

VR Walk: Game

*VR simulator for treadmill as a tool for
constraint-based gait rehabilitation*



Bendik Myrene Bleikli and Lauritz Skålevik Valved

Master's thesis in Software Engineering at

Department of Computer science, Electrical
engineering and Mathematical sciences, HVL

Department of Informatics, UiB

June 2020



**Western Norway
University of
Applied Sciences**



Abstract

The viability of Virtual Reality (VR) is being explored in an increasing number of fields outside its primary use in entertainment. A notable example is VR therapy, where VR-based alternatives to exposure therapy and post-stroke rehabilitation have proven especially effective.

Treadmill-walking paired with monitor-based game activities is a popular gait rehabilitation tool within physiotherapy. While, to the authors' knowledge, no equivalent immersive VR-based solutions currently exist outside of experimental stages, recent research has shown that doing activities in VR while walking on a treadmill notably affects the user's gait. This indicates that VR-based gait rehabilitation might be a viable alternative while also addressing the issues of lacking immersion and motivation for continued training present in current solutions.

This thesis revolves around the development and evaluation of an immersive VR simulator for treadmill usage, intended to further explore the viability of such a solution within gait rehabilitation.

While clinical testing was outside of the scope of the thesis, the resulting software solution was to be evaluated through demonstration for, and a subsequent semi-structured interview with, physiotherapists and VR rehabilitation researchers. However, the outbreak of COVID-19 led to the cancellation of the demonstration and interview. Instead, this thesis will detail the intended processing of the interview feedback and results and how these might have been used to appraise the simulator as a viable tool within gait rehabilitation according to the three primary factors of the Extended Technology Acceptance Model, namely perceived usefulness, perceived enjoyment and perceived ease-of-use. Additionally, feedback received during informal test settings and from the authors' supervisors is reviewed and discussed in the context of the simulator's potential usability.

*The most important step a man can take.
It's not the first one, is it?
It's the next one. Always the next step.*

BRANDON SANDERSON
Oathbringer

Outline

”VR Walk: Game” builds upon the work done during the BSc project ”VR Walk BSc”. The BSc project, VR Walk BSc, was developed as a tool for use in research within physiotherapy. This master’s thesis revolves around further developing VR Walk: Game, with a focus on both entertainment value, user-friendliness and its value as a potential tool for constraint-based gait rehabilitation.

Supervisors

Department of Computer science, Electrical engineering and Mathematical sciences:

- Atle Geitung, HVL
- Harald Soleim, HVL

Physiotherapy at HVL’s Department of Health and Functioning:

- Lars Peder Bovim, SimArena at HVL and Energisenteret for Barn og Unge at Haukeland University Hospital
- Bård Erik Bogen, HVL

Acknowledgements

We would like to thank our supervisors at Western Norway University of Applied Sciences (HVL), Harald Soleim, Atle Birger Geitung, Bård Erik Bogen and Lars Peder Bovim for their excellent support and guidance throughout our work on this thesis. Without their contributions and feedback, this thesis would not have been possible. Lars Peder also has our unyielding gratitude for his continued enthusiasm in presenting and promoting our efforts through presentations and opportunities to exhibit VR Walk: Game.

Thanks to Bodil Moss at the HVL library for her valuable contributions in locating literature, letting us present at the library's VR and research webinar and for her recommendations in steering our writing voice throughout the thesis.

Thanks to Martin Tellevik, who developed the original VR Walk BSc simulator along with us. At least one of us escaped the madness.

We would also like to thank Thomas Fiskeseth Larsen, Aleksander Hurtley, Fred Rune Breisnes and Thasajini Rasakunasingam for their assistance in making us able to take control of the treadmills directly through our software, and thanks to Unity Technologies and the amazing tool and asset developers who made VR Walk: Game possible! n Thanks to our friends at the HVL IT master's lab for making this time such an enjoyable one. It has been great fun bouncing ideas back and forth with you!

Finally, we would like to thank our friends and family members (including very, very recent additions) for their motivation, support and understanding throughout the project!



Division of work

As a general rule, both have contributed to almost every section. This list is intended to fulfill a formal criteria for group theses, as an indication of which author contributed the majority of the writing to any particular part.

Abstract	- Both
Outline	- Both
1 Introduction	- B (Bleikli)
1.1 Thesis outline	- B
1.2 Motivation and problem description	- B
1.3 Research questions	- Both
1.4 Research goals	- V (Valved)
1.5 Methodology	
1.5.1 Ideas	- V
1.5.2 Feedback - development cycle	- V
1.5.3 Final evaluation	- Both
1.6 Prior work	- B
1.7 Related work	
- introduction	- V
- Using VR for Fitness Training - Pilot Study	- V
- How do persons with stroke and therapists experience trialing a VR-based rehabilitation game performed on a tread-mill?	- B
- Addition of a non-immersive virtual reality component to treadmill training to reduce fall risk in older adults (V-TIME): a randomised controlled trial	- V
- Using games for stroke rehabilitation	- B
Immersive virtual reality and Gamification within procedurally generated environments to increase motivation during gait rehabilitation	- V
- Treadmill walking in a fully immersive virtual environment	- V
- Virtual Reality games and gamified exercises in physiotherapeutic treatment of non-specific low back pain patients with kinesiophobia	- B
2 Background	
2.1 Fully immersive VR technology	- V
2.2 Serious game	- B
2.3 VR fitness and exercise	- V
2.4 Unity	- V
2.5 VR Walk BSc	
introduction	- B
2.5.1 The VR Walk BSc software	- Both
2.5.2 Reception	- V
2.6 Constraint-based gait rehabilitation	- B
3 Solution - VR Walk: Game	
- introduction	- Both

3.1	Hardware	- B
3.2	Software	- Both
3.3	Third-party assets	- V
3.4	User safety	- V
3.5	The visuals - Choices, tools and technologies	- V
3.5.1	Vegetation biomes	- B
3.6	Interaction design	
	- introduction	- B
3.6.1	Input handling	
	- SteamVR configuration	- B
	- Controller input	- B
3.6.2	User interfaces	
	- Main menu	- B
	- Treadmill panel	- B
	- In-game statistics for the operator	- V
	- Instructional text boxes	- B
3.7	Treadmill interfaces	
	- introduction	- B
	- Synchronization	- B
	- Rodby RL2500E x 1000 interface	- B
	- Woodway PPS 55 Med interface	- V
	- Dummy interface	- V
3.8	Constraints and activities	
	- introduction & list	- B
3.8.1	Perceptual constraints	
	- introduction	- B
	- Glass bridge	- B
3.8.2	Cognitive dual task constraints	
	- introduction	- V
	- Totem pole memorization and building	- Both
3.8.3	Motor dual task constraints	
	- introduction	- B
	- Coin collecting	- V
	- Axe throwing	- Both
	- Balance ball	- V
3.8.4	Mechanical constraints	
	- introduction	- B
	- Narrow paths	- V
	- Obstacles on the path	- V
	- Pre-distributed footprints	- V
3.8.5	Final score	- V
3.9	Operator choices and the main menu	
	- introduction	- V
3.9.1	The path builder menu	- V
3.9.2	Data persistence	- V
3.9.3	Activities during main menu navigation	- V
3.10	Designing for modularity	- V
3.11	Optimization	- B
3.12	Continuous evaluation as part of the development process	- V
4	Evaluation	
	- introduction	- Both
4.1	Semi-structured interviews with	

	physiotherapists and researchers	- Both
4.2	Evaluating perceived ease-of-use through the System Usability Scale	- V
4.3	Evaluation through testing and conversations with supervisors	- Both
4.4	Feedback and evaluation through informal testing	- V
5	Conclusion	- Both
6	Further Work	
	- Further research	- Both
	- Known issues	- V
	- Further development	- V
	Appendix	
	- Intervjuprotokoll	- Both
	- System Usability Scale	
	- Manual: Oppsett og kalibrering	- V

Acronyms

FPS frames per second. 57

GUI Graphical user interface. 61

HMD Head mounted display. x, 2, 10, 13, 17, 18, 22, 23, 30, 55, 57, 67, 69

HVL Western Norway University of Applied Sciences. iii, iv, 4, 6, 13, 18, 64, 65

IDE Integrated development environment. 19

MMSE Mini-Mental State Examination. 36

NTNU Norwegian University of Science and Technology. 59, 61, 80

PE perceived enjoyment. i, ix, 3–6, 24, 43, 58, 60, 61, 63, 65, 66

PEOU perceived ease-of-use. i, ix, xvii, 3, 5, 6, 29, 44, 51, 58–63, 65–67

PU perceived usefulness. i, ix, 3, 4, 6, 51, 60, 61, 63, 66

REK Regional Committees for Medical and Health Research Ethics. 60

SUS System Usability Scale. xvii, 2, 7, 59, 61, 62, 73, 80

TAM Technology Acceptance Model. ix, 3

VR Virtual Reality. i, x, xi, xvi, 2, 6–14, 16, 17, 19, 21, 25–31, 33, 35, 51, 59, 60, 63–65, 67, 69, 81

Glossary

- asset** In this document, an asset refers to a digital resource used within Unity such as scripts, models, textures and tools. Unity Technologies maintains a digital store with a wide range of these[1]. iv, ix, x, xvii, 5, 12, 20–22, 24, 25
- biome** An enclosed geographical area with shared settings, i.e for asset distribution. xii, xvii, 22, 24, 25
- bradykinesia** Bradykinesia is defined by slow movement and an impaired ability to move the body swiftly on command, i.e having a shuffling gait. Other symptoms can include difficulty with repetitive tasks. It is often associated with Parkinson’s disease or as a side-effect of medications[2]. 15, 66
- collider** The object’s geometrical representation used for physics behaviour or raycasts. 30
- constraint** Limitation or additional challenge put upon an aspect of the act of walking. 2, 4, 9, 14, 15, 35–37, 39, 44–46, 63, 66
- constraint-based gait rehabilitation** A method for assessing a patient’s motor skills, detailed in the Constraint-based gait rehabilitation section. iii, ix, xvi, 2, 3, 6, 14, 63, 66
- constraint-based gait training** See constraint-based gait rehabilitation. 6, 35
- Extended Technology Acceptance Model** TAM is a model proposed to understand the determinants, PU and PEOU, of IT-usage from a user’s perspective. The Extended Technology Acceptance Model introduced PE in order to better reflect the motivations of video game players. Elaborated further in the Research goals section. i, 3, 16, 60, 61, 66
- gait** A person’s pattern of walking. Walking involves balance and coordination of muscles so that the body is propelled forward in a rhythm. i, 2–4, 6, 8, 9, 12, 14, 15, 47, 63, 64, 66, 68, 69
- GameObject** Entities in Unity that act as containers for components, such as lights, cameras, custom scripts or properties, and other GameObjects. Every element of a Unity scene is either a GameObject or attached to a GameObject. Each GameObject has an attached transform which contains the scale of the GameObject and positional and rotational data. x, 19, 26, 33, 50

- HTC Vive** A range of VR hardware created by HTC. 10, 13, 17, 27, 57, 67
- Lighthouse** SteamVR’s Lighthouse tracking system. A 3D spatial laser-tracking system for tracking VR hardware through SteamVR[3]. x, xi, 10, 17, 33
- Perlin noise** A gradient noise algorithm developed by Ken Perlin in 1983[4]. xiii, 40, 41, 47
- prefab** A pre-configured or prepared collection of assets or a GameObject state. Can be instantiated either through scripts or the Unity editor. 57
- raycast** A ray passed from an origin point in a set direction, that will extend until it reaches a set range or until it intersects with a valid object. The intersected object is then returned. ix, 30, 31
- simulation sickness** Nausea caused by movement in a HMD-based VR system. It is especially prominent when real-life movement does not reflect the expected virtual movement. 6, 12, 57
- SteamVR** Software for positional tracking and input handling of VR hardware. Developed by Valve Corporation. x, xii, 17, 21, 26–31, 33, 57, 64, 67, 68, 81, 82, 84
- Unity** A cross-platform game engine developed by Unity Technologies[5]. ix–xi, xiv, 12, 13, 19–22, 24, 26, 30, 39, 53, 57, 68
- UV** 2D-space coordinates used in the UV mapping process, where image texture data is projected onto a (3D-space) object. xi, 21, 36, 39
- Vive tracker** Optional hardware for Lighthouse-based VR hardware. Can be strapped to a user’s legs to provide additional positional and directional information. 10, 13, 17, 23, 30, 47
- VR Walk BSc** The game/simulator developed as part of our bachelor thesis. Used as a tool for gait research within physiotherapy[6, 7]. iii, iv, xii, xiii, 3, 4, 6, 9, 10, 12, 13, 19, 20, 23, 26, 27, 31, 33–35, 44, 46, 51, 53, 63, 64
- VR Walk: Game** The game/simulator developed as part of this thesis. iii, iv, xi, xiv, xvi, 1, 3–6, 10, 12, 16, 20, 23, 24, 34, 44, 51, 53, 55, 58, 60–69
- Worley-Voronoi noise** A noise algorithm achieved by combining Worley noise[8] with Voronoi decomposition[9]. Worley noise by itself looks cellular, somewhat like caviar, when rendering it’s output. With Voronoi decomposition it can be straightened to look more like stone shards (3D) or natural stone tiles (2D). xiii, 40, 43

List of Figures

1.1	An example of a completed card on the Kanban. Screenshot taken from the VR Walk: Game project in Trello.	4
1.2	Opportunities for external testing were given in the form of exhibiting VR Walk: Game at events. Photo taken at Bergen Research Days 2018[10].	5
1.3	Photo: Mirelman, et al.[11]. The key physical components of their foot tracking and monitoring solution are marked in red. . .	8
2.1	An example of fully immersive VR hardware; the HTC Vive Pro with controllers and base stations for the Lighthouse tracking system. During the development of VR Walk: Game, this was the model most frequently used. Photo: HTC Corporation.	10
2.2	A selection of screen technologies under the VR umbrella. Left image: Cave VR, where the environment is projected onto the surrounding walls. Center: A non-immersive VR solution. Right: Fully immersive VR.	11
2.3	Beat Saber is a VR rhythm game that promotes physical activity. Image: Beat Games	11
2.4	An overview of the Unity Editor.	12
2.5	The environment of VR Walk BSc.	13
3.1	A HTC Vive tracker strapped correctly to the player's ankle. Also pictured is one of the shoes used as reference for the virtual representation of the player's feet.	17
3.2	Sculpting a prop for the totem pole memorization and building activity in VR, using Masterpiece VR. At later stages, the real-time optimized versions were created and UV-unwrapped using a combination of Instant Meshes and Blender before they finally were texture painted in Blender. The final prop can be seen in Figures 3.12 and 3.15.	21

3.3	The user steps too far to the left on the treadmill in VR Walk BSc. The treadmill has stopped and the user is alerted to why, with a red grid and an icon on the virtual display.	23
3.4	Vegetation density by biome. Areas along the path where activities would otherwise be obscured are without vegetation. Dark green areas are densely populated while the player is near. Outside this is a transition area before fading into a sparsely populated terrain.	25
3.5	Configuration page for the SteamVR action sets. Each action is given a string identifier and a type for the input value. Simple buttons use boolean values. Vectors are used for pressure sensitive elements and thumb-stick movements.	26
3.6	On the left: the controller representation utilized in VR Walk BSc, an interactive model from the SteamVR plug-in featuring all buttons present on the physical controller. On the right: The controller representation currently used. Hand models from the SteamVR plug-in that animate in response to controller input and can be manually posed.	27
3.7	Custom hand poses for various objects and interactions.	28
3.8	The treadmill front panel, featuring buttons for increasing and decreasing movement speed on the left, a start/stop button at the center, and a statistics field and safety indicators on the right.	31
3.9	An instructional text box, displayed ahead of the balance ball activity.	32
3.10	The detected serial-ports (COM) are listed and can be selected. This, combined with treadmill selection, determines the interface used.	34
3.11	The glass bridge as seen from the starting hill top.	36
3.12	The player builds a miniature totem pole from memory.	37
3.13	Each totem pole piece was 3D sculpted with a distinct silhouette and contrasting features for easier recognition from afar.	37
3.14	A chart over three color palettes suited for various types of dichromacy color deficiencies[12], with the totem paint colors added at the bottom row. While more care could have been taken with regard to tritanopia, it is a very rare condition[13]. . .	38
3.15	The unpainted totem pieces as presented to the player. The card paint marks are displayed in white when the card suit option is enabled to allow players to still use them for recognition.	38

3.16	The player reaches to grab a randomly moving coin, while avoiding the ones with a skull motif.	40
3.17	The shattered version of the activity props are fractured based on a Worley-Voronoi noise algorithm.	40
3.18	By either controlling how fast one traverses along an axis of a Perlin noise-map or by altering the map's scale, it is possible to control how smooth or erratic the motion of a coin is along an axis. The illustrated maps are stretched in their x-axis for illustration purposes.	41
3.19	Possibly the first recorded instance of a coin collector, Caesar Augustus (63 B.C. to A.D. 14)[14]	42
3.20	A wooden bridge serving as an alternative play area for the axe throwing activity. The wider bridge and lower altitude creates a more comfortable environment for users with vertigo.	42
3.21	Targets for the axe throwing activity. The spheres are positive targets and the skulls are negative ones. Clustered target patterns are seen in the distance.	43
3.22	Placing an axe-throwing -or a striking target.	43
3.23	The left image shows the outline of a hand, instructing the player to grab the plate. The right image shows the player where the plate must be held to receive a balancing ball. When the plate is held in the correct position, the ball becomes solid and the physics engine takes control of it.	45
3.24	The player tries to follow the path, while balancing a burning ball on a plate, without the plate getting too hot.	45
3.25	The various path-segments that contain alterations to the path itself, as seen from above. These include path-segments that are partially narrowed or littered with obstacles. From top to bottom: Path narrowing in on the right hand side, on the left hand side, towards the middle of the path and with large rocks on top of the path.	46
3.26	Dance floor decals for learning dance steps. Image: ifloortape.com	47
3.27	(a) shows the player following a sequence of pre-generated footprints with his feet. (b) illustrates the second iteration of the footprint placement algorithm.	48
3.28	The score panel shown to the player at the end of the path. . . .	50
3.29	The path builder as it appeared in VR Walk BSc.	51

3.30	The second iteration of the path builder introduced more granular control of the path settings and descriptive mouse-over tool-tips.	52
3.31	The third iteration of the path builder focused on flexibility, ease-of-use, random generation and storing and loading both settings and paths.	52
3.32	The nested composition of a path builder button. Green: Click-and-drag slot area, Yellow: Main body of the button, Blue: The common elements of the configurable buttons, Red: The activity-specific settings.	56
3.33	A portal entrance used for transporting players to and from other areas of the environment.	56
3.34	Unity's Profiler tool, displaying time use of application processes on a per-frame basis.	57
4.1	Picture from E-LAN 2018	64
4.2	Exhibiting VR Walk: Game for upper secondary school students at Forskningsdagene Ung 2019. Photo: @hvlrehablab. . .	65

List of Code Examples

3.1	The properties and color blending between paint splashes and the underlying totem piece texture. in this code example, 'a' refers to the wooden texture of the totem pieces, 'b' to the '_Color'-painted parts, and 'c' to the resulting combination of these.	39
3.2	The code for the coin movement positions for all possible movement patterns. The coordination -and time offsets are randomized per coin and ensures that the coins and directions do not follow the same cycles.	41
3.3	The code for generating the randomized footprints	50
3.4	The JSON representation of a fully configured path. When the main game-play scene is loaded, a path will be generated from these settings. If loaded into the path builder, these settings will also be loaded.	54

Contents

1	Introduction	1
1.1	Thesis outline	1
1.2	Motivation and problem description	2
1.3	Research questions	2
1.4	Research goals	3
1.5	Methodology	3
1.5.1	Ideas	3
1.5.2	Feedback - development cycle	4
1.5.3	Final evaluation	6
1.6	Prior work	6
1.7	Related work	7
2	Background	10
2.1	Fully immersive VR technology	10
2.2	Serious games	11
2.3	VR fitness and exercise	11
2.4	Unity	12
2.5	VR Walk BSc	12
2.5.1	The VR Walk BSc software	12
2.5.2	Reception	13
2.6	Constraint-based gait rehabilitation	14
3	Solution - VR Walk: Game	16

3.1	Hardware	16
3.2	Software	19
3.3	Third-party assets	20
3.4	User safety	23
3.5	The visuals - Choices, tools and technologies	23
3.5.1	Vegetation biomes	24
3.6	Interaction design	25
3.6.1	Input handling	26
3.6.2	User interfaces	30
3.7	Treadmill integration	32
3.8	Constraints and activities	35
3.8.1	Perceptual constraints	35
3.8.2	Cognitive dual task constraints	36
3.8.3	Motor dual task constraints	39
3.8.4	Mechanical constraints	45
3.8.5	Final score	50
3.9	Operator choices and the main menu	51
3.9.1	The path builder menu	51
3.9.2	Data persistence	53
3.9.3	Activities during main menu navigation	55
3.10	Designing for modularity	55
3.11	Optimization	57
3.12	Continuous evaluation as part of the development process	58
4	Evaluation	59
4.1	Semi-structured interview with physiotherapists and researchers	59
4.2	Evaluating perceived ease-of-use through the System Usability Scale	61

4.3	Evaluation through testing and conversations with supervisors	62
4.4	Feedback and evaluation through informal testing	63
5	Conclusion	66
6	Further Work	67
	Appendices	76

Chapter 1

Introduction

This chapter provides context for the work done in relation to this thesis. This includes a brief detailing of the previous project that this thesis is a continuation of, other related works, the goals for the thesis, and how we worked towards them.

1.1 Thesis outline

Introduction The first chapter introduces the context and motivation for the thesis. Additionally, it contains an overview of related research and previous work, as well as the chosen methodology used during the work of this thesis.

