game's terrain design can refer to real-life mountain bike trails or parks. Therefore, the following aspects need to be implemented.
    - The Terrain should have more features. Possible features like slope and curved walls can be added.
    - Multiple ground surfaces should be considered
    - Terrain should be different between levels
    - It should have clear start and end points

- **Game system:** Participants report that the bike is difficult to control during the lo-fi test. Users have to wait quite a long time to control the bike before it usually behaves again, which leads to a frustrating experience. Thus, improvement should be considered in the aspects of bike performance and interactive feedback.
    - The bike should accelerate slower
    - Brakes should be more effective.
    - The wheels need adjustments to provide better traction and damping.

- **UI:** Participants agree that this game should introduce a rank system with counting score and runtime to track user performance.
    - Score and time system should be available.
    - Should have a menu.
    - Recommended steering and the leaning angle should show on the screen.
    - End game state should be available.

In conclusion, the low-fi prototype evaluation provides clear guidance to improve the whole system and functionality of the prototype. These recommendations will be instrumental in the development of the hi-fi prototype.

Chapter 7 – Improvement of Hi-fi Prototype Development

This chapter will focus on the development process of the hi-fi prototype, outlining its various stages. Additionally, it presents a detailed account of the final evaluation test, test setup, participant selection, evaluation methods, and test results.

7.1 Improvements of the game controller

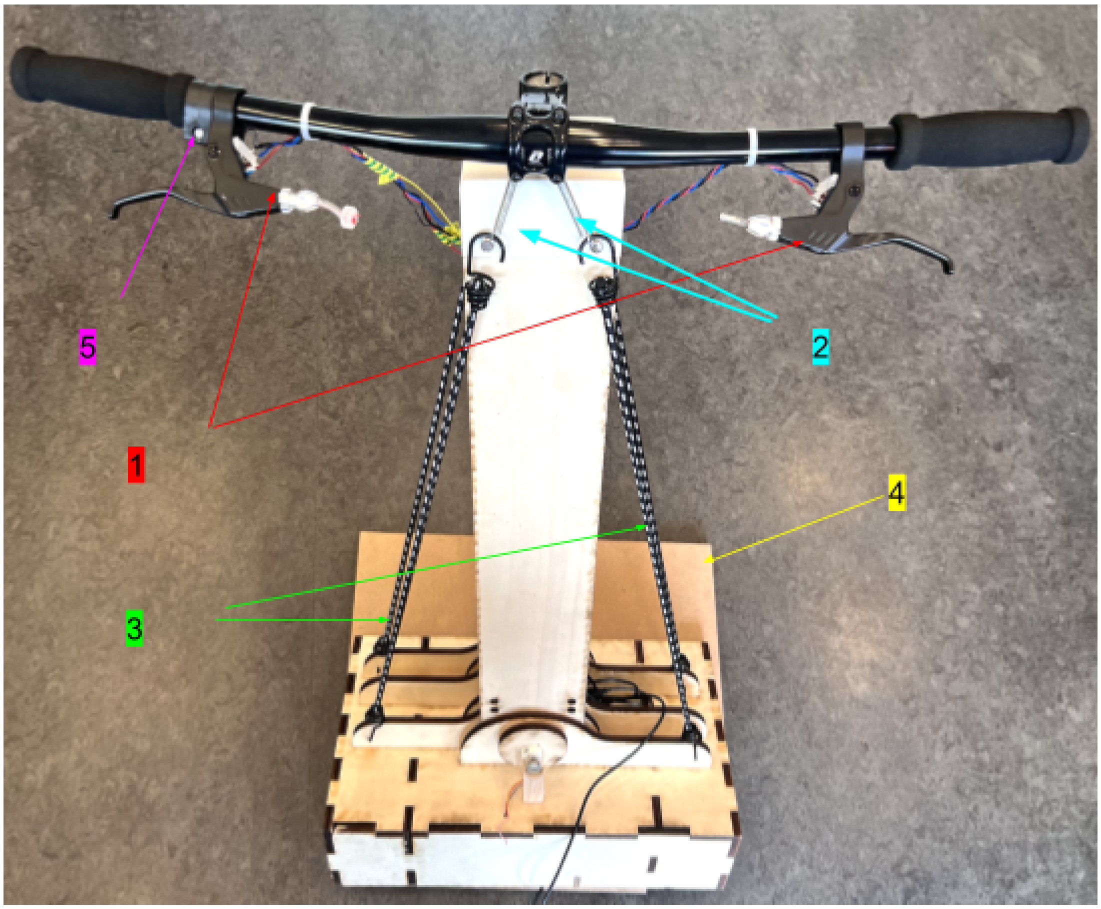


Figure 7.1: The improved prototype

Figure 7.1 shows the finished hi-fi version of the controller, with the changes in the functions, optimized steering and leaning feedback, more stable sensors, and base. The improvement detail will be explained in the following sections. The improved parts are listed in Table 7.2.

Table 7.2: Improved parts in hi-fi prototype.

Part number	**Part name**

1	Brake levers with limited pull travel
2	Coil spring pair for handlebar self-return
3	Elastic strings for self-return leaning
4	Base extension plate
5	Acceleration button

7.1.1 Improvement of the steering control

Figure 7.3: Self-return and rotate limitation achieved by adding two coil springs.

Figure 7.4: limited range of steering angle (45 degrees).

Based on the test results from the low-fi evaluation, the overall build quality and functionality of the controller have been improved to meet the requirements of user-friendliness and easy access. Figure 7.3 shows two coil springs connected to the handlebar stem using two bolts. This change was implemented to address the issue of the handlebar lacking a rotation range limitation and steering feedback. With this modification, users can experience steering resistance, as the range has been set to 45 degrees (see Figure 7.4). The springs also enable the handlebar to return to the neutral position, pointing forward, ensuring a smoother operation.

Figure 7.5: The accelerate button

Figure 7.5 shows that the acceleration button has been added on the right-hand side of the handlebar. It allows users to speed up the bike when needed. The previous low-fi prototype used consistent acceleration input, which led to a difficult control and undesired experience. This setting has been removed and taken over by this acceleration button to provide a better game control experience.

