Factory Reset

The Design and Evaluation of an Accessible Game

Submitted to the Faculty of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Bachelor of Science in

Computer Science

and for the

Degree of Bachelor of Science in

Interactive Media and Game Development

and for the

Degree of Bachelor of Arts in

Interactive Media and Game Development

Authors:

Christian Orion Adler
coadler@wpi.edu

Raymond Dolan
rfdolan@wpi.edu

Elizabeth Kirschner
eekirschner@wpi.edu

Joseph Henry Stadolnik IV
jhstadolnik@wpi.edu

Faculty Advisors:

Prof. Gillian Smith
gmsmith@wpi.edu

Prof. Ralph Sutter
rsutter@wpi.edu

This report represents the work of one or more WPI undergraduate students submitted to the faculty as evidence of completion of a degree requirement. WPI routinely publishes these reports on the web without editorial or peer review.
Abstract

*Factory Reset* is a 3D, first-person, puzzle game set in a Sovietpunk world. Four WPI seniors from the IMGD and CS departments created *Factory Reset* over the course of one academic year. The project incorporated digital art, programming, and accessibility-oriented design to create an immersive video game. The team developed the game using feedback from three rounds of playtesting to inform changes to the game’s design. The team evaluated how well they met their goals through playtesting feedback. Through the project, the team demonstrated their skills with development tools and created a game that was well received by playtesters. The final game features an accessible overall design and offers 14 different types of accessibility features that make it playable by a wide range of people.
Acknowledgements

The team would like to thank their advisors: Professors Gillian Smith and Ralph Sutter for their supportive feedback and trust-filled encouragement over the course of this year-long project. The team would also like to thank the wonderful people who lent their voices to the game: Skyler Emery (Moscow, Russian Federation), Anastasia Nastya (Chişinău, Moldova), and Kristinia Golina (Kiev, Ukraine). Finally, the team would also like to thank all the playtesters who supported the game and gave feedback along the way.

*Factory Reset* would never have been possible without all of these wonderful people.
# Table of Contents

Abstract 1

Acknowledgements 2

Table of Contents 3

List of Figures 8

1. Introduction 14

2. Background 16
   2.1. Accessibility 16
   2.2. Game Inspirations 19
      2.2.1. Portal 2 19
      2.2.2. Half-Life 2 20
      2.2.3. The Legend of Zelda: Twilight Princess 21
      2.2.4. Manifold Garden 23
   2.3. Art Inspirations 24
      2.3.1. Pseudo-Realism 25
      2.3.2. Cyberpunk Genre 26
         2.3.2.1. Ridley Scott’s 1982 Blade Runner 27
         2.3.2.2. Paul Verhoeven’s RoboCop (1987) 28
      2.3.3. The Legend of Zelda: Breath of the Wild 29
      2.3.4. Furi 30
      2.3.5. Sayonara Wild Hearts 31
      2.3.6. Control 32
      2.3.7. Soviet Union Fourth Age of Architecture 33
   2.4. COVID-19 Pandemic 34

3. Design 35
   3.1. Design Goals 35
   3.2. Experience Goals 35
   3.3. Audience 36
   3.4. Accessibility 36
   3.5. Perspective 39
   3.6. Mechanics 39
      3.6.1. Basic Player Controls 39
      3.6.2. Magnetic Feet 40
      3.6.3. Magnetic Hands 41
   3.7. User Interface 42
   3.8. Game Structure 44
3.9. Level Design 46
  3.9.1. Level Design Goals 47
  3.9.2. Level Design Process 47
  3.9.3. Level 1: The Storage Facility 49
  3.9.4. Level 2: The Sky Highway 50
  3.9.5. Level 3: The Factory Lobby 52

3.10. Scope 55
  3.10.1. Dream Design Scope 55
  3.10.2. Streamlined Design Scope 56

4. Gameplay Implementation 58
  4.1. Tools 58
    4.1.1. Game Engine 58
    4.1.2. Version Control 59
  4.2. Mechanics and Controls 59
    4.2.3. Magnetic Feet Development 60
      4.2.3.1. Constraining the Mechanic 60
      4.2.3.2. Physics Challenges 62
      4.2.3.3. Polarity Clarity Challenges 63
      4.2.3.4. Conflict with First Person View 64
      4.2.3.5. Refining the Magnetic Feet 65
      4.2.3.6. Third Person Experiment 66
    4.2.4. Magnetic Hands Development 67
      4.2.4.1. Polarity Boxes 69
      4.2.4.2. Scaffolding 70
  4.3. User Interface 70
    4.3.1. Heads-Up Display 70
    4.3.2. Menu 76
      4.3.2.1. Pause Menu 77
      4.3.2.2. Settings Menu 78
      4.3.2.3. General Settings 79
      4.3.2.4. Controls Settings 80
      4.3.2.5. Graphics Settings 82
      4.3.2.6. Audio Settings 83
    4.3.3. Subtitle System 84
  4.4. Cutscene System 85
  4.5. The Infinite City 85

5. Art Development 89
  5.1. Overview 89
    5.1.1. Defining Sovietpunk 90
    5.1.2. Establishing Factory Reset’s Sovietpunk 91
5.2. High Definition Render Pipeline
5.3. Prop Creation Pipeline
  5.3.1. Conceptualization
  5.3.2. Visual Accessibility Design
  5.3.3. Modeling
  5.3.4. UV Unwrapping & Texturing
5.4. Environmental Art Pipeline
  5.4.1. Conceptualization
  5.4.2. Greyboxing
  5.4.3. Modeling
    5.4.3.1. Original Modeling Design
    5.4.3.2. Modularity Switch
  5.4.4. UV Unwrapping and Texturing
  5.4.5. Environmental Object Placement (Set-Dressing)
  5.4.6. Lighting
  5.4.7. Volumetrics & Post Processing
5.5. Character
  5.5.1. Conceptualization
  5.5.2. Modeling
  5.5.3. UV Unwrapping and Texturing
  5.5.4. First Person View Model
6. Audio
  6.1. Inspirations and Aesthetic
    6.1.1. Basic Computer-Generated Sounds
    6.1.2. Portal
    6.1.3. Music Inspiration
  6.2. Tools
  6.3. Sound Effects
    6.3.1. Feet
    6.3.2. Hands
    6.3.3. Environmental Sound Effects
  6.4. Ambience
    6.4.1. Noise
    6.4.2. Music
  6.5. Voice Acting
7. Testing
  7.1. Testing Methodology
  7.2. Testing Results
    7.2.1. Alphafest
    7.2.2. C Term
7.2.3. D Term

8. Discussion

8.1. Experimental Design Challenges

8.2. Accessibility

8.2.1. Gameplay Accessibility

8.2.2. Hardware Requirements

8.3. Design Goal Evaluation

8.4. Technological Challenges

8.5. Artistic Challenges

8.6. Looking Forward

References

Appendix A: Project Website

Appendix B: Screening Form

Appendix C: Informed Consent Notice - Public Testing

Appendix D: Informed Consent Notice - Supervised Testing

Appendix E: Alphafest Playtesting Survey

Appendix F: Pre-Session Questionnaire

Appendix G: Spring Playtesting Survey

Appendix H: Forums

Appendix I: Factory Reset Sign Up Play Testing Instructions

Appendix J: Sign Up Session Email Template

Appendix K: Sample Reddit Outreach Post

Appendix L: Sample Twitter Outreach Post

Appendix M: README
List of Figures

Figure 1. The logo for Factory Reset.

Figure 2. Informatic showcasing the False Color Equivalence of the three different types of dichromatic vision. (Collinge 2017). Image (Iris, n.d.).

Figure 3: A screenshot of a test chamber in Portal 2 is shown on the left (Valve, 2011). Image (Jongeneel, 2013). A screenshot of Factory Reset is shown on the right for comparison.

Figure 4. The Gravity Gun from Half-Life 2 (Valve, 2004). Image (Burgar, 2021).

Figure 5. The Clawshot item being used in The Legend of Zelda: Twilight Princess (Nintendo, 2006). Image (Zelda Dungeon, n.d.).

Figure 6: The player walking on a magnetic ceiling in the Goron Mines dungeon from The Legend of Zelda: Twilight Princess (Nintendo, 2006). Image (Zelda Dungeon, n.d.).

Figure 7. Screenshot from Manifold Garden (William Chyr Studio, 2019). Image (GameFFine, 2020).

Figure 8: The mood board the team created for conceptualizing the visual design of Factory Reset. Images (Shashkin, 2019), (Bai, 2018), (Suzdal, 1989), (Zubkov, 1959), (Romadin, 1979)

Figure 9: Concept art from Cyberpunk 2077 (CD Projekt, 2020) showcasing a cyberpunk environment. Image (CD Projekt RED, 2020)

Figure 10: Still-frame from Ridley Scott’s 1982 Blade Runner (top) showcasing the sprawling city and flying cars, as inspiration for the city level in Factory Reset (bottom). Image top (Mead, n.d.)

Figure 11: Paul Verhoeven’s RoboCop (1987) standing outside of a police cruiser cast, demonstrating a strong contrast between red and blue lighting. Image (Bryan, 2019).


Figure 13: Examples of a Shrine area in The Legend of Zelda: Breath of the Wild (Nintendo, 2017). Images (Talbot, n.d.)

Figure 14: A screenshot from Furi (left) showing the player character’s emissives. Factory Reset’s player character (right) had emissives to highlight the action being performed with their hands. Image left (Devore, 2016).

Figure 15: The city environment featured in the third major section of Sayonara Wild Hearts (top) and the city environment featured in Factory Reset (bottom). Image top (Flesser, 2020)

Figure 16: Some examples of the brutalist architecture featured in Control (top) and the stark, brutalism-inspired lobby of the third level in Factory Reset (bottom). Image top left (Ewan, 2019) Image top right (Ewan, 2019)

Figure 17: Photographs taken by Frédéric Chaubin, from Cosmic Communist Constructions (Chaubin, 2011). Images (Chaubin, 2011)

Figure 18: Kalinin Prospekt and Novy Arbat, Moscow (Suzdal, 1989). Image left (bouncermangel, 2014).

Figure 19: The positive polarity pad (left) and the negative polarity pad (right).

Figure 20: The positive polarity box (left) and the negative polarity box (right).
Figure 21: The player looking at a polarized surface then pulling onto it using the magnetic feet ability (top), then looking at a polarity pad and pulling from there onto the polarity pad using the feet ability again (bottom). The reticle is visible in the center of each image.

Figure 22. Magnetic hands in Negative (top left), Positive (top right), and Neutral (bottom center) polarity modes. The polarities are supplemented with matching symbols and colored regions on the reticle in the center of the screen.

Figure 23. A visual breakdown of the final HUD.

Figure 24: The trash compactor (top), elevator shaft (middle), and hangar (bottom) areas of the first level in Factory Reset.

Figure 25: The second level of Factory Reset, where the player travels across a group of flying cars in a city.

Figure 26: The lobby (top) and puzzle room (bottom) of the third level in Factory Reset.

Figure 27. Early concept images of environment ideas.

Figure 28. An early sketch of an unused level design idea.

Figure 29. The first complete sketch of the planned layout for the first level of Factory Reset (left) and an early greybox version of the level (right). The player would start in the lowermost area, ascend a triangular elevator shaft using their magnetic abilities, and exit the level by hitching a ride on a departing vehicle in the hangar area in the upper-left.

Figure 30. The initial concept sketch for the second level (top) and an early greybox version of it (bottom).

Figure 31. The initial concept sketch for the third level, showing the basic level layout and some initial sketches of the lobby room.

Figure 32. A render of OLOF-14 (left) and an early look at them in the greybox of the third level (right).

Figure 33. Initial sketches and descriptions of the power cell (upper-left), power supply (lower-left), and movable scaffold (right) puzzle objects.

Figure 34. The three power cell puzzles the team initially designed for the third level. One must be reached by knocking the objects supporting it down (upper-left), one must be reached by moving a scaffold under a polarity pad (lower-left), and one must be reached by moving one scaffold out of the way of another and using a polarity pad attached to it (right).

Figure 35. A prototype polarity target placed on a long metal surface.

Figure 36. A polarized surface that is outside of the player’s range.

Figure 37. If a player repels from a surface in the direction opposite gravity (top), their movement would be purely vertical and very controlled. If they repelled from a surface at a different angle (bottom), they would have a less-intuitive path, especially seen from a first person view.

Figure 38. The special inverted collider volume used to keep the player from stepping off of polarized surfaces.

Figure 39. The third person mock-up test the team performed.

Figure 40. HUD indicators for when an object is out of range (left) and in range (right).

Figure 41. A movable polarity box (left) and movable scaffold (right), both negatively charged.

Figure 42. The pre-Alpha (left), Alpha (middle), and final (right) versions of the HUD reticle.
Figure 43. An early test of Factory Reset’s heads-up display. The reticle is visible in the center, with the hand polarity shown in the arc above it, the polarity of the feet shown in the arc below it, the orientation compass below that, the feet polarity glow along the bottom of the screen, and the graphical limb status readout in the upper-right.

Figure 44. A screenshot from Portal 2 (Valve, 2011) showing its aiming reticle. Image (Steam, n.d.).

Figure 45. An assortment of sketches of possible reticle designs which the team strongly considered (top), the version from the Alpha build (bottom left), and the final refined version (bottom right). This version featured special toggle-able indicators at the top to show when an object could be manipulated with the player’s hands and at the bottom to show when the targeted surface can be pulled to.

Figure 46. The heads-up display from Factory Reset’s Alpha build. The first image shows the player picking up a polarity box with their magnetic hands, while the second image shows the player looking at an interactable switch that is out of their reach and thus currently inaccessible.

Figure 47. The redesigned heads-up display featured in the final build.

Figure 48. The Alpha build’s pause menu (top) and the final pause menu (bottom).

Figure 49. The top-level settings menu.

Figure 50. The general settings menu.

Figure 51. The remappable controls menu.

Figure 52. The graphical settings menu.

Figure 53. The audio settings menu.

Figure 54: Three views of the sky highway level in Factory Reset.

Figure 55: The three models of buildings used to form the city in Factory Reset.

Figure 56: An untextured flying car from Level 2 (left) and the textured flying car (right) within Unity.

Figure 57: Early whiteboxed environments for Factory Reset, showcasing the fusion between Cyberpunk and Sovietpunk.

Figure 58: 3D render of the antagonist, OLOF-14, showcasing the meld of both Soviet iconography and science fiction and futurism.

Figure 59: “Sovietpunk” by Ian Llanas (2016), an art piece showcasing the divide between traditional Russian architecture and a futuristic vision of a Soviet future.

Figure 60: Concept art created for Factory Reset envisioning a Soviet factory.

Figure 61: Concept art created for Factory Reset envisioning an interior of a dilapidated Soviet environment.

Figure 62: Screenshot from Factory Reset showcasing the two variants of polarity boxes: red (positive) and blue (negative).

Figure 63: The two polarity boxes simulated with different forms of colorblindness. Trichromatic (top left), Deuteranopia (top right), Protanomaly (bottom left), and Tritanopia (bottom right).

Figure 64: The two polarity boxes simulated with monochromatic Full Color Blindness, Achromatopsia (left) and Blue Cone Monochromacy (right).

Figure 65: Final environment lighting colorblindness simulations. Trichromatic (top left), Deuteranopia (top right), Protanomaly (bottom left), and Tritanopia (bottom right).
Figure 66: Screenshot from Autodesk Maya, showcasing the different props used (and some which were not used) within *Factory Reset* game environment.

Figure 67: Crate render for showing the detailed texture work, specifically finger prints.

Figure 68: Small detailed prop of a Soviet 1957 Travellers Mug, showcasing the intricate texture work found within *Factory Reset*’s props.

Figure 69: An example of a UV texture map of the “Overhang Light”, showcasing the detailed texture work of grime, scratches, and other discolorations throughout.

Figure 70: Close up view of the door switch found within *Factory Reset* (top). Mao Tse-tung (Chairman of the PRC) and Nikita Khrushchev (Soviet Premier of the Soviet Union) meeting together two years during the historical split in 1958 (bottom left (Unknown, 1958)). Propaganda poster utilizing the two scripts showcasing the “Socialist Fraternal Love” painted by Vikto Ivanovich Ivanov (1958) Image bottom right. (Archive, W. C. D., 1953)

Figure 71: Team sketches of *Factory Reset*’s second and third level showcasing specific assets and the overall rough setup of the environment.

Figure 72: Photo provided by Hayabusa, at the Hayabusa Vertical Parking System. (München, Germany). Image (WWJD, 2016).

Figure 73: Greybox environment of Level One’s first section, showcasing polycount.

Figure 74: Original textured environment for Alphafest (as well as drawn concerns with the pre-modular approach)

Figure 75: Final look at the first segment of Level One showcasing the many changes in texture work, environmental modeling, lighting pass, volumetrics & post-processing, and set-dressing.

Figure 76: *Factory Reset*’s Level One original model atlas, showcasing all five parts (and the flying truck)

Figure 77: Level One’s second segment’s inefficient UV Map.

Figure 78: Screenshot from the Alphafest build, showcasing the problems that existed with the current non-modular switch.

Figure 79: The previous Level One pre-modularity change models (left) and the new modular Level One pieces (right).

Figure 80: Full model atlas of modular level elements created in order to be reused throughout various levels.

Figure 81: Level One’s original texture work, which was later changed.

Figure 82: Level One’s textures applied within Unity’s Development Environment.

Figure 83: Screenshot taken from *Factory Reset*’s first level, final segment, showcasing the use of teal and orange.

Figure 84: *Factory Reset*’s final set dressing pass in Level One.

Figure 85: Another example of clutter during the set dressing stage of the game on Level One.

Figure 86: Level One’s lighting from above, showcasing the contrast between the “warmer” orange side (left), and the “cooler” blue side (right).

Figure 87: Lighting was used to direct the player’s direction to look at certain areas of the map. In this screenshot, the light points towards the polarity pad on Level One.

Figure 88: An example of the first level’s high contrast lighting.

Figure 89: An example of the lighting interacting with volumetric fog profiles mixing orange and teal lights together.
Figure 90: Near-final scene showcasing lighting without volumetric post processing effects.

Figure 91: Final scene showcasing lighting with volumetric post processing effects and custom volumetric profile.

Figure 92: Original concept art for various types of non-humanoid FyED-OR robots. The team eventually settled on the centered variant.

Figure 93: Russian Space Agency’s FEDOR robot on launch to the International Space Station in 2019. (Advanced Research Foundation, n.d.)

Figure 94: Another iteration sketch of a humanoid FyED-OR.

Figure 95: Screenshot from Meshlab showing the final version of the player character FyED-OR.

Figure 96: Final version of FyED-OR painted orange.

Figure 97: Screenshot of an unfinished blue variant of FyED-OR, before the team decided on orange.

Figure 98: A screenshot from Factory Reset showcasing the first person camera and view model.

Figure 99. Album covers from Я люблю тебя, возвращайся домой (Эйфория, 2019) by Эйфория (left) and Побег (Побег, 2018) by Побег (right) which inspired the music for Factory Reset.

Figure 100. The plugins used for the final foot sounds, including the “JS: Flanger” plugin that comes with reaper.

Figure 101. Rolloff graph on a Unity AudioSource component which dictates its volume at certain distances.

Figure 102. Some code responsible for pitch randomization on the flying vehicles.

Figure 103. The timeline for “Don’t Fall, FyED-OR”.

Figure 104. The timeline for “Where am I?”

Figure 105. Playtesting timeline.

Figure 106. Survey results for how much participants agree with the statements “I clearly understood how to use the magnet boots” and “I clearly understood how to use the magnet gloves.”

Figure 107. Survey results for how much participants agree with the statement “movement in the game was disorienting.”

Figure 108. Survey results for how much participants agree with the statements “movement in the game was NOT disorienting” and “I did NOT experience motion sickness while playing the game.”

Figure 109. Survey results from Beta 0.1 and 0.2 reflecting which settings were modified.

Figure 110. Graphs showing how much players agreed with the statements “Movement in the game was NOT disorienting” and “I did NOT experience motion sickness while playing the game.”

Figure 111. Graphs showing how much players agreed with the statements “I clearly understood how to use the magnet boots” and “I clearly understood how to use the magnet gloves.”

Figure 112. Results of running Unity’s built-in profiler on levels one, two, and three respectively.

Figure 113. Word cloud of the 25 most common words from responses to the question “What did you like most about the game?”

Figure 114. Word cloud of the 25 most common words from responses to the question “What did you dislike most about the game?”
Figure 115: An example showing the clear disconnect between level elements texture resolution within the modular system that could only be addressed with increasing texture resolution manually.

Figure 116: Screenshot from the game showcasing the high resolution texture work for the door switch prop, and the much lower resolution level element wall.
1. Introduction

*Factory Reset*, is a single player, first-person, puzzle and traversal game about a magnetic robot named FyED-OR navigating their way through a Soviet-inspired cyberpunk world. The player uses their magnetic feet and magnetic hands to traverse a sequence of three different level environments and solve tricky puzzles along the way. It is intended to function as a demo for a larger hypothetical game. The full experience typically takes about 10 to 20 minutes to play through. The logo for Factory Reset can be seen in Figure 1.

![Figure 1. The logo for Factory Reset.](image)

*Factory Reset* was designed to showcase the skills that the project team gained and developed during their time at WPI, including programming, 3D modeling, texturing, and audio design. The team also developed it with a focus on accessibility, incorporating a range of features and overall design decisions intended to make it accessible to a wide audience. This was to highlight the importance of designing games to be accessible through overall design and through a wealth of gameplay customization options.