Background The background chapter contains theoretical background material needed to complete the developmental and evaluative work done for the thesis.

Solution - VR Walk: Game This chapter goes into detail about the activities and functionalities developed for the simulator, the overall visuals, interaction design and other developmental choices.

Evaluation In lieu of the planned testing, we elaborate here on how testing would have proceeded and how we intended to process the findings.

Conclusion The conclusion chapter summarizes the work completed for the thesis in relation to the corresponding research goals and questions. While no solid conclusion could be drawn in the absence of the intended evaluation, some implications for the work are discussed.

Further Work This chapter contains some reflections about the thesis project and an overview of desired features that did not make it into the final version of the software by the end of development.

Appendix The appendix contains the interview protocol and System Usability Scale (SUS) form intended for evaluating the software. Additionally, a manual for setting up and calibrating the software is also included. The appended documents are in Norwegian only.

1.2 Motivation and problem description

A person's ability to walk is strongly indicated to be linked with capacity for executive function and attention[15]. For some with reduced lower body mobility, the act of walking demands enough focus and attention that simultaneously performing other tasks becomes highly challenging. Conversely, being suddenly distracted might cause the person to halt their walk involuntarily. Individuals with Parkinson's disease, stroke sufferers and, to a lesser extent, unimpeded individuals have been shown to have reduced balance and walking speed when being subjected to cognitive or motoric distractions[16, 17].

In constraint-based gait rehabilitation patients exercise to gradually improve the automation of their gait. Treadmill-based gait training is a frequently used tool in this type of physical rehabilitation. Constraining the training by way of introducing other tasks or altering the premise of the walking activity has been shown to often improve the results of the treadmill training[18]. These added activities, or *constraints*, consist of having the user perform various tasks while walking (dual task) or by adding mechanical or perceptual complexity to the basic walking task[19].

Most available tools for constraint-based gait rehabilitation today are either monitor-based, or do not have virtual environments at all. It has been shown that training using monitor-based virtual environments has given better results in rehabilitation than solutions with no virtual environments [20]. It is generally agreed upon that HMD-based VR is a more immersive experience than its monitor-based counterpart. Therefore, the intent of this thesis is to explore the viability of a fully immersive VR gait training solution.

1.3 Research questions

Reduced higher-order cognitive and executive functions are associated with a higher risk of falls[21]. Gait disorders may be complex and involve both motor and cognitive aspects. Recent evidence suggests that training motor skills and cognitive skills simultaneously may give added benefits[11]. How can we, in a fully immersive VR and treadmill solution, introduce activities that address functions associated with specific motor-cognitive impairments during constraint-based gait?

1.4 Research goals

According to the analysis of Wang and Goh[22], the Technology Acceptance Model (TAM) was proposed to understand the determinants of information technology usage, as seen from a user's perception of the analyzed technology. In this model, the first main factor is the perceived usefulness (PU), meaning the *"subjective probability that using a specific application will increase his or her performance within an organizational context"*. In the case of VR Walk: Game, this is how well suited the solution is for potential use within constraint-based gait rehabilitation or for research within the field of physiotherapy. The second main factor is the perceived ease-of-use (PEOU), meaning *"the degree to which the user expects the target system to be free of effort"*. This includes both the player and simulator operator's intuitive understanding of the software product.

Video games, especially outside of serious games, are hypothesized to be motivated primarily by hedonistic, as opposed to primarily utilitarian, factors. To reflect the motivations of game players, the Extended Technology Acceptance Model for video game acceptance introduced perceived enjoyment (PE). It is defined as *"the extent to which the activity of using the computer(-technology) is perceived to be enjoyable in its own right, apart from any performance consequences that may be anticipated"*. Other keywords relevant to this factor are flow, perceived playfulness, experience and satisfaction. Wang and Goh describes flow as *"the experience that people feel when they act with total involvement"*. This is, among other factors, dependent on the degree of which the player is challenged without experiencing frustration. Satisfaction is described as *"the pleasurable or positive emotional state resulting from cognitive appraisal of the game"*. Perceived playfulness is conceptualized as *"an intrinsic belief that is formed from the individual's subjective experience with the (information) system"*[22].

Taking inspiration from this model, *the goal of this thesis is to develop a software product that, from a user's perspective, satisfies in regards to player safety, as well as the three primary factors of the Extended Technology Acceptance Model[22], namely, perceived enjoyment, perceived ease-of-use and perceived usefulness in terms of treatment value within constraint-based gait rehabilitation.*

1.5 Methodology

1.5.1 Ideas

The planning started in June 2019. An idea workshop centered around VR Walk BSc was organized by Lars Peder Bovim through Energisenteret for Barn og Unge at Haukeland University Hospital. A group of eight experts within gait training, neurology, physiotherapy and rehabilitation met with us

for the workshop, which was held in SimArena’s physiotherapy lab at HVL. The experts were able to try VR Walk BSc, and suggest and discuss which game-play mechanics and activities would be most beneficial to implement in the software for their respective patient groups.

We evaluated these suggestions based on their expected value as a dual task mechanic for gait rehabilitation (PU), their expected ”fun factor” (PE), if the activity played well with existing game-mechanics and how challenging it would be to implement time-wise. Although many of the suggestions were deemed too specific to a very small part of the patient groups, the results from this workshop, in addition to the constraints listed in the sub-chapter ”Constraints and activities”, provided the guidelines for directing the development of the desired mechanics.

While some ideas came from the structured workshops and meetings, most of them originated from casual conversations (and joking) between us, where these conversations sparked the inspiration for a game-play activity or a new way to improve upon one. The most promising ideas were listed in cards on a digital Kanban board, using Trello.com. The ideas were then discussed further and potentially prototyped through a simplified proof of concept implementation of the idea, following the pattern listed in the ”Feedback - development cycle” section below.

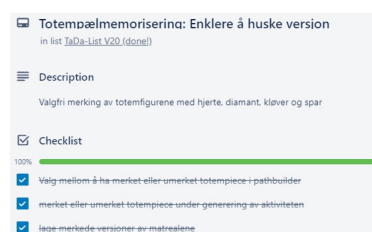


Figure 1.1: *An example of a completed card on the Kanban. Screenshot taken from the VR Walk: Game project in Trello.*

1.5.2 Feedback - development cycle

The development of the thesis project started in August 2019. The first steps in the development was to prepare the bachelor project for expansion, perform optimizations and to integrate support for the Woodway treadmills. A previous bachelor project had revolved around establishing serial-communication with SimArena’s Woodway PPS Med 55[23] treadmill. We expanded upon this work and were able to fully integrate this treadmill in VR Walk: Game. With those tasks completed, new dual-task activities for the player to perform, visual quality, better controls and better, easy to use tools were prioritized.

When developing a new activity, the following iterative pattern was loosely followed:

1. Workshop ideas that covered a constraints category.
2. Explore the most promising ideas by prototyping base functionality with simple props and graphics. This might be with plain cylinders and boxes or other temporary placeholder models (3D geometry).

3. Create cards and lists on the Kanban board, containing the steps expected to be required to complete the activity. Figure 1.1 shows one of these cards. New tasks would be listed as issues arose.
4. Internal testing and adjustments until the activity felt fun and/or satisfying. Failing this, the idea would be either scrapped or reworked.
5. Replace temporary models and graphics assets.
6. External informal testing for determining activity difficulty and perceived ease-of-use.
7. Make adjustments accordingly, without sacrificing perceived enjoyment.
8. Polish activity "looks and feel".
9. Include the activity during internal testing over multiple days in case of infrequently appearing bugs.

Internal testing was performed continuously by ourselves during development. Here we were able to develop ideas from a rough stage to something we could put in front of an external tester.



Figure 1.2: *Opportunities for external testing were given in the form of exhibiting VR Walk: Game at events. Photo taken at Bergen Research Days 2018[10].*

Opportunities for external testing were given in the form of numerous expo-or presentation settings during development (Figure 1.2). Event attendees and random passers by were able to try out the latest iteration of the application on a Woodway[23] treadmill. During these demonstrations, we gathered informal feedback and monitored if any of the presented activities were not as intuitive or at the difficulty-level as first assumed during development, or if there were any other difficulties with the new functionalities.

Throughout development, an emphasis was always put on having the project software available in a usable state. Adopting an iterative development method allowed for both spontaneous and planned testing and demonstration. Added features were shaped through discussion with supervisors at several stages of the features' implementation.

1.5.3 Final evaluation

Towards the end of the development period, it was planned to complete an evaluation through a semi-structured group interview with power-users involved with research within the VR-treadmill field, and specialists within physiotherapy. In this interview, VR Walk: Game would be evaluated based on its user-friendliness(PEOU) both for the user and the operator, its viability as a tool for constraint-based gait training(PU, PE), its future potential and overall impressions.

Due to the outbreak of the COVID-19 virus, these interviews were unfortunately cancelled and the subjects did not have access to the equipment needed for testing VR Walk: Game during this time. In their stead, a detailed account of the intended processing of the interview feedback can be found in the section "Semi-structured interview with physiotherapists and researchers".

1.6 Prior work

During spring of 2018, the first iterations VR Walk BSc[6] were developed as a bachelor thesis in software engineering by Bendik Bleikli, Martin Tellevik and Lauritz Valved, in cooperation with Lars Peder Bovim from SimArena, at Western Norway University of Applied Sciences (HVL).

The aim of the project was to provide Bovim and SimArena with a fully immersive VR simulation with treadmill integration. The simulation would be used to facilitate research conducted by Bovim and would contribute to his master's thesis[7]. At this stage it was utilized as a tool for gathering research data by presenting a fully immersive virtual environment and some light challenges to the user. The gathered data was then used to determine whether gait pattern was altered and if the user was affected by simulation sickness.

Bovim's positive findings (summarized in the "Treadmill walking in a fully immersive virtual environment" subsection) were a solid indication that the viability of constraint-based gait rehabilitation using fully immersive VR should be explored, and being able to use the same simulator as a starting point for this seemed ideal.

1.7 Related work

The following subsections are titled directly by the articles' names and includes a range of studies where games have been used in treatment or exercise with or without any kind of VR and/or treadmill.

Using VR for Fitness Training - Pilot Study

In this recent pilot study, Alce, Hansson and Mårtensson [24] developed a fitness VR application with three exercise activities and a mindfulness meditation scene. Through evaluation by structured interviews and the System Usability Scale (SUS) questionnaire, their initial findings showed that their participants found VR exercise both fun and motivating.

How do persons with stroke and therapists experience trialing a VR-based rehabilitation game performed on a treadmill?

In this master's thesis, also using VR and treadmill, Endresen[25] created a selection of primarily cognitive game tasks based on head movement and hand controllers. 5 persons with stroke and 5 therapists were interviewed about their experience with the game in semi-structured interviews. Both groups reported that they were positive to the continued use of technology-based rehabilitation, and found the experience with VR to be very immersive, novel and motivating.

Addition of a non-immersive virtual reality component to treadmill training to reduce fall risk in older adults (V-TIME): a randomised controlled trial

In this study, Mirelman, et al.[11] found that the addition of a VR solution to treadmill training, as opposed to with treadmill training alone, significantly lowered the rate of falls in elderly with a history of risk of falls. The subjects were adults aged 60-90 years with either a history of idiopathic falls, with mild cognitive impairments or with Parkinson's disease. In the case of this study, VR refers to a monitor-based solution with the participant's feet mapped onto a virtual environment using a modified Microsoft Kinect camera solution (Figure 1.3).

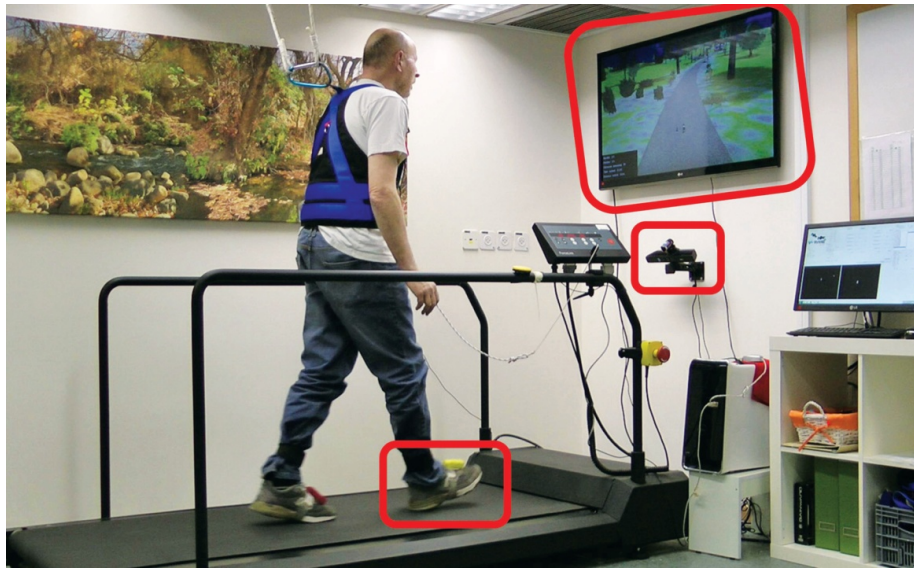


Figure 1.3: *Photo: Mirelman, et al.[11]. The key physical components of their foot tracking and monitoring solution are marked in red.*

Using games for stroke rehabilitation

The work of Alankus et al.[26] provides insight into developing games specifically for rehabilitation. The project revolved around creating a series of light exercise games suitable for stroke victims with a large variance in motor capacity. Feedback was gathered from testing with 4 patients with different degrees of impairment. Their findings indicate that particular care should be taken in accommodating the physical capabilities of the user, and in scaling the difficulty to provide constant challenge. They also indicate that games may aid motivation for repeated motion by having game-play elements revolve around producing the desired motion.

Immersive virtual reality and Gamification within procedurally generated environments to increase motivation during gait rehabilitation

Kern et al.[27] developed a story-based game for treadmill and VR or monitor. By reaching walking distance milestones, the user progresses through the story and gradually transforms the environment from desert to forest. It was tested on 36 healthy subjects divided into groups that tried either the VR version or the monitor-based version. The subjects reported increased decision freedom, increased perceived task meaningfulness, lower anxiety, lower frustration and lower pressure with VR. Their test subjects also reported significantly increased motivation to continue. The article recognizes that the novelty of VR might have colored their results to a degree.

Treadmill walking in a fully immersive virtual environment

In this physiotherapy master's thesis, Bovim was able to determine that gait was measurably altered when subjected to a virtual environment and a selection of VR-presented constraints. These findings were produced by subjecting 28 healthy adults (age: 29 ± 4.5) to the VR Walk BSc virtual environment and a selection of its implemented gait constraints while wearing sensors.

Bovim also found that none of his test subjects reported motion sickness while -or after using VR Walk BSc on a treadmill[7].

Virtual Reality games and gamified exercises in physiotherapeutic treatment of non-specific low back pain patients with kinesiophobia

In a recent thesis, Larsen[28] explores the viability of the use of VR in the treatment of non-specific low back pain in patients with fear of movement. The game-play of the VR software emphasises guiding the user through specific motions by having them draw arrows from a bucket at knee height and fire them.

The findings showed that patients were able to complete motions when using VR that they would otherwise find too painful to perform, which holds interesting implications for other forms of distraction-based VR treatment.

Chapter 2

Background

The following sections describe the terms and technologies used in this project.

2.1 Fully immersive VR technology



Figure 2.1: *An example of fully immersive VR hardware; the HTC Vive Pro with controllers and base stations for the Lighthouse tracking system. During the development of VR Walk: Game, this was the model most frequently used. Photo: HTC Corporation.*

In literature regarding Virtual Reality (VR) and therapy, there is a distinction between "non-immersive VR" and "(fully) immersive VR" [11]. VR encompasses a wider range of technologies (Figure 2.2), including the use of TV screens or computer monitors placed in front of the user, where fully immersive VR refers specifically to the use of specialized Head mounted displays (HMD) in order for the user to be immersed in -and interact with a computer-generated environment. The Oculus Rift and HTC Vive (Figure 2.1) headsets are examples of such HMDs. User inputs can be extended by using controllers and other units, such as Vive trackers. Both VR Walk: Game and the BSc project, VR Walk BSc, are based on fully immersive VR, and any mention of VR will be in reference to this technology.



Figure 2.2: A selection of screen technologies under the VR umbrella. Left image: Cave VR, where the environment is projected onto the surrounding walls. Center: A non-immersive VR solution. Right: Fully immersive VR.

2.2 Serious games

The term 'serious game' is applied to video game software developed for use outside of pure entertainment or other hedonistic purposes. Some of the primary uses of serious games are within training, recruitment, education or, as is the case in this thesis, exercise and rehabilitation[29].

2.3 VR fitness and exercise

A whole sub-genre of VR-gaming, with dedicated communities (such as VR Fitness Insider), has emerged since VR-sets became commercially available. These often play into more or less fantastical power fantasies, emulate movements from sports or borrow from rhythm games. Audioshield and Beat Saber (Figure 2.3) are successful examples of such VR rhythm games[30, 31], while Box-VR combines elements from both boxing exercise routines and rhythm games [32, 33].

It is not uncommon to read reviews and statements from users who supplement their training or diet with VR fitness games, or even from users who, having previously lacked motivation, started exercising through these VR fitness games. A recent VR Fitness Insider article[34] provides details on how to get started and see results, while keeping oneself entertained through VR fitness. The author emphasizes that to see any lasting effects, this must be treated as actual regularly performed exercise combined with proper nutrition, not something to just try out every now and then.

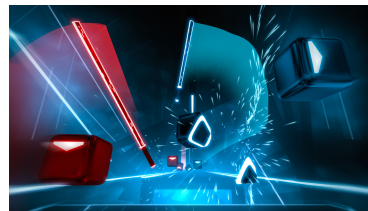


Figure 2.3: Beat Saber is a VR rhythm game that promotes physical activity. Image: Beat Games

2.4 Unity

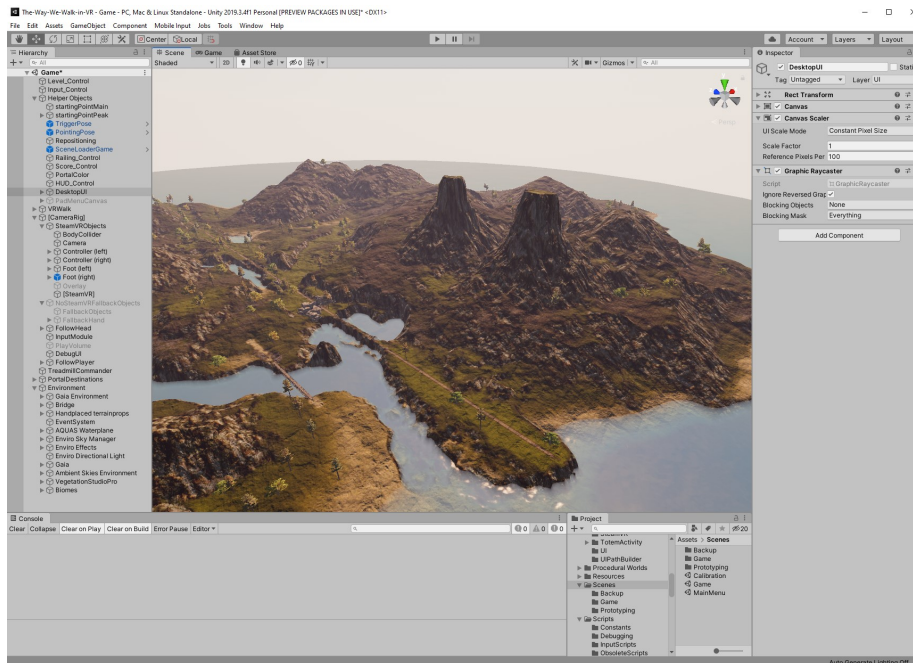


Figure 2.4: An overview of the Unity Editor.

Unity (Figure 2.4) is a cross-platform game engine developed by Unity Technologies[5]. We chose this engine both for VR Walk: Game and VR Walk BSc due to its performance, flexibility and ease-of-use. This is, in large part, due to its wide selection of VR compatible plug-ins and other assets found on the Unity Asset store[1] in addition to its support of the C# programming language. Unity's built-in terrain tools and optimized handling of said terrain was also greatly beneficial to the project.

2.5 VR Walk BSc

VR Walk BSc is a VR treadmill simulator developed as part of a previous bachelor thesis. The simulator was used to gather data regarding gait regularity and simulation sickness.

2.5.1 The VR Walk BSc software

VR Walk BSc is a fully immersive VR simulation with full 3D graphics (Figure 2.5). Its main purpose was to subject users to various physical tasks. The tasks were intended to measurably influence a user's gait. The simulation also contained a basic level-builder where tasks could seamlessly be strung together in a playable sequence. Users were able to interact with the environment using hand controllers and foot-tracking units.