7.1.2 Improvements of the brake lever

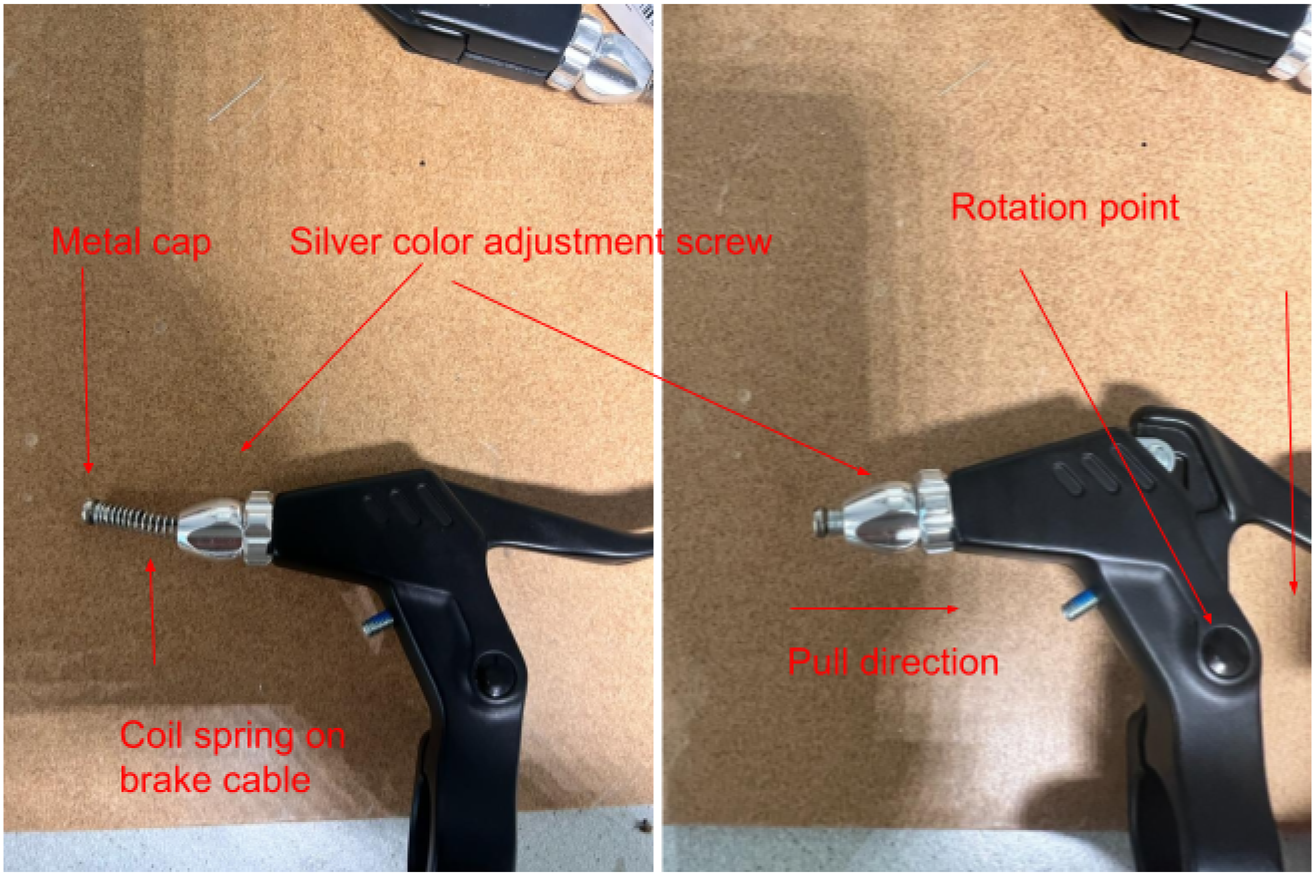


Figure 7.6: Adding a cable and coil to the brake to create a bit-point to the brake lever and proved self-return function.

The improved brake lever follows the same principle of attaching the potentiometer beneath the lever's rotation point. Two adapters have been utilized. These adapters serve the dual purpose of securely clamping the potentiometer shaft and the bolt on the brake lever.

The second modification involves the brake cables. During actual brake operation, the brake lever should pull the cable, causing the brake pads to make contact with the braking disc or rim. This action creates a distinct bite point that riders can feel. In the low-fi test, participants noted that the brakes did not feel realistic due to the absence of a bite point.

A shortened brake cable has been implemented (see Figure 7.6). The cable length has been cut to approximately 1 cm beyond the cable adjustment screw on the lever. One end of the cable is fixed inside the brake lever, while the other end is secured between two components using a larger-diameter metal cap that encloses the silver adjusting screw. Additionally, a coil spring is positioned between the metal cap and the silver adjusting screw.

This design allows for the desired effect: when the user pulls the brake lever, the lever pulls the brake cable. As the far end of the brake cable encounters the collision between the metal cap and the adjustment screw, it is prevented from further movement. Consequently, the brake lever cannot be pulled any further. By implementing this mechanism, the goal of creating a bite point in the brakes and the sensation of resistance while pulling the brakes are simultaneously achieved.

7.1.3 Improvements of the game terrains

Considering the results of the lo-fi prototype test, it was deemed essential to incorporate two game levels featuring distinct terrains. The first level, designed as an easier terrain with wide turns, serves as a practice ground to familiarize users with the game controller. Conversely, the second level consists of a challenging downhill terrain with twisting corners, intended for competitive play where time and score are recorded.

The easier practice terrain map has been constructed in a confined environment (see Figure 7.7), incorporating ground indicators strategically placed to guide players. Those indicators show the ideal line choices for those corners. Following the indicators will allow users to complete the corners smoothly without losing too much speed. Thus, participants are encouraged to follow these indicators while practicing to help them understand the cornering line-picking strategies for different types of corners.

Figure 7.7: Practice terrain with guide indicators.

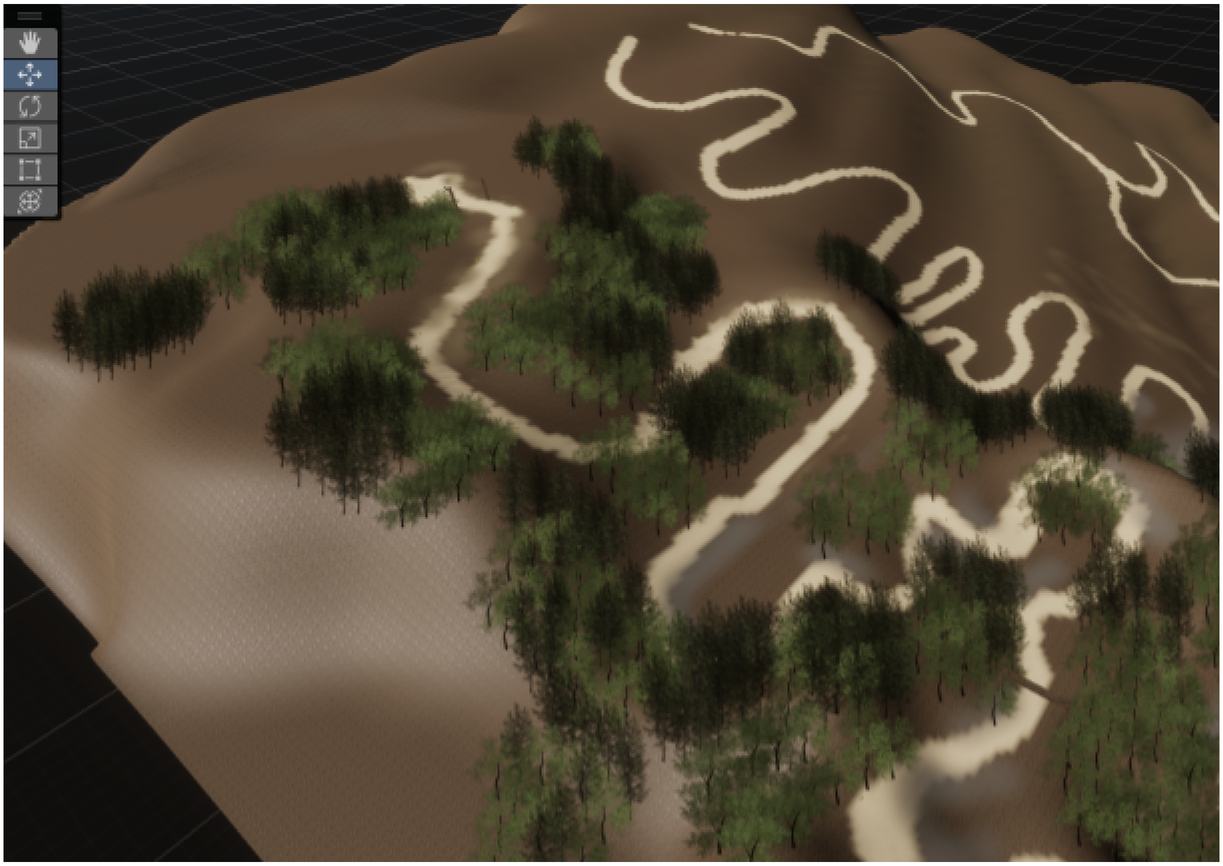


Figure 7.8: Terrain in rank level

The rank terrain map (see Figure 7.8) is constructed within an expansive game environment devoid of cornering guide indicators. This terrain represents a downhill setting, allowing the bike to accelerate at a higher rate than the practice level. The corners on this map are sharper, resembling the terrain in real mountain bike parks.

The rank terrain is designated to have a score and time system. Coins are strategically placed along the optimal cornering path (see Figure 7.9) to encourage players to follow the optimal path to complete the turn and enhance their game experience.