The team originally built the game around the idea of magnetizing to surfaces with magnetic boots. This evolved to include magnetic gloves that allowed for interesting interaction with the environment. Building off this technological theme, the team looked to the cyberpunk genre and the architecture from the Post-Soviet era to develop a Sovieterpunk world and art style for the game.
Through playtesting the team iterated upon the design of *Factory Reset* and evaluated its success. From this testing, the team found that discoverability of settings is critical to their use, first person reorientation inherently brings accessibility challenges, and game performance is a large barrier to play.

This paper is split up into several sections detailing different aspects of the project. The background provides valuable information based on the team’s research, which is required to understand the challenges of the project. The design section covers the concepts and rationale for the mechanics and other critical elements of the game. Key challenges faced and decisions made throughout development are documented in their respective areas by subject matter - technical development, art development, and audio. The findings from playtesting sessions are found in the results and data section. Finally, there is a discussion of how well the team feels that the project achieved its goals and what the team plans to do with the project going forwards.
2. Background

This project builds on prior research and design in game accessibility, 3D physics-based puzzle design, and soviet cyberpunk aesthetic design. Section 2.1 gives a brief overview of the importance and history of accessibility in games. Sections 2.2 and 2.3 cover the games and other media that inspired the gameplay and art for Factory Reset, respectively. Finally, section 2.4 covers the impact of the COVID-19 pandemic on the development of Factory Reset.

2.1. Accessibility

Video games are an important part of modern culture, yet the nature of many games often makes them inaccessible to a large audience of potential players. Games are a fun source of entertainment, but inaccessible design makes games frustrating and unplayable. Instances of unplayability are referred to as disabling situations and occur when there is a mismatch between a person’s ability and the barriers presented by an interaction (The IGDA Game Accessibility Special Interest Group, 2020). Often these barriers are created unintentionally through design and are easily removed. Other times accessible features can easily remove barriers for the players who need them. The goal of accessible features in games is to close the gap between the needs of individual players and the demands of the game. This gap is also known as a ‘digital divide’ that highlights how inequality is present even in the digital word (Wästerfors & Hansson, 2017).

Disabling situations crop up in a large range of games and game features. Although there are a vast number of design choices that can make a game completely inaccessible, it is also common for a game to be entirely playable except for small sequences.

An example of this can be seen in Uncharted 2: Among Thieves (Sony Computer Entertainment, 2009) where players have reported that the game is entirely playable except for a short sequence near the end of the game where players need to press one button repeatedly to open a set of doors (Straub, 2012). Another example is the lock picking mechanic in Assassin's Creed III (Ubisoft, 2012) which forces players to hold both joysticks in a specific direction while repeatedly pressing the triggers to break open the lock (Wästerfors & Hansson, 2017). Both of these disabling situations can be solved with motor accessibility features that allow players to bypass rapid, repeated button presses. These situations can be extremely frustrating to players who would otherwise have a positive and uninterrupted play experience.
Another common form of inaccessibility comes from the visual design of games. Conveying important game information solely through color is inherently inaccessible to those who cannot distinguish certain colors. This issue is solved by conveying critical visual information through multiple formats, including using additional, non-visual, means.

Accessible game features are typically divided into five categories: hearing, motor, cognitive, speech, and visual. Each category covers a large range of features, many of which overlap in their ability to aid players. Hearing features aid in understanding and audio feedback in a game. Motor features typically reduce the need for quick or repeated actions which aids players who cannot press buttons quickly or cannot hold a standard video game controller. Cognitive features help with understandability, puzzle solving, and dyslexia. Speech features typically aid with language barriers and any verbal requirements of a game. Visual features focus on non-visual signifiers\textsuperscript{1} which to make gameplay understandable through non-visual feedback\textsuperscript{2}.

Within the eyes, there are three color receptors which respond to different groups of light wavelengths and thus allow people to distinguish red, green, blue, and combinations thereof. The receptors respond most strongly to their namesake colors, but react to a lesser extent to others in specific patterns. Being able to use all three color receptors is called tritanomal vision; however, this state is not universal (Mukamal, 2017). Potential players may only be able to utilize two, a condition called dichromatic vision. A depiction of this can be seen in Figure 2. Within dichromacy, there are multiple variants: some cannot use their green color receptor, called deuteranopia; some cannot use their red color receptor, called protanopia; and some cannot use their blue color receptor, called tritanopia. As a result, a dichromat has difficulty distinguishing certain pairs of hues.

\textsuperscript{1} Signifiers: indicate to users what actions are possible in each situation. Signifiers are not the action that a user can take, but the indication that there is an action they \textit{can} take (Norman, 2013).

\textsuperscript{2} Feedback: An immediate response to an action taken by a user. Feedback informs the player that their action has been registered by the system they are interacting with, and often communicates the results of said action (Norman, 2013).
Including accessible features in games has a large variety of benefits to games and players. From a broad perspective accessible games are inherently more inclusive and for many disabled gamers; playing video games can also provide a coping mechanism and social refuge (Wästerfors & Hansson, 2017). Making games accessible also helps them to reach the widest possible audience and therefore increase a game’s player base.

Some examples of games with a large range of accessible features can be seen in recent AAA titles such as *The Last of Us Part II* (Sony Interactive Entertainment, 2020) and *Spider-Man* (Sony Interactive Entertainment, 2018). *The Last of Us Part II* (Sony Interactive Entertainment, 2020) incorporated accessible design from the beginning of its development. The final release includes over 60 accessible settings with a focus on motor, hearing and visual features. The game includes feature presets that automatically configure a large number of relevant features to each focus category (“The Last of Us Part II - Accessibility”, n.d.). *Spider-Man* (Sony Interactive Entertainment, 2018) is another great example of an accessible
game, and allows players to autocomplete quick time events\(^3\) (QTE’s), bypass puzzles, and change repeated button presses to button holds.

The team’s accessible design was aided by researching and consulting a range of accessibility guidelines including Game Accessibility Guidelines (Game accessibility guidelines), which breaks down features by the implementation difficulty they pose to developers. Other resources included Accessible.Games (Accessible.Games 2019), Microsoft’s accessible gaming resources (stevewhims et al.), and the International Game Developers Association game accessibility special interest group (IGDA-GASIG) (The IGDA Game Accessibility Special Interest Group 2020).

2.2. Game Inspirations

No game is developed in a vacuum, and Factory Reset is no exception. The team drew inspiration for its mechanics and design from a range of other titles. As the team was developing a first-person 3D puzzle game with physics elements, they were inspired by Portal 2’s (Valve, 2011) environments, traversal mechanics, and physical interactions; Half-Life 2’s (Valve, 2004) physics-based Gravity Gun weapon; multiple movement mechanics from The Legend of Zelda: Twilight Princess (Nintendo, 2006); and the first person orientation shifting in Manifold Garden (William Chyr Studio, 2019).

2.2.1. Portal 2

Valve’s Portal 2 (Valve, 2011), as well as its predecessor Portal (Valve, 2007), were some of the game’s heaviest inspirations. Both games are puzzle-centric first-person shooters where the player is trapped in a high-tech facility and forced to run through a series of test chambers using a gun that creates a pair of portals between surfaces. Figure 3 shows an example of one of these test chamber environments, as well as the portal gun. These portals are the centerpiece of the game’s physics-based puzzles and movement mechanics. The games, particularly the sequel, also feature sections where the player uses the same traversal mechanics to explore the facility environment outside of the test chambers.

\(^3\) A quick time event (QTE) is a gameplay action that players have to perform in a limited timeframe, typically after reacting to an onscreen prompt.
The central concept of manipulating and traversing the world using magnetic powers shares a good deal of DNA with the *Portal* (Valve, 2007) games. Both have players moving between surfaces and thinking about navigation in unconventional, fantastical ways. Both give players tools for manipulating and carrying puzzle objects and leveraging the physics of the game world to overcome challenges. Both even share more superficial similarities like having mechanics that operate around a pair of color-coded actions.

![Figure 3: A screenshot of a test chamber in *Portal 2* is shown on the left (Valve, 2011). Image (Jongeneel, 2013). A screenshot of *Factory Reset* is shown on the right for comparison.](image)

### 2.2.2. *Half-Life 2*

Another Valve title, *Half-Life 2* (Valve, 2004), also served as a significant source of inspiration for *Factory Reset*’s magnetic hands mechanic. It is a first-person shooter with science fiction and horror elements where the player helps fight a rebellion against alien races that have taken over the Earth. Of specific note here is one of the most iconic tools in the game: the Gravity Gun, pictured in Figure 4. The core function of this device is to pick up and launch all manner of props and objects. This is often used as an improvised weapon, flinging any sort of loose object the player can find at targets with destructive force. The two main actions afforded by this tool - pulling objects to the player and launching them away - served as a direct inspiration for the magnetic hands ability in *Factory Reset*, which similarly lets players attract and repel polarized objects in the environment.
2.2.3. The Legend of Zelda: Twilight Princess

Nintendo’s *The Legend of Zelda* series of action-adventure games has a long history of creating explorable puzzle spaces centered around specific traversal or interaction mechanics, referred to as dungeons. *The Legend of Zelda: Twilight Princess* (Nintendo, 2006) in particular features two mechanics from its dungeons that strongly inspired the magnetic feet ability in *Factory Reset*. The first of these is the Clawshot item, shown in Figure 5, which is a variation of a recurring tool in the series where the player can aim at a designated target in range and fire a grappling hook to it in a straight line. If it hits a solid, grippable surface the player will be pulled directly to the point of impact, but otherwise it will try to pull the struck object towards the player. This is used primarily as a traversal ability, but it can also interact with certain switches and even has roles in combat.
The Legend of Zelda: Twilight Princess (Nintendo, 2006) also features a magnetism mechanic in the Goron Mines dungeon, shown in Figure 6. Magnetic surfaces are found throughout this area, and when standing in range of them and equipping the Iron Boots item the player character re-orientates and gets pulled to the surface, walking along regardless of the actual direction of gravity (Nintendo, 2006). The clawshot and Goron Mines magnetism jointly acted as a core inspiration for the magnetic feet ability featured in Factory Reset; the magnetic feet mechanic essentially combines them by allowing the player to aim at a designated target, pull yourself to it, and then walk on it as if it were the floor.
Figure 6: The player walking on a magnetic ceiling in the Goron Mines dungeon from *The Legend of Zelda: Twilight Princess* (Nintendo, 2006). Image (Zelda Dungeon, n.d.).

2.2.4. *Manifold Garden*

The indie first person puzzle-exploration title *Manifold Garden*, by William Chyr Studio, has players navigate surreal, abstract architecture spaces that loop indefinitely or otherwise defy conventional physics (William Chyr Studio, 2019). An example of such a space is shown in Figure 7. The core game mechanic is the ability to walk up to any wall and interact with it to shift the entire world’s gravity so that it functionally becomes the floor. Players have to combine this orientation shifting, various puzzle objects, and the looping nature of the world to figure out how to overcome challenges. The game has no concept of damage or death, so a common traversal technique is to deliberately step off a ledge and exploit the looping world to land somewhere else on the level (even one that seemed “above” the starting location from the starting orientation). The re-orientation effect is made all the more dramatic by the game being played from a first-person view, and this was thus a strong inspiration for the team’s own version of first-person re-orientation in *Factory Reset*. 
2.3. Art Inspirations

A diverse and multifaceted spectrum of media and art inspired the overall aesthetic of Factory Reset. The aesthetic mission of Factory Reset was to create a lived-in cyberpunk world, set in an alternate-history Soviet Union. The overall environment was to seem believable and realistic enough to make the world feel like it could actually exist. Throughout the conceptualization process of Factory Reset’s vision, the team created a mood board that facilitated various inspirations that ended up informing the look of the game, particularly drawing from popular 1980 gilded age cinema, video games, and Soviet architecture. Figure 8 shows this mood board.
2.3.1. Pseudo-Realism

The team used the term “pseudo-realism” to describe something close to reality, but not quite there: photorealism with a touch of the absurd and unbelievable. It is a juxtaposition that rides the edge of convincing and the uncanny valley, as the props, characters, and levels are modeled and textured to look like a believable world, yet still dissonant. One of the biggest areas of creating realistic environments is creating believable environments. This is done by putting forethought at answering questions throughout the artistic process of:

1. **What** exists in the world?
2. **Where** do they exist in the world?
3. **Why** do they exist in the world?
4. **How** do they relate to the overall story of the world?

By answering these questions throughout the assets creation and environmental building pipeline discussed in detail in Section 5, the artist is able to achieve a much more powerful and believable world, while developing an authentically convincing narrative. Pseudo-Realism isn’t just about reaching an artistic parallel to the real world, but gapping the bridge between the game world and the real world.
2.3.2. Cyberpunk Genre

Cyberpunk is best known as a subgenre of science fiction, depicting a hyper-capitalist dystopia. It tends to focus on a “combination of lowlife lifestyles and high tech future societies” (Gibson, 2003). It’s a juxtaposition of incredibly advanced technological and scientific achievements, usually in the manner of artificial intelligence, androids, flying cars, and ecumenopolisi⁴, to a manner of radical classist divide within social order. Cyberpunk as a genre is rooted within the New Wave of science fiction that took place throughout the 1960’s and 1970’s; however, made a massive resurgence during the 1980’s Cinematic Gilded Age (Redmond, 2004). Filmmakers used this new age of cyberpunk and science fiction of the time as a way to preconceive the 1980’s era of physical conservatism and the on-slaught of rising wealth inequality, Reaganism, and corporate greed. These opposing themes find themself being represented visually through stark contrasts, generally through the use of lighting and colors, for instance, menacing dystopian cityscapes juxtapositioned with colorful, vibrant, neon lights.

Figure 9: Concept art from Cyberpunk 2077 (CD Projekt, 2020) showcasing a cyberpunk environment. Image (CD Projekt RED, 2020)

⁴ From the Greek word “οἰκουμένη”, or latinized: oecumene, meaning "world" and the Greek word “πόλις” , or latinized: polis "city", creating the word "a world city", which describes the hypothetical concept of a planet wide city.
2.3.2.1. Ridley Scott’s 1982 *Blade Runner*

![Image of Blade Runner and Factory Reset](image.png)

Figure 10: Still-frame from Ridley Scott’s 1982 *Blade Runner* (top) showcasing the sprawling city and flying cars, as inspiration for the city level in *Factory Reset* (bottom). Image top (Mead, n.d.)

Considered one of the first cinematic, non-comic, forms of media that explored cyberpunk themes, Ridley Scott’s *Blade Runner* (1982), was a massive inspiration for *Factory Reset*. *Blade Runner’s* visual themes of sprawling, seemingly unending city-scapes, inspired many of the visual and artistic themes of *Factory Reset*. One of the most obvious parallels drawn from *Blade Runner* would be found within *Factory Reset’s* second level, in which the player character jumps between flying vehicles throughout a dense urban landscape covered in a dense smog. Figure 10 illustrates this comparison.
2.3.2.2. Paul Verhoeven’s *RoboCop* (1987)

![Figure 11](image)

Figure 11: Paul Verhoeven’s *RoboCop* (1987) standing outside of a police cruiser cast, demonstrating a strong contrast between red and blue lighting. Image (Bryan, 2019).

Another cinematic piece of cyberpunk science fiction that drew inspiration was Paul Verhoeven’s *RoboCop* (1987). The film has been analyzed for having themes of humanity, personal identity, and as an overall rebuke of Ronald Reagan’s then era policies (Fisher 2017). *RoboCop*’s inspiration was mainly drawn from two specific areas of thought: color and artificial-autonomy (Ramsay-Jones, 2020). *RoboCop* is one of the few films of the 1980’s Cinematic Gilded Age to establish a bright neon color palette of purples, violets, and pinks. Notwithstanding, it included the powerful contrast between red and blue, which film scholars such as James Adams describe as the physical divide between authoritarianism and personal freedom, which so happened to be an analogy to American police brutality and the 1980 Miami / 1992 Los Angeles Rodney King riots (2016). The idea of having contrasted colors, as well as adopting the cyberpunk’s full medium of visual color storytelling would eventually find its way into *Factory Reset*. These concepts were integrated by the team, primarily within *Factory Reset*’s lighting and color palette.
2.3.3. The Legend of Zelda: Breath of the Wild

Although the team strayed away from it once the cyberpunk direction was established, Nintendo’s *The Legend of Zelda: Breath of the Wild* (Nintendo, 2017), was an early inspiration for the look and level design of *Factory Reset. Breath of the Wild* (Nintendo, 2017) is a vast free-form third-person open world adventure which includes many self-contained puzzle challenges in the form of Shrines scattered around the world. The team specifically looked to the luminous and stylized ancient technology featured in the game, as shown in Figure 12, as well as the highly modular and artificial puzzle box-like level design of the Shrines. The glowing elements of this technology very prominently feature the contrasting color pair of blue and orange, which beyond their use for striking decorative details also indicate the state of objects: inactive devices have a dominantly orange hue, which switches to blue upon activation.

![Figure 12: An ancient glowing robot in *The Legend of Zelda: Breath of the Wild* (Nintendo, 2017). Image (52katie, 2017).](image)

The interiors of the Shrines are large spaces filled by what essentially amount to obstacle courses that test the player’s puzzle solving ability and mastery of the game’s mechanics. They all share a relatively small but highly modular pool of assets such as blocks and ramps for shaping levels and puzzle elements like movable balls, breakable walls, powerful fans, and switches. This was considered an appealing direction to emulate, as it would minimize the need to create a large amount of complicated art assets while still allowing the team to explore the desired puzzle and traversal mechanics.
2.3.4. **Furi**

*Furi* by The Game Bakers (The Game Bakers, 2016) is a boss-rush style game where the player fights several opponents while unraveling an interesting story. Its powerful emissives and futuristic look were something the team wanted to emulate from very early in the concepting process. The team was particularly drawn to the types of emissives\(^5\) on the player character seen in Figure 14, because they were stylish and made the game elements with emissives stand out from the others. *Factory Reset* ultimately featured emissives on the player to highlight when the magnetic hands were being used.

---

\(^5\) Emmisives are special textures which allow the rendering or game engines to designate where light is to be found within a 3D Model.
2.3.5. *Sayonara Wild Hearts*

Simogo’s *Sayonara Wild Hearts* (Annapurna Interactive, 2019) bills itself as a “pop-album video game” (Flesser, 2020). It is a varied experience comparable to a sequence of playable music videos that employs a splashy, colorful, and stylized look throughout its environments, characters, and effects. While its general aesthetic direction diverges from the team’s goals with the art of *Factory Reset*, the third area of the game features a city of looming buildings with softly glowing windows shrouded in a colorful haze. The team used this environment as an inspiration for the haze-filled city featured in the second level of the game, as fog was a necessary element to mask the endlessly looping cityscape. Figure 15 compares the look of the two city environments.

![Image](image1.png)

*Figure 15:* The city environment featured in the third major section of *Sayonara Wild Hearts* (top) and the city environment featured in *Factory Reset* (bottom). Image top (Flesser, 2020)
2.3.6. *Control*

*Control* (505 Games, 2020), is a third person shooter game set inside a massive, reality-bending government complex. The interior of the building has a huge emphasis on oppressive, blocky, brutalist architecture, which was a good atmospheric inspiration for the buildings and interior in *Factory Reset*. This type of architectural design ultimately manifested most heavily in the third level, where the player enters the large and ominous entrance lobby to the factory they were created in.

![Figure 16: Some examples of the brutalist architecture featured in Control (top) and the stark, brutalism-inspired lobby of the third level in Factory Reset (bottom). Image top left (Ewan, 2019) Image top right (Ewan, 2019)](image-url)
2.3.7. Soviet Union Fourth Age of Architecture

In the book *Cosmic Communist Constructions Photographed*, Frédéric Chaubin documents ninety buildings in the fourteen former Soviet Republics of the USSR. It describes a form of architecture which became an unexpected rebirth of imagination, building on the 1920’s to 30’s artistic trend of constructivist architecture: a burgeoning that took place throughout the 1970’s, until the 1990s. Contrary to the 1920s and 30s traditional constructivist architecture, no ‘school’ or main trend emerges within this new era of Soviet architecture. This era of Soviet architecture is called the “fourth age” of Soviet architecture. The buildings represent a chaotic impulse of an envisage for the future of socialism and communist societies (Chaubin, 2011).
The shapes of these buildings are anarchic, yet still follow some resemblance of order and lawness. Contrary to Brutalism or Brutalist architecture, another similar early twentieth century modernist movement, this fourth age of Soviet Architecture followed more of a humanist direction (Đorđe, 2017). Many of these buildings were designed around the human experience. Modernist Soviet city planning and architecture can still be seen to this day revolving around the idea of designing for people. Adverse to Brutalist architecture, which was meant to feel imposing and authoritarian, this era of Soviet architectural design existed to improve “the people’s continuation of their lifestyles”. Chaubin in his book describes this form of Soviet architecture as “Sovietpunk”, or in other words the Soviet version of the 1980s rebellious, or “punk” culture.

2.4. COVID-19 Pandemic

The Factory Reset team primarily developed this game during the COVID-19 global pandemic. COVID-19 was a global pandemic that came to the United States in March, 2020 and was ongoing at the end of this project. The pandemic had a significant impact on all aspects of the team’s development process. The pandemic also had a significant impact on the design and implementation of the team’s playtesting protocol. The team completed all work on this project remotely, including the playtesting.
3. Design

This section documents the design and rationale of the technical aspects of Factory Reset. It covers higher level concepts and considerations that shaped the team’s approach to development (Sections 3.1 - 3.3), the final forms taken by core game mechanics (Section 3.4), the nature of the game’s first-person perspective (Section 3.5), key decisions made by the team in establishing the scope of the project (Section 3.6), and the ultimate structure of the final game (Section 3.7).