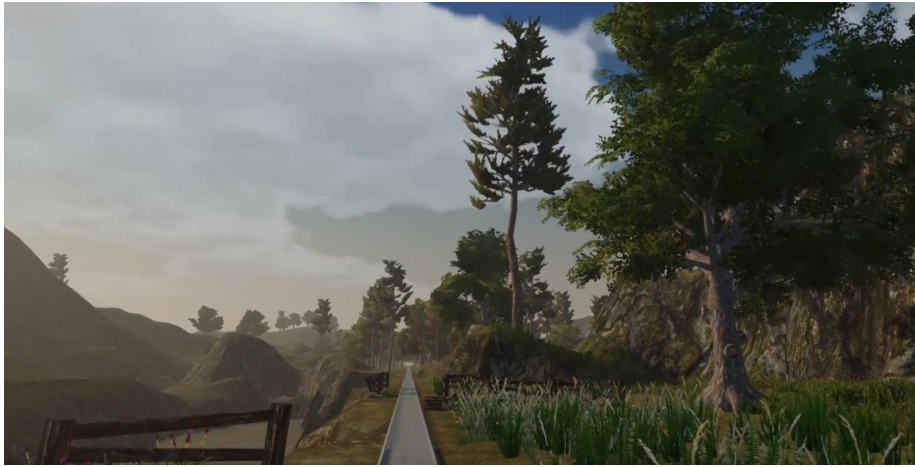


Figure 2.5: *The environment of VR Walk BSc.*

VR Walk BSc was developed for use with the HTC Vive HMD in the Unity[5] game engine and featured full integration for controlling a Rodby RL2500E treadmill[35]. The treadmill could be signaled to start, stop, and be set to specific speeds directly through the game engine. The simulation maintained accurate synchronization between the movement of the treadmill and a user's motion in the simulation environment. This allowed an emergency stop of the treadmill to be triggered by detecting if the VR-headset or the foot-mounted Vive trackers had left their designated safe areas.

2.5.2 Reception

The project was considered a success, both in terms of it being suited for Bovim's master's thesis[7] and based upon positive feedback from users, as well as being a "showpiece" for Western Norway University of Applied Sciences to use in tours, videos and presentations.

It has been demonstrated at multiple venues; for HVL's public bachelor thesis expo, a separate version of the project was developed for quick demonstrations on a smaller treadmill. Here it won best demonstration video among computer students. It was also included in HVL's stand at Forskningsdagene i Bergen (Bergen Research Days), where more than a hundred children were able to experience the simplified version of the product. The stand won the price for best stand of the fair, chosen by a panel of participating children[10].

When Haukeland University Hospital arranged a LAN-party for patients in treatment at their clinic for youth, this project was also included for the participants, friends and parents to try. The project has been featured in HVL's promotional material and tours, numerous presentations, expos, newspapers[36] and in a national physiotherapy publication[37].

In addition to this master's thesis, the BSc project serves as the groundwork for multiple bachelor-level projects both within computer engineering and physiotherapy.

2.6 Constraint-based gait rehabilitation

In physiotherapy, when assessing gross motor skills for rehabilitation it is common to separately evaluate the individual abilities that interact to produce movement. The abilities under scrutiny, being the *perceptual*, *cognitive* and *motor* abilities, can reveal in what areas a patient might have a reduced capability of motion. Assessing the perceptual ability revolves around discovering the extent a patient relies on visual information for balance and motion. A patient's cognitive ability pertains to how much deliberate attention must be given to the process of moving. The motor ability indicates the flexibility of the patient's motion and their ability to anticipate required changes to their motion[19].

Gait, the repeated pattern of motion a person's body follows when moving, has been shown to be affected when constraints have been placed on the person's motor skills in VR[7]. This indicates that there may be merit in constraint-based gait rehabilitation in VR. As such, in this thesis we will make use of constraints targeting the perceptual, cognitive and motoric skills, as well as mechanical constraints.

What follows is an overview detailing what changes each set of constraints introduces:

- **Perceptual constraints:** Changes to the sensory input of the user (i.e. visual distractions or limitations, or sound)
- **Cognitive dual task constraints:** Doing multiple simultaneous tasks where one or more task has a cognitively challenging aspect (i.e. logic puzzles, calculations, memorization tasks, verbalizing imagery or counting backwards)
- **Motor dual task constraints:** Doing multiple simultaneous tasks where one or more task has a physically challenging aspect, separate from the walking motion (i.e. arm movements and picking up or throwing objects)
- **Mechanical constraints:** Changes to the environment (i.e. adding obstacles or altering the walking path)

Motor and cognitive constraints are designated as dual task constraints as these involve activities that are performed as an addition to the walking activity. The mechanical constraint category serves to include all activities that directly manipulate the walking act directly, while the other constraint types interfere by introducing extraneous actions or information.

As an aside, treatment of bradykinesia[2] encompasses gait training with repeated motion and activities of a similar nature as mechanical and motoric constraints.

Chapter 3

Solution - VR Walk: Game

As stated in the research goals, the aim of this thesis was to create a VR simulator that satisfied in terms of the factors of the Extended Technology Acceptance Model, as well as player safety. This chapter details the development decisions we made, and the technologies and equipment we used, in our efforts to accomplish this.

Particular emphasis was placed on working towards creating a modular and expandable activity system, and on deciding which factors to take into consideration when designing interactions for VR.

3.1 Hardware

What follows is an overview of the hardware equipment used for the thesis work. This includes reasoning around equipment choice and notes on how the chosen hardware influenced the course of the work.

VR Equipment

Several factors had to be considered when choosing the VR hardware for the project. Solutions were considered in terms of accessibility, options for foot tracking, visual fidelity, treadmill compatibility and, to an extent, comfort.

The Oculus Quest[38], a standalone VR set with on-board hardware for positional tracking and graphics rendering, was an option available to us for the project. When considering accessibility, it might have been a good choice as it would preclude the requirement for a PC performant enough to support VR applications, and the lack of wires would be an added degree of player comfort. However, the lack of support for tracking units for foot positioning and the limited graphical performance of standalone sets made the Quest a less than ideal fit for our use. Additionally, direct treadmill communication needed to be through cable for the available mills, so movement synchronization between the virtual environment and the treadmill would have had to be done manually.

Another option available to us was to use the HTC Vive, a VR set using an external PC for graphics rendering. This set featured SteamVR support and used Lighthouses for tracking. This allowed for foot positioning through the use of Vive trackers attached to the user's legs, as shown in Figure 3.1. Being able to control the the treadmill from the computer would also be advantageous. In terms of comfort, the attached cable could be reduced to a minor inconvenience by fastening it at a point above or towards the back of the treadmill. Although heavier than the Quest, the HTC Vive was found to be the least front-heavy of the two, partially thanks to the cable. The more centered balance made it seem better suited for longer play-sessions and for handling jerky motions from walking.



Figure 3.1: A HTC Vive tracker strapped correctly to the player's ankle. Also pictured is one of the shoes used as reference for the virtual representation of the player's feet.

In the end, the HTC Vive was chosen for the project. This allowed us to develop for any SteamVR-compatible PC-based Virtual Reality (VR) sets, provided that features were not tailored specifically for the HTC Vive controllers or HMD. The increased performance from the desktop GPU allowed us to push towards creating a more realistic environment. Being bound to a sizable medical treadmill, the lack of portability of the external PC seemed negligible. In choosing the HTC Vive, the use of Vive trackers was also made feasible.

VR hardware

- HTC Vive sets, both original and pro editions with their respective original HMDs and controllers
- 2x Vive trackers with straps sized for use around ankles

Treadmills

Two of SimArena's treadmills were available to us for use within the project: a large Rodby treadmill, with a belt width of 1 m and length of 2.5 m, and a smaller portable Woodway medical treadmill with a belt width of 0.55 m, length of 1.73 m. Both treadmills had handrails going along the full length of the belt.

The Rodby treadmill was outfitted with a non-weight-bearing harness suspended from the ceiling, which would prevent a user from fully falling in the event of a stumble, and could also be used by a user to judge their

position on the treadmill by the tautness of the harness. Attached to the treadmill was an external panel used for adjusting speed and incline, and for running programmed exercise routines. The panel featured an RS-232 serial port, which was used for communicating with a connected computer.

The Woodway treadmill had a significantly shorter and narrower belt, which made it important to keep users centered on the treadmill as they were virtually blindfolded by an HMD. The treadmill featured a dead man's switch in the form of a circuit breaking magnet attached to a rope. The end of the rope would be securely fastened to the user's clothing with a clothespin. Pulling the rope too far would disconnect the magnet, immediately halting the treadmill belt. The magnet was strong enough to resist slight tugs, which could serve as a gentle cautioning to the user that they were falling behind, similarly to the harness of the Rodby treadmill.

The Woodway treadmill's main panel was positioned at the front of the metal frame of the handrails, and had an additional wired handheld controller with buttons for speed and incline adjustments. The main panel was not removable, and it was positioned in roughly the same area as where users might move their arms while playing. This forced the decision to raise the interactive elements of the simulator to around shoulder or head-height, as to avoid user injury or equipment damage. As a consequence, users with limited range of motion of their upper body might find some activities challenging to complete.

As with the Rodby treadmill, communication with the simulator was also done through an RS-232 serial port.

Treadmills

- Rodby RL2500E x 1000[35]
- Woodway PPS 55 Med[23]

Computer equipment

A capable desktop computer was lent to us by the Department of Computer science, Electrical engineering and Mathematical at HVL for use in development. Another desktop computer, situated at the SimArena rehabilitation lab, had moderately lesser specifications and was used for testing or demonstrating the simulator along with the treadmills. Having access to two computers with different hardware specifications provided us with helpful perspectives regarding performance optimization.

PC Hardware

- Primary development desktop PC with a Nvidia GTX 1080TI video card.
- SimArena lab desktop PC with a Nvidia GTX 1080 video card.
- 2x ultrabook PCs

3.2 Software

The software used in development was chosen based on a combination of factors. Among the major ones were availability and whether the software was capable of supporting the intended features and functionalities. To a lesser extent, the choices were influenced by our previous experience and familiarity with a given software, quality-of-life features and cross-software integration.

The following software was used:

Real-time / Video game engine

- Unity from version 2017.3.1 to 2019.3.13

Unity was chosen for the project in large part due to the VR Walk BSc software being created in the same engine. Porting the existing project to a different engine would have been a daunting task in the face of an already fairly time-sensitive development window. The engine's adaptability, robust VR support, well-stocked Asset Store[1] and our prior experience with it made it an ideal choice for the project.

Integrated development environment (IDE)s

- JetBrains Rider 2019
- Microsoft Visual Studio 2017 to 2019

Both Rider and Visual Studio were used as IDEs when writing C# scripts for Unity. Visual Studio, which came bundled with the Unity, was initially used as the sole IDE for the project before gradually transitioning to Rider due to its powerful Unity integration. The debugger in Rider featured a more accessible way of visualizing GameObject states, and was a more powerful tool for running and halting Unity scenes directly from the IDE.

3D asset creation tools

- Blender 2.8x
- Masterpiece VR
- Instant Meshes

2D asset creation tools

- Krita
- Affinity Designer
- Affinity Paint
- Gimp
- Filter Forge 4
- Blender 2.8x
- Adobe Substance Painter

A wide selection of software was used in order to create and manipulate models (Figure 3.2), textures, sprites and other 2D and 3D assets.

Collaboration and version control

- Trello
- Discord
- Github

Trello was chosen for its very accessible card-system, which made it an easy to use, synchronized online Kanban board.

The use of Github for version control was continued from the VR Walk BSc project. It seemed pertinent to have a comprehensive continuous changelog for both projects.

3.3 Third-party assets

Unity provides means of integrating third-party assets into the project. The following lists contains the assets used, and a brief description of what purpose they were used for, during the development of VR Walk: Game.

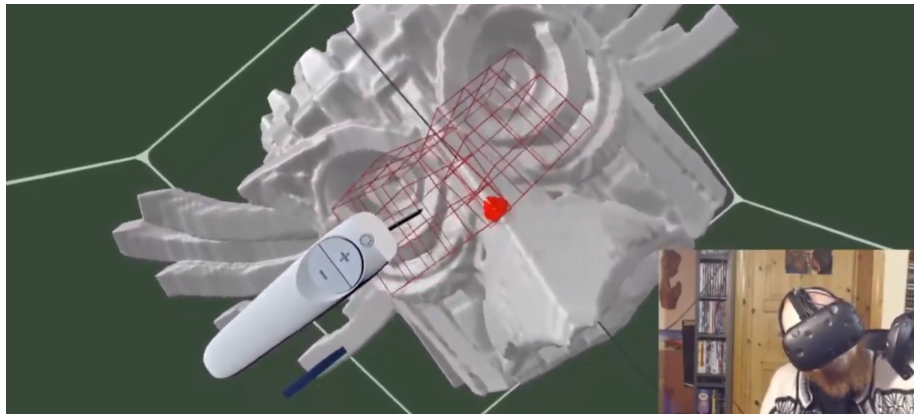


Figure 3.2: *Sculpting a prop for the totem pole memorization and building activity in VR, using Masterpiece VR. At later stages, the real-time optimized versions were created and UV-unwrapped using a combination of Instant Meshes and Blender before they finally were texture painted in Blender. The final prop can be seen in Figures 3.12 and 3.15.*

From the Unity Asset Store[1], parts of the following asset packs were incorporated:

- Valve Corporation - **Steam VR Plugin:**
For two-way interaction between SteamVR and Unity.
- Demigiant - **DOTween (HOTween v2):**
For simplifying setting up a single, or sequence of, simple animations.
- Unity Technologies - **TextMesh Pro:**
For general text creation.
- Unity Technologies - **Unity Particle Pack:**
A wide range of sample particle effects that we used elements from. An example of this is making smashing and collecting objects more forceful and satisfying by adding dust and smoke on breaking.
- Procedural Worlds - **Gaia 2 - Terrain & Scene Generator:**
Used to procedurally create a terrain and distribute textures on it as a starting point before making additional adjustments manually through Unity's built-in terrain tools. Includes several sample asset-packs.
- Procedural Worlds - **CTS 2019 - Complete Terrain Shader:**
Higher quality materials, including shaders and textures, for the terrain.
- Hendrik Haupt - **Enviro - Sky and Weather:**
A procedural sky and weather system. Used for creating the sky for the day-night cycle and for environmental lighting and weather conditions.
- Procedural Worlds - **Ambient Skies - Skies, Post FX, Lighting:**
For setting up over-all color grading and other post-effects.

- **AWESOME TECHNOLOGIES - Vegetation Studio Pro:**
A powerful tool for dynamically scattering(distributing) and culling(hiding and discarding) environmental assets, including grass, rocks, trees, plants and bushes. Used for setting up the biomes, detailed in the "Vegetation biomes" section.
- Dogmatic - **AQUAS Water/River Set:**
Used for shading the water surfaces.
- NatureManufacture - **Meadow Environment - Dynamic Nature:**
Additional grass and foliage to place through Vegetation Studio Pro.
- NatureManufacture - **Advanced Foliage Pack 2.0:**
Additional foliage to place through Vegetation Studio Pro.
- SpeedTree® - **Free SpeedTrees Package** (included with Gaia):
Various trees to place through Vegetation Studio Pro.
- MotuProprio - **Real Rocks** (included with Gaia):
Large rock assets to place through Vegetation Studio Pro or by hand. Used for creating additional points of interest and for the stone gate surrounding some of the portals.
- 3DForge - **Village Exteriors Kit** (included with Gaia):
Includes logs, poles, planks, wooden fences and primitive structures. The roof of two chicken coops served as the platform where the HMD-wearing player is standing during main menu navigation. The throwable axe was sourced from a log splitting prop and the wooden bridge was built by picking apart horse stables.
- Imphenzia - **Universal Sound FX:**
For the majority of the sound-effects used.

From the GameDev Market[39], the following non-Unity specific assets were used:

- GamemasterAudio - **Pro Sound Collection:**
For the remaining sound effects and background audio.
- Rexard - **Gui Icons:**
Used for activity and pathbuilder icons for half of the activities and served as stylistic inspiration while making the remaining activity icons.

3.4 User safety

The safety of the users has been the main deciding factor for the tasks presented to them. During VR Walk: Game, both visual and physical measures were taken. By having a virtual representation of the treadmill handrails, the users knew where to grab on to if they were off balance. There is an option to hide these menus if an experienced user wants to increase their immersion in the virtual environment.

As a visual indicator that the user should re-position themselves, a virtual colored grid-pattern will envelop the user until correctly positioned inside it. If the user's head, tracked by the HMD's position, is too far outside of this grid, an emergency stop will be triggered. As shown in Figure 3.3, the same applies for the user's feet, which are tracked using Vive trackers. When the emergency stop is triggered, both the treadmill and the forward motion in the virtual environment will stop. To avoid an unexpected restart, the user is required to move back inside the "safe area" before the treadmill can be started once more.

When dropping a virtual item during an activity, we observed a natural reflex to want to crouch down to pick the item back up, even if the treadmill was moving. Therefore, several steps were taken to discourage this. Depending on the current activity, the dropped item will either float back into reach, reappear in front of the player or be attracted to the player's hand by holding a button on the controller.

In addition to software features, external safety measures such as using a non-weight-bearing harness, a dead man's switch attached to the user's clothes as well as being monitored, guided and, if needed, steadied by the person supervising the activity.

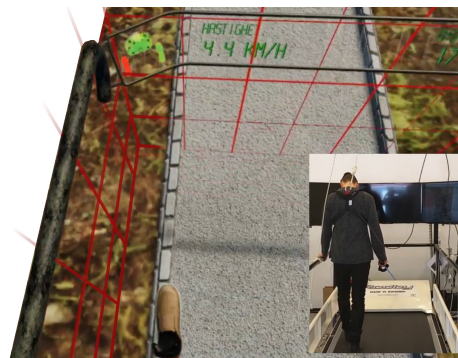


Figure 3.3: *The user steps too far to the left on the treadmill in VR Walk BSc. The treadmill has stopped and the user is alerted to why, with a red grid and an icon on the virtual display.*

3.5 The visuals - Choices, tools and technologies

During early testing when developing VR Walk BSc, we found that determining scale and distances was difficult in abstract or overly simplistic environments. Because of this, we went in a more realistic, if still simplistic, direction. As further detailed in the bachelor thesis[6], the final look was

achieved by using Procedural Worlds' terrain generator, Gaia, as a starting point. The weather and day-and-night cycle was achieved with Hendrik Haupt's Enviro. The water surfaces used Dogmatic's AQUAS Water/River Set shader. Bespoke elements, such as the feet, paths, activity props and portals were created from scratch.

For VR Walk: Game, ambitions for the visuals were set a substantial notch above what was achieved during BSc. As part of this process, about a week was spent experimenting with Unity's less matured scriptable render pipelines. For the high definition render pipeline, we were unable to achieve the same visual quality and performance (in terms of achieving an acceptable frame-rate) without sinking numerous hours into this transition. The light-weight render pipe offered similar frame-rates as the, then referred to as, standard render pipe. However, volumetric lighting was not supported. When beams of sun light shone through the trees and foliage, as shown on the front cover illustration, it would frequently have an observed positive impact the player's perceived enjoyment. For this feature, support for volumetric rendering was a requirement. Neither of these render pipelines have had enough time to achieve the same Unity plug-in support as the "standard", so we shelved further experimentation and returned to the "standard" render-pipe.

As for actually lifting the visual quality, a number of additional add-ons were utilized. Procedural Worlds' CTS 2019 (Complete Terrain Shader), with its quality shaders and textures, had a major impact on how the ground, mountains and other terrain features looked. For color grading and over-all lighting mood, another of Procedural Worlds' tools, Ambient Skies, was used. The vegetation density and quality was upped using Awesome Technologies' Vegetation Studio Pro with a number of asset-packs for a wider selection of visually pleasing grass, plants and shrubs. The increased visual quality had an apparent impact in regards to perceived enjoyment during both internal and external testing.

3.5.1 Vegetation biomes

Shortly after adding higher quality vegetation and foliage assets to the scene it became apparent that the software's performance was greatly affected by the quantity of the assets. After experimenting with reducing the vegetation density across the terrain, we found that at the point where performance became acceptable, the vegetation looked too sparse to be convincing. Instead we decided to increase the density of the vegetation immediately next to the player's path and decrease the density everywhere else. To achieve this, we used Vegetation Studio Pro's biome system. The biome system allowed us to designate sections of the terrain, assign assets to them, and customize the frequency, size and orientation of each asset. The system also allowed for gradual transitions between different biomes, as shown in Figure 3.4.

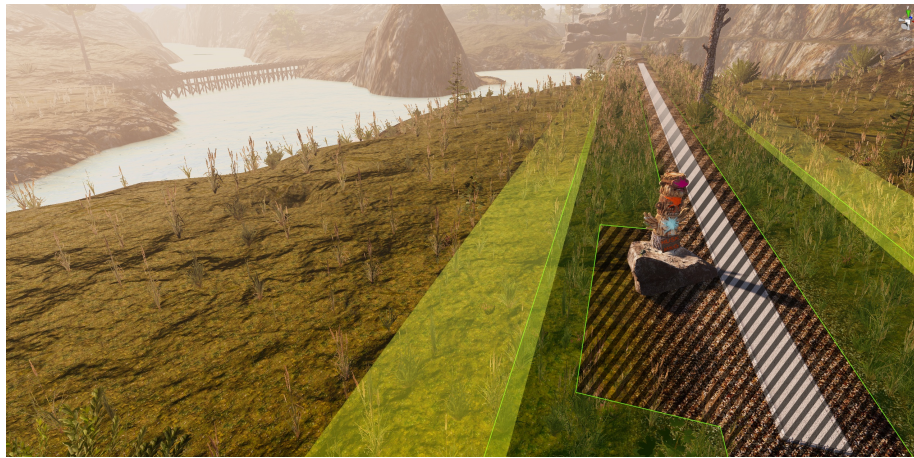


Figure 3.4: *Vegetation density by biome. Areas along the path where activities would otherwise be obscured are without vegetation. Dark green areas are densely populated while the player is near. Outside this is a transition area before fading into a sparsely populated terrain.*

A biome completely devoid of foliage assets was assigned to the entire strip of terrain where the player's path could appear. The biome extended roughly 1 meter to each side of path to prevent any foliage assets from distracting the player, or blocking the view of important game-play objects. Additional biomes were assigned on either side of the path biome. These biomes were set to be roughly 4 meters wide and had a fairly high foliage density. This was found to give us an appropriately convincing amount of vegetation without hampering performance. By having a high frequency of tall reeds and bushes in the foliage, we attempted to give the player the impression that the dense foliage stretched further out from the path than it actually did. In reality, outside the dense biomes, the rest of the terrain used a biome with much more sparse vegetation. However, we made use of the biome transition functionality to avoid having a drastic and unnatural cutoff in the vegetation.