Figure 7.9: Coin on the path

7.1.4 Improvements for the user interface

The game perspective is set to a first-person perspective while on a bicycle (see Figure 7.9). At the same time, various parameters in the game, such as friction, bicycle center of gravity, acceleration, and deceleration, have significantly been adjusted to make the operating experience closer to the actual riding experience.

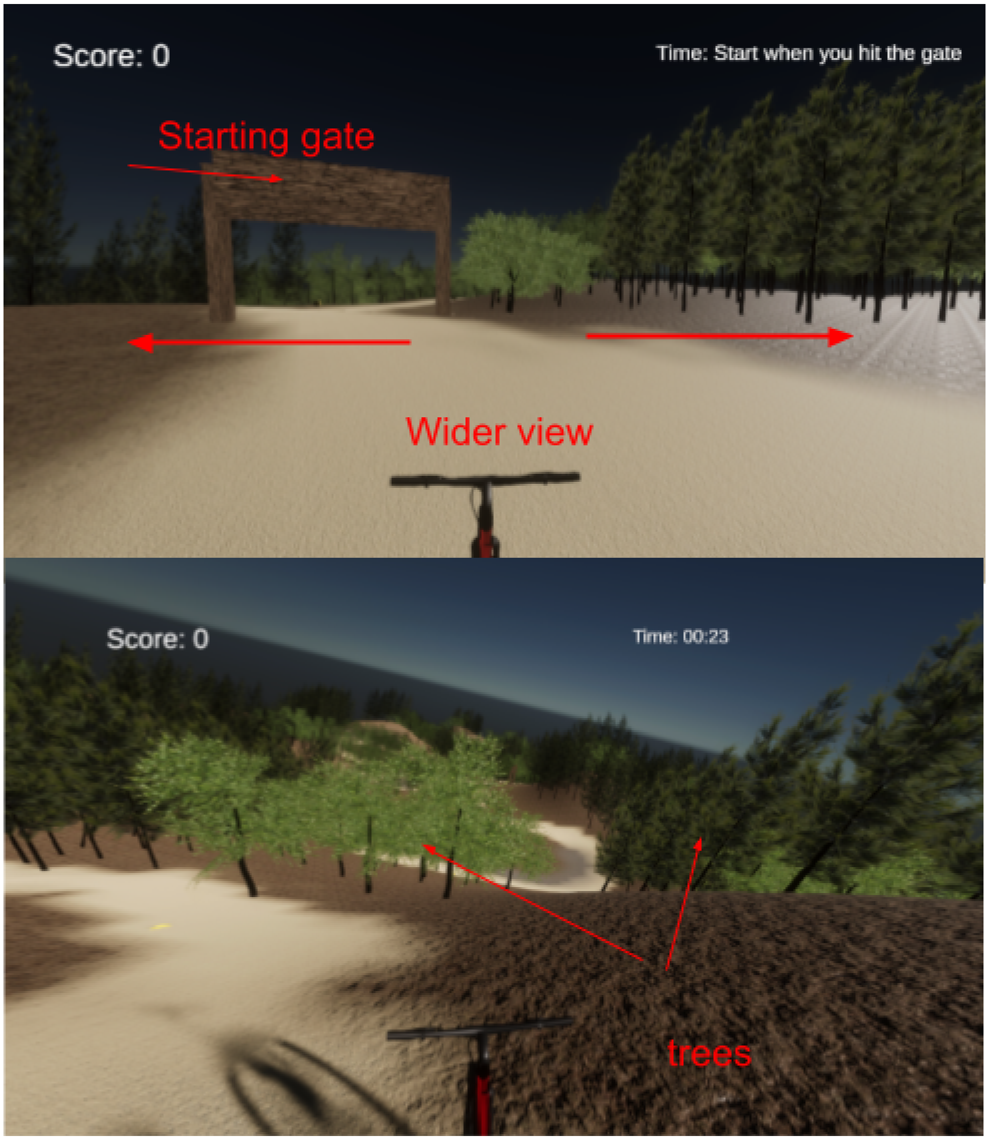


Figure 7.10: The updated view and scene setup.

Once users cross the finish line, their score and run-time will be displayed at the end game state (see Figure 7.11). Which can provide timely result reports, and the ranking system can increase their motivation.

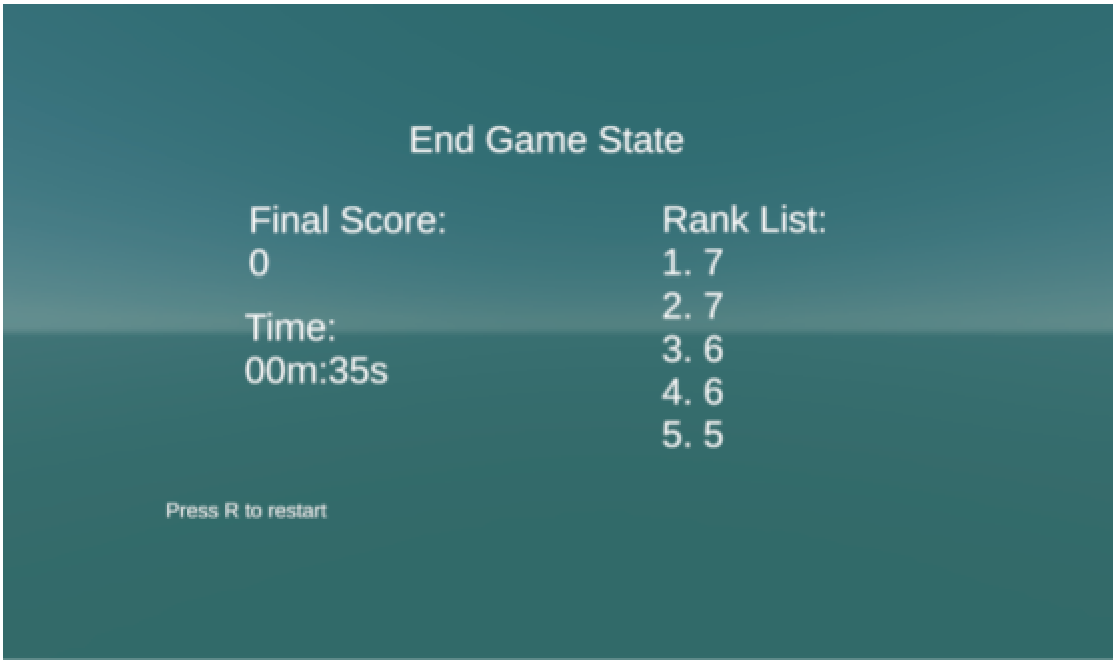


Figure 7.11: End game state shows the score, run-time, and rank position.

7.1.5 Improvements in parameter settings

When looking for ideal variable values, it was found that achieving a satisfactory level of adjustment for material friction, elasticity, and looseness was difficult. Moreover, in user research, Someone mentioned that a large part of the handling of corners depends on the driver's perception and judgment of the grip of different road surfaces. In order to achieve a more satisfactory simulation experience, the road surface of the hi-fi prototype is finally set to simulate a dry hard sandy soil road surface, and its settings in the game can be found in Table 7.11.

Table 7.11: The ground Material Settings.

Property	Function	Used value in the game
Dynamic friction	The friction used when already moving. Usually, a value from 0 to 1. A value of zero feels like ice. A value of 1 will make it come to rest very quickly unless a lot of force or gravity pushes the object	1 (high friction level)

Static friction	The friction used when an object is lying still on a surface. Usually, a value from 0 to 1. A value of zero feels like ice. A value of 1 will make it very hard to get the object moving.	0.9 (high friction level)
Bounciness	A value of 0 will not bounce. A value of 1 will bounce without any loss of energy.	0 (no bounciness).

Meanwhile, the **Angular Drag** of the bike and the **Acceleration** have also been set to an optimal value to prevent unpredictable rotation and loss of control. The description of the setting can be found in Table 7.12.

Table 7.12: The setting on drags and acceleration of the bike.