3.1. Design Goals

The team’s design goals with Factory Reset were to create engaging mechanics, accessible gameplay, and a polished and well-crafted experience. The team wanted to make a game that was enjoyable and straightforward to pick up while being accessible and inclusive to a wide audience. By wide audience, the team means that they want anyone to be able to play the game regardless of their ability or equipment. The team also wanted it to have polished art and mechanics that showcased the team’s abilities and surpassed the expectations of a student game.

3.2. Experience Goals

Experience goals define the way the team wants players to feel while consuming an experience. The team’s biggest experience goals with Factory Reset were to instill a sense of visual awe, intrigue about the world, and satisfaction upon completing puzzles. The team sought to achieve awe through art setpieces like riding flying cars in the city level and encountering OLOF-14 in the imposing lobby of the third level, as well as through the striking visual hook of re-orienting onto walls and ceilings in first person. The team used dialogue and environment design to build up the world’s intrigue. To reach the final experience goal of offering a satisfying experience, the team used clear progression through the game’s levels and puzzles, reinforced by the discovery of the aforementioned set piece moments.
3.3. Audience

The team defined the audience for Factory Reset in terms of the types of games that inspired it, the visual design, the atmosphere the team created, and the included features. The team designed Factory Reset to appeal to players who are interested in puzzle games containing an exploration component, especially ones employing unusual and intriguing traversal mechanics like Portal 2 (Valve, 2011) and Manifold Garden (William Chyr Studio, 2019). Players would also be drawn to the cyberpunk art style and bold lighting design. The game’s accessibility features also make it more approachable and enticing for players who commonly make use of them to improve and customize their gameplay experience.

3.4. Accessibility

The team chose the accessible features implemented in Factory Reset by consulting accessible game development resources and then developing their own prioritized accessible feature list. The key areas on this list were the in-game menu, support for controller remapping, subtitles, audio, game saving, epilepsy mitigation, and animations. The biggest priorities over the course of development were support for external controllers, fully remappable controls, and subtitles. These features were chosen because they are inclusive of many players and allow potential players to create a fully customized play experience. The team’s goals in choosing these options were to focus on features that would reach a wide audience of players and be within the scope of the ten-month project.

For the gameplay design of Factory Reset, the team chose single action gameplay and elected to have no time constraints for player progression through each level. Single action gameplay means that players only need to use one game mechanic at a time. This can be seen in action in the game’s first level, where players need to walk up the elevator to escape. In this context, the hands and the feet never need to be used at the same time to escape the level. These decisions were some of the earliest in terms of gameplay design.

An additional gameplay design choice was to have the hands require holding the respective mouse key by default. The team made this choice based on playtesting feedback which is detailed in Chapter 7. To mitigate the inherent inaccessibility of button holding, the team added a toggleable button hold feature for the magnetic hands mechanic.
Motion sickness was also a concern due to the nature of reorientation when using the magnetic feet mechanic. Through playtesting feedback the team saw that a small but significant number of testers were experiencing motion sickness from using the magnetic feet. The team’s solution was to implement a feature to disable the magnetic feet animation as players are pulled to a surface. Detailed analysis of this feedback is in Section 7.2.

The team implemented a menu that enables players to access accessible features. This menu incorporates several accessible design features to make it more usable. The team designed the menu to be navigable with all common control types: keyboard, mouse, and external controller. Scrolling through the menu produces audio feedback, with a unique sound at the top of each menu to indicate to players their cursor’s current location. A final visual feature is a togglable high-contrast UI option that makes the menu background opaque to aid its readability. A further discussion of the menu’s implementation is in Section 4.3.2.

Keeping the visual aspects of the gameplay design accessible was also a priority. The game’s artist designed the polarity colors used throughout the game to be distinguishable with all forms of single cone color blindness. More details on this can be found in Sections 2.1 and 5.3.2. The team’s artist also designed key assets to have physically distinguishing elements so that players do not have to rely solely on color to distinguish between polarized game objects. The polarity pads and polarity boxes can be seen in Figure 19 and Figure 20 respectively.

Figure 19: The positive polarity pad (left) and the negative polarity pad (right).
The final game includes 14 different accessible features and numerous design choices made with the intent of supporting accessibility. These are mostly motor, visual and cognitive features:

- External controller support
- Fully remappable controls
- Toggleable button holds
- Adjustable look sensitivity
- Toggleable look inversion
- Subtitle system
  - Size
  - Opacity
  - Speed
- High contrast UI mode
- Adjustable FOV
- Reduced camera motion
- Adjustable volume by channel
- Full screen mode
- Adjustable screen resolution

There were a number of accessible features that were discarded due to scope and technical limitations.
3.5. Perspective

*Factory Reset* was designed with a first person perspective in mind. This allowed for intuitive aiming and maximized the dramatic effect of the re-orientation aspect of the magnetic feet mechanic. It also simplified the camera control implementation, as there was no need for special logic to keep objects and walls from interfering with the camera’s movement or positioning. The main downside of this approach proved to be the disorientation it caused players, as pulling between surfaces swung the camera around. This disorientation was compounded by a lack of clear directionality signifiers in the environment, especially in earlier builds.

3.6. Mechanics

The mechanics of *Factory Reset* are themed around magnetism. The player controls a robot, FyED-OR, whose magnetic abilities allow them to interact with and traverse the levels. They use their magnetic feet to pull themselves onto different polarized surfaces, reorienting their view accordingly and even defy gravity by walking along them. They can also use their magnetic hands to push and pull certain objects around the environment. The portrayal of magnetism in the game is simplified and stylized from real life to make these mechanics more easily understood and designed around; of particular note is that magnetic interactions only take place between the player and the objects they aim at, rather than directly between objects in the level.

3.6.1. Basic Player Controls

The player’s basic controls in *Factory Reset* follow the conventions of first-person games. The player moves with either four keys on a keyboard, by default W, A, S, and D, or with a gamepad’s analog stick. They have no ability to change their ground speed. They can also look around the environment by moving the mouse or the opposite gamepad stick. Notably, the player cannot jump, as the team instead opted to focus the level design on requiring the player to solely use the magnetic feet ability for spatial traversal.

Camera control sensitivity proved to be an unexpectedly challenging aspect of the project to implement satisfactorily. The initial implementation of the camera control system used the
default Unity function \texttt{Input.GetAxis()}, but was converted to use \textit{Unity}'s new input system before Alphafest to accommodate support for external controllers. This change revealed mouse sensitivity issues that are device dependent, and more specifically dependent on screen resolution. Several attempts were made to solve these issues including normalizing the raw mouse input vector, and scaling the input by time. Despite changes made to mitigate this issue, look sensitivity remained a high cause of frustration during playtesting feedback.

3.6.2. Magnetic Feet

If the player aims their reticle at a polarized surface, they can press a dedicated button to activate their magnetic feet and pull themselves to the surface, re-orienting to stand on it as if it were the ground. While attached, the player can freely walk along the surface but is prevented from accidentally stepping off. To disengage the magnetic feet, the player presses their button again while not aiming at a polarized surface. The player can also pull themselves directly from one polarized surface directly to another. Examples of these movements are shown in Figure 21.

The surfaces used for this would be specially marked with colors and symbols to indicate their polarized or metallic nature.

As noted in Sections 2.2.3 and 2.2.4, this mechanic was jointly inspired by the Clawshot item and Goron Mines dungeon from \textit{The Legend of Zelda: Twilight Princess} (Nintendo, 2006) and the first person re-orientation mechanic in \textit{Manifold Garden} (William Chyr Studio, 2019). This ability to re-orient onto surfaces serves as \textit{Factory Reset}'s gameplay feature with the strongest visual hook, as twisting one’s perspective to see the world in a totally new direction is inherently attention grabbing.
3.6.3. Magnetic Hands

The player can, at any time, set their hands to either a positive or negative polarity using the proper button. While the hands have a polarity set, aiming at any interactable polarized object, such as a polarity box, will cause it to react to the polarity based on its own and its proximity - if the polarity matches it will be pushed away, if it is opposite it will be attracted, and the strength of both forces diminishes with distance. These objects would also be visually marked in a similar manner to the polarized surfaces, with color-coded emissives indicating their polarity and supplemented by plus and minus symbols when the objects have different polarity variants. The various polarity states of the ability, as indicated on both the player character’s hands and the reticle, are shown in Figure 21. As described in Section 2.2.2, the concept of this ability was primarily inspired by the Gravity Gun item in *Half-Life 2* (Valve, 2004).
Figure 22. Magnetic hands in Negative (top left), Positive (top right), and Neutral (bottom center) polarity modes. The polarities are supplemented with matching symbols and colored regions on the reticle in the center of the screen.

There are two modes for how to control this ability: with the default “hold” mode, the player must hold down the button for the appropriate polarity and it will remain active until they release it. With the alternative “toggle” mode they press one and the hands will remain in that polarity state indefinitely until the polarity is switched or the button is pressed again to deactivate them. This latter system was the default up until the last Beta builds because it is more physically accessible since some forms of motor disabilities make it difficult to hold buttons down. The team switched the default away from this because players would consistently accidentally leave the magnetic hands active, and they suspect that players’ tendency to do so was part of what led to confusion between the functions and purposes of the feet and hand abilities.

3.7. User Interface

A game’s user interface is very important, as it is through this that players are provided information about the current game state and what options are available to them at any given moment. This is thus also an area where accessibility considerations are particularly critical, as if players face barriers in interpreting a game’s interface they will have a much more difficult time enjoying or even understanding how to play it in the first place.
One very critical element of the user interface for both accessibility and general gameplay understanding is the heads-up display (HUD). The team had a significant amount of information they considered important to convey through this:

- Whether the player was attached to the surface with their magnetic feet.
- Whether the player’s magnetic hands were active and what polarity they were set to.
- Whether the player could interact with an object they were looking at, such as a switch.
- Whether the player was blocked from making an action, such as by being out of range of the object they were targeting.
- What the player’s current orientation was relative to the world’s true gravity direction.

Our final HUD incorporated these features as shown in Figure 23.

**Figure 23.** A visual breakdown of the final HUD.
3.8. Game Structure

*Factory Reset* follows a linear structure with three distinct levels, which are designed to showcase the core mechanics, visual design, audio work, and narrative. It is intended as the intro and demo of a hypothetical expanded game delving deeper into the gameplay, world, and story.

The first level opens with the player character powering on in a trash compactor, which they must escape from using their magnetic feet. They then realize they are in the base of a tall storage facility and must find a way out by using both their hand and feet abilities to ascend the building’s central shaft. To do this, they must locate a series of three buttons along the way to restore power to the elevator platform and open the path upwards. The level ends with the player finding a newly-arrived flying vehicle and hitching a ride on top of it. Mechanically, this level serves as an introduction to the player’s abilities in a confined space.

![Image of the first level](image1)

**Figure 24:** The trash compactor (top left), elevator shaft (top right), and hangar (bottom) areas of the first level in *Factory Reset.*

The second level begins with the player riding on the vehicle from level one while it flies along a sky highway crowded with other vehicles. The player’s character recognizes the large vehicle at the end of the vehicle pack, and so the player must try to reach it by pulling themselves
to polarized surfaces on different cars and even dropping from one to another. This sequence acts as a transitional space, narratively linking the first and third levels, while also increasing the perceived scope of the world and creating a visually striking moment. Its purpose from a mechanical standpoint is to highlight the traversal aspect of the magnetic feet ability.

![Figure 25: The second level of Factory Reset, where the player travels across a group of flying cars in a city.](image)

The third and final level has the player arrive at an entrance to the facility they have been trying to reach. Inside, they cross through a cavernous and ominous lobby, have a run-in with OLOF-14, the intimidating robot looming over the lobby desk, and find the elevator down is out of power. They must venture into the room to the side, which is in disarray from construction work, and locate three power cells to attach to a generator to activate the elevator. This level presents the most involved puzzle in the game, as well as new mechanics related to the magnetic hands: scaffold devices they can push and pull along rail, and the aforementioned polarized power cells they can attract and repel. It mechanically serves as a showcase of the magnetic hands ability and how it can interact with the magnetic feet mechanic.

The demo ends on a cliffhanger note, with the player descending down into the facility and the elevator door opening with a blinding white light.
3.9. Level Design

Level design for Factory Reset was a significant learning process for the team. When the project was initially conceptualized and proposed, the team was interested in exploring highly modular and artificial design like that seen in the puzzle chambers in the Portal (Valve, 2007) series and the Shrines in Nintendo’s The Legend of Zelda: Breath of the Wild (Nintendo, 2017). The spaces would have been primarily focused around the gameplay and puzzles rather than being realistic, plausible environments. After the team’s artist joined the team, the team’s goals shifted to put much more emphasis on showcasing his artistic abilities through believable,
detailed environments. This made the team’s initial idea for how to approach level design a less-ideal fit, so the team instead shifted to focus on creating more plausible places with both a gameplay and narrative purpose.

3.9.1. Level Design Goals

Early on the team established that they wanted a mix of environmentally-focused and puzzle-focused sections to strike a balance between a design approach well-suited to exploring both the mechanics and the world. They also wanted to introduce the gameplay mechanics in an approachable, intuitive manner and use them in increasingly complicated ways over time.

3.9.2. Level Design Process

The team’s general level design process was to develop and sketch a concept collectively in a group meeting, create a greybox version of the level in *Maya* or *Unity* based off of it, populate the space with gameplay elements in *Unity*, then revise the level with final art assets, lighting, sound, and environmental settings. This process ensured that the team was in collective agreement and was able to divide the work effectively.

While this approach proved effective, it was challenging for the team to initially develop a successful strategy for the conceptualization part of the workflow. The team’s initial approach to brainstorming level concepts was to work individually, sketching and describing different potential gameplay spaces and puzzles like those seen in Figures 27 and 28. This was intended to generate a wide range of possible ideas in order to help identify a gameplay direction, as early in development the game lacked one besides the desired mechanics and the three-level structure. In this approach, the team’s artist focused on visually conceiving possible environments while the remainder of the team focused on the design work.
Figure 27. Early concept images of environment ideas.
This arrangement proved to be difficult to work with, as there was limited cohesion between the team’s ideas and nothing generated from it inspired the team with a strong direction. There also proved to be an imbalance in the work on the environment concepting side which was exacerbated by the lack of clear direction, as the team’s artist found it difficult to juggle concepting spaces alongside early asset creation.

To address these difficulties, the team shifted their approach to instead holding a collective meeting to brainstorm and plan the level together. This focused, team-wide discussion format was far more fruitful and efficient, and was thus used for all future level planning.

3.9.3. Level 1: The Storage Facility

The team wanted the first level to serve as an effective introduction to the mechanics, the setting, and the narrative of Factory Reset through the gameplay and levels. As illustrated in Figure 29, the level was conceptualized as a storage facility composed of three sections: an opening room where the player wakes up in a trash compactor and must use their magnetic feet to escape and clear polarized obstacles out of the way with their magnetic hands, a vertical triangular elevator shaft lined with platforms for storage that the player ascends by re-powering elevators and using both their main abilities, and a final hangar room where they escape the level by riding on a departing cargo vehicle. This met all of the team’s goals: it gave a narrative context for the player’s actions, it offered a mix of gameplay and environment-focused sections,
and it could demonstrate both the mechanics and aesthetic direction. The elevator shaft was a particularly important set piece, as re-orienting to walk along the central column and seeing the vertical space become horizontal would be a very striking visual. The final look of the first level is shown in Figure 29.

![Figure 29](image)

**Figure 29.** The first complete sketch of the planned layout for the first level of *Factory Reset* (left) and an early greybox version of the level (right). The player would start in the lowermost area, ascend a triangular elevator shaft using their magnetic abilities, and exit the level by hitching a ride on a departing vehicle in the hangar area in the upper-left.

3.9.4. Level 2: The Sky Highway

The first level had concluded in a very open-ended manner, with the player hitching a ride on a departing vehicle. Narratively, the team wanted the player character’s end goal to be returning to the factory they were built in. However, the team felt that jumping from leaving the interior of the first level’s facility directly arriving inside their goal one would result in the team missing significant opportunities for showcasing the game’s world. To remedy this, the team came up with the idea of having the second level represent the player’s trip through the city instead of immediately being the introduction to the goal facility. Specifically, the player would use their magnetic feet power to traverse a highway of flying futuristic cars as they collectively travelled through the city. This would provide a big set piece moment to convey more of the setting and scale of the world while presenting the game’s main movement mechanic in a far more open context.
Translating this design into a playable level for the game took some special consideration. In order to convincingly sell the idea that the cars were flying, it was necessary to have them both idly shift relative to one another as well as continuously appear to move forward relative to the buildings around them. The details on how this was achieved are covered in Section 4.5. The final look of the second level is shown in Figure 30.
3.9.5. Level 3: The Factory Lobby

For the third and final level, the team wanted to bring the narrative to a compelling but open-ended stopping point: the player character arriving at their home factory’s lobby and descending into the depths of the facility through an elevator. The team also wanted to use this level to showcase the game’s mechanics, especially the magnetic hands, in a more involved puzzle room. To marry these goals, the team came up with the concept that the lobby was partly under construction and the elevator door had lost power. To progress, the player would need to go into a side room to restore the power. This inclusion of both an environment-centric space and a puzzle-centric space also meets the team’s level design goals. The final look of the third level is shown in Figure 31.

![Figure 31](image)

**Figure 31.** The initial concept sketch for the third level, showing the basic level layout and some initial sketches of the lobby room.

The team also further reinforced and expanded the narrative while building on the experience goal of awe by adding a massive, ominous robot known as OLOF-14 that watches
over the lobby and speaks to the player throughout the level. It swivels around to watch them upon arrival, and speaks in a deep, menacing computer-like Russian voice.

![Figure 32](image.png)

**Figure 32.** A render of OLOF-14 (left) and an early look at them in the greybox of the third level (right).

For the actual gameplay of the third level, the team designed three new puzzle elements: polarized power cells, a power supply, and movable scaffold objects, illustrated in Figure 32. These were intended to further develop the magnetic hands ability, as they were all interacted with using it. The team designed this section to have a nonlinear approach where the player must figure out how to use their powers and the movable scaffolds to reach and collect three power cells and attach them to the power supply, illustrated in Figure 33. This allowed the team to efficiently fit in three different puzzles that explored the interactions of the mechanics. The team also added an introductory scaffold that the player must discover how to interact with to enter the puzzle area and progress, which is visible on the left side of the screenshot in Figure 34.
Figure 33. Initial sketches and descriptions of the power cell (upper-left), power supply (lower-left), and movable scaffold (right) puzzle objects.

Figure 34. The three power cell puzzles the team initially designed for the third level. One must be reached by knocking the objects supporting it down (upper-left), one must be reached by moving a scaffold under a polarity pad (lower-left), and one must be reached by moving one scaffold out of the way of another and using a polarity pad attached to it (right).
One additional important decision related to the third level was establishing its ordering in the game. The team’s advisors pointed out the possibility that the city level, with its dramatic spectacle, might be a striking note to end on. This would also make it more likely that players reached the lobby level, helping to ensure that they encountered the game’s most interesting puzzle content if they stopped playing early. The team deliberated on this but decided to ultimately preserve the original order with the lobby level being the end point. This allowed the narrative to play out more completely and matched the team’s original vision for the game more closely; finishing on the sky highway would have left a huge amount of ambiguity as to where the game could go next compared to the more focused cliffhanger of descending into the goal facility. The team also felt that ending on an elevator ride felt more tense and less jarringly abrupt than just reaching a giant vehicle. From a gameplay perspective, this ordering also preserved the design goal of increasing the complexity of the use of mechanics over time.

3.10. Scope

Early in development, the team set out to establish the planned scope for the full project. As a senior project developed over the course of a full school year, the team had to strike a balance between making something that showcased the team’s abilities and something that was achievable in the timeframe. The team sat down to compile a dream wish list of features they would want in the game if they had no restrictions on time (Section 3.8.1), and then pared this list down to a second one which they felt was the simplest version of the project that met their needs (Section 3.8.2). The team carried forwards this streamlined design as the basis for the final game.

3.10.1. Dream Design Scope

In order to hone in on the final set of features and mechanics for Factory Reset, the team decided to first define the upper extreme of what they would want to include in the game if they had no restrictions and unlimited time. This idealized vision of the game would feature:

- A wide range of mechanics, with the ability to push and pull from surfaces with magnetic feet and attract and repel objects with magnetic hands at the core of the experience.
  Supplementary mechanics could include switches and pressure plates that manipulate the world when activated, disruptions fields that prevent the player from using certain
abilities in specially-defined regions, and magnetic prop objects that are attracted to each other.

- A variety of enemies that players must employ the main game mechanics to defeat. These would have puzzle-like designs where the player must dismantle key parts of them using their magnetic hands, and they would even have to climb onto large foes using their magnetic feet.
- A suite of levels that fully and iteratively explore each mechanic.
- A robust assortment of accessibility features improving motor, visual, auditory, and cognitive access to the game. Beyond the feature list described in Section 3.5, the team also wanted to have support for changing the gameplay speed, implementing positional visual cues for sound effects, and supporting multiple languages besides English.
- A richly developed Sovietpunk world, with gritty brutalist industrial architecture, an emphasis on glowing neon visual elements, and ample environmental storytelling and worldbuilding through details like posters.
- A heavy emphasis on ambient sound, with music being primarily splashed in to punctuate moments with dramatic effect. There would also be a robust set of sounds for player and object actions and interactions.
- A wealth of splashy and stylish visual effects like particles.
- A narrative following a world-weary robot as they descend through the many floors of the facility they were created in. Along the way they must overcome challenges like malfunctioning security robots and obstacles resulting from the facility falling into disrepair. The upper floors would be ruined and abandoned but deeper areas would be still intact and operational, creating a strong sense of visual progression.