Additional biomes were used for reducing or removing foliage on hill-tops, around rock formations, the reference totem pole pedestal for one of the activities and the area around a wooden bridge.

3.6 Interaction design

The following sections delve into the various instances of interaction between the software and the user, the technologies used to facilitate them, and how the VR setting shaped their implementation.

3.6.1 Input handling

SteamVR configuration

The SteamVR plug-in for Unity is the interface between Valve Corporation’s SteamVR software for VR hardware support and the Unity game engine. The plug-in provides 3D models of hands and VR controllers and input handling for the controllers.

The SteamVR software handles input, tracking, positioning and image projection for VR sets that require an external computer for graphics rendering. SteamVR couples the VR hardware with any software or application that has integrated it’s support. SteamVR uses the OpenVR[40] API to modularly connect with any supported VR equipment.

The version of the SteamVR plug-in used in the VR Walk BSc project required that scripts using controller input needed access directly to a SteamVR script attached to the controller’s GameObject. Additionally, the way input was registered required that each script (that needed input) checked the state of the corresponding button signal for each frame rendered by the game engine.

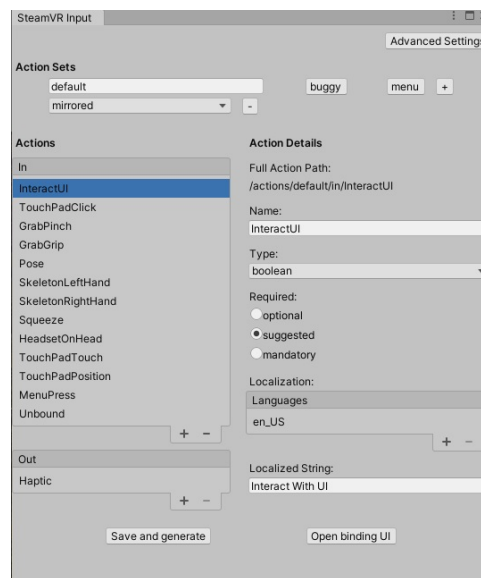


Figure 3.5: Configuration page for the SteamVR action sets. Each action is given a string identifier and a type for the input value. Simple buttons use boolean values. Vectors are used for pressure sensitive elements and thumb-stick movements.

By the time development began for the thesis project, the SteamVR plug-in had seen considerable changes and updates. The change most immediately noticeable was that the previously used system for input registration had been made obsolete. It had been replaced by a system using interchangeable action sets where each action could be freely bound to preferred buttons by the end user. The action sets were created and configured from a dedicated view in the Unity editor, shown in Figure 3.5. This new system made it possible to remap input bindings and use controllers with different buttons and layouts without having to rewrite any code. This made it considerably less complicated to configure the inputs for additional VR controllers.

With the new input system in place, the first task of development was to migrate all instances of input handling to it. While the previously used

method of checking the state of each input for every frame was still supported by the actions system, we decided instead to implement another method that would have notably fewer method calls per rendered frame. The method we used was centered around having a globally accessible singleton containing a reference to the current action set and its individual actions. Methods that handled input were then subscribed to their corresponding actions as event listeners, and unsubscribed when the method should no longer be invoked. The methods could be set to listen for specific states, such as a button being pressed down, held or released.

Another notable feature with the new system was the ability to enable separate action sets for different contexts. The action sets were ordered by priority, and when multiple sets were enabled each entered input would use the action set with the highest priority. By having an action set with a higher priority, but a limited number of actions, we were able to contextually change what certain inputs did, without interfering with other input functionality. This was put into use for the laser pointer used for the in-game treadmill panel. This allowed us to use the same trigger button for menu interaction while the laser pointer was active, and picking up interactive objects when it was not.

Controller representation



Figure 3.6: *On the left: the controller representation utilized in VR Walk BSc, an interactive model from the SteamVR plug-in featuring all buttons present on the physical controller. On the right: The controller representation currently used. Hand models from the SteamVR plug-in that animate in response to controller input and can be manually posed.*

In the VR Walk BSc project, the handheld HTC Vive controllers were represented in the virtual environment by detailed and interactive models of the actual controller, seen in Figure 3.6. This was the default controller representation provided by SteamVR, and we argued that it was suitable for the software as it provided users unfamiliar to VR and HTC Vive controllers with virtual representations of the button positions on the physical controller.

There were several reasons why a visible controller model was desirable. In particular, because the button used for starting and stopping the treadmill was located on the underside of the grip, around where the middle joint of the middle and ring fingers might rest, and was slightly challenging to press. These properties were also the reason why the button was chosen as the treadmill toggle in the first place; it was thought less likely to be pressed by accident. At the same time, several users had found the button difficult to discover, even when given spoken instructions. For some users, we observed them tensing up during a challenging activity and accidentally holding in the grip button, resulting in the treadmill stopping unexpectedly.

When transitioning to the latest version of SteamVR at the start of development for the thesis project, we discovered that high quality, fully posable hand models had been added to the plug-in, as seen in Figure 3.6. After experimenting with including the hand models we experienced that they aided player immersion and decided to implement them into the simulator. There are also indications that having morphologically accurate hands in VR gives an increased sense of ownership of them to the user [41], which might also improve the sense of immersion. It was also initially decided to keep the virtual representation of the controllers along with the hands.



Figure 3.7: *Custom hand poses for various objects and interactions.*

As new activities were added to the simulator, poses for the hand models, shown in Figure 3.7, were created for the objects a player could pick up and manipulate. Interacting with objects required the virtual controller model to be hidden for the duration the other object was held. This was slightly jarring for the immersive experience, and was one of the deciding factors in choosing to remove the controller models entirely.

Later in the project, the input method for controlling the treadmill and player movement was changed from using specific controller buttons to using a laser pointer and virtual buttons. This negated the strongest argument for keeping the controller model visible, namely giving users an accessible view of the relevant buttons. Removing the controller model would also permit smoother transitions between hand poses when picking up objects, and might better serve immersion overall by removing the persistent reminder to the user that they are holding a game controller.

There was some concern that the default splayed pose of the controller-less hand would feel less convincing than the previous pose, which closely mimicked a hand holding a controller. However, no feedback from play-tests or demonstrations indicated that it was a notable issue, thus the hand models without visible controllers was the representation we decided upon. The only situation that calls for having the controller displayed are the rare occasions where the player is prompted to hold a button when an expected input is not given. This might be when a thrown object is not recalled or when an activity-object is not picked up. In these cases, the controller would be shown with a semi-transparent material, while the relevant button would be indicated by flashing in yellow.

Controller input

An important part of the perceived ease-of-use is how the user physically interacts with the software.

The simulator was intended to be usable to user groups of various levels of video game experience. Therefore we worked towards making the controller inputs as simple as possible. As a result, the majority of in-game interactions are done with one button. The chosen button, the trigger button, is universally present on all current VR controllers, and is situated on the underside of the given controller, in a position where the index finger naturally rests. This button is primarily used for interface- and menu navigation and interacting with game-play objects.

A separate designated menu button is used for toggling an extended menu for the VR user during a play-through. This type of button is present on contemporary controllers, though their positioning varies. While these are the default bindings, buttons can be rebound using the SteamVR Bindings interface. In theory (as we have not had the opportunity to test this), this should allow the use of any SteamVR-supported controller to be mapped to the game.

Through SteamVR we have accurate positional and rotational tracking of the VR equipment, including the controllers. This tracking allows users to complete activities that require precise and complex motions in VR. The SteamVR plug-in also provides an object interaction system that, among other features, simplifies picking up and throwing objects with believable velocity calculations.

3.6.2 User interfaces

This section describes the various instances of interaction and information exchange between the user and the simulator.

Main menu

The main menu is accessible to users both when using VR and when using keyboard and mouse with a desktop monitor. On the monitor the menu is positioned in screen space, meaning that the size of every element is defined in pixels relative to the window resolution. In VR, placing user interface (UI) elements in screen space is discouraged, as it gives the impression of the element being uncomfortably close to the viewer's eyes. Instead, the menu in VR uses world space for its positioning. This allows the UI elements to be positioned in 3D-space, allowing the user to view them in a more natural perspective.

The desktop menu makes use of Unity's UI toolkit. Among other things, the toolkit includes buttons, sliders, toggles, and text fields, all of which can execute and be accessed by custom scripts. While the VR menu also makes use of the same UI toolkit, a laser pointer from the SteamVR plug-in was used to allow the user to interact with the UI elements.

Ideally, a desktop user and a VR user would access the same instance of the menu, but as the VR menu needed to be positioned in world space, two separate menus were used instead. The two menus are partially synchronized; the opening of sub-menus and basic setting changes are reflected on both instances, but certain features, such as the path builder menu, were deemed too complex to be feasible for use in VR. This was partially due to the precision required for certain elements and the need for text input.

Treadmill panel

During a play-through, in addition to being surrounded by virtual representations of the treadmill's handrails, the user will also have access to an interactive information panel. As seen in Figure 3.8, this panel features button controls for starting or stopping the treadmill, and for adjusting the walking pace. Additionally, there are indicators for whether the HMD and foot-mounted Vive trackers are inside the safe play-area and text displays showing the current set pace, distance walked and remaining, and statistics for the activities.

The control buttons are activated by a raycast laser pointer emitted from the hand controllers. When the raycast hits a collider on a specified layer, a pointer indicator appears on the surface of the collider's object. When the indicator touches a button, the button is highlighted and can be activated by pressing the controller's trigger. As additional feedback to the user, the hand



Figure 3.8: *The treadmill front panel, featuring buttons for increasing and decreasing movement speed on the left, a start/stop button at the center, and a statistics field and safety indicators on the right.*

models attached to the controller adopt a pointing pose to indicate where the raycast will hit, and a trigger-pull is indicated by a thumb gesture.

Treadmill controls were initially bound to button-presses on the controllers themselves, but to avoid issues with other controller designs than the ones available during development, the controls were redesigned to be controller-independent, and subsequently compatible with all VR motion controllers supported by SteamVR.

In-game statistics for the operator

By hitting the F1-key during game-play, the operator can pull up a statistics overlay . This overlay displays the states of settings, the set speed of the treadmill, time elapsed, step counts and statistics for the various activities in the current play-through.

The overlay was added as it was thought to be beneficial for personnel monitoring a VR user to have access to more in-depth statistics that would have made the VR interface more cluttered and overwhelming. During development it also became helpful in troubleshooting bugs.

Instructional text boxes

Throughout a play-through, players will encounter situations or interactions where all necessary information can not easily be gathered from the environment alone. While we strove to craft interactions that were intuitive and simple to grasp, feedback received during the VR Walk BSc revealed that certain situations left users confused or unprepared for what was happening. To remedy this, simple text instructions were added.

Initially, in the BSc project, the text was displayed in a large font with a dark outline, floating around 5 m in front of the player at eye level. During this thesis' development period, the display format was revised to make it both more legible and less obtrusive. The text had previously tended to blend with the color of the sky in certain conditions, and it also obscured items of interest further ahead on the path. Instead, the text was moved to an angled plane with a strongly contrasting background, positioned slightly above the treadmill panel at roughly 1.5 m height, as shown in Figure 3.9.



Figure 3.9: An instructional text box, displayed ahead of the balance ball activity.

This change made the text legible in all lighting conditions, and allowed users to shift their attention away from the text once they were finished with it.

3.7 Treadmill integration

A vital function for the project software was the ability to simultaneously and synchronously control both motion in the virtual environment and the treadmill the user would walk on. A seamless way of achieving this was to allow the game engine to send commands directly to the attached treadmill. This was facilitated by configuring interfaces for the treadmills, where input registered in the game engine could be relayed as commands to the treadmill by serial communication. The commands required included starting and stopping the treadmill and being able to set a specific treadmill running speed. While both of the treadmills supported inclining the belt to an upward slope, we opted not to utilize this as it would be incompatible with the path system, as is detailed in the section on modularity. The connected treadmill had to be selected from a drop-down menu, shown in Figure 3.10 in the main menu settings.

Synchronization

The different treadmills accelerated and decelerated at different rates, so the in-environment acceleration was made to reflect the corresponding rate depending on which treadmill interface had been assigned.

The system for calibrating the directional alignment between the virtual environment and the treadmill was carried over from VR Walk BSc and reconfigured for SteamVR's refurbished input system.

The calibration system takes full advantage of the precise positional data of the Lighthouse-tracked motion controllers. The controllers are placed on symmetrical markers on either side of the treadmill belt and the controller positions are registered. A vector is drawn between the two registered points. The cross product between this vector and a vector facing directly upwards in the 3D environment yields a new vector that is parallel to the treadmill band. This resulting vector is then reversed or left unaltered depending on the orientation and left/right assignment of the controllers. A placeholder GameObject representing the virtual treadmill's orientation in the 3D environment is rotated towards the treadmill vector and repositioned to be directly between the two registered controller positions. This rotation and the GameObject's position relative to the VR play area are then stored in the Windows Registry. When starting a new play-through of the simulator, the values are fetched and used to orient the VR play area and virtual treadmill representation accordingly.

A full walk-through of the calibration sequence is included in the appendix' setup and calibration manual (in Norwegian).

Rodby RL2500E x 1000 interface

The interface for the Rodby[35] treadmill was carried over from VR Walk BSc, for which it was provided by a former student. It features one-way communication from the computer to the treadmill. The interface uses a buffer for commands, where a small delay is placed between each command to avoid data loss. It contains a dictionary of commands for the treadmill, including the relevant commands for starting and stopping the treadmill, and parameterized setting of treadmill speed. The interface and Rodby treadmill have proven to be reliable throughout both the BSc and the thesis project, having very few, if any, instances of dropped commands.

Woodway PPS 55 Med interface

VR Walk BSc originally supported only the Rodby interface. The Rodby commands are incompatible with the Woodway PPS Med. While demonstrating on a Woodway treadmill, we resorted to writing input-macros

for controlling the treadmill through Woodway’s official control software as a quick and temporary, but fragile solution.

During the time between VR Walk BSc and VR Walk: Game, a group of bachelor students were tasked with implementing the controls for the Woodway PPS Med treadmill[23]. They were able to put together a brief command line proof of concept. As such, one of the very first tasks tackled during VR Walk: Game’s development was to fully integrate direct controls of the Woodway treadmill, without the need to pass the commands through another software.

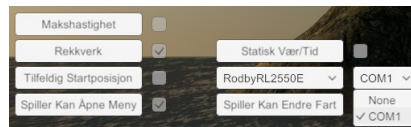


Figure 3.10: *The detected serial-ports (COM) are listed and can be selected. This, combined with treadmill selection, determines the interface used.*

There were issues with both the official control software and the aforementioned proof of concept, where giving the start-command was unreliable, sometimes up to 6 times in a row. This resulted in the motion through the virtual environment starting, but without any movement from the treadmill. This disconnect from the expected behaviour was both immersion breaking and uncomfortable for the player. We looked into if there were any command acknowledgements we could listen for over the protocol, but in the end, we remedied this by repeating the start-command 6 times with a fraction of a second in between. While a bit of a hack, this remedy has not yet failed.

Dummy interface

In cases where no COM-ports are detected when requesting to have them listed, or if "None" is manually selected in the port-selection drop-down list, a dummy treadmill interface will be utilized. This interface will accept the same commands as the previously mentioned interface, but will make no attempt at serial communication. This way, the virtual treadmill interface will only affect the virtual forward motion. This was useful when none of the supported treadmills were present and connected. It also left room for manually synchronizing the running speed of an unsupported treadmill with the motion through the virtual environment.

3.8 Constraints and activities

Several game-play activities were developed during the course of the project. These activities were designed with a dual purpose; to engage and challenge the user with varied tasks of scaling difficulty, and to serve as appropriate exercises in constraint-based gait training (see 2.6). Interactions with the environment during the activities were crafted to be tactile and responsive in order to reinforce the immersion of the VR experience.

Index of activities arranged by constraint category:

- **Perceptual constraints**
 - Glass bridge
- **Cognitive dual task constraints**
 - Totem pole memorization and building
 - Coin collecting
 - Axe throwing
- **Motor dual task constraints**
 - Coin collecting
 - Axe throwing
 - Balance ball
- **Mechanical constraints**
 - Narrow paths
 - Obstacles on the path
 - Pre-distributed footprints

3.8.1 Perceptual constraints

Significant changes to what the user perceives are classed as perceptual constraints. In the context of VR, the perceptive input is limited to what the user is seeing or hearing, as well as haptic controller feedback (vibration).

Glass bridge

One of the visual constraints carried over from VR Walk BSc was to introduce users to an exposed glass bridge suspended between two hill tops at roughly 200 meters above the ground, shown in Figure 3.11. The bridge walkway is a semi-transparent glass material and the bridge makes use of the full width of the treadmill, while offering no handholds.



Figure 3.11: *The glass bridge as seen from the starting hill top.*

Access to the bridge is given by walking through a portal-plane placed on a corresponding path segment and being transported to the starting hill top. The portals were given a refractive surface with a ripple-motion (animated through the use of a sprite-map technique with UV-offsets) and a distinct sound as the player is approaching. When traversing the portal, the user's vision is briefly faded into a solid color to lessen discomfort when being rapidly relocated. The fade color is approximately matched to the current brightness of the skybox, in an effort to avoid unpleasant transitions between highly contrasting colors.

The exit portal-plane is, by default, placed on the opposite hill top, but it can be placed within certain limits on the glass bridge itself as a way to shorten the bridge sequence. Both Bovim[7] and ourselves found, during testing, that the users got accustomed to the change of height after only a few dozen meters. Therefore, it was sometimes beneficial to shorten this section, unless performing other activities for the remainder of the bridge length.

3.8.2 Cognitive dual task constraints

Cognitive dual task constraints are activities where tasks with a cognitive aspect are done concurrently with walking. Cognitive tasks include word-to-image matching, memorization, mental calculation and similar activities.

In geriatrics, a commonly used tool for assessing a patient's cognitive abilities is the Mini-Mental State Examination (MMSE). Among the suggested tasks is to have the patient memorize a list of words and recall them at a later point in the survey. Another suggested task is to describe the properties of an object[42]. Rather than using a floating list of words, we wanted to make this a more immersive and tactile task, thus the totem pole memorization activity was conceived.

Totem pole memorization and building



Figure 3.12: *The player builds a miniature totem pole from memory.*

The totem memorization activity was developed primarily as a cognitive constraint. A totem pole appears alongside the path while the user is occupied with other activities. The totem pole is generated from a set of totem pieces with distinct silhouettes (Figure 3.13), designed to be told apart at a distance. The user is tasked with memorizing the order and appearance of the totem pieces. Later in the path, the user is then presented with a set of miniature totem pieces and is challenged to recreate the previously presented totem pole from memory by stacking them, as seen in Figure 3.12.

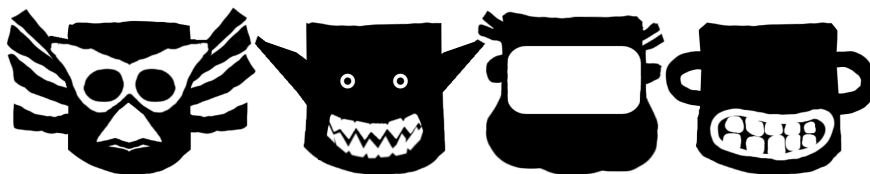


Figure 3.13: *Each totem pole piece was 3D sculpted with a distinct silhouette and contrasting features for easier recognition from afar.*

As an optional element of difficulty, the pieces of the presented totem pole may be individually painted and the user is tasked with painting the miniature pieces the correct colors after successfully stacking the totem pieces together.

The colors for the paints were chosen based on charts[12] over color distinction for color blindness. One of the charts used is shown in Figure 3.14. The colors warm orange, magenta and pale cyan were selected as they appeared to be clearly distinguishable for the majority of dichromacy color deficiencies.

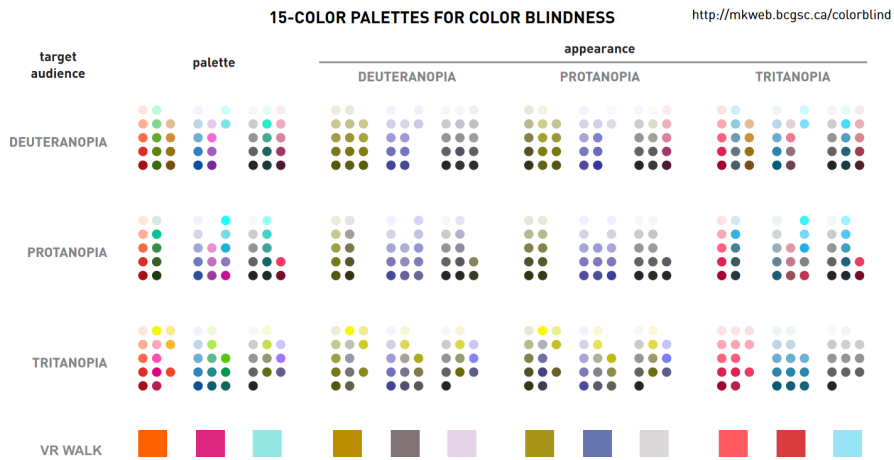


Figure 3.14: A chart over three color palettes suited for various types of dichromacy color deficiencies[12], with the totem paint colors added at the bottom row. While more care could have been taken with regard to tritanopia, it is a very rare condition[13].

From testing, we found that users not familiar with the shapes of these pieces still struggled with recalling their presented order. This led to the addition of a difficulty-setting where each shape was paired with the image of a card suit, prominently stamped onto its face. This way the user could choose to remember a sequence as, for instance, cyan-heart on top of magenta-diamond. The stamped and unstamped versions of the totem pieces are shown in Figure 3.15.