Property	Function	Picked Value
Angular drag	The angular drag applies to rotational movement and is set up separately from the linear drag that affects positional movements. A higher value of angular drag will cause an object's rotation to come to rest more quickly following a collision or torque.	10 (low angular drag)
Acceleration	How many newtons of torque are applied to the object	30 (low acceleration value)

Chapter 8 - Hi-fi prototype user evaluation

Figure 8.1 shows the user test procedure for the hi-fi prototype evaluation. The test involved five mountain bike riders who had a history of injuries and expressed fear of cornering, as identified in the user research conducted in **section 2.4.** This targeted selection aimed to gather valuable insights from individuals who could provide unique perspectives on addressing cornering challenges.

8.1 Testing method and test environment

Figure 8.1: Hi-fi prototype tested by participants.

Participants were recruited and formally invited to participate, emphasizing the importance of their feedback in improving the prototype's design. Before the evaluation, participants received a detailed orientation about the hi-fi prototype's purpose, features, and evaluation goals. A demonstration of the prototype was conducted to ensure participants understood its functionality.

The hi-fi prototype evaluation begins with user guidance and an introduction to the functions of the bike controller. Initially, the tester verbally explains the controller's functions and provides an overview of the game system. Subsequently, a brief demonstration is conducted to ensure participants better understand the system's working principles. Once the demonstration is complete, the controller is handed over to the participants.

The evaluation process involves participants engaging in a practice model set in the practice terrain. During this phase, participants familiarize themselves with the accelerator button and two brake controls and learn how to steer and lean. After the tutorial's completion, participants are allowed to practice in the practice model and can transition to the rank model upon request.

In the rank model, participants are instructed to complete a level by following a designated path. They are encouraged to collect coins to accumulate a higher score and aim to finish the level quickly. Subsequently, participants are asked to provide feedback by responding to questions regarding their overall impression of the setup, their gaming experience, and their personal thoughts.

This evaluation process allows for comprehensive feedback gathering, ensuring participants' experiences and perspectives are considered for further analysis and improvement of the hi-fi prototype. The following formed questions were asked during the after-test interview.

Interview Questions

1. Could you share your overall impressions and experiences of this game setup?

2. What are your thoughts on the design in terms of user-friendliness and aesthetics?

3. Do you think the idea behind this simulator is innovative and unique? Please explain.

4. How do you assess the potential of this simulator in improving your cornering skills? If so, in what ways?

5. Can you elaborate on any training value you found using this simulator?

6. Do you think using this simulator can reduce your fear of reinjury while cornering?

7. Do you see yourself using this setup? Why and why not?

8. What improvements or changes would you suggest to improve this simulator?

9. Would you recommend this game setup to other mountain bikers? Why or why not?

8.2 Test result

The test result shows that all five participants were interested in using the simulator and had positive comments on the design and the concept of practicing cornering skills. From the results shown in Table 8.2, participants commented that the simulator's appearance, design, and user experience are close to real mountain bikes. Additionally, using real mountain bike handlebars is helpful for this experience. However, two out of five participants mentioned that the speed of the bicycle in the game is difficult to perceive, which will cause mistakes in corners due to excessive speed.

Table 8.2: Interview results about controllers and game impressions.

Participant number	**Overall impression of the controller**	**Impression of the game**
1	The simulator looks good, and the design is unique and conforms to the shape of the bicycle handlebar.	The game control was quite realistic, but the speed was difficult to perceive.
2	Steering feedback feels real.	The speed is hard to feel, which makes some corners hard to turn.
3	Steering and leaning are realistic, and the brakes work well. The design seems unique.	The game is fun. The coin system helps to improve the user's awareness of line pickings.
4	The controller is easy to use since it looks and moves like a real bike handlebar.	The game and terrain design are fun. It made me want to ride the bike.
5	It is a high-quality prototype, and the self-return system is helpful for providing steering feedback. But there was no vibration on the handlebar.	It is easy to use, and the operation is very smooth, but the terrain design is rather simple and should be improved.

As for user-friendly reports (see Chart 8.3), all participants agreed that this simulator's appearance and functional design are close to the actual mountain bike, which requires little study effort and is considered user-friendly. However, it is worth mentioning that one participant mentioned that the acceleration button on the handlebar is too small and feels too light, which is easy to touch accidentally and affects the gaming experience.

Chart 8.3: Do you think this simulator is user friendly?

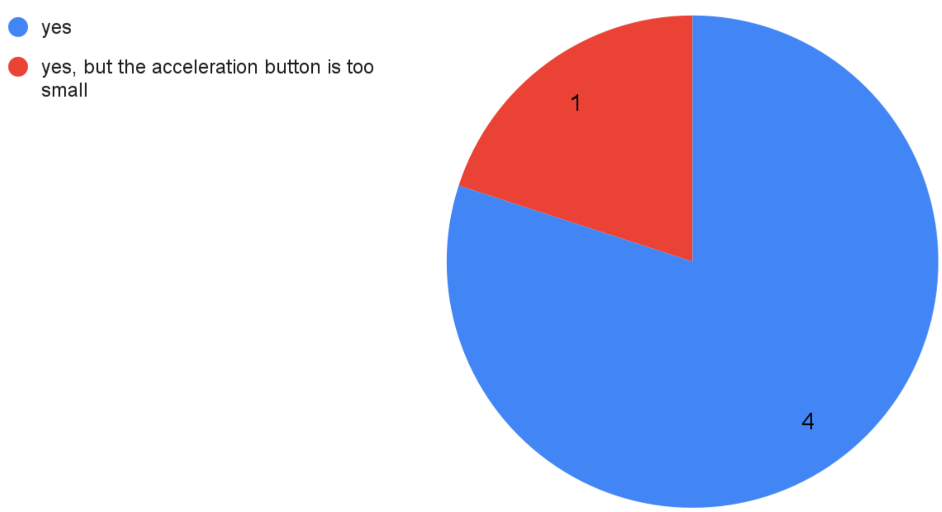


Table 8.4: The interview results of the training value and fear reduction effectiveness test.

Participant number	**Thoughts on the value of training**	**Thoughts on the effectiveness of fear reduction at cornering**
1	This device is helpful for training on speed control when cornering.	It has the potential to fear reduction training. This device allows user training in a risk-free environment, which can help improve overall user confidence and reduce fear.
2	It can help the riders improve their awareness of leaning the bike while cornering and	The direct effect is not obvious, but the increased motivation can greatly contribute to the fear level reduction.

	practice general cornering skills. Most importantly, it can greatly improve the motivation for training.	Decreased riding and training motivation is prevalent. Using this simulator revises the joy of biking, and the riding motivation is back.
3	Thanks to the realistic feel of steering and leaning, it can help riders train for their cornering skills.	The fear reduction effect could be more sensible. The current simulator does not include the ground feedback to the user, which cannot reproduce the scary feeling when losing traction to users.
4	Valuable on the speed control training.	There is no direct impact on fear reduction, but it has the potential to improve the skills and confidence level to help the riders reduce the fear of re-injury.
5	There is great potential for training the control of the front and rear wheel brakes and the timing of acceleration and deceleration.	Practicing leaning and speed control can increase the confidence of the riders, which is very helpful in reducing fear, and this real riding experience can greatly improve the driving force of riding and training.

Table 8.4 shows all participants' answers regarding the mountain bike simulator's training value and fear reduction effectiveness. Participants agree that this simulator effectively enhances their training and riding motivation, and they think it has the potential for cornering skills and speed control training.

Regarding the influence on fear reduction, four out of the five participants noted that they did not observe an immediate impact on their fear levels. However, they acknowledged that fear reduction is a long-term process influenced by various factors.

Table 8.5: Answers from participants about the use value.

Participant number	Do you see yourself using this setup?
Participant 1	Yes, the cornering skills training can help me apply them in real-life riding.
Participant 2	Yes, it can help me improve my cornering skills.
Participant 3	Yes, using this simulator during my recovery can reduce my stress level.
Participant 4	Yes, I prefer to use this simulator when I am injured.
Participant 5	Yes, even if I am not injured, I can use it during the off-season when the weather is not suitable for mountain biking.