3.10.2. Streamlined Design Scope

In refining their scope to be achievable with their resources and time, the biggest changes the team made was paring the length down to three levels and removing the concept of enemies altogether. This end result would be essentially a demo of the idealized game scope the team had envisioned. Since the team had only one dedicated artist, it was critical to reduce the number of unique art assets required for the game. Enemies were particularly important to remove as they would be very time-consuming to implement, the team had very limited 3D animation ability,
and no significant experience implementing computer-controlled enemies in 3D. They were also unnecessary, serving only as an additional application for the game’s core mechanics, and worked against the accessibility goal of putting no time pressure on the player. The last major scope reduction in the plan was to minimize extraneous mechanics, and so the team scrapped disruption fields and all magnetic interaction that was not player-driven.

The team further adjusted these scope goals in the actual implementation. Most notably, they broadened the setting to include two separate interior environments in different locations separated by an exterior sky highway level in the city. The team also removed the ability for the player to repel themselves from a surface with their magnetic feet, as this was difficult to convey in a physically believable manner in first person, as further detailed in Section 4.2.3.3. Additionally, the team scrapped pressure plates from the final game because they made little logical sense in the environments.
4. Gameplay Implementation

This section details the considerations and decisions made in the process of implementing the technical aspects of Factory Reset, as well as the challenges encountered and overcome along the way. It starts with an overview of key tools used.

4.1. Tools

In order to collaboratively implement the gameplay and levels of Factory Reset, the team made heavy use of the Unity game engine, Git, and GitHub.

4.1.1. Game Engine

The team developed Factory Reset using the Unity game engine. Unity is one of several popular high-performance game engines available for public use, which handle the internal architecture and rendering of the game in addition to providing an environment to develop it in. One of the biggest technical decisions at the start of Factory Reset's development process was to choose between using Unity and the Unreal Engine.

The technical side of the team had previously worked together on a substantial Unreal Engine project, and also had significant individual experience working in Unity from other projects. Unreal Engine has a better default lighting system and easy ways to create volumes that control the appearance of the game. It also comes pre-packaged with useful features like a fully-functional first person player controller; however, it suffers from high instability and frequent crashing, requires lengthy compile times when programming in C++ that impair productivity and make programming tedious, and in their previous project experience proved to have some aspects that the team found awkward to work with like the cinematics system. The availability of Unreal Engine documentation was often inconsistent when the team had previously worked in it. They were also concerned that they would have less direct control over the physics system in Unreal Engine, or at least a lack of familiarity with how to handle it, which may make it more challenging to implement the desired player and puzzle mechanics.

Unity, on the other hand, is an engine that the team was more deeply familiar with. The team knew how to effectively work within it and control its physics system. In its base form it tends to be more stable than Unreal Engine, and it has comparatively very fast compile times.
after changing scripts, so it would be much more fluid to work in. There is a wealth of robust documentation and useful forum discussions to use as resources for solving any Unity issue the team encountered. The engine also offers the High Definition Render Pipeline project type, which provides better support for Unreal-like features such as environmental volumes. The main drawbacks were that getting the project functional from scratch would take more implementation work and that the team was less familiar with how to effectively handle its lighting and emissives. Given these considerations, the team ultimately chose to develop the game in Unity.

4.1.2. Version Control

To enable the team to work on the project together asynchronously, the team used Git and GitHub. Using a version control system like this allowed the team to keep track of changes and collaborate on the project without having to manually combine different parts of the project that people were working on. Having a remote GitHub repository was essential for the project because the team was working from completely separate locations as a result of the COVID-19 pandemic. Each of the team’s programmers had significant experience using Git from various programming courses, and some already had experience using Git with the Unity game engine. Because of this, the team chose to use Git to maintain versions of the project and allow all of the team members to collaborate. The game’s file sizes were ultimately small enough that the team was able to avoid using the common Git Large File Storage (LFS) feature. The only issue encountered with this decision was an occasional issue where the team accidentally created some texture maps that were too large for GitHub to handle, which the team resolved by reducing their sizes.

4.2. Mechanics and Controls

The process of developing and refining the mechanics and their controls proved to be one of the largest challenges in the technical side of Factory Reset. While the team had a clear concept direction from early on, there were many wrinkles to iron out in their specific implementation.
4.2.3. Magnetic Feet Development

The magnetic feet system proved to be one of the largest design and implementation challenges in the project. Even on the conceptual side, the team ran into some early challenges when designing the player’s ability to pull themselves to a surface.

4.2.3.1. Constraining the Mechanic

The team’s first major concern was that if the player could pull themselves to any designated polarity pad they could see, designing any linear section of a level would require placing arbitrary walls to block this and control their progression through the environment. This would be awkward to do, as it could interfere with visibility and in many cases would further limit the team’s ability to make the environments appear as plausible spaces. It could also potentially lead to players clipping into walls or out of bounds if they were able to pull themselves into a corner or nook.

The team tried a couple approaches to addressing these issues. To reduce the need for arbitrary walls and bypass the issue of players pulling themselves into tight corners, the team explored using designated “target points” which the player could pull themselves directly to the center of. These would be polarized and use an emissive glow to draw the player’s attention. They could be supplemented by metal panels that the player can continue walking along once they have pulled to the polarity target but cannot pull themselves onto directly. The team’s other main approach to solving this issue was to limit the range at which the player could pull themselves to a polarity pad. An example of a prototype polarity target and metal surface can be seen in Figure 35.
Pulling to consistent, dedicated points worked well, although the metal surfaces proved troublesome. The team ran into issues where stepping between surfaces, such as from a polarity pad onto a metal one, sometimes would return players to the effects of normal gravity. Ultimately the team scrapped the metal surfaces and dedicated polarized points in favor of a more general system that allowed players to pull to anywhere they aimed at on a designated polarized surface. This bypassed the issue of the feet disconnecting while changing surfaces by ensuring that the player was never stepping between surfaces. The team also compensated for the risk this re-introduced of the player clipping into unintended surfaces through careful level design that avoided creating those situations in the first place.

Limiting the range at which the player could pull was far more straightforwardly effective, and is still fully implemented in the final version of the game. It solved the issue of needing to place barriers between polarized surfaces by instead getting the same effect by merely spacing them apart. The biggest downside of this was that it turned some aspects of level design into a matter of tweaking the positions of various objects until they were just out of reach enough that they could be only interacted with in the intended order. It did also require an addition to the heads-up display in order to indicate to the player that their target was not reachable, but that was very straightforward to create and implement. Figure 36 shows the player aiming at a polarized surface that is too far away to reach.
4.2.3.2. Physics Challenges

During initial prototyping, the team found they were unable to rely on Unity’s built-in Player Controller system, as they found its interactions appeared to be hardcoded to assume the default gravity orientation with no support for changing it. To compensate, the team had to implement their own movement system which used the robust collision detection already inherent to Unity’s Rigidbody physics system but supported arbitrary gravity direction changes.

The team’s initial approach to pulling the player to a surface was entirely physics-based. If the player collided with an obstacle along the way, they would get stuck to it due to the re-oriented gravity. To remedy this, the team initially implemented the magnetic feet to automatically disengage and return the player to normal gravity if the game ever detected that they were no longer standing on a surface they could stick to with their current foot polarity. This implementation was chosen because it seemed intuitive, handled cases where the player stepped off of a surface or landed on one that they could not stick to, and because the team was focused on getting the system functional quickly so they could experiment with it.

The problematic nature of this implementation became glaringly obvious once the team finally built their first level. Because the system automatically returned the player to normal gravity when they collided with a surface, level elements that obscured the path to a target became problematic. Attempting to pull to a point without a perfectly clear path would result in
the player colliding with the obstacle, stopping part-way to their target, and dropping, much to their confusion. There was no quick solution to this that the team could think of during the Alpha phase of development since it would require re-implementing the entire magnetic feet mechanic while the team was under significant time pressure to get a playable version of the game, so this issue went unaddressed in the initial build.

4.2.3.3. Polarity Clarity Challenges

The initially polarity-based nature of the magnetic feet system also proved problematic. Originally, the player’s feet could be set to either positive or negative polarity directly on command. The player had to look at their target and press the button for the polarity type opposite its charge in order to pull themselves to it. This initial system bore some accuracy to how magnets and polarized charges work in real life, since opposite charges attract and like charges repel, but it made it less intuitive to play the game. Not only did it make the act of pulling to a surface controlled by two separate buttons, but the team found even when testing the game internally that the player’s instinct was to select the polarity that matched the target to try to interact with it.

The team initially used this two-polarity pulling system in part because they also wanted to let players repel themselves from surfaces by setting their feet to the polarity that matched them. While this was interesting in concept, it proved to be challenging for several reasons. For one, the reticle-based system currently in place for targeting surfaces would make things awkward to control, as it would be rather impractical to launch off of a surface by looking down at it and then re-adjust your aim while re-orienting in the air. While that could have been circumvented with additional logic to automatically detect and repel from objects beneath the player’s feet, that would fail to address the other main issue with it: gravity’s influence. While repelling against gravity from a polarized floor would be fairly straightforward and intuitive, from any other angle gravity would cause the player to be launched in a trickier to understand arc with far less control over how they moved, as shown in Figure 37. Collision detection at this stage of development was prone to having the player clip through surfaces in some cases, so controlling the movement as closely as possible also would help prevent that. Conceivably the team could have implemented the system such that gravity was ignored while pushing or pulling the player from a surface, but they had other concerns beyond the further physical implausibility
and implementation challenge of that. The final nail in the coffin for repelling with the magnetic feet was the acknowledgement that the timing element and added difficulty in interpreting the visual feedback from launching off of a surface with the camera re-orienting worked against the team’s accessibility goals. Ultimately, the team decided to scrap the ability for players to repel themselves from a surface due to these challenges before even designing the first level.

Figure 37. If a player repels from a surface in the direction opposite gravity (top), their movement would be purely vertical and very controlled. If they repelled from a surface at a different angle (bottom), they would have a less-intuitive path, especially seen from a first person view.

4.2.3.4. Conflict with First Person View

Implementing the magnetic feet mechanic was also the first area where the team experienced friction between their experience goals and the actual gameplay. The team strongly felt that having the game viewed in first person was important, as this heavily emphasized the dramatic perspective change when pulling onto a surface and minimized concerns with the camera clipping into objects compared to a third person camera approach. Having the camera in
first person, however, meant that it would tend to swing dramatically when changing orientation, and thus could be disorienting to players and would make it difficult to meaningfully manually adjust its angle during this process. The team worked to mitigate these challenges by completing the orientation shift promptly so the player has more time to process their new orientation, as well as slowing the rotation speed to make it easier to follow and less jarring. This issue was also an inciting factor in the decision to remove timing-based execution challenges from the game, as they would become more punishing and stressful if the player also had to cope with disorientation and effectively losing camera control for frequent periods of time.

4.2.3.5. Refining the Magnetic Feet

Especially given the finicky nature of the feet pulling system at the time, it was clear that the mechanics needed some overhauls to be readily usable by actual players. The team replaced the feet polarity system with a unified “pull to surface” button. Since the team had scrapped the ability to repel from surfaces, having to recognize and choose the correct polarity to pull to a surface just made the game more frustrating to play. This let the team map it to just a single button, which was much more intuitive since it simplified the decision making process, reduced the potential for player error, and made the mechanic feel closer to a familiar gameplay action like jumping. The user interface still displayed the current foot polarity and the game still tracked it internally, but this greatly simplified the game’s demands on the player to control it.

Following the feedback from Alphafest, detailed in Section 7.2.1, the team further overhauled the entire magnetic feet pulling ability. With the new implementation, pulling to a surface brought the player into a special “Pulling” state where collisions were disabled and they attempted to automatically steer around obstacles, which completely bypassed the earlier issues of getting caught on obstacles on the way while minimizing cases of the player visibly clipping through walls. To address the issue of accidentally stepping off of polarized surfaces after pulling to them, he also added a special inverted collider mesh to them, shown in Figure 38.
These were invisible in-game, but once the player arrived at a polarized surface with one they changed into a special “Attached” state which prevented them from leaving the collider until they disengaged their feet, returning to the “Neutral” state and standard gravity, or used them on another polarized surface and returned to the “Pulling” state. He also reworked the internal tracking of the player’s orientation state to a unified parameter that could be easily referenced and automatically changed the necessary settings when its value was updated, which made the player state system much easier to work with and understand as a programmer.

4.2.3.6. Third Person Experiment

After creating the Alpha build and a 3D model for the player character, the team did briefly explore the concept of using a third-person camera. With this perspective they could see their whole character model at all times, as shown in Figure 39. The team’s initial concerns of it being incompatible with the existing level design due to the somewhat restricted nature of spaces in the first level’s compactor and storage column areas proved to be largely unfounded, as a simple third person camera mockup behaved surprisingly well. It also may have helped counter disorientation issues, as it would give players a more concrete point of reference for their placement in their surroundings. However, this perspective made the player character appear far smaller than intended, to the point where level collision and structure may need to be redesigned to make them appear as intended. The team also did not have any team members who were very
experienced in animation, and Given the timeline of the project, the team opted to prioritize refining their original first-person vision instead of devoting the extra resources to properly implementing a third person version.

![Figure 39. The third person mock-up test the team performed.](image)

4.2.4. Magnetic Hands Development

The magnetic hands that the player could use to push and pull objects were fairly straightforward to design but difficult to implement properly. From the beginning the idea of polarities was important, and while the magnetic feet being polarized did not make it into the final game, the hands being polarized did. This meant that the player could choose to use either positive or negative hands depending on what they wanted to do to the object before them. Early in development the team decided to add a limited range to the hands similarly to how they added a limited range to the feet because it made sense that the player would not be able to influence things that were very far away. Initially there was no way to know whether or not an object could be interacted with, but the team added indicators to the HUD for when the player was looking at an object they could push or pull that was in range and for when the object they were looking at was out of range.
Another decision made early on in the project was the decision to have the player toggle the hands on and off rather than having them have to hold the associated button to use them. The team made this decision primarily for accessibility reasons, since holding down a button can be problematic for those with motor disabilities. Later on in the process the team added the option to have the hands trigger on a button hold rather than toggle because it makes the game feel a little more natural.

Early in development the team decided that there would be both visual and auditory feedback for the player using their magnetic hands. When activated, a space on the reticle would light up and a sound associated with the hands being used would play so that the players would know when the hands were on and which polarity they had activated. This created some issues during the alpha build where players would not realize that the hands were on due to limited visual feedback, and would express frustration about the constant noise. Later iterations solved this by adding arms for FyED-OR which lit up the corresponding color when the player was using a magnet.

The hands were originally only able to be toggled on and off, but later builds defaulted to only activating the hands while the button was held, with an option to have them toggleable. Defaulting to this helped alleviate the issue of players accidentally leaving the hands on, while still maintaining the option for players to turn off the button holding if they did not wish to hold a button. The magnetic hands rely greatly on the object being interacted with. In the final game there were two such objects: the polarity boxes and the scaffolding, seen in Figure 41.

![Figure 40. HUD indicators for when an object is out of range (left) and in range (right).](image-url)
4.2.4.1. Polarity Boxes

The polarity boxes were the first object to be implemented and were simply boxes that could be pushed and pulled around freely with FyED-OR’s magnets. These facilitated the creation of several physics based puzzles including moving the polarity boxes out of a hallway, moving the polarity boxes to see a polarity pad to jump to, and knocking down other polarity boxes by launching one.

The initial implementation of this polarity box pushing and pulling was fairly buggy and allowed the player to do a number of unintended things including clipping themselves through the floor and flying on top of polarity boxes. This was present in the early alpha build, but was fixed with code that protected the player from doing these things. There was also a glitch where the polarity boxes would not be dropped when a player turned off their hands, but this was also fixed later in development to make interaction with the polarity boxes seamless.

One persistent issue the team encountered was a tendency for the player to clip through the floor the moment anything collided with them from above, which was disorienting and immersion-breaking at best and at worst forced players to reset the game. The team countered this by implementing a simple system to apply a counteracting force whenever the player’s collider clipped into the floor, as well as placing colliders in every scene that would automatically respawn the player if they accidentally fell out of the map. Another recurring issue
the team addressed was a glitch where players could essentially fly by carrying a polarity box in a certain way, which could allow them to bypass certain challenges. The team remedied this by restricting the range of angles that polarity boxes can be carried in. They also revised the aim detection system to use a more intelligent method with a detection radius around the reticle position instead of just a singular point, allowing the player to have much more leniency in their aiming when targeting polarized surfaces and obstacles. These fixes made the moment to moment gameplay significantly more robust and reliable, invisibly reducing player frustration.

4.2.4.2. Scaffolding

The scaffolding worked differently than the polarity boxes, as they only moved in a straight line along a predetermined path. These allowed for the creation of puzzles involving moving the scaffolding to give the player platforms to walk across and to gain access to higher up places. The scaffolding could also have polarity pads attached to them, allowing the player to move a polarity pad to another spot to get a better vantage point.

4.3. User Interface

The design of an understandable user interface was key to creating an elegant and accessible experience. The team focused on two main elements for this: the HUD and the menus.

4.3.1. Heads-Up Display

The team experimented with a range of ways to convey the HUD information in a clear way that did not interfere with the gameplay. Figure 42 shows an example of the progression of the HUD reticles. The team wanted to minimize the amount of text required as much as possible, to keep the interface readable to people regardless of their language, reading ability, or screen size. Because the game was in first person and the team did not have visible models representing the player’s body at this time, it was especially critical to convey all necessary information about their state through the user interface.
The team’s initial approaches to a user interface were not very unified. There was a simple dot in the center of the screen showing where the player was looking, a readout of the current polarities of the player’s limbs, and a colored glow at the bottom of the screen reinforcing the state of the player’s feet. The limb readout used a simple graphical representation of the player’s character, and was intended to also help give context to the player about their character given the lack of a visible model. To show the player’s orientation, the team started with a simple three-dimensional “compass” made of axes that rotated to match the world orientation, with the downwards direction marked differently. This would allow players to easily tell what direction they would fall if they disengaged their magnetic feet while attached to a surface. All of these elements can be seen in Figure 43. While this all was technically functional, it was not very cohesive and was scattered all around the screen. It also did not convey any information regarding whether or not the player could interact with targets, as this was before interactable objects like switches were implemented.
Figure 43. An early test of Factory Reset’s heads-up display. The reticle is visible in the center, with the hand polarity shown in the arc above it, the polarity of the feet shown in the arc below it, the orientation compass below that, the feet polarity glow along the bottom of the screen, and the graphical limb status readout in the upper-right.

To refine this scattered interface and make it more cohesive, the team looked into ways to consolidate its elements into a more unified reticle. The team looked to Valve’s Portal (Valve, 2007) game series in particular for inspiration, as they use a stylized visual reticle, show in Figure 44, to convey which of two polarities of portals the player has placed in the world, which was comparable to what the team wanted to convey with Factory Reset’s reticle.

Figure 44. A screenshot from Portal 2 (Valve, 2011) showing its aiming reticle. Image (Steam, n.d.).

The core things Factory Reset’s reticle needed were also present in the Portal’s (Valve, 2007): a central marker indicating what the player is aiming at, and regions around that which can be filled with a color to indicate the selected polarity of both the player’s hands and feet. After sketching numerous possible layouts and shapes, the most promising of which are shown in Figure 45, the team settled on a finalized design. They chose the last one shown in that listing
because it had a clear separation between the hands readout (the top arc) and the feet readout (the bottom arc), and gave ample space for additional indicators like a target being possible to pull to or interact with.

![Figure 45. An assortment of sketches of possible reticle designs which the team strongly considered (top), the version from the Alpha build (bottom left), and the final refined version (bottom right). This version featured special toggle-able indicators at the top to show when an object could be manipulated with the player’s hands and at the bottom to show when the targeted surface can be pulled to.](image)

For the Alpha Build, the team streamlined the user interface down to the essentials. With the switch to a single-button version of the magnetic mechanic it became unnecessary to convey the exact polarity of the feet to the player as it made no difference on how they used them. This rendered the secondary indicators of the feet state as just unnecessary clutter so the team removed them. They did, however, keep the reticle’s polarity indicator as a way to help convey the concept of the player attaching to surfaces with their feet. The team was also forced to remove the orientation compass, as the approach they used to overlay it onto the screen was unavailable in Unity’s High Definition Render Pipeline. Also, the team modified the reticle to feature an indicator for when an interactable like a switch was looked at, shown by a pair of markings to the sides of the reticle, and to indicate when an object was out of range to interact with, shown by a prominent “X” shape over the center of the reticle. These were important additions as they would help minimize frustration by giving players greatly improved feedback about when they could and could not take actions. The final heads-up display seen in the Alpha build appeared as shown in Figure 46.
Figure 46. The heads-up display from Factory Reset’s Alpha build. The first image shows the player picking up a polarity box with their magnetic hands, while the second image shows the player looking at an interactable switch that is out of their reach and thus currently inaccessible.
The HUD underwent a substantial redesign for the Beta phase of development. Taking into account the difficulties players encountered with interpreting the heads-up display, the team made several revisions and additions to it as shown in Figure 47. To minimize confusion between the two-polarity state of the hand readout and the functionally on-off state of the feet readout, the team changed the feet readout from displaying the polarity of the feet with a red or blue marking to just a white marking indicating whether they were active. They also reshaped the regions for the hands to further distinguish them from the feet, splitting the top arc down the middle into two pieces to mirror the visual separation of the hands. In an effort to make it more clear when players could interact with an object with their hands or their feet, the team revised the respective markings that appear to be bolder, appear both inside and outside the reticle, and taper in the general direction. To make it more readable against light surfaces, the team added a translucent dark outline around its white elements.