Figure 3.15: The unpainted totem pieces as presented to the player. The card paint marks are displayed in white when the card suit option is enabled to allow players to still use them for recognition.

The acts of picking up, placing and painting the totems may be challenging enough in itself to certain users, that this task skirts the line of serving as a motor dual task as well.

To be able to add the paint on top of the totem piece texture, a shader with blending between the main color texture and masked, painted color was created using Unity’s Surface Shader system[43]. With this approach, boiler-plate lighting code can be inherited from Unity’s standard shaders, resulting in a complete HLSL-shader (High Level Shading Language shader). In the customized shader, the painted part is masked off using a black and white stencil, which is only placed once and not repeated anywhere else on the model, hence the additional UV parameters. To have the painted colors visible under all lighting conditions, the emissive (light-up) property was added to the shader to make sure the paint splashes were distinguishable under all lighting conditions. The remainder of the material properties were inherited from the default Unity shader, resulting in totem pieces that could look like either the left column of Figure 3.12 or the bottom row of Figure 3.15.

```

1  fixed4 mask = tex2D (_PaintMask, IN.uv_PaintMask) * _PaintStrength;
2  fixed4 a = tex2D (_MainTex, IN.uv_MainTex) * (1 - mask);
3  fixed4 b = _Color * mask;
4  fixed4 c = a + (1 - a) * b; // texture-map a (partially) replaced with b where
   ↪ masked by stencil
5  o.Albedo = c.rgb;
6  if (_EmissionStrength > 0){
7      o.Emission = mask * _Color * _EmissionStrength;
8  }
9  o.Metallic = _Metallic;
10 o.Smoothness = _Glossiness;
11 o.Normal = UnpackNormal (tex2D (_BumpMap, IN.uv_BumpMap));
12 o.Alpha = c.a;

```

Code Example 3.1: *The properties and color blending between paint splashes and the underlying totem piece texture. in this code example, 'a' refers to the wooden texture of the totem pieces, 'b' to the '_Color'-painted parts, and 'c' to the resulting combination of these.*

3.8.3 Motor dual task constraints

Motor dual task constraints are activities where walking and tasks of a motoric nature are done simultaneously. Any task requiring motion apart from the act of walking falls within this category.

Coin collecting

Coin collection (Figure 3.16) is a primarily motoric dual task with a cognitive aspect. The user is tasked with gathering coins by touching them with their hands. There is a correct and an incorrect variant of coins that the user must distinguish from each other. Collecting a correct coin yields points, while collecting an incorrect one is penalized. The correct variant is a shiny golden coin with a stylized portrait. The incorrect coin is more dim and corroded



Figure 3.16: *The player reaches to grab a randomly moving coin, while avoiding the ones with a skull motif.*

than its counterpart. It is also embossed with a skull face, intended to make it less appealing to collect.

When a coin is collected, it shatters, leaving debris and a particle puff of dust accompanied by a randomly modulated sound effect. The sound plays a role in giving the player feedback on if something positive or negative occurred. If a correct coin is collected, a gentle metallic jingling sound is played. Skull coins give a harsh ice crushing sound accompanied by a floating, red cross when collected. Both in the case of the coins and the targets of the axe throwing activity, the shattered versions are created by making cuts into the models, creating additional geometry in the cut areas. The broken pieces are determined beforehand by creating cuts and splitting up the model geometry based on a Worley-Voronoi noise algorithm[8, 9], resulting in models such as the ones shown in Figure 3.17.



Figure 3.17: *The shattered version of the activity props are fractured based on a Worley-Voronoi noise algorithm.*

There are four different movement patterns, either in both x and y-axis or only in the x-axis, that the coins can follow based on settings from the path builder. The first two determines their movement by reading off values from a Perlin noise-map, similar to the illustrations in Figure 3.18. By either controlling how fast one traverses along an axis of the map or how scaled down this map is, one can control how smooth or erratic the motion of a coin is along an axis. The map consists of values between zero and one. Values

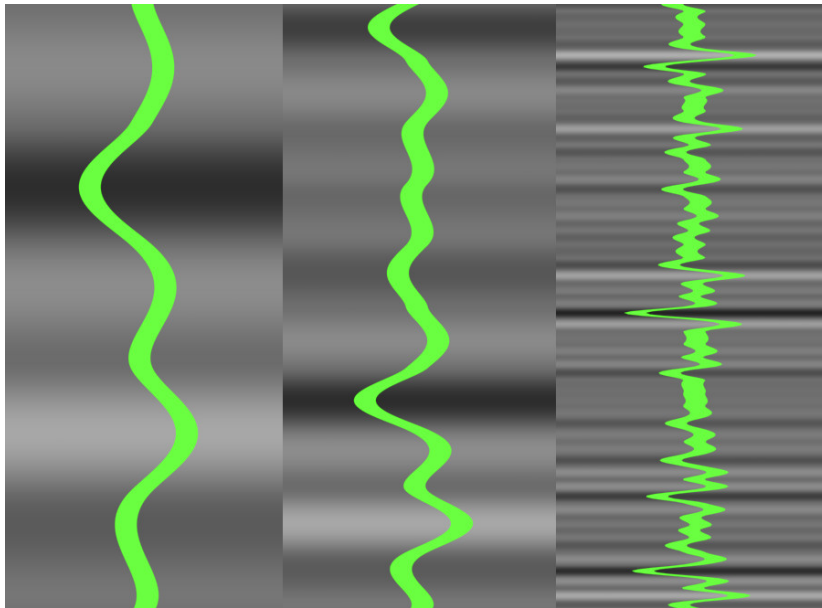


Figure 3.18: *By either controlling how fast one traverses along an axis of a Perlin noise-map or by altering the map's scale, it is possible to control how smooth or erratic the motion of a coin is along an axis. The illustrated maps are stretched in their x-axis for illustration purposes.*

below 0.5 are treated as offset to the left, and above 0.5 to the right. The range of motion is determined by multipliers on these values. The next two options follow a regular motion. For this, the offset values are based on a sine-curve rather than a noise map. There is also an option to have no motion.

```

1  if (_movementPattern == CoinMovementPattern.Random1Axis || _movementPattern ==
    ↪  CoinMovementPattern.Random2Axis) {
2      pos.x = _xOffset * (2 * Mathf.PerlinNoise((Time.time + _timeOffset) *
    ↪  moveSpeed, _coordOffset) - 1f) + _startOffsetX;
3      if (_movementPattern == CoinMovementPattern.Random2Axis)
4          pos.y = _yOffset * (2 * Mathf.PerlinNoise((Time.time + _timeOffsetHalf)
    ↪  * moveSpeed, 3f * _coordOffset) - 1f) + _startOffsetY;
5  }
6  else if (_movementPattern == CoinMovementPattern.Regular1Axis ||
    ↪  _movementPattern == CoinMovementPattern.Regular2Axis) {
7      pos.x = Mathf.Sin((Time.time + _timeOffset) * moveSpeed) * _xOffset +
    ↪  _startOffsetX;
8      if (_movementPattern == CoinMovementPattern.Regular2Axis)
9          pos.y = Mathf.Sin((Time.time + _timeOffsetHalf) * moveSpeed) * _yOffset
    ↪  + _startOffsetY;
10 }

```

Code Example 3.2: *The code for the coin movement positions for all possible movement patterns. The coordination -and time offsets are randomized per coin and ensures that the coins and directions do not follow the same cycles.*

As for why coins were decided on for this activity, coins have a long history as a collectible for accruing points in video games, dating back to at least the early 80s[44] (and the association can be dated back 2000 years[14] if considering coin collecting in the real world, as elaborated in Figure 3.19), and will likely be recognized as such by users who are familiar with video games. The skull motif for the "bad" coins was chosen because of its symbolic meaning as hazardous, i.e on warning labels for chemicals. The eye-holes make the "bad" coins' silhouettes separable from the "good" coins in difficult lighting conditions, while at the same time, not being too easily distinguishable.



Figure 3.19: *Possibly the first recorded instance of a coin collector, Caesar Augustus (63 B.C. to A.D. 14)[14]*

Axe throwing

One of the more intensive motoric tasks developed for the project is the axe throwing activity. It challenges the user to swing or throw a large axe with precision to hit near and distant targets. As an added cognitive aspect, the user is encouraged to avoid hitting certain indicated targets. As with the coin collecting activity, negative targets are recognizable because of their skull shape, but with a material similar to the positive targets.



Figure 3.20: *A wooden bridge serving as an alternative play area for the axe throwing activity. The wider bridge and lower altitude creates a more comfortable environment for users with vertigo.*

By default, the axe activity is intended to be performed once the user is familiarized with walking on the elevated glass bridge, though for users averse to heights it is possible to do the activity in a less vertically challenging environment (Figure 3.20).

In this activity a randomized pattern of targets (Figure 3.21) is generated. The selection of target formations is based on the chosen difficulty level. On the lowest difficulty, the generated pattern will only consist of single targets within easy reach of the path. On higher difficulties, larger clusters of targets may be added to the generated pattern. Targets may also be placed further

from the path, requiring the user to throw the axe in order to hit them. In order to disincentivize users from trying to pick dropped items from the ground while walking, a "fetch" behaviour was added to the axe. The behaviour involves having the user press and hold a button to send the axe flying back to their hand. This also allows the axe to hit targets on its return, adding another layer of potential mastery.



Figure 3.21: *Targets for the axe throwing activity. The spheres are positive targets and the skulls are negative ones. Clustered target patterns are seen in the distance.*

When a target is hit, it shatters violently, potentially setting off a chain explosion due to the shattered pieces striking secondary targets. In terms of perceived enjoyment, this feels particularly rewarding. A selection of randomly modulated sounds of rocks and ice shattering accompanies the explosion, as does a particle puff of dust. As with the coin collecting activity, the shattered version of the targets are determined beforehand by creating cuts and splitting up the model geometry based upon a Worley-Voronoi noise algorithm.

Through the difficulty settings or sliders in the path builder, the operator is able to alter the amount (in path length) and density of the targets, percentage of negative/invalid targets and how far from the path both the striking and the throwing targets are spawned.

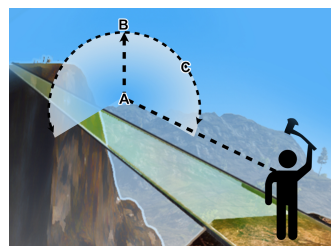


Figure 3.22: *Placing an axe-throwing -or a striking target.*

Figure 3.22 illustrates a simplified view of how one such target is, or cluster of targets are, positioned; From approximately shoulder height, every step-length of A in forward direction, the position is offset upwards within the randomized range of B and then rotated around A by a randomized amount of C. Throwing and striking targets uses different ranges for B.

Balance ball

One of the first constraints developed for VR Walk BSc was an adaption of the classic egg-and-spoon race. As such, this task has benefited from several rounds of observation and user feedback. The goal of the activity is to balance a ball on top of a plate for 30 meters, between a start and a finish-line. The challenge is to keep a steady hand and sufficient eye-hand coordination while walking on the treadmill. The intuitiveness of this challenge (PEOU) was observed during expos to be highly dependent on the user's age-group and nationality. Norwegians above a certain age are likely to have encountered egg-and-spoon races during the Constitution day celebrations, and such, were able to understand the goal of the challenge intuitively.

At this challenge's first iterations, a plate simply appeared, attached to the user's virtual controller, and a new ball was dropped from directly above the plate each time the ball was dropped onto the ground. For safety reasons, this happened quickly, before the user was able to reflexively crouch down to pick up the dropped ball. This sequence was also part of a tutorial at the start of any given path, but this tutorial was axed in VR Walk: Game in favor of giving verbal instructions to the users as too many were unable to take in the instructions given during the tutorial.

The next sizable improvement was to make the user pick up and hold the plate. This change was done in order to give the task a more "tactile feel" and in order to unify the input rules between each task, i.e holding the trigger-button to hold an object. This meant that the user was able to drop or throw the plate. To remedy this, the user can attract the plate back to its hand by holding the grip-trigger.

This additional step to the task sequence added a layer of complexity, where another step had to be understood in order to complete the challenge. The first time this was tested in an expo setting, it sometimes took almost the entire length of the path before the user was able to figure out how to handle the plate. To remedy this, a yellow, ghostly hand that mimed picking up the plate (Figure 3.23) was shown to the user until he or she successfully picked up the plate, while simultaneously lighting up and flashing the trigger-button on the virtual controller.

Likewise, for retrieving the ball, a transparent version of it is visible where it will drop from until the plate is held underneath it, as shown in Figure 3.23. A stylized animated character was placed on top of the starting-line, demonstrating either how to hold the plate relative to the ball or how to throw and catch the ball depending on the chosen difficulty. The lessons learned from the feedback and observations from this task was transferred to the other tasks where difficulties understanding (perceived ease-of-use) a specific task was observed.



Figure 3.23: *The left image shows the outline of a hand, instructing the player to grab the plate. The right image shows the player where the plate must be held to receive a balancing ball. When the plate is held in the correct position, the ball becomes solid and the physics engine takes control of it.*

To be able to adjust the challenge-level of the task, the ball can be swapped for a fireball, as shown in Figure 3.24. If left on the plate for too long, the plate will smolder away, dropping the ball to the ground. To successfully keep this from happening, the user must throw the ball in the air and catch it again on the plate to allow the plate to "cool off". For adjustable challenge-level, the time the ball can rest on the plate before shattering it, can be adjusted.



Figure 3.24: *The player tries to follow the path, while balancing a burning ball on a plate, without the plate getting too hot.*

The final score for this task is based on the longest continuous distance the user is able to walk while successfully balancing the ball on the plate.

3.8.4 Mechanical constraints

Mechanical constraints are all activities that directly introduce changes to the walking activity, either by involving new elements, or by altering the environment (Figure 3.25).

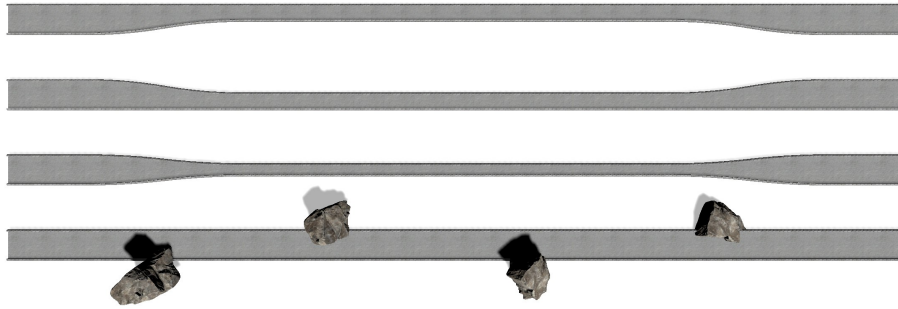


Figure 3.25: *The various path-segments that contain alterations to the path itself, as seen from above. These include path-segments that are partially narrowed or littered with obstacles. From top to bottom: Path narrowing in on the right hand side, on the left hand side, towards the middle of the path and with large rocks on top of the path.*

Narrow paths

One of the constraints developed during VR Walk BSc was the ability to narrow the virtual pathway. For this we created different path-sections where the pathway was narrowed in a way where the user would either have to walk on either the center, left, or right side of the treadmill. Because of the risk of falling when stepping outside the sides of the treadmill, this activity was heavily constrained by the available width of the treadmill used. As a result, the only option available for the narrowest treadmills is the one where the user is constrained to the center of the treadmill.

As feedback to the user that they have stepped outside of the path, the footprints made outside of the paths are replaced by red crosses. For safety-reasons, the changes to the pathway for these sections are gentle, and thus these are considered very easy for the majority of the users. Scoring for this activity is simply a comparison between the number of steps taken on the allowed path and the total number of steps taken during the narrowed segment.

Obstacles on the path

Another constraint in the mechanical category is path-sections with obstacles on the ground. In this case, parts of the pathway has large rocks approximately half a meter in diameter distributed along its length. The feedback the user is given if they step inside one of the rocks is the same red indicator as when stepping outside of the pathway. This is a slightly more challenging option compared to the variable path-width, but still considered easy for the majority of the users due to the relatively low frequency of these rocks. Scoring operates similarly to the narrow paths. Steps taken that do not hit an obstacle are compared to the total number of steps taken during the activity.

Pre-distributed footprints

The purpose of this activity is to break up the participant's gait pattern by generating a sequence of semi-randomly placed footprints that they are encouraged to try to mimic (Figure 3.28a). By using Vive trackers attached to the participant's feet, we can determine if the footprints have been hit successfully. In its final iteration the footprint generator can create very challenging patterns. Due to the risk of losing balance, a safety harness or an alert simulator operator is considered necessary.

The idea for this activity originated in how dancing schools sometimes attach footprint-stickers, such as those shown in Figure 3.26, on the floor for assistance when learning dancing steps. This idea was brought up as early as the idea workshop, detailed in the "Ideas" section, and was considered one of the more promising ones.

The algorithm for generating the sequence of pre-placed footprints underwent two major iterations. The first of these was based on offsets determined by setting an initial position for the left and right "footprint track" and traversing a procedurally generated Perlin noise-map in forward direction. This cloud-like map consists of values between 0 and 1, where a value of 0.5 was treated as no offset towards any side, values below as offsets towards the left and above to the right. The level of challenge was determined by the frequency of this map. As shown in Figure 3.18 in the "Coin collecting" section, by "zooming" in on a feature, we would follow a gentle swaying pattern and by "zooming" out, the results would be more erratic. By combining this with offsets from a random number generator we could achieve a scaling difficulty level. The downside of this approach was how unpredictable the challenge-level was on the higher difficulty settings. Variables that we predicted would generate exciting patterns would sometimes create a more straight-forward path.

For the second iteration, the lab's whiteboard was used, in order to encourage feedback and suggestions from fellow students. The goal was to come up with an idea for an algorithm that satisfied the following criteria: From easy to hard settings, the walking pattern had to start from being straight and regular and to gradually increase in challenge, while also following the speed of the treadmill and not exceeding the safe boundaries of the treadmill. In addition, we wanted to allow for the legs to cross from step to step at the harder difficulties. While sketching on the whiteboard, we were successful in receiving input and ideas from the students we shared the lab with. These were considered, but ultimately we settled on implementing one of our own ideas. Here, each step was only influenced by the step before it.

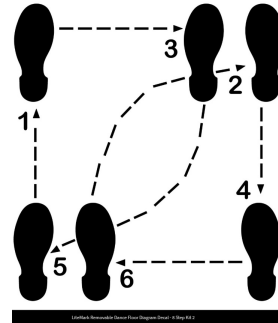


Figure 3.26: *Dance floor decals for learning dance steps. Image: ifloortape.com*

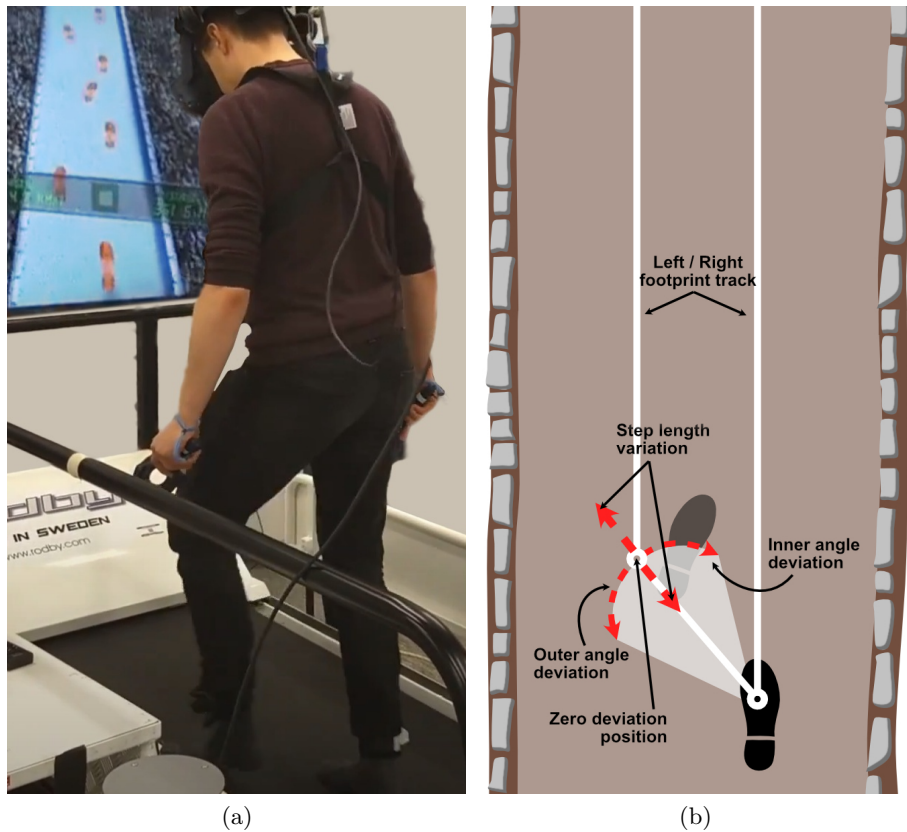


Figure 3.27: (a) shows the player following a sequence of pre-generated footprints with his feet. (b) illustrates the second iteration of the footprint placement algorithm.

The algorithm, shown in Code Example 3.3 and illustrated in Figure 3.28b, starts with a base-line step length with a randomized offset, labeled step length variation. A zero deviation angle for the step is then set as the angle where the step length, as the hypotenuse, intersects with the opposite footprint track. From there, we allowed for an inner and an outer deviation angle, ranging from crossed to wide steps. A limit towards the edges of the treadmill width was also included as part of the formula. The magnitude of all required parameters can be set through sliders in the path builder. From this new algorithm, we were able to satisfy all our self-imposed criteria.