At the end of the test, participants were asked whether they see themself using this simulator. The answers from all five participants were positive (see **Table 8.5**). They agree that this simulator can be an effective indoor training device, In addition to its influence on cornering skills and maintaining training motivation. One participant mentioned that mountain bikers could use the device even if they are not injured, such as being unable to bike outside due to external reasons, especially during the off-season when the weather is not suitable for mountain biking.

Table 8.6: Suggestion to improve the simulator.

Objects	Suggestions of improvements
Game setting	Add a motion blur effect for a better sense of speed.
Controller	Add an adjustable length function to the stand.
Terrain design	Add a pointer to the finish line.
Controller	Vibration can be considered as an additional interaction.

In order to further improve the design, participants were asked to provide suggestions on how to improve the prototype to provide a promised user experience and fill the requirements in fear reduction training. In all the responses, the participants gave their opinions and new requirements on the game settings, visual effects, and controller design (see Table 8.6). They believe that the most critical factors for improving the game's realism are road grip, bumpiness, and feedback from the bike speed. Other requirements such as adding multiple maps and terrain features, can improve its game experience, but these are secondary to visual and physical feedback.

Chapter 9 – Discussion & Future Work

This study aimed to evaluate the effectiveness of a mountain bike simulator in reducing the fear of re-injury for injured mountain bike riders. The results from the user tests shed light on the potential of the simulator as a valuable training tool and its impact on fear reduction. Even though the results did not directly influence reducing the fear of re-injury, since it requires long-term observation, they provide important insights into the psychological and motivational aspects of using the simulator.

The simulator's role in improving training and riding motivation emerged as a critical aspect. Participants reported that the simulator reproduced the mountain biking experience, which increased their training and riding motivation. Participants agree that this simulator can potentially reduce fear but currently cannot be approved. Those findings highlight the interplay between skill development and training motivation, suggesting that the simulator is vital in promoting positive psychological factors among riders to help them possibly reduce the fear of re-injury in the future.

However, it is crucial to recognize the limitations inherent in this study. The test group size was small, comprising only five participants, which limits the generalizability of the findings to the broader mountain biking community. Additionally, all participants had a personal connection with the research, which introduces the possibility of biased or less objective test results. It is imperative for future studies to address these limitations by involving a larger and more diverse participant pool to ensure greater representativeness and minimize potential biases associated with personal connections to the research.

Furthermore, it is important to acknowledge potential issues with the chosen testing method. The current study solely relies on post-experience interviews as the testing approach, which introduces concerns about the accuracy of participants' recollections and the variability of their emotional state during the interviews. Moreover, this method can be time-consuming, costly, and potentially emotionally taxing for participants. Considering these limitations, future studies should explore additional testing methods that provide more objective and reliable data while minimizing the burden on participants.

The final test was limited in diversity as it solely included male participants, thereby lacking results from female users. Additionally, the age range of participants was limited to individuals between 20 and 40 years old, which hampers the study's representation of age differences. Future studies will aim to involve a more extensive and diverse sample of users. This will ensure a more comprehensive understanding of the simulator's performance and validate its effectiveness across various user demographics.

For the limitation of the design, one participant raised concerns about the simulator's current limitations in reproducing the feeling of losing traction and the associated fear. The absence of ground feedback was identified as a potential area for improvement to enhance fear reduction. Incorporating realistic feedback mechanisms in future iterations of the simulator could provide a more immersive experience and further aid in addressing fear-related issues.

Furthermore, based on the test results, it is evident that users' satisfaction with the game experience is directly proportional to the simulation realism. Therefore, future improvements should prioritize enhancing the simulation feedback. The current design uses a normal screen for game display since VR technology has been proven to provide users with a realistic experience and has been used in sports such as football and boating. Therefore, in future improvements, the application of VR will be regarded as the primary goal, with secondary goals such as improving terrain designs, adding motion blur to improve the sense of speed, and sound feedback to improve realism and enhance the user experience.

In summary, the current study has identified several product design, research methodology, and testing approach limitations. Furthermore, future work will prioritize addressing these limitations by incorporating feedback received, making design improvements, conducting more focused and specific research, and implementing wider, more diverse, and more detailed user testing protocols. These efforts aim to enhance the overall quality and reliability of the findings.

Chapter 10 – Conclusion

In conclusion, this research project has highlighted the potential of a mountain bike simulator in addressing the fear of reinjury among mountain bikers. The simulator offers a controlled and immersive training experience that helps riders enhance their cornering skills and boosts their training motivation.

Although the current results did not directly indicate a significant impact of the simulator on fear reduction specifically related to cornering, it is important to note that overcoming the fear of reinjury is a long-term process requiring comprehensive and professional training involving multiple parties. Therefore, the effectiveness of this mountain bike simulator should not be dismissed outright. Instead, it demonstrates its capability to increase training and riding motivation for injured mountain bikers.

Future research will focus on refining game and controller designs, optimizing research and testing methodologies, and fostering collaboration among researchers, trainers, and riders. These efforts aim to unlock the full potential of reducing the fear of re-injury in mountain biking and further enhance the efficacy of the mountain bike simulator.

References

[1] W. Schrader, “Spinal Cord Injury in mountain biking: What is the risk?” *Idaho Sports Medicine Institute*, 05-Jul-2022. [Online]. Available: https://www.idsportsmed.com/spinal-cord-injury-in-mountain-biking-whats-the-risk/. [Accessed: 01-May-2023].

[2] D. C. Flanigan, J. S. Everhart, A. Pedroza, T. Smith, and C. C. Kaeding, “Fear of reinjury (kinesiophobia) and persistent knee symptoms are common factors for lack of return to sport after Anterior Cruciate Ligament Reconstruction,” *Arthroscopy: The Journal of Arthroscopic & Related Surgery*, vol. 29, no. 8, pp. 1322–1329, 2013.

[3] Hsu C-J, Meierbachtol A, George SZ, Chmielewski TL. Fear of Reinjury in Athletes: Implications for Rehabilitation. Sports Health. 2017;9(2):162-167. doi:10.1177/1941738116666813

[4] C.-J. Hsu, A. Meierbachtol, S. Z. George, and T. L. Chmielewski, “Fear of reinjury in athletes,” Sports Health: A Multidisciplinary Approach, vol. 9, no. 2, pp. 162–167, 2016.

[5] Ehn, Meredith, et al. “The National Interscholastic Cycling Association (NICA) Mountain Biking Injury Surveillance System (ISS): Analysis of 66,588 Student Athlete-Years of Injury Data.” *International Journal of Environmental Research and Public Health*, vol. 18, no. 11, 29 May 2021, p. 5856, https://doi.org/10.3390/ijerph18115856. Accessed 28 Sept. 2022.

[6] C. D. Rojas Ferrer, H. Shishido, I. Kitahara, and Y. Kameda, “Read-the-game: System for skill-based visual exploratory activity assessment with a full body virtual reality soccer simulation,” *PLoS One*, vol. 15, no. 3, p. e0230042, 2020.

[7] M. Isogawa et al., "What Can VR Systems Tell Sports Players? Reaction-Based Analysis of Baseball Batters in Virtual and Real Worlds," 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Tuebingen/Reutlingen, Germany, 2018, pp. 587-588, doi: 10.1109/VR.2018.8446073.

[8] Hämäläinen, P. (2004, October). Interactive video mirrors for sports training. In Proceedings of the third Nordic Conference on Human-computer Interaction (pp. 199-202).

[9] E. W. Trejo and P. Yuan, "Recognition of Yoga Poses Through an Interactive System with Kinect Device," 2018 2nd International Conference on Robotics and Automation Sciences (ICRAS), Wuhan, China, 2018, pp. 1-5, doi: 10.1109/ICRAS.2018.8443267.

[10] M. M. Jensen, M. K. Rasmussen, and K. Grønbæk, "Design sensitivities for Interactive Sport-Training Games," Proceedings of the 2014 conference on Designing interactive systems, 2014.