The team also experimented with adding supplementary icons and readouts to help convey and reinforce important information. When the player is attached to a polarized surface, the HUD shows a symbol of a downward arrow intersecting a horizontal line. When their hands are set to a polarity, the appropriate polarity symbol (a plus or minus sign) is shown above the reticle. The team also added back in a new orientation compass in the form of a three-dimensional arrow that is drawn at the top of the screen and always points in the direction of real gravity. Player feedback, particularly during supervised sessions, indicated that these helped as intended to at least some extent, but were met with some initial confusion. Some players found it difficult to trust and process the orientation arrow at a glance, and one even mistook it for an indicator of the direction to travel in to proceed in the game. Similarly, while players did clue into the meaning of the indicator for being attached to a surface, some assumed its arrow-like symbol was conveying the direction of gravity despite being a 2D image in a fixed orientation.
To provide further feedback beyond the reticle and help convey the player character’s robotic nature and their actions, the team also added visible models of the player’s hands. Although technically static models, these were animated through code to convey different details. When the player has their hands set to a polarity they glow with the appropriate color and shift forward to lock in place, and when they interact with a switch the right arm moves in a simple button-pressing motion. They also move in a cyclical motion while the player walks, pull backwards when they come into contact with a wall, and subtly shift as the player looks around. Combined, these give the player’s actions much more physicality and context in the world.

4.3.2. Menu

The team developed the pause menu and associated options menu continuously through the project, going through a few redesigns and frequent updates to support more accessibility options. The team implemented all menus to be fully navigable with keyboard, mouse, and controller. This made them accessible through any control method. At first the team designed every menu to wrap around when players attempted to move the selection off an edge, so that if
the player got to the bottom of the menu and hit down they would return to the top, accompanied by a differently tuned sound to indicate this special action. This became less clear as the menus got more complicated, so the ability to wrap around once reaching the edges of the menus was removed.

4.3.2.1. Pause Menu

The team developed the pause menu first. It was designed to set the timescale of the game to zero and pause all non-menu sounds while active to pause the action. This menu allowed players to continue the game, view the settings, return to the main menu, and quit the game. These were the only options presented on this menu to keep it simple and understandable. As a consequence of prioritizing other aspects of the project to get the game to a functional state, the original design of the menu just relied on the default button styles that came with Unity instead of ones the team designed. These worked at the time, but were iterated upon after Alphafest to make the aesthetic of the game more cohesive. Both the original version and the final version of the pause menu can be seen in Figure 48 below.
4.3.2.2. Settings Menu

This menu, pictured in Figure 49 below, functioned as a bridge between the main pause menu and the more specific settings sub-menus. It allowed players to navigate to one of five other menus: general settings, graphics settings, controls settings, audio settings, or back to the primary pause menu. Each sub-menu also contained the option to return back to the previous menu.
4.3.2.3 General Settings

The general settings menu housed the subtitle settings, which included adjustable font-size, transparency for the background of the subtitles, and an option to extend the duration that subtitles showed for. The settings menu also featured a sample subtitle that showed players a preview of their changes in real time. These settings were chosen because they would improve the readability of the subtitles, in particular to accommodate people who could not read the subtitles at their default size or speed. The final version of that menu can be seen in Figure 50.
4.3.2.4. Controls Settings

A core motor accessibility feature in *Factory Reset* was remappable controls. This is a vital and powerful way to help ensure that the gameplay mechanics work for a wide range of players. Players may wish to use these for any number of reasons, not just limited to ergonomic comfort or simple preferences. If players have a form of motor disability, they may require a specially-customized control layout to adapt to their body’s abilities. The team deliberately extended the control remapping system to include gamepads input as well, which enables even more flexibility as players are free to use any combination of keyboard, mouse, and gamepad input to control the game.

The team used *Unity’s* improved Input System as the basis for the control remapping feature. This featured built-in support for changing control input mappings at runtime without needing to adjust any of the code that responds to them, which made implementing the remapping straightforward. The main aspect of this the team designed ourselves was the actual user interface for assigning key and button mappings, shown in Figure 51.

![Figure 50. The general settings menu.](image)
Another vital aspect of the input system that players could customize was the camera controls. The team provided support for inverting both horizontal and vertical look directions to accommodate the varied control preferences of different players. More crucially, the controls menu included sliders to allow players to customize the sensitivity of their look inputs. The controls menu also had separate sensitivity sliders for mouse and joystick camera control, which allowed players to customize their settings by input device instead of needing to adjust them every time they switch. These sensitivity settings were particularly important due to the first person aiming central to the game - if players are unable to aim at targets due to oversensitive input or find the camera movement sluggish, it will negatively impact their experience of the game or even make it too frustrating to reasonably play.

Finally, the team implemented an option to make all places in the game where the user would have to hold a button replaced with toggles. This was to aid players who may have had a hard time holding down buttons.
4.3.2.5. Graphics Settings

The graphics settings menu contained a number of options that were each added at different stages of development to improve performance and provide options to players who may have performance issues from rendering the game. The ability to toggle fullscreen on and off came first, and was added to this menu due to the performance issues some players encountered. The team believed playing the game in windowed mode would help mitigate these issues for at least some players. Shortly after, the team added the ability to change the resolution to several supported resolutions. This was also done to further improve performance, particularly since the team was unable to identify a way for players to manually resize the game window, and this addition was seen to noticeably improve framerate on several playtesters’ computers. The option to hide the camera rotation of the magnetic feet pulling was added in an effort to reduce motion sickness. It was originally called “Boot Fade” but was changed to “Reduced Camera Motion” for the final build to make it more clear what the setting’s intent and function was. The menu’s backgrounds normally have partial opacity so that the player can see the paused game world a little bit through the menu. For players who did not like this or found that it interfered with their understanding of the menus, however, the team added a “High-Contrast UI” option that made all the backgrounds solid colors with no transparency. Finally, after some playtesters requested it a camera field of view (FOV) slider was added for players who preferred to play with a different FOV. All these options can be seen in Figure 52 below.
4.3.2.6. Audio Settings

Audio settings were added to allow players to control the levels of the types of sounds in the game. Originally the players were given settings for master, sound effect, environment, and UI volume. Eventually the environment volume slider was removed because the team realized that most of the sounds covered by that setting fit more clearly into the sound effects category. Also, after the first iteration sliders were added for music, which had not been in the game when the audio settings were introduced, and for dialog. The sliders besides master covered every sound in the game, allowing players to adjust them to the levels they found most convenient. The final list of audio sliders can be seen in Figure 53.

Figure 52. The graphical settings menu.
4.3.3. Subtitle System

Subtitles are a very useful feature for a range of players. They are generally used to provide a visual transcription of audio such as speech, making it easier to understand what is happening in the game regardless of volume levels, disruptive noise, or individual difficulties with auditory processing (Game Accessibility Guidelines, n.d.). In the case of Factory Reset, the only voice acting in the game is in Russian, the player’s character has no voice at all, and the game was developed for a primarily English-speaking audience, so subtitles were critical for players to understand the story component of the game. The team also repurposed the subtitle system for control prompts, as this was an efficient way to use existing resources to convey this information. This addition to the subtitle system made use of features in Unity’s new input system to display the name of the currently mapped control for each control prompt.

The team had to implement a system to ensure that the most important subtitles always displayed in place of others. Each subtitle created had a time duration after which they would disappear as well as a priority level. Only the current subtitle with the highest priority level. This priority system made it simple to control the display of subtitle messages regardless of context.
4.4. Cutscene System

To reinforce its story elements and make the presentation of the game more cinematic, the team created a cutscene system that built off the existing subtitle system. The basic implementation of it was simply a different use of the system - in order to queue up sequences of messages, the team created them with increasingly long durations and high but incrementally reduced priority, so the moment one was resolved the system would immediately switch to displaying the next one in the sequence. Due to the time pressure when developing the Alpha build the team had initially implemented the code to initiate cutscenes rather hastily, in a manner that hard-coded it into the player’s controller script. While this worked then, it was problematic for any future testing as there was no convenient way to disable the opening cutscene from playing.

While developing the Beta version of the game, the team addressed these issues and fully revamped the system to have much greater utility beyond them. The team created a special type of scriptable object that defined a cutscene’s data and properties and could be assigned to the player at the start of a level or during gameplay. The team also made it so that any cutscene could be set to play the “boot-up” animation used at the start of the first level, as well as end in a reversed version of it. Coupled with the ability to load the next level or reload the current one at the end of a cutscene, this allowed the team to roll both smooth level transitions and respawning directly into the cutscene system. Whereas before the level would just abruptly change when the next one loaded, now the screen would smoothly turn black from the fade-out animation, automatically change or reload the scene, and then smoothly fade back in.

4.5. The Infinite City

The second level of Factory Reset, the sky highway, required the most involved custom programming to implement out of the three levels in the game. As seen in Figure 54, the level takes place on a group of flying cars travelling through a hazy city of towering buildings with luminous windows. In order to convincingly sell that they were flying in traffic, it was necessary to have the cars appear to continually move relative to both the city and each other. However, the team had programmed the magnetic feet mechanic with the expectation that the surfaces the player pulls themselves to would remain stationary, which complicated both of these forms of
movement. The team’s solutions to this were to have the movement through the city be an illusion and to make the vehicles idly move at a distance but settle into place when approached.

Figure 54: Three views of the sky highway level in Factory Reset.
The team created the city effect by randomly generating a cityscape from a set of building models, shown in Figure 55, that continually scroll past the player. With this set up, the player character moves only under the player’s control and the flying cars remain stationary aside from their localized idle motion. Whenever a building gets a certain distance behind the player, they teleport far in front and change their position, shape, and gradient pattern. This effect is hidden from the player by a combination of environmental fog, which obscures the non-glowing parts of them while they are far away from the player, and special code to adjust the emissive glow of the building windows. It was necessary for the windows to fade out before respawning and fade back. To further prevent awkward spawning, the buildings also have logic to relocate themselves further away if they respawn overlapping another building, keeping the city feeling more believable. The end result of this is the illusion that the player is riding the flying cars through an endless cityscape, no matter how long they spend in the level.

Figure 55: The three models of buildings used to form the city in Factory Reset.
The team implemented the flying car movements with proximity-detecting code. At a far distance they continually shift around between different positions within a specified spatial range, but as the player approaches within a certain range they settle into a single fixed position. This allows them to idly move while being observed but remain stationary while the player is near enough to interact and require it. While this does have the quirk that players can control the movement of cars in certain places by stepping towards and away from them, in general it is a convincing effect.
5. Art Development

This section covers the artistic techniques and tools that were used during the development of Factory Reset, as well as the various justifications for why certain artistic decisions were made. In each of the aforementioned areas, this section will go in-depth on the asset creation pipeline for three critical artistic elements: props and environmental assets (Section 5.3), level creation (Section 5.4), and finally character creation (Section 5.5). Within each section, the pipeline will be expanded in detail from conceptualization to the final game-ready state, following the core steps within their respective asset creation pipelines.

5.1. Overview

The creation of three dimensional assets is a multidisciplinary field, in which the artist (or team of artists) must be able to work within a varied environment of visual toolkits and programs. For Factory Reset, a plethora of dedicated 3D and 2D visual art programs were used. Before any work was done, all designs were explored through concept art. This process was primarily done within Gimp, an open source GNU Image Manipulation Program. Following concept art came the physical asset creation. Most importantly in the artistic pipeline for Factory Reset was Autodesk Maya 2020; an industry standard, 3D modeling and animation suite. Autodesk Maya 2020 is where 96% of the game’s physical 3D assets were created, and without it there would be no visual component to the game. Within Autodesk Maya 2020, the team created and UV unwrapped⁶ all the level assets, level props, game props, and characters. Texture work was done within Adobe's 2021 Substance Suite, which is comprised of Substance Painter, Substance Alchemist, and Substance Designer. Creating specialized materials for certain surfaces of the polygonal meshes was primarily done within Substance Alchemist and Substance Designer, both material creators and editors. After the creation of specialized and realistic textures through the process of Physically Based Rendering (PBR)⁷, the 3D models and materials were imported to Substance Painter, in which they were hand painted. Finally, all modeled and

---

⁶ UV Unwrapping is the process in which a 3D artist creates a flat representation of the surface of a 3D polygonal mesh, in order to produce a canvas to texture said 3D model. The U and V in UV Unwrapping refer to the horizontal and vertical axis of the 2D space within the texture sheet.

⁷ Physically Based Rendering or PBR for short is the texture workflow that aims to simulate how light reacts with a model in an attempt to simulate real world materials.
textured 3D models were imported into the *Unity* game engine, where the textures and materials of the polygonal meshes created in the aforementioned programs were integrated with each other, which can be seen in Figure 56.

![Figure 56: An untextured flying car from Level 2 (left) and the textured flying car (right) within Unity.](image)

Lastly, within Unity, it is the job of the artist to set dress: the process of populating an environment with assets in a realistic (or game specific) manner. Throughout set dressing an environment, it is important to be able to create a visually thematic story of why props are placed where they are. Through the process of placing assets thoughtfully, the artist is able to achieve a more lived-in quality to the environment, thus improving the overall realism of the game’s world.

5.1.1. Defining Sovietpunk

![Figure 57: Early whiteboxed environments for *Factory Reset*, showcasing the fusion between Cyberpunk and Sovietpunk.](image)
Taking heavy inspiration from Frédéric Chaubin’s “Sovietpunk”, the *Factory Reset* team wanted to meld both cyberpunk and Sovietpunk together and create a world in which cyberpunk’s aesthetics existed in the Soviet realm. Chaubin’s definition of the Soviet fourth age of architecture and Sovietpunk can be seen in parallels to the American 1980’s cinematic gilded age too. Many buildings which Chaubin show as part of his book, contain similarities to contemporary cyberpunk aesthetics, such as the use of prominent geometric shapes, colors, and the “powerful and imposing adoption of lighting”. These elements so happen to be found in the genre of cyberpunk. Seeing these similarities, the team decided to utilize a juxtaposition of cyberpunk’s themes of capitalism and urban decay, and mesh it with communism and themes found throughout this fourth age of Soviet architecture.

5.1.2. Establishing *Factory Reset*’s Sovietpunk

*Figure 58:* 3D render of the antagonist, OLOF-14, showcasing the meld of both Soviet iconography and science fiction and futurism.

*Factory Reset’s* take on Chaubin’s Sovietpunk is a bit different, as it takes a much more hyperstylized approach accepting a lot of artistic license, which can be illustrated with Figure 58. This is because the fourth age of Soviet architecture took place during the 1970s through to the
1990s. *Factory Reset* on the other hand takes place in a future in which the Soviet Union still exists. Modeling off of the architecture that can be seen today, *Factory Reset*'s environments are much more futuristic, using Chaubin’s definition as a launching ground. Throughout the game, the art style is mixed with elements of far-future science fiction: autonomous robots and androids, flying cars, and artificial intelligence. Unlike Chaubin’s definition of Sovietpunk, *Factory Reset*'s definition of Sovietpunk is much more focused on taking a Soviet approach to cyberpunk, adopting elements of Chaubin’s fourth age of rebellious Soviet architecture.

5.2. High Definition Render Pipeline

To give the game cutting edge graphics that took advantage of the latest in graphical technology, the team used *Unity*’s High-Definition Render Pipeline (HDRP). HDRP is entirely different from *Unity*’s other render pipelines. It uses lighting models based on the real world and also takes more advantage of the GPU to render (*Unity*, n.d.). This results in a much more realistically lit environment. Because of the more complicated lighting, lower-end computers may have trouble keeping up with the demands of HDRP.

Developing *Factory Reset* using *Unity*’s HDRP was instrumental in reaching the final look featured in the game, although it did bring several challenges. HDRP is designed with high-graphical fidelity in mind, and provides powerful lighting and rendering options as well as customizable environment settings and volumes to give fine-grained control of effects like fog, bloom, exposure, and color adjustments. Brightly glowing objects in HDRP have a distinctive, realistic look where their color appears as a bright white with a tinted glow, while also being configurable to better preserve their raw color if desired. Given the emphasis the team wanted to put on lighting and graphical fidelity, HDRP was thus a good fit for the project goals. The team made the decision to switch over to HDRP from the default *Unity* project in the middle of B Term 2020, while the team was exploring how they wanted to handle lighting in the project.

The conversion process was slow and would take longer the further along in the project the team became, so given the team’s artistic and scope ambitions it was important to try HDRP early on in development to see if the team approved of it and then commit to one path. While the team did ultimately choose to keep it for the artistic benefits and control it afforded, the changing to HDRP did cause several issues for the team. On one team member’s computer *Unity* was still
usable but began crashing frequently, while another was running *Ubuntu* at the time of conversion, which was incompatible with HDRP. The only reason this was not a critical reason to avoid HDRP was that he had plans to acquire a *Windows* computer soon after and had mostly audio left to work on for that level and thus did not need to use *Unity* itself. The conversion also interfered with the team’s initial approach to the orientation compass in the HUD, as this originally used the *Unity* Camera Stacking feature to overlay one the compass image onto the screen. The team eventually found an alternative approach where the compass was rendered to a texture drawn on screen, but this was not implemented until the Beta version created in C Term and was thus omitted from the Alpha build.

### 5.3. Prop Creation Pipeline

The next step was to establish the pipeline for prop creation which would conceptualize and realize necessary level-related objects that would populate the world of *Factory Reset* (5.3.1). This started with creating concept art to explore the game’s worldbuilding and aesthetic. This was important to establish from the outset, as it helped ensure a consistent art style across the game’s assets and gives a better idea of what props are necessary and critical to the game. After conceptualizing came addressing key visual-accessibility concerns with the planned props (5.3.2). This was necessary as props that are game mechanic objects, such as the polarity boxes and magnetic attachment points, are needed to be seen and distinguished by any player, regardless of visual impairments such as color blindness and dichromatic vision. Following this, the props were modeled within *Autodesk Maya 2020* (5.3.3) and finally unwrapped to texture them within *Substance Painter* (5.3.4).
5.3.1. Conceptualization

Figure 59: “Sovietpunk” by Ian Llanas (2016), an art piece showcasing the divide between traditional Russian architecture and a futuristic vision of a Soviet future.

Before any work began on actual modeling, the prop production pipeline began with gathering reference images of real-life environments that Factory Reset’s environments would draw inspiration from, such as Figure 59. This was done through the creation of a Mood Board, a slideshow with various environments the team wanted to capture. This included elements of 1980 Soviet Design, Sovietpunk aesthetics, as well as elements from cyberpunk-derived worlds.
Figure 60: Concept art created for *Factory Reset* envisioning a Soviet factory.

Figure 61: Concept art created for *Factory Reset* envisioning an interior of a dilapidated Soviet environment.
Drawing from these sources, concept art was sketched to capture general designs, shapes, features, and ideas the team wanted to incorporate in the final game, such as in Figure 60 and Figure 61. This allowed for two crucial developments: firstly, it allowed the team to create a refined and visually cohesive art style, and secondly, it allowed the team to create a rudimentary asset list of all necessary props and environmental assets. Having this list early on allowed for a rough estimate of assets necessary to get the game into a playable state, thus refining the artistic scope from the initial planning stages of Factory Reset.

5.3.2. Visual Accessibility Design

Visual accessibility was an imperative area that the team addressed at the start of development. Out of the many visual accessibility concerns was choosing a well designed color palette that would work well with common forms of color blindness and people with visual impairments. The team next designed many of the important gameplay props, such as the polarity boxes, with this in mind.

Figure 62: Screenshot from Factory Reset showcasing the two variants of polarity boxes: red (positive) and blue (negative).

For Factory Reset, the core gameplay colors of red and blue were chosen to distinguish between polarities. This choice was done for three fundamental reasons. Firstly, red and blue are often portrayed on magnets in general, red generally depicted for positive, and blue for negative.
Next, red and blue are a strongly-contrasting color pair. This allows the player to distinguish polarities much more easily, regardless of the surrounding environment. On core gameplay objects, these colored areas also feature emissive textures that allow them to glow brightly and stand out more from the environment.

![Figure 63: The two polarity boxes simulated with different forms of colorblindness. Trichromatic (top left), Deuteranopia (top right), Protanomaly (bottom left), and Tritanopia (bottom right).](image)

In terms of visual accessibility, red and blue are one of the few contrasting color pairs that can be distinguished with all of the various forms of dichromatic vision, the most common forms of color blindness. In *Factory Reset*'s case, the use of red and blue to represent the polarities was designed to allow any potential player with dichromacy to still be able to differentiate between them and still play the game, as this is a pair of colors that can be distinguished with any two functioning types of cones. Figure 63 illustrates the simulation of the different forms of *Factory Reset*'s polarity boxes with different types of color blindness.
Figure 64: The two polarity boxes simulated with monochromatic Full Color Blindness, Achromatopsia (left) and Blue Cone Monochromacy (right).

In some rare circumstances, some people may have a monochromatic view, in which they can only detect contrast. Monochromatic “Full Color Blindness”, specifically, Achromatopsia causes the player to not be able to distinguish color, and Blue Cone Monochromacy can only distinguish between contrast. In order to allow for important gameplay elements like the polarity boxes to still be seen, the artist added non color indicators to the polarity boxes in the form of a positive symbol and a negative symbol. These can be seen in Figure 64.

Figure 65: Final environment lighting colorblindness simulations. Trichromatic (top left), Deuteranopia (top right), Protanomaly (bottom left), and Tritanopia (bottom right).

Gameplay-specific assets’ visual accessibility was not the only area of visual accessibility with forethought. The entire scene, including overall environmental color palette and lighting
were chosen to work with any form of colorblindness, as well as the aforementioned, more rarer forms of full color blindness. This can be seen within Figure 65.