In addition to the footprint pattern properties, the operator can choose between different approaches for estimating an average step -or stride length. The choices are between a set step length, calculating step length from a set step frequency and velocity or from a set step frequency and automatically read velocity from the treadmill speed while in-game. All of these options boil down to the formula: $\text{Step length (meters)} = \text{Velocity (meters per second)} \div \text{step frequency (steps per second)}$. An exaggerated step frequency and a wider angle -and step-length variation will contribute to increasing the challenge level for the activity.

The player's score is calculated to a percentage value by comparing the number of generated footprints correctly stepped on and the number of total generated footprints for the activity instance.

```

1  /// <summary> sets lane positions (footprint tracks) and path edges based upon
   ↳ path properties </summary>
2  internal void InitializeLanes()
3  {
4      // Sets up the left and right lane positions with the lesser of either a
       ↳ third of the path width or 40cm in-between.
5
6      // Sets up the left and right limits to 5cm inside the edge of the treadmill
7
8      // Sets lastPos left and right on top of the start of the path before
       ↳ spawning in the steps.
9  }
10
11 /// <summary> Generates the randomized pre-placed footprints </summary>
12 /// <param name="dist"> The length of the path where generation shall be
   ↳ performed </param>
13 /// <param name="averageStepLength"> The average player step length either set
   ↳ or calculated </param>
14 /// <param name="stepLengthDeviation"> How much the step length can randomly
   ↳ deviate </param>
15 /// <param name="innerAngle"> The maximum inwards random angle deviation from
   ↳ the zero-angle </param>
16 /// <param name="outerAngle"> The maximum outwards random angle deviation from
   ↳ the zero-angle </param>
17 public void SpawnSteps(float dist, float averageStepLength, float
   stepLengthDeviation, float innerAngle, float outerAngle)
18 {
19     var rightFoot = true; // start with the right foot
20     var footprintCount = 0; // number of footprints for registering max-score
       ↳ possible
21     var halfStepLengthDeviation = stepLengthDeviation * .5f;
22     var tempTransform = new GameObject().transform; // can't create a transform
       ↳ unless on a gameobject
23
24     while (Mathf.Max(_lastPosRight.z, _lastPosLeft.z) + averageStepLength <= dist +
       pathStart.transform.position.z) // while there is still room for more steps
       ↳ along the path distance
25     {
26         var offsetStepLength = Random.Range(averageStepLength -
           halfStepLengthDeviation, averageStepLength + halfStepLengthDeviation);
27
28         tempTransform.position = tempTransform.eulerAngles = Vector3.zero;
29         GameObject newFootprint;
30
31         // angle to non-rotation-offset next footprint position. Clamped in case of
           ↳ too short step length
32         var zeroAngle = Mathf.Asin(Mathf.Clamp(pathWidth / offsetStepLength, -1f,
           1f));
33
34         if (rightFoot)
35         {
36             var lastPosRightTrack = new Vector3(_rightLanePos, _lastPosLeft.y,
               ↳ _lastPosLeft.z);
37             var angle = Random.Range(-innerAngle, outerAngle) + zeroAngle;
38             tempTransform.position = lastPosRightTrack + Vector3.forward *
               ↳ offsetStepLength;
39             tempTransform.RotateAround(lastPosRightTrack, Vector3.up, angle);

```



```

40     var position = tempTransform.position;
41     position = new Vector3(Mathf.Clamp(position.x, _leftLimit,
42         ↪ _rightLimit),
43         position.y, position.z );
44
45     // angle of footprint is determined by the position of previous same
46     ↪ side footprint
47     var lastFootPos = footPrintCount < 2 ? _lastPosRight + Vector3.forward
48     ↪ * -3 : _lastPosRight; // to make the first footsteps point forward
49     newFootprint = Instantiate(rightPrint, position,
50     ↪ Quaternion.LookRotation(position - lastFootPos), pathStart);
51     _lastPosRight = newFootprint.transform.position;
52 }
53 else // leftfoot
54 {
55     // the behaviour for the right foot mirrored
56 }
57 _spawnedSteps.Add(newFootprint);
58 footPrintCount++;
59 rightFoot = !rightFoot; // swap active foot
60 }
61 // ... cleanup and register maximum possible score
62 }

```

Code Example 3.3: *The code for generating the randomized footprints*

3.8.5 Final score

In gamification, having one’s performance assessed through a score value serves as an important motivational factor[29]. This is especially important for players who want to increase their performance from play-through to play-through, or for those who want to compete against their friends.

When the end of the path sequence has been reached, the player is prompted to grab onto the railing and prepare for the treadmill to stop. The treadmill and forward motion in the virtual environment will then gradually slow down until coming to a full stop. The player is then met with floating board containing a final assessment of his or her performance, both per activity and a total percentage of best possible combined score (Figure 3.28).

Behind the scenes, the scores are fetched from a scriptable GameObject that all corresponding activity controllers writes to during game-play.



Figure 3.28: *The score panel shown to the player at the end of the path.*

3.9 Operator choices and the main menu

When launching VR Walk: Game, the user and operator are met with the main menu. In these menus, they can determine all the necessary variables for setting up the game how they want it to play.

3.9.1 The path builder menu

During the development of VR Walk BSc, a basic tool for stringing together an operator-chosen sequence of specific path-blocks, shown in Figure 3.29, was created. This tool was integrated into the main menu to be able to freely select a sequence of activities for the simulation, unless the operator had chosen to run a randomized path through another sub-menu.

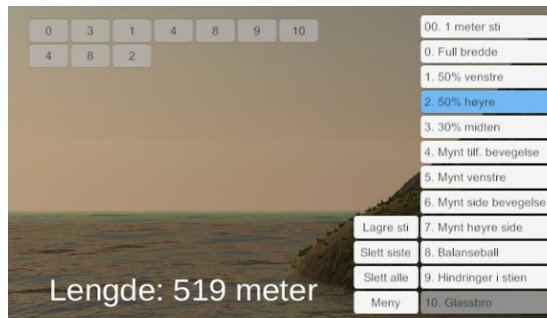


Figure 3.29: *The path builder as it appeared in VR Walk BSc.*

Because of the input precision needed to navigate the path builder tool, it was limited to the desktop and mouse interface (as opposed to the VR interface).

For VR Walk: Game, the path builder underwent two major revisions as a result of feedback from the physiotherapy supervisors. For the second iteration (Figure 3.30), we allowed the operator to set detailed instructions for the procedural generation of the path, i.e for having faster coin movement, creating the throwing targets further away or to set a more challenging footprint placement. Behind the scenes, we transitioned from using hard-coded buttons and path lists, to using a common list of settings that both the path generation and the button generation utilized. This greatly improved the ease of expanding or altering the available path activities.

While the second iteration made the path builder a much more powerful tool, the complexity increased as well. Feedback from the physiotherapy supervisors indicated that perceived usefulness was well covered. However, in terms of perceived ease-of-use, building a sequence now was a much more daunting task. The goal of the third iteration (Figure 3.31) was therefore to present a more end-user friendly experience while building a path. To achieve this, a set of preset difficulty values was created, ranging from easy to medium and difficult. If the operator wants more granular control of the settings, a panel with options, similar to those from the second iteration, could be opened for each button. Inside this panel, the operator can save these settings to disk for later retrieval through a drop-down list.

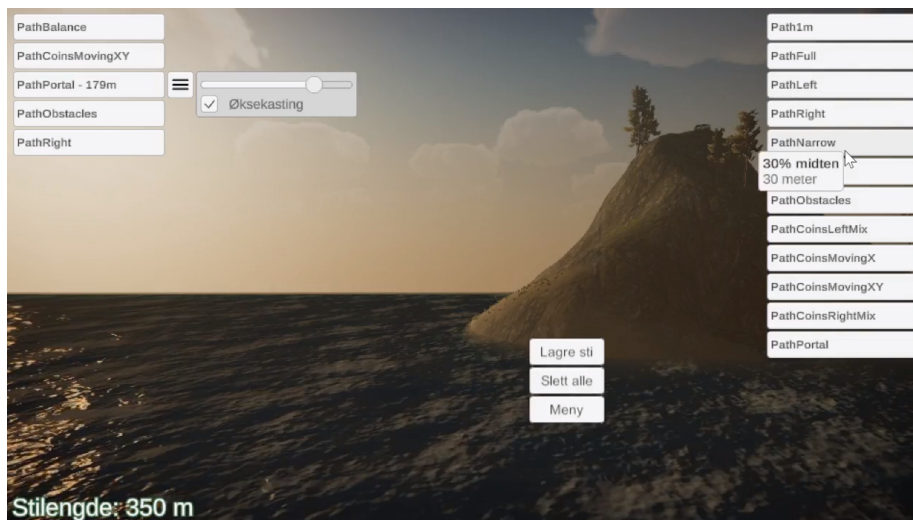


Figure 3.30: The second iteration of the path builder introduced more granular control of the path settings and descriptive mouse-over tool-tips.

To further increase usability while creating a path, we wrote a slot-based drag-and-drop system in order to freely change the order of the selected path pieces, and to supply an alternative means to distribute more path-pieces. This meant that the operator could either drag and drop from the pool of available path sections anywhere in the selected sequence, or simply click one in the pool of available path blocks (Seen at the right hand side of Figures 3.29, 3.30 and 3.31) to place at the last spot in the sequence.

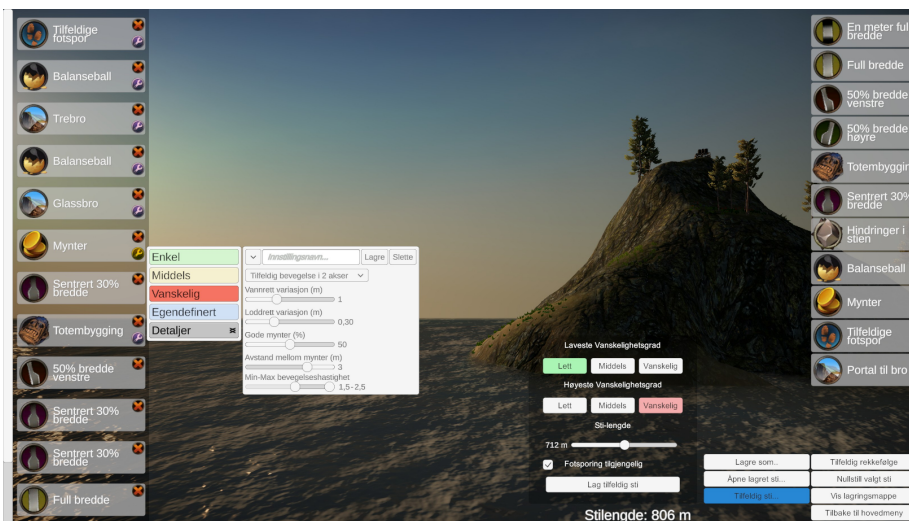


Figure 3.31: The third iteration of the path builder focused on flexibility, ease-of-use, random generation and storing and loading both settings and paths.

During this iteration, we discarded our previous system for generating a random path and integrated these settings into the path builder. As the path

builder became more rich in features, the system for generating random path on the fly in VR Walk BSc had become obsolete. This transition allowed the user to automatically generate a path based upon the desired path length, the desired difficulty level and availability of foot-tracking while also allowing the user to tweak the results to their liking using the path builder tools. The transition to this system meant that we were able to simplify the path generation in the main game scene, since both systems now utilized the same saved path system (detailed in the data persistence section below).

In addition to storing custom difficulty-settings, any number of completely configured paths can be named, saved and retrieved at will.

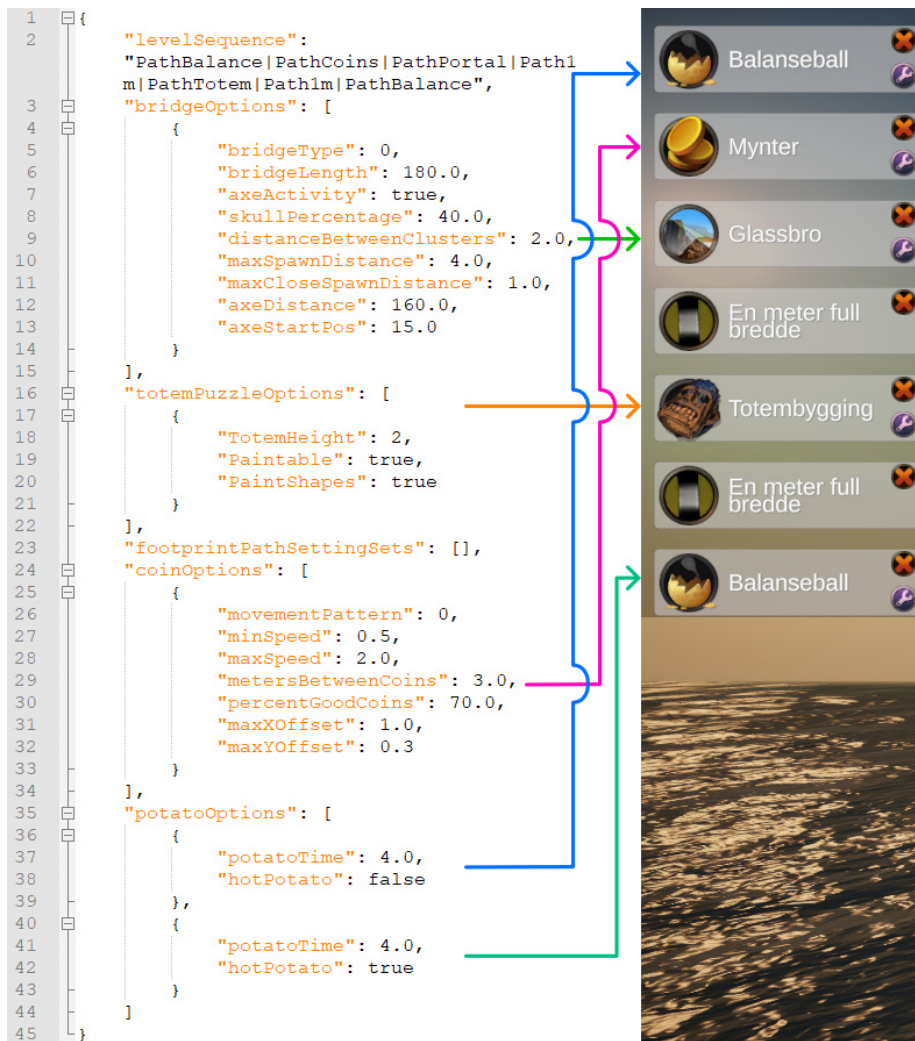
For determining a randomly selected path, each path-piece, recognized by it's activity, has an assigned starting weight. This weight is based on how interesting or unique the activity was in relation to the other available activities. This meant that similar paths, such as the three narrow variants and the obstacles path all received a lower starting weight. A random number between zero and the sum of all the weights is then rolled. The next piece to add to the random path is then chosen by looping through path-pieces and adding up their weights, one after another, until the rolled number has been surpassed and the loop aborts. The piece the loop stopped on is then chosen as the next piece, and it's weight is reduced to 25% of it's current weight to reduce the chance of repeats during the selection of the rest of the path. The difficulty of this piece is, if applicable, randomly set to a difficulty level between the chosen minimum and maximum. A new total weight sum is calculated and the process is repeated until the desired approximate path length has been met.

3.9.2 Data persistence

For persisting data, we opted for a combination of storing settings in the Windows registry through Unity's Player preferences system, and storing data in JSON files.

Fully configured path sequences can be saved in one such JSON file, as shown in Code Example 3.4. They can then be retrieved in the path builder or when launching the main game-play scene in order to generate the selected sequence in addition to other settings, such as the parameters for generating each path segment.

The first time VR Walk: Game launches, the activity difficulty presets are written into JSON files. At subsequent launches, these files will be used for loading difficulty level presets for each activity. As these are human readable, an added side-benefit to this is being able to alter these presets by editing or replacing these files. This approach is not necessarily recommended, as it is possible to ignore the constraints of each setting, but might be desirable if the operator has a preference for a particular custom activity setting. The difficulty presets for certain activities are also stored in a human-readable format in JSON files, which allows an operator to overwrite these with their own or to supplement the settings list with more options.



Code Example 3.4: The JSON representation of a fully configured path. When the main game-play scene is loaded, a path will be generated from these settings. If loaded into the path builder, these settings will also be loaded.

Less frequently altered settings, such as calibration data and treadmill settings are stored in the Windows registry.

3.9.3 Activities during main menu navigation

The user wearing the Head mounted display (HMD) only has a limited selection of settings available in the main menu. Due to the amount of settings available, precise inputs required, and in order to save on development time, we chose to limit the path builder and seldom changed settings to the desktop interface. In an operator and user/patient setting, this meant that the user wearing the HMD was free to explore the environment in the main menu scene while the operator is setting up a suitable path. This serves several purposes; It allows the user to get accustomed to viewing the virtual environment, since the treadmill is not running at this point. It also allows the user to make necessary adjustments to the HMD-straps in a low pressure setting.

Additionally, the exploratory setting allowed us to introduce ways for the user to get accustomed with some of the interactive objects of the path activities. The user would be able to try out coin collecting and axe throwing with continuously reappearing and repositioning targets, and they would get the chance to learn which targets to aim for and avoid during these. A third activity available in this environment is a totem set, where the user can learn how to stack and paint totem pieces before encountering the activity during the main game.

3.10 Designing for modularity

Due to the large scope of VR Walk: Game in relation to available development time, the only way to have the desired variation and flexibility was to generalize as much as possible into modules. The path builder, with its numerous configurations, the ability to interact with different treadmill interfaces and the ability to freely combine any sequence of path segments and have them generate their activities based upon the correct parameters are all examples of this.

To ensure that *any* stretch of the walking path with *any* activity could be followed by *any* other activity at *any* point in the terrain, there were several limitations to account for. The terrain itself needed to be static, i.e it could not be re-shaped to match the path layout. This made introducing slopes and treadmill elevation unfeasible as this would make the path float higher and higher above the terrain, as there are no practical means to achieve downward slopes on the treadmills on demand since only the front part can be raised. Some treadmills can reverse the running direction, making the rear end the raised one, but the required setup for performing this was not considered practical in our case. The angle limitations and speed with which the treadmills are raised are also quite varied between treadmill models.



Figure 3.32: The nested composition of a path builder button. *Green*: Click-and-drag slot area, *Yellow*: Main body of the button, *Blue*: The common elements of the configurable buttons, *Red*: The activity-specific settings.

All of these factors resulted in the decision to have all path-segments flat and only distributed along a straight, pre-flattened strip along the center of the terrain. As a positive side-effect, this meant we were able to focus our efforts in making visual interest features only where the player would be able to see from these areas. The transitions between sections of the paths could also be made visually seamless.

The straight, flat area available for activities presented a challenge in how to introduce the height exposure constraint. In this case the limitation directly informed the decision to use portals that transported the player to other areas that are similarly flat and straight, but in a different location, i.e at the mountain top before the glass bridge or on the beach before the wooden bridge. This decision was also beneficial in making sure these sections could be positioned at any point along the main path as they only required to have their portal entrances (and exits) there, as shown in Figure 3.33.



Figure 3.33: A portal entrance used for transporting players to and from other areas of the environment.

Each path section is responsible for containing the logic required to correctly generate its own activity props and to report its statistics for the purpose of the final score. Likewise, each path builder block is responsible for writing and retrieving its settings in the format shown in Code Example 3.4.

On the topic of the path builder buttons, these also consist of multiple nested modules (Figure 3.32). This was achieved by using Unity's newly released nested prefabs feature, where each layer of these prefabs does not lose its respective state when nested inside other prefabs. As each activity has a varied set of restrictions and parameters, class inheritance was heavily utilized.

3.11 Optimization

Throughout development, we attempted to maintain a balance between visual fidelity and performance. In order to assess the simulator's performance, we used Unity's Profiler (Figure 3.34), an analysis tool which captures the time used by individual processes as each frame of the software is rendered.

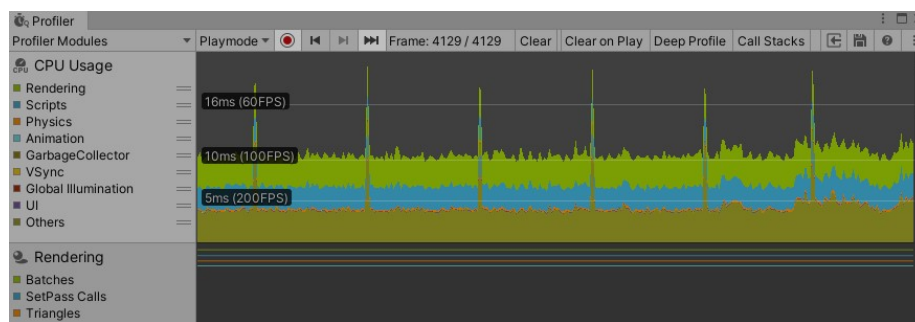


Figure 3.34: *Unity's Profiler tool, displaying time use of application processes on a per-frame basis.*

Some seemingly anomalous frame timings discovered while experimenting with this tool lead us to SteamVR's Motion Smoothing feature. The purpose of the feature was to synthesize transitional frames in order to preserve the smoothness of a user's head movement and reducing the chance of simulation sickness, should the frame rate fail to match the current HMD's refresh rate.

At the times when the frame rate of the simulator dropped significantly below 90 rendered frames per second (FPS), corresponding to the HTC Vive display's refresh rate of 90 Hz, the Motion Smoothing feature would cause information on the semi-transparent treadmill panel to appear warped, and any text on the panel became illegible. Disabling Motion Smoothing, however, made for a significantly less comfortable experience, with the user's view appearing to jump from position to position at times.