[11] M. Ludvigsen, M. H. Fogtmann, and K. Grønbæk, "Tactowers," Proceedings of the 8th ACM Conference on Designing Interactive Systems, 2010.

[12] S. Rauter, "New approach for planning the mountain bike training with virtual coach - Wielkopolska Biblioteka Cyfrowa," Strona główna - Wielkopolska Biblioteka Cyfrowa, 06-Jul-2018. [Online]. Available: https://www.wbc.poznan.pl/dlibra/publication/536132/edition/443839/content. [Accessed: 25-Apr-2023].

[13] D. H. Kotler, A. N. Babu, and G. Robidoux, "Prevention, evaluation, and rehabilitation of cycling-related injury," *Current Sports Medicine Reports*, vol. 15, no. 3, pp. 199–206, 2016.

[14] B. Clarsen, T. Krosshaug, and R. Bahr, "Overuse injuries in professional road cyclists," *The American Journal of Sports Medicine*, vol. 38, no. 12, pp. 2494–2501, 2010.

[15] S. Apsingi, C. U. Dussa, and B. M. Soni, "Acute cervical spine injuries in mountain biking," *The American Journal of Sports Medicine*, vol. 34, no. 3, pp. 487–489, 2006.

[16] M. P. Schwellnus and E. W. Derman, "Common injuries in cycling: Prevention, diagnosis and management," *South African Family Practice*, vol. 47, no. 7, pp. 14–19, 2005.

[17] C;, P.D.-J.G. (no date) *Enduro World Series (EWS) mountain biking injuries: A 2-year prospective study of 2010 riders*, *International journal of sports medicine*. Available at: https://pubmed.ncbi.nlm.nih.gov/33348388/ (Accessed: 05 June 2023).

[18] *Acting against the fear of re-injury* (no date) *Sports Injury Bulletin*. Available at: https://www.sportsinjurybulletin.com/diagnose--treat/acting-against-the-fear-of-re-injury (Accessed: 06 June 2023).

[19] J. J. Hamson-Utley and L. Vazquez, "The comeback: Rehabilitating the psychological injury," *Athletic Therapy Today*, vol. 13, no. 5, pp. 35–38, 2008.

[20] D. C. Flanigan, J. S. Everhart, A. Pedroza, T. Smith, and C. C. Kaeding, "Fear of reinjury (kinesiophobia) and persistent knee symptoms are common factors for lack of return to sport after Anterior Cruciate Ligament Reconstruction," *Arthroscopy: The Journal of Arthroscopic & Related Surgery*, vol. 29, no. 8, pp. 1322–1329, 2013.

[21] C.-J. Hsu, A. Meierbachtol, S. Z. George, and T. L. Chmielewski, "Fear of reinjury in athletes," Sports Health: A Multidisciplinary Approach, vol. 9, no. 2, pp. 162–167, 2016.

[22] F. R. Connolly, L. M. Aitken, and M. Tower, "An integrative review of self-efficacy and patient recovery post acute injury," *Journal of Advanced Nursing*, vol. 70, no. 4, pp. 714–728, 2013.

[23] Francis SR, Andersen MB, Maley P. Physiotherapists' and male professional athletes' views on psychological skills for rehabilitation. J Sci Med Sport Sports Med Aust. 2000;3:17-29

[24] Hamson-Utley J, Vazquez L. The comeback: rehabilitating the psychological injury. Athl Ther Today. 2008;13(5):35-38.

[25] Woods MP, Asmundson GJ. Evaluating the efficacy of graded in vivo exposure for the treatment of fear in patients with chronic back pain: a randomized controlled clinical trial. Pain. 2008;136:271-280

[26] Rees T, Mitchell I, Evans L, Hardy L. Stressors, social support and psychological responses to sport injury in high- and low-performance standard participants. Psychol Sport Exerc. 2010;11:505-512.

[27] B. Lopes and L. McCormack, *Mastering mountain bike skills 3rd edition*, 3rd ed. Champaign, IL: Human Kinetics, pp.87-89, 2017.

[28] "At Home Cycling & Running Virtual Training & Workout Game App," *Zwift*. [Online]. Available: https://www.zwift.com/ca. [Accessed: 04-May-2023].

[29] "Get faster with TrainerRoad," *TrainerRoad*. [Online]. Available: https://www.trainerroad.com/. [Accessed: 04-May-2023].

[30] Jung Hwan Jeong et al., "The Development of a New Training System for Improving Equilibrium Sense Using a Virtual Bicycle Simulator," 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, Shanghai, 2005, pp. 2567-2570, doi: 10.1109/IEMBS.2005.1616993.

[31] R. Ranky, M. Sivak, J. Lewis, V. Gade, J. E. Deutsch and C. Mavroidis, "VRACK — virtual reality augmented cycling kit: Design and validation," 2010 IEEE Virtual Reality Conference (VR), Boston, MA, USA, 2010, pp. 135-138, doi: 10.1109/VR.2010.5444798.

[32] Henry, M.I. (no date) *Identifying the interest in virtual reality as an injury rehabilitation program amongst collegiate athletes*, *CORE*. Available at: https://core.ac.uk/display/323397697 (Accessed: 02 July 2023).

[33] C. Sun and Z. Qing, "Design and construction of a virtual bicycle simulator for evaluating sustainable facilities design," Adv. Civ. Eng., vol. 2018, pp. 1–10, 2018.

[34] Jong Yun Kim, Chul Gyu Song and Nam Gyun Kim, "A new VR bike system for balance rehabilitation training," Proceedings Seventh International Conference on Virtual Systems and Multimedia, Berkeley, CA, USA, 2001, pp. 790-799, doi: 10.1109/VSMM.2001.969747.

[35] F. Grani and J. R. Bruun-Pedersen, "Giro: better biking in virtual reality," 2017 IEEE 3rd Workshop on Everyday Virtual Reality (WEVR), Los Angeles, CA, USA, 2017, pp. 1-5, doi: 10.1109/WEVR.2017.7957711.

[36] 4DmIn and L. U. M. A.-iD. Ltd, "Danny Macaskill's VR Ride experience - luma-ID," LUMA, 25-Apr-2022. [Online]. Available: https://www.luma-id.co.uk/danny-macaskills-vr-ride-experience-2/. Accessed: 20-Apr-2023.

[37] R. Ranky, M. Sivak, J. Lewis, V. Gade, J. E. Deutsch and C. Mavroidis, "VRACK — virtual reality augmented cycling kit: Design and validation," 2010 IEEE Virtual Reality Conference (VR), Boston, MA, USA, 2010, pp. 135-138, doi: 10.1109/VR.2010.5444798.

[38] Rimer, E., Husby, L.V. and Solem, S. (2021) *Virtual reality exposure therapy for fear of heights: Clinicians' attitudes become more positive after trying Vret*, *Frontiers*. Available at: https://www.frontiersin.org/articles/10.3389/fpsyg.2021.671871/full (Accessed: 02 July 2023).

[39] F. Schramka, "Development of a virtual reality cycling simulator," *Journal of Computers*, pp. 603–605, 2018.

[40] J. Mittelstaedt, J. Wacker, and D. Stelling, "Effects of display type and motion control on cybersickness in a virtual bike simulator," Displays, vol. 51, pp. 43–50, 2018

[41] Ref. Mader A.H. and Eggink W., “A Design Process For Creative Technology”, Proceedings of the 16th International Conference on Engineering and product design education, Enschede, The Netherlands, 2014, pp. 568-573

[42] “What is the MoSCoW Method?,” SearchSoftwareQuality. https://www.techtarget.com/searchsoftwarequality/definition/MoSCoW-method#:~:text=The%20 MoSCoW%20method%20is%20a

[43] A. L. Mendelow, “Association for Information Systems AIS Electronic Library (AISeL) Environmental Scanning-The Impact of the Stakeholder Concept”, Accessed: Apr. 20, 2022. [Online]. Available: http://aisel.aisnet.org/icis1981/20

[44] J. Darlow, “Mountain Bike Geometry explained,” *MBR*, 15-Feb-2021. [Online]. Available: https://www.mbr.co.uk/news/mountain-bike-geometry-326498. [Accessed: 04-May-2023].