5.3.3. Modeling

The next step in the game’s prop creation pipeline was to model the assets in *Autodesk Maya 2020*. The props themselves had to be on par with the balance between complex geometry, while balancing polycount so that the game would still perform well on a variety of different computers. All models were modeled to a uniform world scale, allowing for easy prop placement in the environmental level set dressing. This also allowed the artist to reuse assets across different levels by being modular. Keeping these assets modular allowed for them to be repurposed without the need to create more level-specific, helping reduce the overall time spent producing 3D assets for the game, which can be seen in Figure 66.

![Figure 66](image.png): Screenshot from Autodesk Maya, showcasing the different props used (and some which were not used) within *Factory Reset* game environment.

One fairly important aspect in creating realistic props is making sure to bevel edges. Hard edges on any surface of a mesh, as much as they may lower the polycount, will always look much worse in the grand scheme of things. Most importantly, hard edges cause a plethora of aliasing issues and baked lighting artifacts, and generally cause more harm to visual fidelity. No
hard edge, such as a ninety-degree angle, is mathematically perfect in real life. Adding bevels and slightly rounding corners allowed for the props to look much more lived-in, and provided a much higher feeling of “belonging” in an environment.

5.3.4. UV Unwrapping & Texturing

All assets were UV unwrapped within Autodesk Maya 2020 before they were exported as .obj files. This was necessary in order to create any sort of textures, as it was required to map the 2D textures onto 3D polygonal meshes. These unwraps were then rotated and repositioned in the UV space to best utilize the texture space available.

Following the UV Unwrapping, the texturing process began. This was done in tandem with Substance Painter, Substance Designer, and Substance Alchemist. Within Substance Painter, each of the models were hand painted utilizing hand-generated materials created with Substance Designer, Alchemist, and Painter. Assembling a collection of textures and materials prior to this stage is crucial to reducing the time cost of texturing.

![Figure 67](image_url): Crate render for showing the detailed texture work, specifically finger prints.

Creating realistic textures requires a lot of small details such as scratches, dirt and dust, to even reflective fingerprint grease, which is showcased in Figure 67 and 68. As this is a laborious task to handle manually, some of these micro-details were automated within Substance Painter by creating specialized SmartMaterials that apply custom material and texture data from.
Substance Designer and Alchemist through custom generators. However, bulk of the texture work was still done by hand in order to ensure the desired graphical fidelity and realism.

Figure 68: Small detailed prop of a Soviet 1957 Travellers Mug, showcasing the intricate texture work found within Factory Reset’s props.

Figure 69: An example of a UV texture map of the “Overhang Light”, showcasing the detailed texture work of grime, scratches, and other discolorations throughout.

Once these textures were completed, they were individually exported at different resolutions determined by the prop’s importance. Higher resolution, and thus higher detailed textures were saved for important elements that the player would see often, whereas lower resolution ones were used for objects the player would not frequently interact with like distant objects and props scattered on the ground. Above all entryways and exits the player had to go
through, exist what was dubbed as an “Overhang Light”, or in other words a model of bright text in Cryllic, and some flashing red lights to help draw the player’s attention to this area.

![Image of door switch and propaganda poster]

**Figure 70:** Close up view of the door switch found within *Factory Reset* (top). Mao Tse-tung (Chairman of the PRC) and Nikita Khrushchev (Soviet Premier of the Soviet Union) meeting together two years during the historical split in 1958 (bottom left (Unknown, 1958)). Propaganda poster utilizing the two scripts showcasing the “Socialist Fraternal Love” painted by Vikto Ivanovich Ivanov (1958) Image bottom right. (Archive, W. C. D., 1953)

There is more than meets the eye in each object texture in *Factory Reset*. The artist used texturing as a continuation of visual storytelling by adding specific details to each item that would thematically make sense in the game’s world. Each asset in-itself tells a story, and that story helps create a more believable world. For instance, the “door switch”, that’s often found next to entrances throughout the levels, includes various markings on it. The expected Russian characters and Soviet symbolism exist across this inconsequential environmental asset; however, on closer inspection the player will realize there are also Chinese markings blazoned across the texture. In this hypothetical world that *Factory Reset* finds itself within, it’s is implied that the

5.4. Environmental Art Pipeline

This subsection takes an in-depth look at the creation of Factory Reset’s level design and the reasoning behind the approaches taken with it. Although it has a few key distinctions, the environment pipeline is overall similar to the prop creation pipeline. Firstly, there is the conceptualization stage (5.4.1) in which environments are drafted with additional level design sketches. The level design sketches include how the puzzle elements would function, and are closely tied to the gameplay of Factory Reset. Next, is the greyboxing stage (5.4.2), in which the level is set up using placeholder geometry that will be replaced in the modeling stage (5.4.3) and the subsequent UV unwrapping and texturing stage (5.4.4). After the physical level is completed, the environmental set dressing stage begins (5.4.5) in which the artist goes around the level placing the props created in section 5.3 in a realistic fashion. Following set dressing comes the placement and baking of lighting objects within the level (5.4.6), and the creation of customized environment profiles to further refine the look of the level (5.4.7).

5.4.1. Conceptualization

Much of the concept art created in the prop creation pipeline also acted as environmental conceptualizations; however, a few additional dedicated environmental concept pieces, shown in Figure 71, were also created to get a better sense and feel of the world.
Figure 71: Team sketches of Factory Reset’s second and third level showcasing specific assets and the overall rough setup of the environment.

The core of the conceptualization stage was dedicated to creating level designs for Factory Reset’s three levels. In each of the level designs, it was discussed what type of environment the player would find themselves in, but more importantly why the environment exists. Defining the reason the player would be participating in an environment, based off of the potential puzzles the level could have, was substantially helpful in the creation of level designs. For example, in the first level, the player finds themselves in a vertical storage facility centered around an elevator platform, based off of real designs of vertical car parks in metropolitan cities in Europe, such as the Hayabusa Vertical Parking System in München showcased in Figure 72.
Figure 72: Photo provided by Hayabusa, at the Hayabusa Vertical Parking System. (München, Germany). Image (WWJD, 2016).

Using that preliminary information allowed the artistic design to carry over onto level two, in which the player finds themselves exploring flying cars through an unidentified Soviet city. Keeping a thematic, but also gameplay-orientated approach to the conceptualization stage was imperative to the overall level and environmental pipeline.
5.4.2. Greyboxing

Unlike the prop creation pipeline, the greyboxing stage of the environmental pipeline was needed in order to create the rough approximations of what the level would look and play like. Utilizing simple geometry, the team was able to create the outline of what the level would end up looking like, and were able to iterate upon it easily without spending time recreating assets. Throughout this process; however, the team could also figure out the overall feel of the gameplay in a rough environment. This allowed the team to test pacing and better develop puzzles for the game in an environment that could be later iterated upon. For instance, the height of the vertical segment in Level One, changed frequently after internal-testing found that it was too tall and took too much time to get to the top. Greyboxing the environment was also the first part of recreating the 2D conceptualizations from section 5.4.1 into an actual playable 3D environment.

Figure 73: Greybox environment of Level One’s first section, showcasing polycount.

Figure 74: Original textured environment for Alphafest (as well as drawn concerns with the pre-modular approach)
5.4.3. Modeling

Once the greybox environments were all agreed upon, the next step of modeling the final iterations of the environments began. The goal was to create much more detailed, visually appealing level models which would replace and decorate the previous greybox environment.

5.4.3.1. Original Modeling Design

Originally, the team had modeled the main portions of the environment in a very interconnected manner. For instance, within level one the team modeled the environment in five separate sections: the initial starting area, the lowermost section of the elevator, the elevator shaft...
segments, the topmost part of the elevator, and the large hangar room. These five segments were then exported to *Unity* and arranged to form the full level structure.

![UV Map](image1)

**Figure 77:** Level One’s second segment’s inefficient UV Map.

While this initially seemed to be a workable approach, the team quickly ran into a substantial problem: the use of the UV texture space for these segments was incredibly inefficient, as shown in Figure 77. The team recognized that after the segments were later textured, which revealed the apparent resolution of these segments to be incredibly low. The only solution at the time was to increase the resolution of the texture maps to 8k, which ended up lowering the performance of the game, and increasing the overall file size of the project to 32GB. This was unacceptable for a vertical slice, and had to be addressed. Such issues can be seen in Figure 78.
5.4.3.2. Modularity Switch

Realizing that the five environmental segments for level one was going to be a complication for future development, it was recommended that the team switch over to a more modular system. Instead of having only five specific segments, the segments were broken into much more disconnected pieces, reaching a total of nineteen much smaller segments. These segments then replaced the original five segmented level one sections near perfectly, as seen in Figure 79.
Switching over to the more modular environment and level system allowed for several benefits. Firstly, the UV texture spaces for each modularity segment became much more efficient. This allowed for higher resolution textures at 4k wrapped on the textures during the texturing stage without increasing the actual screen-space resolution to 8k. This served to improve not only the performance of the game, but also aided at lowering the project file size to only 5.4 GB. Lastly, the biggest improvement of switching to a modularity system was that it allowed for easier set dressing in subsequent level design by reusing level one assets. This allowed the team to develop the third level from the greybox stage much more quickly and easily. A full atlas of these modular pieces is shown in Figure 80.

**Figure 80:** Full model atlas of modular level elements created in order to be reused throughout various levels.
5.4.4. UV Unwrapping and Texturing

Following the team's decision to switch to the modular system, began the second UV unwrapping and texturing stage of Factory Reset’s environmental development process. Much like the UV unwrapping and texturing stage in the prop creation pipeline, the environment was textured using Substance Painter with aid from Substance Designer and Alchemist. The initial texture work, eventually scrapped, can be see above in Figure 81.

**Figure 81:** Level One’s original texture work, which was later changed.
Since the modularity system was intended to be reused, many pieces such as the wall segments had to be more-or-less generic. The artist removed many of the pre-painted posters and markings that would distinguish certain segments as level specific. This made each piece a bit more universally applicable to the game’s levels, as seen in Figure 82.

One large decision the artist made was choosing an environmental color palette, as seen in Figure 83. Much like how the gameplay props of 5.3 chose the color combination pair of red and blue, the level assets were given the complementary pair of orange and blue. These colors
not only are highly contrasted in general as complementary colors are, but also were easily distinguishable with any dichromatic vision impairments. This allowed the levels themselves to incorporate contrasting hues to better distinguish certain areas of the level, but also create areas of interest to guide the player through defining areas of warmth and coldness. Lastly, these colors served to be thematic of the game’s overall art style, as the USSR used teal and orange in machines, robotics, and interiors of ministry buildings - all of which were featured in *Factory Reset* through its Sovietpunk aesthetic.

5.4.5. Environmental Object Placement (Set-Dressing)

![Figure 84: Factory Reset’s final set dressing pass in Level One.](image)

Once the first environment was game-ready, the artist began set dressing. The overall level, despite being modeled and textured, still looked devoid of signs of life or use. Using the props created in section 5.3, the artist went through the level and began the placement of the props throughout the level. This can been in the final version of Level One’s first room, in figure 84.
Figure 85: Another example of clutter during the set dressing stage of the game on Level One.

The set dressing stage allows for the details that bring a sense of life to the environment. This includes the placement of construction equipment, railings, light fixtures, signage, and the like, as well as small props like debris and ground scatter like broken cans, discarded paper, and metal mug. Combined together, these allow the level to feel a lot more grounded in reality. One of the largest parts of set dressing is that it allows the artist to visually storytell and create a sense of history and world building within the environment. An example of this is foreshadowing elements for future levels, such as having parked and discarded flying cars in level one’s vertical elevator section matching ones later seen flying, which gives the player a sense that the world created for Factory Reset could be a real, authentic environment. This can be seen via the myriad of props placed in Figure 85.
5.4.6. Lighting

![Figure 86: Level One’s lighting from above, showcasing the contrast between the “warmer” orange side (left), and the “cooler” blue side (right).]

The second to last, the longest step in the environmental pipeline was implementing lighting within *Unity*. Lighting can easily make or break a game, and require many iterations to get the intricate balance between visual fidelity and game performance in check. Within *Unity*, there are three types of lighting profiles: *real-time*, *mixed*, and *baked*. Real-time lights are calculated live while the player is in the environment. As a result of these lights being calculated real-time, they update all lighting and shadow profiles within the environment whenever it changes. For example, if the player moves a polarity box, the shadow cast by the polarity box needs to update live. Ideally, all lights would be real-time; however, real-time lights are the most performance expensive, and were used very sparingly. Baked lights are the opposite: they exist pre-calculated before the game is built and finished. In other words, these pre-calculated lights directly affect the colors of the textures and materials within the *Unity* environment. Lastly, mixed lighting is a mix of the two, in which the calculations are updated at runtime, albeit much less frequently, and still include the pre-calculated data of baked lighting. For *Factory Reset*, a majority of the lights were baked, with a handful of mixed lights that were placed near movable objects and realtime lights used in places where the light source would be moving or otherwise changing state.
Figure 87: Lighting was used to direct the player’s direction to look at certain areas of the map. In this screenshot, the light points towards the polarity pad on Level One.

From the perspective of gameplay, lighting was used to help the player distinguish between certain areas of the levels and guide their attention, as well as breaking up the monotony of the color palette of orange and teal from 5.4.4. This allowed the artist to expand the visual interest of the levels. This can be seen in Figure 88.

Figure 88: An example of the first level’s high contrast lighting.
For example (see Figure 87), at the start of level one, the player begins in a cramped trash compactor with no way out but up. By having a light facing upwards (with the assistance of volumetrics, further discussed in the next section: 5.4.7), it encourages the player to focus their attention upwards to the polarity pad they need to interact with to escape.

5.4.7. Volumetrics & Post Processing

Figure 89: An example of the lighting interacting with volumetric fog profiles mixing orange and teal lights together.

The final part of the environmental pipeline is finalizing environmental settings like volumetric fog and post-processing effects. The atmosphere in Factory Reset was important to help sell its world, as the places the player finds themselves in are supposed to be more-or-less worn down, eerie, environments. Having the levels filled with an ominous fog that meshes well with the lighting system within the game was seen as important to building a realistic world. On top of world building, volumetrics also helped lighting create stark contrasts without actually lighting up any surfaces. Since the shape of the glow cast by a light source would show up in volumetric fog, it can be used to direct their attention. See Figure 89 for an example, as well as Figure 90 and 91 for a before and after.
In the second level of Factory Reset, the volumetric fog effect was especially important as it was used to preserve the illusion of flying endlessly through a city. The illusion was created by having buildings constantly scrolling past, and then teleporting ahead and changing their appearance whenever they moved too far past the player. With fog effects the team was able to mask this effect by having the buildings fully obscured by the time they respawn, although this also required implementing special logic to make the emissive illuminated windows of the buildings (which were not obscured by the fog) dim and fade out while this happened.

Figure 90: Near-final scene showcasing lighting without volumetric post processing effects.

Figure 91: Final scene showcasing lighting with volumetric post processing effects and custom volumetric profile.

Factory Reset also made use of the post-processing features offered with Unity’s HDRP. It allowed the team to customize the exposure and bloom settings, which were important for really selling the emissive glow used on many of the props and environments. It also let the team increase the saturation of colors in the levels and thus counteract the tendency of HDRP emissive glows to wash out colors where the glow was brightest, which inadvertently had made it harder to identify the colors of certain polarized objects.
5.5. Character

Since Factory Reset’s environments are not populated by non-playable characters, the need for other character art was considered trivial and largely outside of the game’s scope. The player character was a key exception to this; the character creation pipeline was still needed in order to successfully develop a well thought out player character that would exist in the environment. Even though the game was intended to be in first person, a player character model was useful both for providing hand models that could be seen while playing and help convey what the player was doing, as well as for creating promotional materials. The character was also designed during the period when the team was considering experimenting with switching to a third person view. The process of developing them began with preliminary conceptualization through sketches (5.5.1). Next, the team’s artist created the player character in 3D by first modelling (5.5.2) then UV unwrapping and texturing them (5.5.3). Lastly, the rigging process of the character’s first person model was implemented within Unity (5.5.4).

5.5.1. Conceptualization

Factory Reset’s player character began its inception during the research and conceptualization stage. It was critical that the player character was grounded within reality as to bring more life to the world that Factory Reset is set in. Given the magnetic nature of their abilities and the technologically-themed setting, the artist chose to have the player character be a robot because it fit the technologically-themed setting. The team made several draft sketches of different robotic player characters, shown in Figure 92.
During research, the team learned about the autonomous Russian humanoid robot named FEDOR, shown in Figure 93, which was sent to the International Space Station in 2019 (Russian Foundation for Advanced Research Projects, 2017). The purpose of this robot was purely scientific, allowing for its manufacturer, Android Technics, to simulate and replicate the movements of an operator. Seeing that Factory Reset’s player character was to be some sort of experimental magnetic android, the choice of adapting FEDOR to better fit the world of Factory Reset began.

---

**Figure 92:** Original concept art for various types of non-humanoid FyED-OR robots. The team eventually settled on the centered variant.

---

8 Russian: Фёдор. FEDOR is the Russian given name and backronym for “Final Experimental Demonstration Object Research” (Russian Foundation for Advanced Research Projects, 2017)
Drawing inspiration from this real-world robot, the team developed their player character, would eventually be named FyED-OR, or Фёдор in cyrillic. FyED-OR’s conceptualization and concept art stage of the character pipeline followed closely to the real-life counterpart of FEDOR. Adjustments were made such as to better situate FyED-OR to the world of *Factory Reset*. Elements such as large horseshoe-shaped magnetic hands, in-line nuclear reactor, and several optics, allowed the character to look like it existed in a true environment, but more importantly keeping it true thematically. It was also chosen that FyED-OR would be a bipedal robot as opposed to the other forms of robotic locomotion as aforementioned, as the team figured most players would naturally understand the form of locomotion attached to those with humanistic and bipedal traits. A revised sketch of this character concept is shown in Figure 94.
Figure 94: Another iteration sketch of a humanoid FyED-OR.
5.5.2. Modeling

![Figure 95: Screenshot from Meshlab showing the final version of the player character FyED-OR.](image)

Before modeling could begin, the team’s artist divided FyED-OR into four segments: the head, the torso and body, the arms, and finally the legs. They modeled each element separately to allow for a more fluid character building pipeline. This was done in part to minimize the amount of large-scale iterations that would need to be made on the character - if a segment needed to be changed, the rest of the character could still remain exactly as it was in the scene instead of requiring a full re-import. This decision was also chosen for future proofing the game’s visual storytelling. Throughout level one, the player can find discarded pieces of past FyED-OR units, some intact; whereas others in shambles.

Throughout the modeling process, FyED-OR became increasingly detailed beyond the original concept art. One goal was to make the robot look more “friendly” and stray away from killer-robots such as the Terminator from 1984 American science fiction film by James Cameron, *Terminator*, as FyED-OR was supposed to be a simple and friendly magnetic robot. This was achieved by adding clumsy elements, such as loose dangling wires and cables, non-expressive facial features, and an overall graceless and gauche body proportion.
5.5.3. UV Unwrapping and Texturing

![Figure 96: Final version of FyED-OR painted orange.](image)

The character was UV unwrapped separately for each bodily component, continuing the idea of separating body parts in case of iterations and level set dressing. A core design objective when texturing was to keep FyED-OR looking non-hostile, so the color scheme had to represent something bright and happy. Originally, FyED-OR was painted blue (as seen in Figure 97); however, the team deemed the color blue to be a bit confusing since polarity colors were red and blue. Seeing this dilemma, the team eventually settled on the color orange (as seen above in Figure 96). This was done for two reasons:
Firstly, as was mentioned in section 5.4.4 and 5.4.6, the levels used clever texturing and lighting tricks to focus the player into looking in certain directions. Warm colors, such as orange, were located in areas the player weren’t supposed to go towards, whereas blue and teal (colder colors) were important to grab the player’s focal attention. Following the color theory of complementary pairs, if the areas where the player were supposed to go were colder colors such as blue and orange, then it would make sense for the player to be orange. Not only that, but as illustrated in section 5.4.4, this color pair works with any form of dichromatic color blindness. Although relatively subtle with the first-person perspective used in the game, this helped the player stand out in environments they were supposed to head towards due to the contrast between the player character and the environment.

Secondly, orange as a color influences humans to be more cautious. The color orange is understood as a signal word which conveys a natural emotion of caution and hazard (Braun et. al, 1993). During a study by Prof. Braun and his team realized that unlike colors such as red, orange didn’t convey a feeling of lethality, but a composite variable of perceived warning. Since FyED-OR is presumed to be an experimental logistic android, the team figured it would make sense if the color scheme matched that roll, much like construction and real-world logistic equipment such as traffic cones, caution signs, and high visibility vests.
5.5.4. First Person View Model

During the development of *Factory Reset*, there was a constant back-and-forth between whether the game would be in third person or in first person. Despite having a fully modeled and rigged character, the team decided it would be best to have the game take place within a first person camera view. There were many reasons why; however, the most notable is that the majority of the game takes place in closed and confined spaces. Having a large third person character model and a pulled out third person camera, would make the spaces feel even more confined.