This allowed us to establish 90 FPS as an ideal performance baseline for our given hardware.

Several aspects of the simulator were optimized in an effort to reach this baseline. The implementation of the vegetation biomes centered around adding as much visual foliage as possible while minimizing the performance impact. Similarly, post-processing effects were disabled or tuned to the minimal point where they still had a desirable visual effect without compromising performance.

3.12 Continuous evaluation as part of the development process

Each newly developed activity or feature was tested constantly by ourselves and improved upon until the activity was found to be interesting enough and behaving as expected.

VR Walk: Game was exhibited at multiple events throughout the course of development. These were opportunities to receive feedback from the general public of a great range of ages, and to make adjustments and feature changes. This served especially as an effective indication of perceived ease-of-use and perceived enjoyment.

The details around each specific event, and the changes that were direct responses to feedback and observations from these, are elaborated on further in the "Feedback and evaluation through informal testing" section of the Evaluation chapter below.

Chapter 4

Evaluation

The evaluation was intended to be a combination of the following:

- Semi-structured interview with physiotherapists and researchers.
- The results of the System Usability Scale (SUS) form from the same interview session, detailed in the "Evaluating perceived ease-of-use through the System Usability Scale" section.
- Observations from informal testing sessions.

Due to the outbreak of COVID-19, the travel arrangements for the interview were cancelled. A remotely guided demonstration was considered as an alternative evaluation, but the preventative measures taken against the outbreak restricted access to the facilities where the required equipment was located. This applied to both the intended interview subjects and ourselves, and as a result, no equivalent evaluative testing could responsibly be carried out.

The following sections contain a detailed account of how the demonstration and semi-structured interview were to be conducted, and how the results were to be processed. This is followed by notes on the feedback received from our physiotherapy supervisors and informal test sessions.

4.1 Semi-structured interview with physiotherapists and researchers

As part of the evaluation of the application, our physiotherapy supervisors had made arrangements for us to conduct a software demonstration and semi-structured group interview of 3 experts with a background within physiotherapy and VR research. The interviewees were involved in the development of a separate VR treadmill rehabilitation application at Norwegian University of Science and Technology (NTNU). The interview was scheduled for the 30th of March, 2020 and travel arrangements had been made. Due to the COVID-19 outbreak, the demonstration and interview were unfortunately cancelled, but not before a number of preparations had been made.

At the time of cancellation, we had been able to construct an interview protocol, apply for and receive the required permissions from the Regional Committees for Medical and Health Research Ethics (REK) and perform a recorded trial interview with our physiotherapy supervisors. The interview protocol included notes on how the demonstration would be organized, a set of preliminary questions intended to establishing the background and related experience of each interviewee, and a comprehensive checklist of topics related to the application. The protocol can be found in the appendix (in Norwegian).

Semi-structured interviews, as detailed in Hansen's book *Successful Qualitative Health Research*[45], follow a natural conversation style with broad non-leading questions, as opposed to structured interviews that adhere to a more strict question-answer flow. During a semi-structured interview, the conversation might tend to move freely between topics. This allows for the possibility of the interviewee following trains of thought to topics unanticipated by the interviewer. The interviewer might then choose whether to pursue the topic if it seems relevant, or steer the conversation towards subjects that should rather be addressed.

We had planned to maintain an overview of the topics that had been covered by following the checklist from the interview protocol. This would have allowed us to ensure that all notable items were discussed, and we would have been able to lead the conversation back to a previous topic if it had not been sufficiently explored.

As per the protocol, the subjects were to try VR Walk: Game, including the path builder, and make observations amongst themselves before the interview. The interview was to be audio-recorded. During the opening questions we would map out their relevant credentials including the subjects' experiences with fully immersive VR, physiotherapy, physical rehabilitation and game-based rehabilitation. The topics of the interview were chosen partly based on the primary factors of the Extended Technology Acceptance Model[22]. As such, emphasis was put on collecting feedback on the perceived usefulness (PU), perceived enjoyment (PE) and perceived ease-of-use (PEOU) of the software.

For the main part of the interview, we wanted to touch on the following topics:

- Impressions of VR Walk: Game as a whole, i.e ease of use (PEOU), fun-value (PE), software performance (PE), visual quality (PE) and intuitiveness (PEOU).
- Opinions regarding the different path-sections and activities (all primary factors).
- Opinions regarding the difficulty-levels, i.e in regards to the activities' difficulty scaling or if something else was too easy or difficult (PE and PU).

- If they felt they understood the instructions and other feedback given during the run (PEOU and PE).
- Opinions about the Graphical user interface (GUI), menus, path builder, treadmill controls or other controller interaction (all primary factors).
- And most importantly, VR Walk: Game’s potential usefulness (PU) both in it’s current form or if developed further whether within research, physical therapy or other fields of use.

The interview was to be transcribed (in Norwegian) and included in the appendix. The interview would then be summarized, and points made regarding the rehabilitative potential of the simulator would be subject to further discussion. Suggested improvements and observed shortcomings would be noted, discussed and condensed into a list of suggestions for further work with the simulator.

Additionally, the full transcript was to be examined through the lens of the Extended Technology Acceptance Model in order to evaluate to which degree the research goals were successfully met.

4.2 Evaluating perceived ease-of-use through the System Usability Scale

As a supplement to the interview, the interview subjects were to individually fill out the System Usability Scale (SUS) form.

The form contained a set of questions relating to each subject’s experience with using the software. Each question would be framed as a statement, and each statement would be rated according to how much a subject agreed with it. The statements alternate between having a positive attitude and having a negative one. The scores from the negative statements would then be inverted in order to organize all scores within a scale that would range from negative experience to positive experience.

The Norwegian translation of the SUS form by Dag Svanes (NTNU) is included in the appendix.

As described by Toftøy-Andersen and Wold[46], the form’s score is calculated by the following procedure:

- Each question is given a score ranging from 1 to 5 by the subject
- Each given score is converted to a negative-to-positive range going from 0 to 4:
 - Question 1, 3, 5, 7 and 9 (positive attitude) are given the initial score’s value minus 1.
 - Question 2, 4, 6, 8 and 10 (negative attitude) are given the value of 5 minus the initial score’s value.

- The scores from each question are summed up.
- This sum is then multiplied by 2.5 to get the final SUS score in the range of 0 to 100.

The usability can then be interpreted using the following estimates[47]:

SUS Score	Letter Grade	Adjective Rating
Above 80.3	A	Excellent
Between 68 and 80.3	B	Good
68	C	OK
Between 51 and 67	D	Poor
Below 51	F	Awful

Although a sample size of only three would not nearly be sufficient for a full assessment of whether VR Walk: Game was ready for use in rehabilitation, the responses would serve as a relevant addition to the assessment of the product's perceived ease-of-use.

4.3 Evaluation through testing and conversations with supervisors

At semi-regular intervals, our physiotherapy supervisors had the opportunity to have VR Walk: Game demonstrated to them. During these sessions, the state of the software was discussed, and suggestions for improvements and simplifications were presented. These conversations were also helpful in determining which dual-task activities to focus our efforts on.

The pathbuilder menu was often a focal point of the meetings. The feedback received made it clear that particular care should be taken in making the builder sufficiently user-friendly for operators with limited computer knowledge. In order to achieve both this and to improve the path builder in general, a number of usability improvements and features were discussed with the supervisors and subsequently implemented. This included drag and drop functionality, recognizable icons, pre-made difficulty values, a reduction of interface clutter and the ability to save and retrieve saved paths and activity settings.

The activities with cognitive aspects were also largely shaped during these meetings, as was the activity with the pre-distributed footprints. The option to display the totems of the totem building activity with more recognizable card suit iconography was added in response to feedback from the supervisors. The opinion was that the silhouettes of the totem pieces might be too complex to be immediately recognizable to users when they were first introduced to the activity.

A collaborative effort was made for the pre-distributed footprints in order to have the names and functions of the activity parameters conform with terminology used in gait rehabilitation.

During the pilot interview, it was suggested by the supervisors that the VR environment in the main menu might benefit from including an overview of correct and incorrect targets for the coin collecting and axe throwing activities. This led to the inclusion of interactive coins and axe targets in the main menu activity familiarization.

Our supervisors expressed that their overall impression with the simulator was that the activities sufficiently covered the constraint categories of constraint-based gait rehabilitation, and it seemed feasible to guide a completely inexperienced VR user through a play session using the activities and path segments available. The easy, medium and hard difficulty levels, however, were found to be too inconsistent between activities, and it was suggested that large scale user testing might be beneficial in evaluating and improving the consistency of the levels.

Since the semi-structured interview and System Usability Scale fell through, the conclusion and final assessment regarding its perceived usefulness within constraint-based gait rehabilitation are largely based upon the opinions and evaluation from these conversations.

4.4 Feedback and evaluation through informal testing

During the first months of development, numerous participants were able to try the latest versions of either VR Walk BSc or VR Walk: Game. These opportunities were especially helpful in gauging perceived enjoyment and perceived ease-of-use. throughout these, we observed where there were difficulties knowing what to do or where there were other issues. The feedback from these tests was taken into account and, in most cases, implemented into the final product.

2018

September 21 - 22.: Bergen research days 2018 took place between the bachelor and master project periods. This was the largest test in number of participants, where hundreds of children were able to try VR Walk BSc. Due to this amount, most were only able to experience and provide feedback on the environment and a portion of the glass bridge sequence while walking on the treadmill.

Many tensed up and found it scary or exciting as they crossed onto the bridge, but this usually wore off after some meters and they were able to enjoy the view. We used this feedback in determining that a latter part of the bridge sequence was suited for additional activities, which informed the decision to combine the bridges with the axe-throwing activities. The expo stand we were a part of had a common gait theme. The participants enjoyed themselves so much that the stand was awarded the price for best of show by the attendees[10].

October 26 - 27.: E-LAN was a video game event at Energisenteret for Barn og Unge[48] of Haukeland University Hospital (Figure 4.1). Youth and children, who were receiving treatment there, were able to bring their own computers and friends to play video games. We were invited to bring VR Walk BSc, including a Woodway treadmill[23] and a VR-set. These attendees were originally our main target group, as the project was originally prioritizing motivational exercise. Because of these priorities, the feedback was greatly valued. Some attendees commented on how they forgot how they technically were exercising on a treadmill.



Figure 4.1: *Picture from E-LAN 2018*

At this time, there was a non-interactive axe positioned within reach at the start of the balancing activity. These video-game savvy attendees echoed a feature request we had heard many times before, which was how much they would like to be able to bring that axe with them. As work on VR Walk: Game started, this was quickly remedied, and that was the start of the development of the axe throwing activity.

2019

September 11.: Forsknings Expo (Helse og idrett) HVL was the earliest opportunity to show VR Walk: Game after having started development.

At this point, the main improvements from VR Walk BSc were an updated input-system based upon SteamVR v2.x and having implemented start, stop and speed-controls for the Woodway treadmill. Some of the participants struggled to understand the current way the balance ball activity was performed, to such a degree that this activity was the first to be revised.

September 26.: Forskningsdagene Ung. This was an event, held at HVL as part of Bergen research days 2019, mainly for upper secondary school students to listen to lecturers and project supervisors HVL present interesting

projects within exact sciences. Bovim’s presentation included VR Walk: Game, with an invitation to meet us outside the classroom where we had the treadmill and VR-equipment set up (Figure 4.2).

At this point, an early version of the axe-throwing activity was a clear favourite, in terms of perceived enjoyment, amongst both those who were trying the game, and for the spectators who were taunting or cheering for their friends.

October 5.: Bergen Open Smash Series was a gaming event at AdO Arena[49] in Bergen. It was arranged by Bergen diving club, PlayWell[50] and GG Bergen[51] and was intended to draw a parallel between physical well-being and performance in video-game tournaments. We were recruited by Idrettscampus Bergen[52] to bring a complete setup for VR Walk: Game and let participants and visitors partake as part of their ”AktiveGamere” (Gaming and E-sports) project. Our efforts were picked up by the sports section of the TV channel TV2’s televised news and briefly shown during their sports broadcast.



Figure 4.2: *Exhibiting VR Walk: Game for upper secondary school students at Forskningsdagene Ung 2019. Photo: @hvlrehablab.*

As for feedback and observations, we were able to let the participants test the newly developed randomized placement of axe-throwing targets and the burning ball version of the balance ball activity. The participants were mostly younger players with prior gaming knowledge. The axe-throwing was well understood, but the balance ball activity still had ways to go in regards to the perceived ease-of-use factor concerning how to use the plate to interact with the balancing ball.

October 23.: Realfagsdagen (loosely translated, ”The day of exact sciences”) at HVL was our first opportunity to let a larger amount of expo visitors try the freshly developed cognitive dual-task, totem pole memorization and building. While most found it intuitive to stack the totem pieces, very few were able to remember a stack of three figures and their colors. Because of these observations, we introduced the colored card-suit shapes on the figures as the default difficulty setting, as previously seen in Figure 3.15. Again, the glass bridge with the axe throwing activity appeared to be both the participants’ and audience’s favourite in terms of perceived enjoyment.

Chapter 5

Conclusion

This thesis summarized the decisions made during the development of a fully immersive VR and treadmill solution intended for use within constraint-based gait rehabilitation. The product was also intended to satisfy within player safety, as well as the three primary factors of the Extended Technology Acceptance Model, namely, perceived enjoyment (PE), perceived ease-of-use (PEOU) and most importantly, perceived usefulness (PU).

Particular emphasis was placed on attempting to create activities functionally equivalent to constrained gait tasks found in non-immersive VR solutions currently used in gait rehabilitation. The activities were designed to adhere to the characteristics of the perceptual, cognitive, mechanical and motoric constraint categories (PU) of constraint-based gait rehabilitation. Additionally, the activities were purposely varied and featured scaling difficulty to aid motivation for further use (PE). A powerful path designing tool was developed to achieve variation and to enable therapists to easily make custom paths suited to the abilities of their respective patients (PEOU and PU).

Due to the unforeseen setback in regards to the lack of testing and evaluation during the outbreak of COVID-19, we were unable to firmly determine if VR Walk: Game met all the criteria in terms of potential use within constraint-based gait rehabilitation (PU). From conversations with physiotherapy supervisors and gait experts we can, however, surmise that VR Walk: Game offers the opportunity to let individuals train their gait under different constraints that may target their impairments specifically. Individuals with cognitive impairments may be challenged by the memorization task of matching the totem pole. Individuals with Parkinson's disease, who may be suffering from bradykinesia, can benefit from the rapid and irregular movements required to hit the virtual footsteps or the repeated motions[2] from reaching for flying coins. Individuals with vertigo or vestibular disorders may find the height exposure from the glass bridge challenging.

In terms of and general quality, we were able to observe, during external testing, multiple cases where participants were so immersed in the virtual environment that they entirely forgot that they were actually walking on a treadmill (PE and PEOU).

Chapter 6

Further Work

Further research

While VR Walk: Game was developed with the intention that it would be a ready to use product, it would need to be clinically assessed before being applied in a therapeutic setting. One potential strategy would be to conduct a preliminary evaluation, similar to the intended assessment for this thesis, and decide, based on the results of this evaluation, whether to proceed with a larger scale quantitative research project.

Although an initial qualitative evaluation was intended for this thesis, a quantitative user testing of a sufficient scale would have been outside the scope of this thesis, and perhaps better suited for a physiotherapy thesis. With this in mind, ease of use (PEOU) was one of the top priorities.

Known issues

There are still some known faulty elements with VR Walk: Game. These are some of them:

- The length of the path is currently not limited by neither length nor activity count, meaning it is possible to construct a main path that exceeds 1.35 kilometers and continues over the edge of the island.
- The interaction with the virtual front panel is, reportedly, not working "out of the box" when not using Vive controllers. The front panel interaction is using a separate SteamVR action set, called "menu". Only HTC Vive controllers have properly configured bindings for this set. When using other controllers, an action called "InteractUI" would need to be bound to the controllers' trigger press in the SteamVR bindings setup.
- The treadmill space to VR space calibration scene currently lacks an option to return to the main menu from the HMD interface unless a new calibration has been set.

- If the axe throwing activity occurs in the middle of the night, the darkness makes it challenging to see where the axe is being thrown. This would be remedied by either having "emissive" features that lights up parts of the thrown object or by having a less aggressive color grading during night time.

Further development

In terms of development, certain features and changes were not completed for the product in time. Should the simulator be selected for further research or practical application, the items on this list should be reviewed and considered for implementation:

- Options for multiple languages. VR Walk: Game is currently only available in Norwegian.
- A path-set where the width of the path would be controlled by sliders in the options in order to support a wider range of treadmills. The treadmill security-grid would also need to be adjusted accordingly.
- To change the throwable axe into another item, where more users would be able to intuitively understand by association that the object can be used for both striking and throwing and that it can be returned to the user. The hammer, Mjölmir, from Norse mythology was considered.
- Adding a rhythmic portion to the pre-distributed footprints activity, possibly combined with the coin collecting activity, in order to make the player perform various poses, while following a beat.
- An option for treadmills that could not be controlled directly from software. The player would be assisted in syncing up the simulator and the treadmill speed manually. This together with the option to dynamically adjust the path width would enable most commercially available treadmills with the option to run at a set speed to be used.
- Additional environments. An urban environment and a construction site were among the considered options. The reasoning being that for a select groups of patients, distractions from a busy environment factors into their gait and balance.
- Settings for adjusting foliage, sky and rendering quality.
- The ability to record statistics and path setups per patient or player through an account and database system.
- Unity's solution for OpenVR[40] is scheduled to be deprecated in the next major release, resulting in the current SteamVR plugin (v2.2+) being incompatible. Valve is currently updating their SteamVR plugin[53] to support the new Unity XR Plugin Framework[54]. Valve recommend against releasing an application with the current beta

release (v2.6.0.b1). It is unknown (by us) which parts of the VR interaction would have to be re-implemented in VR Walk: Game once this is ready.

- A monitor-based mode to play through activities without wearing an HMD, but while still using the controllers to interact with the environment. This might serve as suitable common starting grounds for research directly comparing the effects of VR-based and monitor-based gait training tools.
- A thorough fine-tuning of the various difficulty settings of the activities, in order to make each "easy", "medium" setting etc. be more in line with each other.

Bibliography

- [1] “Unity asset store - the best assets for game making,” 2020 (Retrieved 2020-05-14). <https://assetstore.unity.com>.
- [2] D. K. Wells and S. Falck, “Bradykinesia: Definition, symptoms, and treatment,” 5 2017 (Retrieved 2020-05-08). <https://www.healthline.com/health/parkinsons/bradykinesia>.
- [3] P. James, “Road to VR: 10 things you didn’t know about Steam VR’s Lighthouse tracking system,” 2015 (Retrieved 2020-03-24). <https://www.roadtovr.com/10-things-you-didnt-know-about-steam-vrs-lighthouse-tracking-system>.
- [4] “Perlin noise,” (Retrieved 2020-05-14). https://en.wikipedia.org/wiki/Perlin_noise.
- [5] “Unity real-time development platform — 3d, 2d vr & ar visualizations,” 2019 (Retrieved 2019-09-18). <https://unity.com>.
- [6] B. Bleikli, M. Tellevik, and L. Valved, “VR Walk,” 2018. <https://hdl.handle.net/11250/2656066>.
- [7] L. P. Bovim, B. E. Gjesdal, S. Mæland, M. K. Aaslund, and B. Bogen, “The impact of motor task and environmental constraints on gait patterns during treadmill walking in a fully immersive virtual environment,” *Gait & Posture*, vol. 77, pp. 243 – 249, 2020.
- [8] “Worley noise,” (Retrieved 2020-05-14). https://en.wikipedia.org/wiki/Worley_noise.
- [9] “Voronoi diagram,” (Retrieved 2020-05-14). https://en.wikipedia.org/wiki/Voronoi_diagram.
- [10] “Institutt for helse og funksjon fra høgskulen på vestlandet vant prisen til årets beste stasjon på forskningstorget,” 2018 (Retrieved 2020-03-03). <https://www.forskningsdagenebergen.com/hogskulen-pa-vestlandet-vant-prisen-for-arets-beste-stasjon-pa-forskningstorget-2018/>.
- [11] A. Mirelman, L. Rochester, I. Maidan, S. Del Din, L. Alcock, F. Nieuwhof, M. O. Rikkert, B. R. Bloem, E. Pelosin, L. Avanzino, G. Abbruzzese, K. Dockx, E. Bekkers, N. Giladi, A. Nieuwboer, and J. M. Hausdorff, “Addition of a non-immersive virtual reality component to treadmill training to reduce fall risk in older adults (v-time): a randomised controlled trial,” *The Lancet*, vol. 388, pp. 1170–1182, Sep 2016.