[45] *Retrospec Ascent mountain bike - 27.5" - matte silt 2022* (no date) *Playtri Westminster*. Available at: https://playtriwestminster.com/products/retrospec-ascent-mountain-bike-27-5-matte-silt-2022 (Accessed: 29 June 2023).

[46] *How to attach a potentiometer to a micro:bit* (no date) *teachComputing*. Available at: https://www.teachwithict.com/potentiometer.html (Accessed: 09 June 2023).

[47] Arduino. (2023). Arduino - Home. [Online]. Available: https://www.arduino.cc/ [Accessed: June, 09, 2023].

[48] Unity Technologies. (2023). Unity. [Software]. Available: https://unity.com/ [Accessed: June, 09, 2023].

[49] Autodesk. (2023). Maya. [Software]. Available: https://www.autodesk.com/products/maya [Accessed: June, 09, 2023].

[50] *Circuit diagram* (no date) *Circuit Diagram Web Editor*. Available at: https://www.circuit-diagram.org/editor/ (Accessed: 12 June 2023).

[51] *Bikes and bicycles Unity.pdf* (no date) *Google Drive*. Available at: https://drive.google.com/file/d/1qRg7jX4yIkbLTU8tkPDABW_v3DM-6P4v/view?pli=1 (Accessed: 09 June 2023).

[52] V-Kudryashov (no date) *V-kudryashov/bikebalance: Bike balance by steering in Unity*, *GitHub*. Available at: https://github.com/V-Kudryashov/BikeBalance#bikebalance (Accessed: 12 June 2023).

[53] Technologies, U. (no date) *Wheel collider component reference*, *Unity*. Available at: https://docs.unity3d.com/Manual/class-WheelCollider.html (Accessed: 14 June 2023).

[54] *Mobile tree package: 3D trees* (no date) *Unity Asset Store*. Available at: https://assetstore.unity.com/packages/3d/vegetation/trees/mobile-tree-package-18866 (Accessed: 09 June 2023).

Appendix

Appendix 1

Code in Arduino

```
int steerPin = A0;      // Analog pin for the rotation potentiometer
int leanPin = A1;       // Analog pin for the lean potentiometer
int brake_FPin = A2;
int brake_BPin = A3;  // Analog pin for the brake
int acceleratePin= 2;

void setup() {
  Serial.begin(9600);             // Start the serial communication
  pinMode(brake_FPin, INPUT);
  pinMode(brake_BPin, INPUT);// Configure the switch pin as input with a pull-up resistor
  pinMode(acceleratePin, INPUT);
  digitalWrite(acceleratePin, HIGH);
  pinMode(steerPin, INPUT);         // Configure the rotation potentiometer pin as input
  pinMode(leanPin, INPUT);
  // Configure the lean potentiometer pin as input
}

void loop() {
  int brakeRaw_F = analogRead(brake_FPin);
  int brakeRaw_B = analogRead(brake_BPin);// Read the value of the switch (HIGH or LOW)
  int steerRaw = analogRead(steerPin);       // Read the raw value of the rotation potentiometer
  int leanRaw = analogRead(leanPin);         // Read the raw value of the lean potentiometer
  int speedUp= digitalRead(acceleratePin);

  float steer = map(steerRaw, 500, 850, -45, 45);   // Scale the rotation value to the desired range (-1 to
1)
  float lean = map(leanRaw, 480, 750, -45, 45);   // Scale the lean value to the desired range (-90 to 90)
  float brake_F = map(brakeRaw_F, 0, 50 , 0, 50);
```

```
  float brake_B = map(brakeRaw_B, 0, 50, 0, 50);// Scale the brakevalue to desired range(0 to 70)
  Serial.print(steer);      // Send the steering value to Unity
  Serial.print(",");        // Separate the values with a comma
  Serial.print(lean);     // Send the lean value to Unity
  Serial.print(",");  // Send the switch value to Unity as well
  Serial.print(brake_F);
  //Serial.print(brakeRaw_F);
  Serial.print(",");  // Send the switch value to Unity as well
  Serial.print(brake_B);
 // Serial.print(brakeRaw_B);
  Serial.print(",");
  Serial.println(speedUp);



  delay(100);   // Adjust the delay according to your needs
}
```

Unity code 1: Bike physics

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;


public class BicyclePhysics : MonoBehaviour
{
	float horizontalInput;
	float vereticalInput;

	public Transform handle;

	Rigidbody rb;

```

```
public Vector3 COG;

[SerializeField] float pedalforce;

float currentbrakeForce;

float steeringAngle;
public float currentSteeringAngle;
[Range(0f, 0.1f)] [SerializeField] float speedteercontrolTime;
[SerializeField] float maxSteeringAngle;
[Range(0.000001f, 1)] [SerializeField] float turnSmoothing;

//[SerializeField]float maxlayingAngle = 45f;

[Range(-90, 90)]public float layingammount;
[Range(0.000001f, 1 )] [SerializeField] float leanSmoothing;

[SerializeField] WheelCollider frontWheel;
[SerializeField] WheelCollider backWheel;

[SerializeField] Transform frontWheeltransform;
[SerializeField] Transform backWheeltransform;

[SerializeField] TrailRenderer fronttrail;
[SerializeField] TrailRenderer rearttrail;

public bool frontGrounded;
public bool rearGrounded;
[Range(0,10)]public float speed ;
//[Range(-1, 1)]public float steerinput;
//[Range(-1, 1)]public float fowardinput;
//[Range(-90,90)]public float targetlayingAngle;
public float steerInput = 0f;
```

```
public float targetlayingAngle = 0f;
public float forwardInput; // Range(0, 1)
        public bool braking = false;
public float brakeForce;



// Start is called before the first frame update
void Start()
        {
                StopEmitTrail();
                rb = GetComponent<Rigidbody>();
        }

        // Update is called once per frame
        void FixedUpdate()
        {
                GetInput();
                HandleEngine();
                HandleSteering();
                UpdateWheels();
                UpdateHandle();
                lean();
                DownPresureOnSpeed();
                EmitTrail();
        }

        public void GetInput()
        {

                horizontalInput=steerInput;
                vereticallInput=forwardInput/5;
                //vereticallInput = Input.GetAxis("Vertical");
```

```
            //horizontalInput = Input.GetAxis("Horizontal");

            braking = brakeForce>=20;
    }

    public void HandleEngine()
    {
            backWheel.motorTorque = vereticallInput * pedalforce;
            currentbrakeForce = braking ? brakeForce : 0f;
            if (braking)
            {
                    ApplyBraking();
                    forwardInput = 0;
            }
            else
            {
                    ReleaseBrakibg();
                    forwardInput = 3;

  }
    }

    public void DownPresureOnSpeed()
    {
            Vector3 downforce = Vector3.down;
            float downpressure;
            if (rb.velocity.magnitude > 5)
            {
                    downpressure = rb.velocity.magnitude;
                    rb.AddForce(downforce * downpressure, ForceMode.Force);
            }

    }
```

```
public void ApplyBraking()
{
        //frontWheel.brakeTorque = currentbrakeForce/2;
        frontWheel.brakeTorque = currentbrakeForce* 2;
        backWheel.brakeTorque = currentbrakeForce*2;
}
public void ReleaseBrakibg()
{
        frontWheel.brakeTorque = 0;
        backWheel.brakeTorque = 0;
}

public void SpeedSteerinReductor()
{
        speed = rb.velocity.magnitude;
        if (rb.velocity.magnitude < 5 )
        //We set the limiting factor for the steering thus allowing how much steer we give to
the player in relation to the speed
        {
                maxSteeringAngle = Mathf.LerpAngle(maxSteeringAngle, 90,
speedteercontrolTime);
        }//set the max steering angle to 90 degree when speed is less than 5
        if (rb.velocity.magnitude > 5 && rb.velocity.magnitude < 10 )
        {
                maxSteeringAngle = Mathf.LerpAngle(maxSteeringAngle, 60,
speedteercontrolTime);
    }//set the max steering angle to 60 degree when speed is less than 10
    if (rb.velocity.magnitude > 10 && rb.velocity.magnitude < 15 )
        {
                maxSteeringAngle = Mathf.LerpAngle(maxSteeringAngle, 45,
speedteercontrolTime);
    }//set the max steering angle to 45 degree when speed is less than 30
```