Therefore, work began on the first person view model. Fortunately, because the team’s artist modeled the player character in bodily segments, it was straightforward to integrate the arms onto the player in *Unity*. The team then converted this into an adaptive and fluid first person view model. The left and the right arm of FyED-OR were taken and attached to the first person camera through hidden objects that served as joints. These were positions to be readily visible at all times through the camera. These arms were then animated to interact with the environment, as summarized at the end of section 4.3.1. When the player character walked, the arms would move in opposing cyclical loops in order to simulate a walk cycle. If the player engaged the magnetic features of the hands, they would shift forwards and lock into position while glowing with the appropriate polarity color. When interacting with a switch, the player’s right hand would briefly reach out to press it. If the player looked around, the hands would subtly shift to match as if leading the camera. And lastly, in order to prevent the arms from clipping through geometry such walls, the view models would be independently pulled back away from the obstacle in question while detected to collide with it. Figure 98 shows the hands in the game with the positive polarity activated.
Figure 98: A screenshot from *Factory Reset* showcasing the first person camera and view model.
6. Audio

This section covers the audio in the game. It focuses on the inspirations for the soundscape, the tools used, and specific breakdowns of the sound effects, music, and voice acting. To enhance the atmosphere of Factory Reset, the team created and implemented a cohesive and complete soundscape. This included sound effects for the player, contextual sounds for the environment, music tracks and voice acting.

6.1. Inspirations and Aesthetic

Because of the nature of creating sounds that do not exist in the real world, the team looked mostly to other games for inspiration, while also considering certain aspects of the actions in the game that would have real-world analogues.

6.1.1. Basic Computer-Generated Sounds

Because of the focus on technology and robotics, inspiration was taken from artificial computer-generated sounds. Specifically, the team used basic electronic sounds like sine waves used as the basis for many of the sounds in the game. Electronic music commonly uses manipulation of basic waves like the sine wave to create interesting tones (Sievers, 2006). Because these sounds do not occur in their pure forms in nature, they sound fairly artificial, which was perfect for the soundscape the team hoped to create.

6.1.2. Portal

As one of the main inspirations for the game overall, Portal (Valve, 2007) also influenced the soundscape. Portal (Valve, 2007) had a number of sounds that did not exist in real life, such as the sound of a portal gun firing. The team took inspiration from these player sounds to create the sound effect for the player’s actions. It also included speakers that gave the player exposition, which inspired level one’s PAs, which spout general safety warnings into the environment.
6.1.3. Music Inspiration

When creating the music for *Factory Reset*, the team aimed to combine the heavy synthesizers that are commonly associated with cyberpunk and sci-fi with the guitars of Russian post-punk. By combining the two, a new musical texture was created that was indicative of Sovietpunk. Once again, *Portal* (Valve, 2007) was used as an inspiration, as it has relatively subtle music tracks that work to enhance the atmosphere rather than underscore action. After listening to a lot of Russian post-punk, the bands Эйфория and Побег inspired the game’s music the most. The two most influential albums, *Я люблю тебя, возвращайся домой* (Эйфория, 2019) by Эйфория and Побег (Побег, 2018) by Побег, can be seen in Figure 99. They had steady but powerful guitar tracks with slight distortion, which the team ended up using as inspiration for the music in *Factory Reset*.

![Figure 99. Album covers from Я люблю тебя, возвращайся домой (Эйфория, 2019) by Эйфория (left) and Побег (Побег, 2018) by Побег (right) which inspired the music for Factory Reset.](image)

6.2. Tools

To create and edit the audio for the project, the team used the digital audio workstation *Reaper*. Most audio was made using sounds from freesound.org which the team edited to produce the desired effects. The team chose to use *Reaper* because they had prior experience with it and felt confident using it to create audio. The team mainly used the plugin *Bias Amp 2* for guitar fx and *Dexed* for synthesizer sounds. The versions of these plugins that the team used were free and powerful, allowing for the team to dial in the tone they wanted.
The guitar used for the music was an Epiphone ES-339, which provided a nice hollow-body sound. To capture output from the guitar, the team used a Focusrite Scarlett Solo audio interface. This allowed for clean recording of the raw guitar audio that could be monitored and manipulated in real time to better facilitate creation of interesting music.

6.3. Sound Effects

Many of the sound effects in the game were difficult to conceptualize because they either did not have real-world analogues, or the real world versions would not work with the game’s atmosphere. For example, the only real-world analogue for the sound of the player’s magnetic feet pulling to a surface is the low magnetic hum of powerful magnets, which does not have the punch that the team wanted to associate with that action. One commonality between all the sounds was that they were all either electronic in nature or highly edited to sound that way. This better sold the idea of all the futuristic technology of the world.

6.3.1. Feet

The magnetic feet have two sounds associated with them: one for when the player attracted themselves to a surface and one for when they released themselves from a surface. This was one sound that was especially hard to imagine, but the team knew that they wanted it to have some punch since it was one of the main sounds, but not be so involved that it would become annoying. The first version of the sound was implemented for the alpha build of the game. It was a powerful bass drop noise with a bit of sustain which helped it feel like it was having an effect on the environment. This sound was well received, but the team felt it was too much, especially for a sound that would be heard so often.

For the second and final iteration of the foot sound, the team used a highly edited bass tone. By applying reverb, delay, and a flanger, seen in Figure 100, the team created a sound with powerful attack, interesting decay, and a subtle enough overall timbre that it would not be grating after hearing it for an extended amount of time.
6.3.2. Hands

The player’s magnetic hands required a sound that could be looped, as they could potentially be left in an activated state indefinitely. The team ended up making a loop using two modified sine waves at different pitches for each of the two hand polarities. Since the sine wave is a basic building block of electronic sound and music, it fits into the game very well. During playtesting however, several players reported being annoyed by the sound, with one tester remarking that “the constant magnetized humming [was] a bit much.” This happened because instead of only turning on the hands when they wanted to interact with objects, they had left the hands on and the sound was playing constantly. Rather than update the sound, the team made holding the button the default to reduce the likelihood of this happening accidentally as well as reinforce the intended use of the mechanic. The team also added adjustable volume sliders for sound effects that would help reduce the harshness of the hand sounds if the player chooses.

6.3.3. Environmental Sound Effects

The team created a number of contextual environment sounds for the game as well. In the first level’s trash compactor the team used some servo noises, paper crinkling, and the sound of a fist slamming into a desk to create the sound of the trash compactor closing in, only to be stopped by some errant debris. Also in this level were buttons which played a simple two tone beep when pressed. The central elevator in the first level also had a looping sound associated with it, which played while it was ascending up the shaft. Finally for that level, the flying car that
pulls up at the end had an associated sound which was made using audio from a real car that was edited to sound more airy and futuristic.

In the second level, the team used environmental sounds to make the flying cars feel more authentic. Using the flying car sound from the first level, the team created an idle loop that could be applied to these cars. To make the sounds less overpowering, spatialization was used so that the sounds would only be audible from a certain distance. That way, the player would only be able to hear the cars when they were close enough to interact with. Figure 101 shows the graph for the rolloff for one of the cars.

![Figure 101. Rolloff graph on a Unity AudioSource component which dictates its volume at certain distances.](image)

Additionally to add some variety, the team used pitch randomization. Figure 102 below shows the simple code that was used to randomize the pitch of the cars upon their creation. Because a custom range could be inputted, the team made it so that larger trucks made lower sounds while standard cars made higher sounds. This contributed authenticity to the overall soundscape of that level.
In the third level, a contextual sound was created for when the power cells were slotted into their proper places. The team created this sound with a stapler’s “ka-chunk” and a little synthesized tune to convey the physicality of the action and the mechanical response. This gave players feedback for their actions, rewarding them for a job well done.

6.4. Ambience

The team wanted to create a sense of mystery and tension in the game through the atmosphere, as well as a little bit of wonder when the player uses their powers to fly around. To set this tone, the team used a combination of ambient noise and music.

6.4.1. Noise

For the second playtesting build, the team implemented an ambient noise for the first level. This noise was a low tone that very slowly phased back and forth. While not very involved, it helped to add some character and life to an otherwise empty environmental soundscape. In the final build it still exists, but evolves into the music part way through.

6.4.2. Music

The team made two music tracks for Factory Reset: “Where am I?” and “Don’t Fall, FyED-OR.” “Don’t Fall, FyED-OR” was the first track to be made, and was created for the second level, in which players travel through a busy skyway of flying cars. Since this level carried with it an inherent sense of wonder, the team wanted to capture that in the music. Taking inspiration from the guitar of Russian post-punk, the team used a delayed guitar with a steady Amaj7/D/E pattern as the foundation of the track. As it progressed, more instruments were layered on top, including a low synthesizer, light cymbals, a high synthesizer, and a kick drum. This resulted in a three and a half minute loop that plays uninterrupted during the second level, the final timeline for which is shown in Figure 103.
The second track to be made, “Where am I?”, plays during the first and third levels after certain conditions are met. Less elaborate and more moody, this track is meant to convey the mystery and tension the player feels while exploring the facilities in the aforementioned levels. Figure 104 shows the timeline for “Where am I?”, on which one can clearly see the three different parts where new instruments come in, which in actuality are separated out as loops in the game. As the player progresses through the level and pushes buttons, the ambient noise mentioned in section 6.4.1 slowly fades into just the slightly-distorted guitar of this track, with a hefty amount of reverb. Then, once another condition is met, the growling bass synthesizer fades in, and then after that a final simple drum track. This gives the effect that in the first level, as the player climbs the elevator shaft, the music builds up. In this third level a similar thing happens, though only near the end of the level, playing music just as the player takes their final elevator ride downwards.

6.5. Voice Acting

*Factory Reset* contains only a few lines of dialogue, which were recorded in asynchronous audio recording sessions. These voice lines were found in specific areas of the
game such as a Russian PA announcement in the first level and OLOF-14’s voice lines in level three. The team wanted the recordings to be spoken in native Russian, not text-to-speech Russian, to keep the game’s environment feeling more natural. Since nobody on the team spoke Russian, voice acting was outsourced abroad to three native Russian speakers that were personal friends of one of the team members: Skyler Emery, based in Moscow, the Russian Federation; Anastasia Nastya, based in Chișinău, Moldova; and Kristinia Golina, based in Kiev, Ukraine. Because the voice acting team was based in Europe, the recording sessions were done asynchronously. This proved to be a challenge as recording in different locations creates variability in terms of sound quality because of varying recording setups. Some microphones were better suited for video game voice recording than others. Fortunately, these voice lines are muffled and scrambled to better suit the environment (especially for OLOF-14), and with audio effects the team made them sound natural.
7. Testing

To gather accurate information about the game, the team developed a thorough testing methodology consisting of asynchronous surveys and supervised playtesting sessions which were distributed periodically throughout Factory Reset’s development.

7.1. Testing Methodology

The team’s main goal throughout playtesting was to ensure that Factory Reset was meeting its experience and design goals related to awe, intrigue, satisfaction, and accessibility. By getting feedback from players on the game while developing it, the team was able to make changes to improve the quality of the experience.

Since this project was completed during the COVID-19 pandemic, in-person playtesting was not feasible to do safely. Because of these limitations the team decided to use surveys distributed over the internet as the main source of obtaining public playtesting feedback. Many large game studios use surveys as a way to get bulk feedback on games since they can be easily distributed, filled out, and analyzed. One benefit to this approach was that it allowed participants to play the game on their own machine, just like how they would be playing the final release. This also allowed the team to test for any issues the game had with customized setups designed to fit the needs of the player. The team developed the surveys with the questions specifically tailored to give feedback on the issues the team was interested in, particularly the quality of the overall game experience and the accessibility features.

The team recruited playtesters from outside the WPI community with promotional posts on Reddit and Twitter, as well as reaching out on various Discord servers. Reddit is a popular social media forum platform where users can post, view, and interact with others. The list of subreddits posted on can be seen in Appendix H. Sample Reddit posts can be seen in Appendix K. Twitter is a popular short form social media platform where users can post, like, share, and comment on posts. Sample Twitter posts can be seen in Appendix L. The Discord servers the team reached out on are listed in Appendix H.

In order to ensure participants had given proper consent before taking part in research, the team had both a screening form and a consent form they needed to complete before playtesters were able to download the game or access the feedback surveys. The screening form
warned players about any special considerations or potential risks associated with playing the game, and is included in Appendix B. The consent form was how players confirmed that they consented to participate in the study. There are two versions of the consent form, one for public unsupervised testing and one for supervised testing. These are included in Appendices C and D.

Figure 105 shows the timeline of playtesting rounds that the team conducted. The first playtesting round was in November 2020 during WPI’s Alphafest event. During this event, students and faculty from WPI playtested the game either privately or while sharing their gameplay screen over a video call while members of the development team taking notes. Afterwards, the participants filled out a survey to collect feedback on the game mechanics and experience, which is summarized in Appendix E.

The team conducted additional playtesting within the WPI community over scheduled Zoom meetings. Playtesters signed up for thirty minute playtesting sessions run by members of the development team. Sign-ups were run through the online service Sign Up Genius. Once signed up for a session, playtesters received an email confirming their session, shown in Appendix J, and the playtesting instructions document included in Appendix I. This document instructed players to fill out the pre session questionnaire, download the game files, and familiarize themselves with the included readme. During the session, playtesters could choose between communicating verbally or non verbally with the developer and receiving verbal and/or
written instruction from the developer. Playtesters also could optionally allow the gameplay video and audio from the session to be recorded.

The sessions were broken into three parts. In the first section, lasting five minutes, the developer confirmed the consent of the playtester and delivered instructions in the preferred format. In the next section the playtester played the game for fifteen uninterrupted minutes while the developer observed. The playtesters were encouraged to talk out loud during this portion of the session so the developer could understand the playtesters' thinking. During this period developers remained as silent as possible to avoid influencing the playtester. In the third and final section of the sessions, playtesters participated in a discussion about their experience with the developer. After their discussion the Zoom session was ended, and playtesters were instructed to fill out the playtesting survey.

7.2. Testing Results

At each round of playtesting the team gathered the results and made changes to the game based on feedback.

7.2.1. Alphafest

The team collected a large amount of valuable gameplay feedback from this first round of playtesting. This was the first time the team had the game in a state where other people could play it, so it was particularly useful for identifying areas of frustration and places where the team had made incorrect assumptions about their design.

The mechanics and design needed some fine-tuning and modification. While players were technically able to complete the level and praised some aspects of it, like having an intuitive path, it was not without frustration. There were reports of disorientation when pulling onto surfaces with the magnetic feet ability, as well as complaints and confusion about the orientation and gravity resetting when colliding with obstacles while pulling to a surface. Players found the ability to step off of polarized surfaces and automatically return to the effects of regular gravity to be frustrating. These issues were particularly frustrating because the vertical nature of the level design meant that falling unintentionally meant the players would have to climb all the way back up the level again. Some players also felt that the hands were under-utilized, though the team did observe that it was remarkably fun to throw polarity boxes
around using the player’s magnetic hands, so the team made a point to consider leaning into that for future levels. There were also a few reports of collision issues in particular places, an unintended gameplay sequence break, and players getting pushed through floors when objects collided with their heads.

It became clear from playtesting that the team needed to rework the visual indicators on the reticle, which was evidently too abstract and unclear from a glance. Players consistently confused the foot and hand polarity indicators and left the hands continually on, with the sound effect for them looping. This was exacerbated by the fact that they used the red and blue hand polarity colors despite the feet functionally being independent of this. The markers for when an object could be affected by an ability or interacted with were also difficult to notice, especially against bright surfaces. Although the controls were described in the included README file, the lack of symbolic images accompanying input prompts was also noted as an issue to address. Players also requested more clear ways to tell their orientation, like clear environmental indicators or a user interface element, so it was clear that removing the orientation compass hampered the game experience. One tester in particular completely failed to understand most aspects of the heads-up display, which highlighted its problematic lack of intuitive clarity.

23 people filled out the post-play survey for Factory Reset’s Alphafest build. From the survey data the team assessed what was and was not working. The team’s likert scale questions gave insight into how much players understood certain mechanics. As can be seen in Figure 106, the mode of responses for the magnetic feet was on “somewhat agree” and for the magnet hands, it was on “strongly agree.” This indicated that the mechanics were altogether easy to understand, though there were some testers who indicated that they did not understand the mechanics.
Figure 106. Survey results for how much participants agree with the statements “I clearly understood how to use the magnet boots” and “I clearly understood how to use the magnet gloves.”

Not all the feedback at this stage was positive however. Figure 107 shows that a majority of players either agreed or strongly agreed that the movement in the game was disorienting. While some disorientation was part of the idea behind the team still wanted to minimize player discomfort.

Figure 107. Survey results for how much participants agree with the statement “Movement in the game was disorienting.”
Another piece of negative feedback came from a player reporting that “as a person with low vision, [the subtitles] were too small.” In future iterations the team made the size of subtitles adjustable. A final result which was observed was that when pulling to a surface, players would frequently bump into parts of the environment, causing them to fall out of the animation. This made the game very frustrating as many moves were much more precise than they ought to be. This was fixed in the next build by revising the feet system to be absolute and ignore collisions with the environment.

Finally, the team received a good amount of feedback on the environmental design of the game at this stage. Multiple playtesters commented positively on the “very polished environment.” One tester noted that “the glowing icons stood out really well, making it very clear where to go and what could be interacted with.” Having lighting guide the player to their destination was one of the team’s goals with the visual design, and it is clear that it worked for this playtester.

7.2.2. C Term

At the end of the third quarter of the school year, the team finished the core game content and entered the phase of iteration based on playtesting feedback. In preparation for public playtesting, they worked to address this feedback and implement some additional changes from the team’s wishlist in order to maximize the number of players who could play the game. One vital change at this stage was the addition of multiple resolution settings, as the game was rather taxing to render and resulting in a large amount of lag on some computers. By offering the ability to change the screen resolution, the team made it possible for people with a wider range of computer strengths to reasonably play the game. This was useful even for internal testing, as the game was problematically slow on some of the developers’ computers. The team also reworked the mouse look sensitivity settings, as it tended to default to be frustratingly high, had some inconsistencies between the vertical and horizontal axes of motion, and the ranges of the sliders controlling sensitivity were biased towards making the sensitivity even more extreme instead of toning it down.
14 people played the Beta 0.1 build of *Factory Reset* and responded to the survey. One key result from this round of playtesting was that movement was still fairly disorienting. A majority of testers disagreed with the statement “movement in the game was NOT disorienting,” and while most testers did not experience motion sickness, at least one person did as seen in Figure 108. To help counteract this disorientation, the team added a compass overlay to the following build which always indicated which direction was down. To help combat motion sickness, the team added an option to have the screen fade out during the camera rotation when pulling to a surface.

![Chart 1](chart1.png)  
**Q1_6** - “Movement in the game was NOT disorienting.”

![Chart 2](chart2.png)  
**Q1_7** - “I did NOT experience motion sickness while playing the game.”

*Figure 108.* Survey results for how much participants agree with the statements “movement in the game was NOT disorienting” and “I did NOT experience motion sickness while playing the game.”

Another major finding from this playtest was the issue of physics objects pushing the player character through the floor and walls, which several players cited as a frustration with the game. The team fixed this in the next build by stopping the player from pushing themselves through the ground or flying upwards on physics objects.
Art feedback in this round of playtesting remained positive, with one tester calling out the newly added second level in which you travel along flying cars “especially interesting.” Another tester remarked that they liked “the high amount of contrast between the dark surfaces and neon lights,” which once again ties into one of the team’s goals with the art direction of Factory Reset. Finally, multiple testers noted that the game seemed especially dark for them. This turned out to be an issue with the customizable resolution settings which was fixed for the following iteration.

7.2.3. D Term

In both Beta playtests, they survey asked players what settings they modified. Figure 109 shows the results of asking playtesters which settings they modified in the C and D term playtesting. One issue that the team ran into was with discoverability of settings; only two participants modified settings in the Beta 0.1 playtest. To increase the discoverability of the settings the team showed the settings menu upon booting up the game before players started. After this change, 18 settings were modified across 28 playtesters, compared to the 2 settings modified across 14 playtesters in C term. The team believes this increase in settings modified to be indicative of an increase of the discoverability of the settings.
The issue with playtesters being disoriented while playing persisted into the final iteration. As shown in Figure 110 below, a majority of playtesters reported that they found the game’s movement disorienting. Thankfully, just like in the C Term playtesting results, a majority of players reported that they did not experience motion sickness while playing. This was after the team added the option to hide the pulling animation with a fade to black, but there was no way to tell through the survey whether or not this had an impact on players.
One area where the game had major issues was its performance. On lower-end to mid-range computers, computers without dedicated graphics cards, and 4k screens, performance was greatly reduced, with players reporting that there was a “low framerate,” and that it was “lagging with even the lowest graphics.” This was even worse in the second level, where some players reported that they “couldn't play [the] second level because even with the lowest resolution [their] game would play at 3 fps.”

The team also received feedback on the menu’s noises, with one user reporting that they were “very loud.” Because of this, the team reworked the menu sound multiple times to use a different sound that was more pleasurable. The team decreased the decay of the sound so that it would only play briefly, and made it more interesting by changing the wave from a sine wave to a sawtooth wave.
One tester remarked the following: “Subtitles were tiny! I think they ought to be bigger by default.” This was a good point, as changing the resolution did not affect the size of the subtitles even though it should have. Having customizable settings is good, but the default settings should be such that the largest number of people are satisfied with them.

In this build the team also received a number of comments about the look sensitivity. One tester remarked that “the lowest [look speed setting] was not nearly low enough.” This led to the team changing the range of look speed values to encompass a wider range so that anyone would be able to set the sensitivity to a setting they were satisfied with.

Art feedback was once again positive for this iteration. Not many playtesters gave actionable feedback on the art. One tester noted that “the environment was on the darker side, which allowed the red and blue of objects that you can interact with stand out more.” This tied into the team’s goal of guiding the player with the brighter areas of the environment.