- [12] M. Krzywinski, “Data visualization, design, science and art - martin krzywinski/ genome sciences center.” <http://mkweb.bcgsc.ca/colorblind/>, (Retrieved 2020-05-04).
- [13] D. Flück, *Color Blind Essentials*, p. 10. Colblindor, 2012.
- [14] “History of coin collecting,” 2020 (Retrieved 2020-05-22). <http://www.coinspot.org/history.html>.
- [15] G. Yogev-Seligmann, J. M. Hausdorff, and N. Giladi, “The role of executive function and attention in gait,” *Movement Disorders*, vol. 23, no. 3, pp. 329–342, 2008.
- [16] A. Bowen, R. Wenman, J. Mickelborough, J. Foster, E. Hill, and R. Tallis, “Dual-task effects of talking while walking on velocity and balance following a stroke,” *Age and Ageing*, vol. 30, pp. 319–323, 07 2001.
- [17] S. O’Shea, M. E. Morris, and R. Ianseck, “Dual Task Interference During Gait in People With Parkinson Disease: Effects of Motor Versus Cognitive Secondary Tasks,” *Physical Therapy*, vol. 82, pp. 888–897, 09 2002.
- [18] Y.-C. Liu, Y.-R. Yang, Y.-A. Tsai, and R.-Y. Wang, “Cognitive and motor dual task gait training improve dual task gait performance after stroke - A randomized controlled pilot trial,” *Scientific Reports*, vol. 7, 2017.
- [19] A. Geurts, T. Mulder, R. Rijken, and B. Nienhuis, “From the analysis of movements to the analysis of skills. bridging the gap between laboratory and clinic,” *Journal of Rehabilitation Sciences*, vol. 4, pp. 9–12, 01 1991.
- [20] I. J. de Rooij, I. G. van de Port, and J.-W. G. Meijer, “Effect of Virtual Reality Training on Balance and Gait Ability in Patients With Stroke: Systematic Review and Meta-Analysis,” *Physical Therapy*, vol. 96, pp. 1905–1918, 12 2016 (Retrieved 2019-11-11).
- [21] N. Fritz, F. Cheek, and D. Nichols-Larsen, “Motor-cognitive dual-task training in persons with neurologic disorders: A systematic review,” *Journal of neurologic physical therapy : JNPT*, vol. 39, pp. 142–153, 07 2015.
- [22] X. Wang and D. H.-L. Goh, “Video game acceptance: A meta-analysis of the extended technology acceptance model,” *Cyberpsychology, Behavior, and Social Networking*, vol. 20, no. 11, pp. 662–671, 2017. PMID: 29125789.
- [23] WOODWAY GmbH, *WOODWAY Laufbänder für die Medizin*, 2020 (Retrieved 2020-03-26). <http://www.woodway.de/medicaltreadmills/ppsm.html>.
- [24] G. Alce, A. Hansson, and K. Mårtensson, “Using vr for fitness training – pilot study,” in *Virtual Reality and Augmented Reality* (P. Bourdot, V. Interrante, L. Nedel, N. Magnenat-Thalmann, and G. Zachmann, eds.), (Cham), pp. 97–115, Springer International Publishing, 2019.

- [25] M. Endresen, “How do persons with stroke and therapists experience the try out of a vr-based rehabilitation game performed on a treadmill?,” 6 2019. <http://hdl.handle.net/11250/2610836>.
- [26] G. Alankus, A. Lazar, M. May, and C. Kelleher, “Towards customizable games for stroke rehabilitation,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2113–2122, ACM, 2010.
- [27] F. Kern, C. Winter, D. Gall, P. Pauli, M. Latoschik, and I. Käthner, “Immersive virtual reality and gamification within procedurally generated environments to increase motivation during gait rehabilitation,” 04 2019.
- [28] T. F. Larsen, “Virtual reality games and gamified exercises in physiotherapeutic treatment of non-specific low back pain patients with kinesiophobia,” 10 2018. <http://hdl.handle.net/1956/19626>.
- [29] T. Susi, M. Johannesson, and P. Backlund, “Serious games : An overview,” Tech. Rep. HS-IKI-TR-07-001, University of Skövde, School of Humanities and Informatics, 2007. <http://www.diva-portal.org/smash/record.jsf?pid=diva2%3A2416&dswid=35>.
- [30] S. Hayden, ““beat saber” sells over one million copies.,” 2019 (Retrieved 2020-03-30). <https://www.roadtovr.com/beat-saber-sells-one-million-copies>.
- [31] ““best of 2017 virtual reality.,” 2018 (Retrieved 2020-03-30). https://store.steampowered.com/sale/2017_top_vr_titles.
- [32] “Boxvr - fitxr,” 2017 (Retrieved 2020-03-27). <https://fitxr.com/boxvr>.
- [33] P. Lewis, “Boxvr game review – duck, dodge, punch and jump!,” 10 2017 (Retrieved 2020-03-30). <https://www.vrfitnessinsider.com/review/box-vr-game-review>.
- [34] G. Moss, “How does playing vr make you fit?,” 3 2020 (Retrieved 2020-03-27). <https://www.vrfitnessinsider.com/how-does-playing-vr-make-you-fit>.
- [35] Rodby Innovation AB, *Treadmill RL2500E x 700 — RODBY INNOVATION*, 2019 (Retrieved 2019-09-18). <https://www.rodby.com/en/products/treadmill-rl2500e>.
- [36] Ø. Vik and S. Muhaisen, “Nå kan de minste gå på tur i egen stue,” 2019 (Retrieved 2020-03-26). <https://www.bt.no/sport/i/d0PvEA/se-video-naa-kan-de-minste-gaa-paa-tur-i-egen-stue-ville-ikke-anbefal>, <https://www.aftenposten.no/sport/i/0n0982/se-video-naa-kan-de-minste-gaa-paa-tur-i-egen-stue-ville-ikke-anbefalt-det-til-mine-barn>, <https://www.adressa.no/100Sport/andredretter/Se-video-Na-kan-de-minste-ga-pa-tur-i-egen-stue--Ville-ikke-anbefalt-det-til-mine-barn-279589b.html>.

- [37] M. Sigerseth, L. P. V. Bovim, B. Bogen, T. F. Larsen, H. Soleim, A. Geitung, R. A. Monsen, T. Stanton, J. S. Skouen, G. Indredavik, K. Wilhelmsen, L. H. Magnussen, M. Kristin, M. Flesland, H. Andersen, and K. Fersum, “Bruk av virtuelle miljø i fysioterapi praksis?,” *Fysioterapeuten*, pp. 22–26, 09 2018 (Retrieved 2020-04-22). <https://fysioterapeuten-eblad.no/dm/fysioterapeuten-9-18/22>.
- [38] “Oculus quest: All-in-one vr headset — oculus,” 2020 (Retrieved 2020-05-27). <https://www.oculus.com/quest>.
- [39] “Gamedev market — game assets for indie developers,” 2020 (Retrieved 2020-05-14). <https://www.gamedevmarket.net>.
- [40] Github, *API Documentation · ValveSoftware/openvr Wiki*, 2020 (Retrieved 2020-05-19). <https://github.com/ValveSoftware/openvr/wiki/API-Documentation>.
- [41] F. Argelaguet, L. Hoyet, M. Trico, and A. Lecuyer, “The role of interaction in virtual embodiment: Effects of the virtual hand representation,” in *2016 IEEE Virtual Reality (VR)*, pp. 3–10, 2016. <https://ieeexplore.ieee.org/document/7504682>.
- [42] C. Strobel and K. Engedal, *Norsk revidert mini mental status evaluering (MMSE-NR)*. Norsk Helseinformatikk AS (FHI), 2020 (Retrieved 2020-05-06). <https://nhi.no/skjema-og-kalkulatorer/skjema/geriatripleie/mmse-nr2-mms-norsk-revisjon>.
- [43] Unity Technologies, *Unity - Manual: Writing Surface Shaders*, 2017 (Retrieved 2020-05-29). <https://docs.unity3d.com/Manual/SL-SurfaceShaders.html>.
- [44] “Coin games.,” 2020 (Retrieved 2020-04-03). <https://www.giantbomb.com/coin/3055-219/games>.
- [45] E. C. Hansen, *Successful Qualitative Health Research: A Practical Introduction*, pp. 98–102. Open University Press, 2006.
- [46] E. Toftøy-Andersen and J. G. Wold, *Praktisk brukertesting*, pp. 147–149. Cappelen Damm akademisk, 2011.
- [47] “System Usability Scale (SUS) is a 10-item questionnaire with five response options that provides a reliable tool for measuring perceived usability,” 2020 (Retrieved 2020-05-11). <https://www.usabilitest.com/system-usability-scale>.
- [48] “Energisenteret for barn og unge,” 2019 (Retrieved 2019-03-30). <https://helse-bergen.no/avdelinger/energisenteret-for-barn-og-unge#les-meir-om-tilboda-til-energisenteret-for-barn-og-unge>.
- [49] “Ado arena,” 2020 (Retrieved 2020-05-04). <https://adoarena.no/en/about-ado-arena>.
- [50] “Playwell,” 2020 (Retrieved 2020-05-04). <https://en.playwell.no/about>.

- [51] “Gg bergen,” 2020 (Retrieved 2020-05-04). <http://www.ggbergen.org>.
- [52] “Idrettscampus bergen,” 2020 (Retrieved 2020-05-04). <https://www.idrettscampusbergen.no>.
- [53] Valve Software, *SteamVR Unity Plugin v2.6.0b1 (sdk 1.11.11)*, 4 2020 (Retrieved 2020-05-31). https://github.com/ValveSoftware/steamvr_unity_plugin/releases/tag/2.6.0b1.
- [54] Unity Technologies, *Unity Manual - XR Plug-in Framework*, 5 2020 (Retrieved 2020-06-01). <https://docs.unity3d.com/Manual/XRPluginArchitecture.html>.

Appendix

The appendix contains the interview protocol and system usability scale form intended for evaluating the software.

Additionally, a manual for setting up and calibrating the software is also included.

The appended documents are in Norwegian only.

Appendices

INTERVJUPROTOKOLL

Demonstrasjon av produktet før intervjuet

Før intervjuet vil intervjuobjektene få prøve produktet i plenum. Her vil de få muligheten til å finne ut av hvordan det fungerer og diskutere seg imellom. Vi vil på forhånd kalibrere VR-oppsettet så spiller er korrekt plassert og orientert når de er i spillet. Intervjuobjektene vil instrueres i hvordan fot-sporingsenhetene skal plasseres og gi en kortfattet demonstrasjon av stibyggerverktøyet før den første av deltakerne skal teste. Ideelt sett finner deltakerne ut av ting selv når de kommer inn i spillet. Vi vil bistå med å svare om det er noe de opplever som utydelig.

De vil og få anledning til å bygge sin egen sti med egenvalgte vanskelighetsnivå og innstillinger for å kunne evaluere denne funksjonaliteten senere i intervjuet.

Kartleggingsspørsmål:

1. Hvilken bakgrunn har du i innen fysioterapi og fysisk rehabilitering?
2. Hva slags erfaring/bakgrunn har du med VR?
3. Har du erfaring innen spill-basert rehabilitering?
4. Noe mer dere anser relevant for dette prosjektet?

Intervjuguide:

(Krysse av underveis og se om det er nødvendig å ta opp manglende punkter)

Diskusjonspunkter:

Helhetsinntrykk Hvilket helhetsinntrykk fikk du da du prøvde VR Walk?

- Visuell kvalitet
- Generell kvalitet
- Ytelse (ytelsesproblemer?)
- Ease of use
- Opplevelse (moro?)

Øvelser Gjorde du deg noen meninger om noen av oppgavene du fikk underveis?

- Fotsporing og følging av fotspor
- Bro og kasting
- Følge sti og unngå hindringer
- Mynter
- Balanseball
- Memorering og sammensetting av totempæl
- Variasjon

Vanskelighetsgrad Var det oppgaver som virket for enkle, unødvendige eller kanskje for vanskelige?

- Om det var overflødige oppgaver (for enkle?)
- Var det for vanskelige oppgaver for målgruppen
- Skalering

Forståelse Var det noe som var vanskelig å forstå om bruken av VR Walk?

- Om noen av oppgavene var vanskelige å forstå
- “Feedback” (lydbruk, symbolbruk)
- Instruksjer
- Poenggiving

Grensesnitt Noen tanker angående menyene eller interaksjon ellers?

- Grafisk grensesnitt (GUI, menyer)
- Stibyggerverktøyet
- Styring av tredemøllen
- Bruk av kontroller

Nytteverdi Gjorde du deg noen tanker om den potensielle nytteverdien til VR Walk?

- Innen rehabilitering som produktet fremstår i dag
- Evt. mangler før potensiell bruk innen rehabilitering
- Som grunnlag for videre forskning
- Andre potensielle områder innen trening eller rehabilitering

Noen spørsmål om systemet du har brukt.

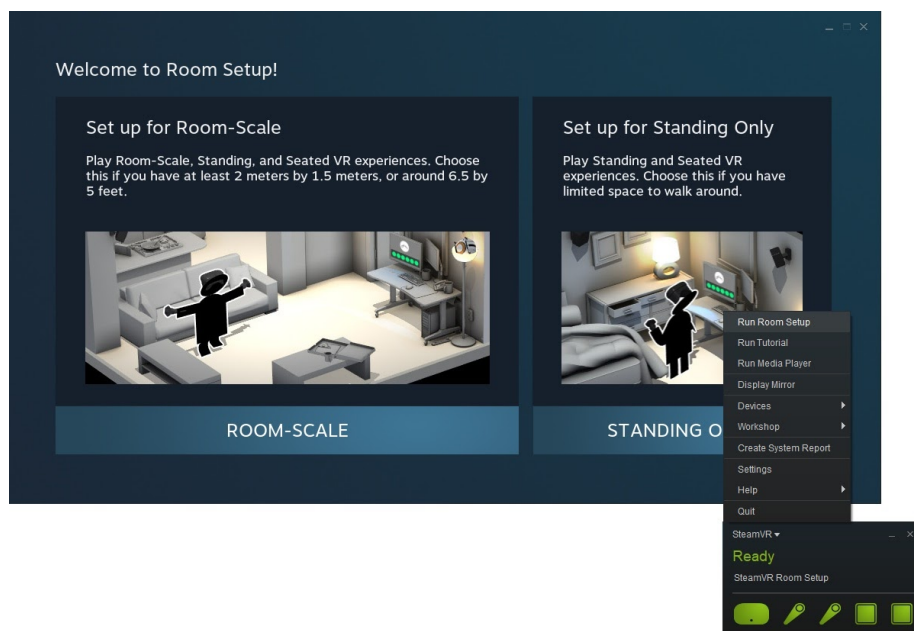
Vennligst sett kryss i kun en rute pr. spørsmål.

- | | Sterkt
uenig | | | | Sterkt
enig |
|---------------------------------------------------------------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| 1. Jeg kunne tenke meg å bruke dette systemet ofte. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 2. Jeg synes systemet var unødvendig komplisert. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 3. Jeg synes systemet var lett å bruke. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 4. Jeg tror jeg måtte trenge hjelp fra en person med teknisk kunnskap for å kunne bruke dette systemet. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 5. Jeg synes at de forskjellige delene av systemet hang godt sammen. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 6. Jeg synes det var for mye inkonsistens i systemet. (Det virket "ulogisk") | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 7. Jeg vil anta at folk flest kan lære seg dette systemet veldig raskt. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 8. Jeg synes systemet var veldig vanskelig å bruke. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 9. Jeg følte meg sikker da jeg brukte systemet. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |
| 10. Jeg trenger å lære meg mye før jeg kan komme i gang med å bruke dette systemet på egen hånd. | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| | 1 | 2 | 3 | 4 | 5 |

System Usability Scale (SUS). Norsk oversettelse ved Dag Svanes, NTNU[46]

MANUAL: OPPSETT OG KALIBRERING

Før programstart



Før oppstart av programmet, bør SteamVR Room-Scale konfigurering gjennomføres. Det er da viktig at basestasjonene er plassert med god sikthet til både hode, hender og føtter. Ved oppsett av gulvhøyde skal håndkontrollerene plasseres på tredemøllebåndet. Om mulig; få VR-området til å dekke hele møllebåndet. Tredemøllen skal ikke ha noen helling på båndet.

Kalibreringsprosessen

Programmets kalibrering sikrer godt samsvar mellom den fysiske tredemøllen og VR-miljøet. Velg “Kalibrering”-knappen i hovedmeny. Dette tar deg til kalibrerings-scenen.

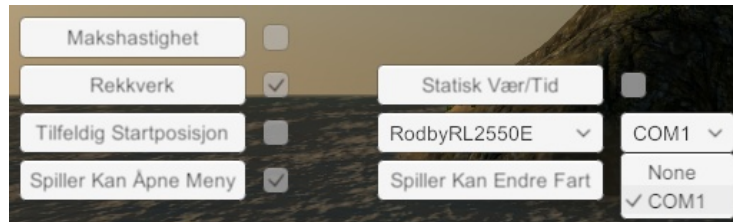
Pass på at begge håndkontrollene er skrudd på, og at de spores av basestasjonene. Dette kan du kontrollere enten ved å se om kontrollernes modeller beveger seg i scenen, eller i SteamVR-applikasjonen. I SteamVR vil kontrollersymbolene ha en jevn grønnfarge hvis de spores tilstrekkelig. Symbolene vil blinke om sporingen er dårlig.



Kalibrering steg for steg:

1. Plassér hver av håndkontrollene på hver sin side av tredemøllebåndet nøyaktig like langt fremme. Midt mellom vil være den foretrukne posisjonen å stå i spillet. Pass på at de peker fremover i gangretning.
2. Deretter trykkes enten 'Enter' på tastaturet, eller touch-platen på en av kontrollene for å kalkulere kalibreringen.
3. Dersom dette ser ut til å stemme godt, kan kalibreringen lagres med "Lagre"-knappen. Hvis ikke, kan man trykke den andre knappen, og begynne fra steg 1 igjen.
4. Kalibreringen er ferdig, og man kan gå rett til kjøring, eller gå tilbake til hovedmeny om man vil endre på innstillinger. Kalibreringsdata lagres lokalt på maskinen, så det skal ikke være nødvendig å kjøre kalibrering på nytt, med mindre SteamVR Room-Scale har blitt endret på noen måte, f.eks ved at en basestasjon har blitt flyttet.

Innstillinger



Fra hovedmenyen kan en trykke seg inn på Innstillinger. Her vil en få de følgende valgene:

- **Makshastighet:** Med denne valgt, vil ikke hastighet kunne stilles høyere enn 5 km/h.
- **Rekkverk:** Om det virtuelle rekkverket skal være synlig ved oppstart. Dette kan overstyres under kjøring enten gjennom spillerens menyknapp eller ved at operatør trykker 'R'.
- **Tilfeldig Startposisjon:** Om stien skal starte et tilfeldig sted langs hovedstrekningen på øyen eller ved syd-spissen av den.
- **Spiller kan åpne meny:** Tillater spilleren å selv åpne opp en meny med innstillinger for rekkverk, fotplassering, fartssperre og å vende tilbake til hovedmeny. Denne menyen vil kunne åpnes med menyknappen på kontrolleren.
- **Statisk Vær/Tid:** Om spillet skal la være å benytte dag-og-natt syklus og endring av værforhold.
- **Spiller Kan Endre Fart:** Når denne er valgt, vil spilleren selv kunne kontrollere ganghastigheten ved å peke på knappene til venstre på det virtuelle displayet og holde inne trigger-knappen. Operatøren vil ha mulighet til å stille opp og ned hastighet med 'Pil opp' og 'Pil ned' på tastaturet uansett valg.

Valg av tredemølle

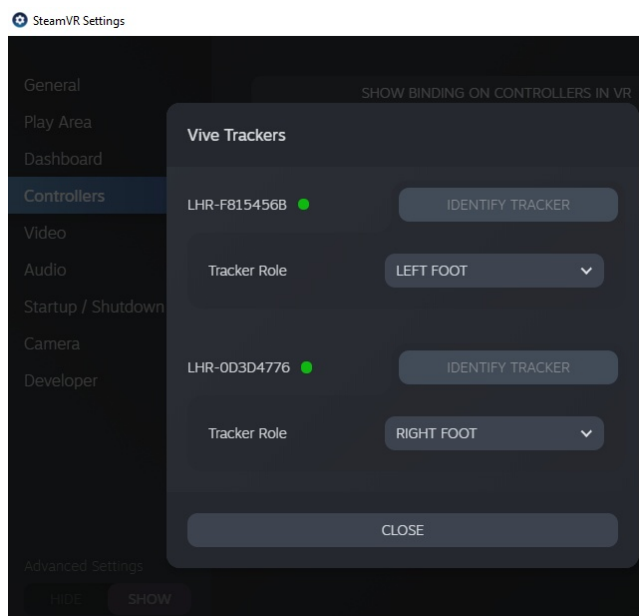
Under innstillinger vil en få valg mellom å enten benytte Rodby eller Woodway, i tillegg til å velge hvilken COM-port denne er koblet til. Dersom ingen av de støttede tredemøllene er koblet til, vil en ikke kunne styre tredemøllen direkte fra spillet og bør ha None valgt som COM-port.

Dersom ingen tredemølle er tilkoblet vil tredemøllevalget kun påvirke stibredden og utvalget av aktiviteter, da sti-settet for Rodby tredemøllen er bredere.

Kalibrering av føtter



For å kunne benytte føtter i spillet, må en ha HTC Vive trackere tilgjengelig og stroppet til utsiden av begge anklene, som vist på bildet. Disse er nødvendige for å kunne gis poeng under fotspor-aktiviteten eller trekkes for poeng ved trakking utenfor stien. De vil og utløse nødstopp dersom føttene kommer for nær kanten på tredemøllebåndet. For at sporingen skal fungere, må rollene til trackerene være satt til "LEFT FOOT" og "RIGHT FOOT" i SteamVR under Settings → Controllers → Manage Vive Trackers.



Oppretting av føtter

Oppretting av føtter skjer først etter en har startet opp en løype.

Dersom det er en operatør tilstedet, skal operatør be spilleren stille seg opp med begge føttene samlet, slik at begge peker rett fremover. Deretter trykker operatøren 'F' på tastaturet og ber spilleren kontrollere at føttene ble riktig plassert. Dersom en tracker løsner eller forskyves underveis, kan en stoppe tredemøllen og gjenta prosessen. Dersom en operatør mot anbefaling ikke er tilstede, kan spilleren selv åpne opp en meny ved å trykke menyknappen på håndkontrollen og selv velge å plassere føttene, men da må "Spiller Kan Åpne Meny" være valgt under innstillinger.

Siden sporing av føttene ofte ikke er like stabil som annen sporing, kan operatøren velge å slå av at nødstopp skal utløses av føttene ved å trykke 'S' på tastaturet ved startet løype.