```
        if (rb.velocity.magnitude > 15 && rb.velocity.magnitude < 20 )
                    {
                            maxSteeringAngle = Mathf.LerpAngle(maxSteeringAngle,  30,
speedteercontrolTime);
        }//set the max steering angle to 30 degree when speed is less than 20
        if (rb.velocity.magnitude > 20)
                    {
                            maxSteeringAngle = Mathf.LerpAngle(maxSteeringAngle,  25,
speedteercontrolTime);
        }   //set the max steering angle to 25 degree when speed is greater than 20
    }

        public void HandleSteering()
        {
                SpeedSteerinReductor();

                currentSteeringAngle = Mathf.Lerp(currentSteeringAngle, maxSteeringAngle *
horizontalInput/90, turnSmoothing);
                //the steering angle is rely on mutiple factors, speed will influence the max steering
angle, while input value from user will is the
                //direct factor
                frontWheel.steerAngle = currentSteeringAngle;



        }

        private void lean()
        {
                Vector3 currentRot = transform.rotation.eulerAngles;// update the current lean angle

                if (rb.velocity.magnitude < 1)
                {
                        layingammount = Mathf.LerpAngle(layingammount, 0f, 0.05f);
```

```
                    transform.rotation = Quaternion.Euler(currentRot.x, currentRot.y,
laying gammount);
                    return;

                }

                if (currentSteeringAngle < 0.5f && currentSteeringAngle > -0.5  ) //We're stright
                {
                    layingammount =  Mathf.LerpAngle(layingammount, 0f, leanSmoothing);

                }
                else //We're turning
                {
                    layingammount = Mathf.LerpAngle(layingammount, targetlayingAngle,
leanSmoothing );
                    rb.centerOfMass = new Vector3(rb.centerOfMass.x, COG.y,
rb.centerOfMass.z);
                }

                transform.rotation = Quaternion.Euler(currentRot.x, currentRot.y, layingammount);
            }

            public void UpdateWheels()
            {
                UpdateSingleWheel(frontWheel, frontWheeltransform);//leaning and spinning the
wheels while moving
                UpdateSingleWheel(backWheel, backWheeltransform);//
            }
            public void UpdateHandle()
            {
                Quaternion sethandleRot;
                sethandleRot = frontWheeltransform.rotation;     //allowed the handelbar transform
follow the steering input
```

```
                handle.localRotation = Quaternion.Euler(handle.localRotation.eulerAngles.x,
currentSteeringAngle, handle.localRotation.eulerAngles.z);
        }

        private void EmitTrail()
        {
                frontGrounded = frontWheel.GetGroundHit(out WheelHit Fhit);
                rearGrounded = backWheel.GetGroundHit(out WheelHit Rhit);

                if (frontGrounded)
                {
                        fronttrail.emitting = true;
                }
                else
                {
                        fronttrail.emitting = false;
                }

                if (rearGrounded)
                {
                        rearttrail.emitting = true;
                }
                else
                {
                        rearttrail.emitting = false;
                }

                //fronttrail.emitting = true;
                //rearttrail.emitting = true;
        }
        private void StopEmitTrail()
        {
                fronttrail.emitting = false;
```

```
            rearttrail.emitting = false;
        }

        private void UpdateSingleWheel(WheelCollider wheelCollider, Transform wheelTransform)
        {
            Vector3 pos;
            Quaternion rot;
            wheelCollider.GetWorldPose(out pos, out rot);
            wheelTransform.rotation = rot;
            wheelTransform.position = pos;
        }


}
```

Unity code 2: Controller

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System.IO.Ports;

public class ArduinoController : MonoBehaviour
{
    public BicyclePhysics bicycleVehicle;

    public float steerInput = 0f;
    public float targetLayingAngle = 0f;
    public float brakeForce_F = 0f;
    public float brakeForce_B = 0f;
```

```
    public float accelerate = 0f;

    private SerialPort serialPort;

    // Start is called before the first frame update
    private void Start()
    {
        serialPort = new SerialPort("/dev/cu.usbmodem2101", 9600); // Replace "COM3" with the
appropriate serial port
        serialPort.Open();
    }

    // Update is called once per frame
    // Update is called once per frame
    private void Update()
    {
        if (serialPort.IsOpen && serialPort.BytesToRead > 0)
        {
            string data = serialPort.ReadLine();
           // Debug.Log("Received data: " + data); // Print received data to console for debugging
            string[] values = data.Split(',');

            if (values.Length >= 5)
            {
                steerInput = float.Parse(values[0]);
                targetLayingAngle = float.Parse(values[1]);
                brakeForce_F = float.Parse(values[2]);
                brakeForce_B = float.Parse(values[3]);
                accelerate = float.Parse(values[4]);

                // Update parameters in the bicycleVehicle script using received data
                bicycleVehicle.steerInput = steerInput;
                bicycleVehicle.targetlayingAngle = -targetLayingAngle;
```

```
            bicycleVehicle.brakeForce_F = brakeForce_F ;
            bicycleVehicle.brakeForce_B = brakeForce_B ;
            bicycleVehicle.forwardInput = accelerate;

            // Debug.Log("Steer Input: " + steerInput); // Print the parsed values to Unity console
            // Debug.Log("Target Laying Angle: " + targetLayingAngle);
            //Debug.Log("BrakeForce_F: " + brakeForce_F);
            //Debug.Log("BrakeForce_B: " + brakeForce_B);
          }
        }
      }

      private void OnApplicationQuit()
      {
        if (serialPort != null && serialPort.IsOpen)
        {
          serialPort.Close();
        }
      }
    }
```

Appendix 2:

Survey used in user research:

1. How would you rate your skill level ?
    a. Beginner
    b. Intermediate
    c. Expert
    d. other
2. Have you ever got injured?
    a. Yes, but small injuries
    b. Yes, have had bad Injuries

    c. No
3. Which situation best describes the cause of your injury?
    a. Excessive riding leads to Overuse injury
    b. Bike crash on jumps
    c. Bike crash while cornering
    d. wrong brake operation
    e. Crash on drops
    f. Other
4. Which skill do you think is the most dangerous?
    a. Cornering
    b. Jumps
    c. Drops
    d. downhill
5. Do you have the fear of getting re-injury?
    a. Yes, I do
    b. Not at all
6. How much does this fear affect your performance?
    a. Scale 0 to 5(0 represent no, 5 represent extreme)
7. Have you received any training or therapy for overcoming your fear?
    a. Yes
    b. No
    c. Other
8. When you get injured, do you want to train at home with a training device?
    a. Yes, I believe it's safer and that can help me get back my physical condition
    b. No, I'd rather rest still fully recovered
    c. No, I'd rather train outdoors even if it's risky
    d. Others
9. Do you think specific training for the mistakes you made will help you develop confidence and overcome fear?
    a. Yes
    b. No
    c. Maybe
    d. Other
10. Do you think overcoming the fear of re-injury could benefit from physical activity?

    a. Yes
    b. No
    c. Other
11. Do you think a simulative bike trainer for mountain bike skills can overcome the fear of re-injury?
    a. Yes
    b. No
    c. Maybe, I give hope on this
    d. Maybe, but I don't give hope on this
    e. Other
12. Would you like to simulate training on your previous crashes/mistakes?
    a. Yes
    b. NO
    c. Other
13. Which skill do you want the interactive bike trainer to target?
    a. Pump track skill
    b. cornering skill
    c. posture correction
    d. Other