Similar to the Alphafest results, the majority of players indicated that they understood the feet mechanic and hands mechanic, although a few more players indicated ‘disagree’ than in the previous rounds of playtesting. Despite this, the team observed through live sessions that players were misinterpreting how to use these mechanics and many playtesters thought they were required to be used together. The full spectrum of responses is shown in Figure 111.
During the live sessions, the team observed that playtesters had a tendency to activate the corresponding hand color before attempting to use the feet to pull themselves to a surface. This led to significant confusion for some players. “Did I just use a negative to pull myself to this negative[?]” one player asked. This use of the mechanic also caused players to be confused when polarized objects attracted to them while they were attempting to pull to a surface.

**Figure 111.** Graphs showing how much players agreed with the statements “I clearly understood how to use the magnet boots” and “I clearly understood how to use the magnet gloves.”
8. Discussion

This section covers the team’s conclusions and experiences from working on this project as well as future plans for Factory Reset.

8.1. Experimental Design Challenges

The team encountered a number of challenges with the design of the playtesting surveys. Specifically, one thing that the team wishes they had kept track of was where playtesters were from. Since the final round of playtesting included outreach beyond the WPI community, it would have been nice to know how many external playtesters the team received. Also, data from these testers may have been interesting to analyze given the fact that they were not playtesting for any reason other than simply to play the game. This could have been solved by adding a question asking whether or not testers were from WPI.

Another thing the team wishes they kept track of was repeat playtesters. The team did not make it mandatory to enter an email with playtesting, but doing so would have been nice purely to see how playtesters’ opinions changed across iterations.

The final issue with the survey was that it was mainly designed to inform the team on what they needed to change and add in subsequent builds, not to evaluate how well Factory Reset had met its goals. The issue with this was that when it came time to evaluate the success of the game, the team did not have many metrics that were specifically designed to do so. Part of this is because the team had originally wanted to playtest with more gamers with disabilities. Partially because of the lack of identifying information collected in surveys and partially because of the difficulty of getting testers with no funding, the team was unable to do so. The team feels that if they were able to test with this demographic, they would have been able to more accurately assess Factory Reset’s success in creating a game that is accessible to the widest possible audience.

8.2. Accessibility

The design of an accessible game and research study proved to be a significant challenge in the development of the game. Accessibility terminology and standard development practices
are constantly evolving and finding consistent development recommendations proved difficult. Although the team was able to include a significant number of accessible features in the final build there could always be more features that accommodate more players.

8.2.1. Gameplay Accessibility

The team found that the magnetic feet mechanic, which involved reorienting the player, presented many challenges when it came to accessibility. Despite the team’s efforts, playtesters in every iteration reported that they were disoriented when using the magnetic feet to pull to surfaces. The nature of the mechanic made disorientation nearly impossible to avoid, which speaks to the struggle of designing a mechanic that challenges gaming conventions while still making it accessible. Luckily, only 3 out of the 21 playtesters from the final round of playtesting reported any amount of motion sickness, which the team had aimed to counteract by blocking the turning part of the pulling animation with a brief fade to black.

8.2.2. Hardware Requirements

The team fell short of making Factory Reset universally playable because on some players’ computers the game ran at a very poor framerate. Device-related performance was a major barrier to accessibility for many playtesters. The team came to this conclusion based on the numerous comments from playtesters and the issues the team encountered during development of the game. It didn’t matter how many accessibility settings the team added to aid with the gameplay if people could not even run the game. This was something the team overlooked until later builds of the game. Figure 112 shows the results from running Unity’s profiler on each of the three levels. They show that the majority of processing load is coming from rendering the game. With more extensive graphical settings, the team hopes to be able to allow the game to run on computers with less processing power.
Figure 112. Results of running Unity’s built-in profiler on levels one, two, and three respectively.

8.3. Design Goal Evaluation

One of the team’s initial goals for the project was to create a game that surpassed the expectations of a “student game,” with polished, fun, and engaging gameplay. While this is largely a subjective quality, the team can ascertain their success from the commentary received from players. Playtesting feedback surveys were geared more towards evaluating and refining the clarity and effects of Factory Reset’s gameplay mechanics than collecting direct data about player’s impressions of the game, so the only points of information on this subject come from observations and comments made during supervised playtesting sessions or provided in text form on surveys. Also, the vast majority of playtesters were within the WPI community or knew the team personally, which may have introduced some bias into responses.

Nevertheless, players did demonstrate that they enjoyed the gameplay concept of Factory Reset. Figure 113 below shows a word cloud of the ten most common words or phrases from responses to the question “what did you like most about the game?” in D Term. This shows that most players praised the game’s concept, mechanics, and levels, with players remarking things like “the mechanic is unique and it is implemented rather well,” and “[the] main mechanic of using polarity to move objects and traverse the level was interesting.”
Figure 113. Word cloud of the 25 most common words from responses to the question “What did you like most about the game?”

The biggest negative observation the team noticed regarding gameplay polish was player frustration. Figure 114 below shows a word cloud of the ten most common words or phrases from responses to the question “what did you dislike most about the game?” in D Term. The word cloud shows many players remarked that the game was disorienting and hard. One playtester remarked that “the switch in perspective when you use the magnet boots was super super disorienting.” Additionally, not captured in this word cloud, 7 out of the 28 playtesters reported technical issues of lagging, crashing, or both.
Another potential metric for polish is how many significant bugs are readily-discovered in the game. The only major bugs the team observed in the final round of playtesting were players being able to pull onto the sides of polarity pads and, as a result, sometimes move in unintended ways like walking on walls or getting stuck on the pads.

Continued research on the game could further evaluate player reactions with targeted questions for gauging their enjoyment, engagement, and emotional responses. This form of testing would be most effective with anonymous public players from outside the WPI community, as they would have minimal pressure to bias their responses and evaluate the game unfairly positively.

8.4. Technological Challenges

Despite all of its advantages, the team had some problems with Unity HDRP. Because some shaders used in Unity HDRP are not supported on Ubuntu, which one of the team members was running on their only computer up through the Alpha stage of development, that team member was unable to directly modify the Unity project after the switch to HDRP. They were able to work on scripts, exportable Unity objects, and audio assets, but having to go through another team member whenever they wanted to add something to the game greatly slowed their contributions leading up to and following Alphafest. Eventually, they got a new computer that ran Windows 10, allowing them to contribute directly once again.

One of the other team members began experiencing frequent, persistent crashes while working on the Unity HDRP project. While these crashes were not so frequent that the software was impractical to use, they were significant enough to reduce productivity somewhat and the team never identified their root cause.

The team also ran into issues with merging large changes to the project using GitHub. While the individual files were generally small enough to transfer fine, the team ran into issues with large pushes of art assets. Github has a push size limit of 100MB, and due to the large size and large quantity of the Factory Reset’s texture files, the team had to perform a large number of pushes containing a few files at a time to get them into the games repository. This was a very
time consuming task and Github would often fail a push after several minutes of progress, which slowed progress at some points.

8.5. Artistic Challenges

Figure 115: An example showing the clear disconnect between level elements texture resolution within the modular system that could only be addressed with increasing texture resolution manually.

Throughout artistic development of *Factory Reset*, the artist ran into a few challenges. One of the largest areas worth addressing is an inconsistent texture resolution for all the level assets, including the new modularity system put in place. In several parts of the levels, there are areas where textures look much more lower resolution than in other parts of the same level. In certain cases, such as the below image the difference between resolution would be very drastic.

Figure 116: Screenshot from the game showcasing the high resolution texture work for the door switch prop, and the much lower resolution level element wall.
This inconsistency exists as a legacy of the former pre-modular level elements carrying over into the modular level elements. This is because the artist had to manually convert the pre-modular elements to become modular as described in Section 5.4.3.2. Had the level elements been developed originally to be modular, this could have been moderately avoided through better and more efficient UV texture wrapping, as can be seen above in Figure 115 and Figure 116.

Another worthwhile challenge was the Factory Reset team only had a single artist in charge of creating the majority of the visual assets. Having a single artist on an ambitious project created many scope issues and timing constraints. One of the largest time constraints that took the artist a majority of time was the conversion of pre-modular level elements to the current modular level elements. Each piece had to be remodeled, remapped, retextured, and readded to the levels manually, which took time away from the artist to create new props, assets, and environmental world building. Having a large team of artists would have allowed for faster asset creation and the ability to reach scope goals.

Another obstacle with having a single dedicated artist was that the artist had to cover many areas of artistic expertise within the artistic development pipeline that they may not feel the most comfortable with. For instance, the artist primarily focused on 3D modeling, texturing, and world building; however, was not trained for concept art and preliminary visual design throughout the early conceptualization stages. The project would have benefitted from a more diverse team of artists who could individually focus on their respected areas of strength, as to bolster the overall quality of the artistic pipeline.

8.6. Looking Forward

The team intends to release a final build of Factory Reset beyond the time scope of this project. This build will be intended to address the game’s performance issues, improve the game controls, and add additional accessibility features. The main control change the team plans to implement is to add a dedicated button to disengage the magnetic feet. Along with this, pressing the button to activate the feet without a valid target would now cause no change instead of deactivating them. This is an effort to reduce player confusion and frustration, as in playtesting the team consistently observed players falling accidentally when trying to activate the feet without properly aiming at a target in range.
This version of the game will be released for free on the itch.io marketplace, with a targeted release date by the end of May 2021. The release will be free because Factory Reset was developed as an academic project, takes only 10 to 20 minutes to play, and the team developed it using software for free in an academic capacity. The main method of promotion for the game will be through the team’s Twitter account, as it is where they have cultivated a follower base. The team intends to use this project as a portfolio piece that demonstrates their capabilities.
References


CD Projekt (2020) Cyberpunk 2077 (PC version) [Video game]. CD Projekt RED.


Эйфория. (2019.) [Album cover for Я люблю тебя, возвращайся домой] [Photograph] *Spotify*. Retrieved from https://open.spotify.com/album/1oqNq3mM5uJagVIxtOwe5x


IGDA-GASIG. (n.d.) The IGDA Game Accessibility Special Interest Group. IGDA-GASIG. Retrieved from https://igda-gasig.org


Nintendo (2017) The Legend of Zelda: Breath of the Wild (Switch version) [Video game]. Nintendo EPD.


https://i.redd.it/z58fu4jmar541.jpg


Sony Interactive Entertainment (2020) *The Last of Us Part II* (PlayStation 4 version) [Video game]. Naughty Dog.

Steam (n.d.) [active Portal reticle] [Photograph]. Steam. Retrieved from 
https://steamcommunity.com/sharedfiles/filedetails/?id=163045846

stevewhims, joannaleecy, msatranjr, ScottMunroMS, KevinAsgari, mijacobs, &amp; DCtheGeek. (n.d.) *Making games accessible*. Microsoft. Retrieved from 
https://docs.microsoft.com/en-us/windows/uwp/gaming/accessibility-for-games


The Game Bakers (2016) Furi (PC version) [Video game]. The Game Bakers.


505 Games (2020) Control (PC version) [Video game]. Remedy Entertainment.
Appendix A: Project Website

Potential playtesters will access our game and its associated survey materials on this website: https://rfdolan.github.io/factory-reset/index.html

Home
Discarded with a factory facility's scrap, the robot FyED-OR awakens with a single directive: return to their pre-programmed home. Wielding their magnetic limbs, they must push and pull their way through puzzles in an enigmatic cyberpunk Soviet facility.

>Click here to playtest our game!<

Updates
4-7-2021 Beta Build 0.2 & Playtesting!
We have created our second beta build of Factory Reset and have begun another round of playtesting! We would love to see your feedback. If you have a Windows PC and would like to playtest, just go to this link and follow the instructions! Thank you!

3-17-2020: Beta Build 0.1 & Playtesting!
We have created a beta build of Factory Reset and have begun playtesting with it! We would love to see your feedback.

11-20-2020: Alphafest!
We created an alpha for our game for WPI's Alphafest and got to show it off! We got a lot of great playtesting feedback and are excited to incorporate it into the game.
Appendix B: Screening Form

Before being able to access the download or survey links for playtesting, participants will have to read and agree to the following screening form found on this website: https://rfdolan.github.io/factory-reset/playtest_screening_public.html

Playtest

There are some things you should know before playing our game:

1. Playtesting this game is voluntary. You are free to stop at any time for any reason.
2. The game will require 2GB of space to install. Currently only Windows is supported.
3. This game could be unsafe for players with photosensitive epilepsy or other photosensitivity-related conditions. Please stop playing and consider contacting a physician if you experience any unusual or concerning symptoms.
4. This game is played from a first-person perspective and features some potentially disorienting motion. Please stop or take a break if you experience significant symptoms of motion sickness.
5. We will collect your email address if you indicate that you are a WPI student who needs playtesting credit. This data will not be disclosed publicly, and will only be used to confirm your participation.
6. If you are under the age of 18, you must have a parent or guardian complete this form and the following Consent Form before playing.

[ ] By checking this box I indicate that I have read and understood the above warnings.
Appendix C: Informed Consent Notice - Public Testing

In order to take part in public self-driven playtesting, participants needed to first complete the following consent form:

Consent Form: Public Playtesting

Thank you for your interest in testing our game! We are a small team of college students from Worcester Polytechnic Institute who are working to create an enjoyable game that is widely accessible. To help us in this study we will ask you to:

1. Download the game from a Google Drive link.
2. Unzip the downloaded folder and run the .exe file inside to start the game.
3. Play the game for 5-15 minutes or until you decide to stop.
4. Complete the provided survey after you stop. This should take about 10 minutes.

Our game is a first-person puzzle game where you are a robot with magnetic hands and feet who must use them to move and solve puzzles. The movement in the game may disorient you, so if you feel dizziness or nausea at any point when playing please take a break or stop playing. The game is not expected to present any other significant risks. If you understand and accept the risks that come with testing this game, then feel free to proceed with playtesting!

[ ] By checking this box I indicate that I have read and understood the above warnings.
Appendix D: Informed Consent Notice - Supervised Testing

In order to take part in supervised playtesting, participants needed to first complete the following consent form:

Consent Form: Supervised Playtesting

Thank you for your interest in testing our game! We are a small team of college students from Worcester Polytechnic Institute who are working to create an enjoyable game that is widely accessible. To help us in this study we will ask you to:

1. Download the game from a Google Drive link.
2. Unzip the downloaded folder
3. Join our Zoom call using the link provided at your reserved time. You do not need to turn on your camera.
4. Play the game for 5-15 minutes or until you decide to stop. Share your screen while you do this, and if you are comfortable with it, narrate your thought process as you play. A study representative will take notes on your comments and gameplay but will generally not offer guidance unless asked.
5. Complete the provided survey after you stop. This should take about 10 minutes.

Our game is a first-person puzzle game where you are a robot with magnetic hands and feet who must use them to move and solve puzzles. The movement in the game may disorient you, so if you feel dizziness or nausea at any point when playing please take a break or stop playing. The game is not expected to present any other significant risks. If you understand and accept the risks that come with testing this game, then feel free to proceed with playtesting!

[ ] By checking this box I indicate that I have read and understood the above warnings.
Appendix E: Alphafest Playtesting Survey

Open Response
What did you like most about the game?
What did you dislike the most about the game?
What stood out to you about the environment?
List three words that reflect the narrative of the game.
What, in your own words, did the different visual indicators on the aiming reticle mean?

Likert
How much do you agree with the following statements?
(Rating Scale: Strongly Disagree, Somewhat Disagree, Neutral/No Opinion, Somewhat Agree, Strongly Agree)

"I clearly understood how to use the magnet boots."
"I clearly understood how to use the magnet gloves."
“It was easy for me to understand the layout of the level.”
“It was easy for me to understand how to move through the level.”
“The menu was easy to navigate and use”
“Movement in the game was disorienting”
“I experienced motion sickness while playing the game”

Accessibility
What did you use to control the game?
[x] Keyboard
[x] Mouse
[x] Gamepad / Game Controller
[x] Other [specify optionally]

Did you encounter any difficulties with the subtitles? If so, what?

Were there any other settings you would have liked to see? If so, what?

Optional
Do you have any other comments or suggestions?

Appendix F: Pre-Session Questionnaire

Please answer the following questions to the best of your ability. Your response will have no impact on your ability to participate in playtesting.

If you have any questions or concerns please email rfdolan@wpi.edu.
Q1 Are you comfortable with your play testing session being recorded?
   o Yes
   o No
Q2 Are you able to/willing to use your microphone for verbal communication during the play testing session?
   A "no" answer exempts you from verbally communicating during your play testing session
   o Yes
   o No
Q3 Please select the format you wish to receive instructions during your session.
   o Verbal
   o Written
Q4 Please enter the email address you used to sign up for your playtesting session.
Appendix G: Spring Playtesting Survey

This section of the survey covers general gameplay feedback.

---

Q1 How much do you agree with the following statements?

<table>
<thead>
<tr>
<th>Statement</th>
<th>Strongly disagree (1)</th>
<th>Somewhat disagree (2)</th>
<th>Neither agree nor disagree (3)</th>
<th>Somewhat agree (4)</th>
<th>Strongly agree (5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;I clearly understood how to use the magnet boots.&quot; (1)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;I clearly understood how to use the magnet gloves.&quot; (2)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;It was easy for me to understand the layout of the level. (3)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;It was easy for me to understand how to move through the level.&quot; (4)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;The menu was easy to navigate and use.&quot; (5)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;Movement in the game was NOT disorienting.&quot; (6)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&quot;I did NOT experience motion sickness while playing the game.&quot; (7)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Open Response Section

Q2 What did you like most about the game?

Q3 What did you dislike most about the game?
Q4 What stood out to you about the environment?

Q5 What are three words you would use to describe the narrative of the game?

Q6 What, in your own words, did the different visual indicators on the aiming reticle mean?

Page Break

Accessibility Section

This section of the survey covers feedback on the accessibility of the game.

Page Break

Q7 What did you use to control the game?

- Keyboard (1)
- Mouse (2)
- Gamepad / Game Controller (3)
- Other (please specify) (4) ____________________________________________

Q8 Which options did you use? Choose all that apply.

- Remappable controls (1)
- Subtitle settings (2)
- EDITOR’S NOTE: ADD THE REST (3)

Q10 Were there any additional settings or options that you would have liked to see or that you think would have aided you in your experience?

Q9 Did you have any issues while playing the game that impeded you progress, including...

- ...issues with the visuals? (1)
- ...issues with the sound? (3)
- ...issues with controlling the game? (4)
...issues with the difficulty of executing certain parts of the game? (5)
...issues with understanding where to go or what to do? (6)

**If:** Did you have any issues while playing the game that impeded your progress, including... = ...issues with the visuals?

Q9A What specific visual issues did you encounter? How did they impede your progress?

**If:** Did you have any issues while playing the game that impeded your progress, including... = ...issues with the sound?

Q9B What specific sound issues did you encounter? How did they impede your progress?

**If:** Did you have any issues while playing the game that impeded your progress, including... = ...issues with controlling the game?

Q9C What specific controls issues did you encounter? How did they impede your progress?

**If:** Did you have any issues while playing the game that impeded your progress, including... = ...issues with the difficulty of executing certain parts of the game?

Q9D What specific issues did you encounter with executing certain parts of the game? How did they impede your progress?

**If:** Did you have any issues while playing the game that impeded your progress, including... = ...issues with understanding where to go or what to do?

Q9E What specific issues did you encounter with figuring out where to go / what to do? How did they impede your progress?

Page Break-----------------------------------------------------------------------------------------------

Q11 Do you have any other comments or suggestions?
Q12 Are you a WPI student who requires playtesting credit?
   - Yes (1)
   - No (2)

**If:** Are you a WPI student who requires playtesting credit? = Yes

   Q12a If you are a WPI student who needs playtesting credit, enter your email below.

Q13 I would like to receive updates via email about the game.
   - Yes (1)
   - No (2)
Appendix H: Forums

We recruited playtesters from the following subreddits:

- r/DisabledGamers - https://www.reddit.com/r/disabledgamers/
- r/Gamedev - https://www.reddit.com/r/gamedev/
- r/GameDesign - https://www.reddit.com/r/gamedesign
- r/Games - https://www.reddit.com/r/Games/
- r/Playtesters - https://www.reddit.com/r/playtesters/
- r/WPI - https://www.reddit.com/r/WPI/

We recruited playtesters from the following discord servers:

- Game Accessibility (IGDA-GASIG)
- Reddit/r/gamedev
- WPI IMGD
Appendix I: Factory Reset Sign Up Play Testing Instructions

Before Your Session:
1. Fill out pre session questionnaire
   a. [Appendix F]
2. Download the game files
   a. [LINK]
3. Familiarize yourself with the README provided in the game files
   a. [Appendix M]

What to Expect During Your Session:
4. Enter the Zoom call at your assigned time
5. Confirmation of consent and session instructions (5 minutes)
6. Play game (15 minutes)
   a. If able, you will be asked to narrate your thoughts while playing
7. Discussion (10 minutes)
8. Leave Zoom call and fill out post survey
   a. [Appendix G]
Appendix J: Sign Up Session Email Template

Subject: Factory Reset Play Test Session
Location: {ZOOM}

This email is to confirm you have signed up to play test the MQP Factory Reset on _____BOLD DATE AND TIME_______ . The Zoom link to your session is below.

{ZOOM LINK}

BEFORE YOU SESSION: Please follow the pre session instructions found in [Appendix I].

If you have any questions or concerns please don’t hesitate to reach out to us at _____EMAIL____.

Thank you

The Factory Reset Team
Appendix K: Sample Reddit Outreach Post

Title: Factory Reset (Playtesters Requested!)

We are a team of students from Worcester Polytechnic Institute looking for playtesters on our senior project.

In Factory Reset, play as a lost robot wielding the power of magnetism on a journey back to their home factory. Manipulate the world with your polarized hands and walk on special walls and ceilings as you push and pull your way onward!

The game has also been designed to incorporate a range of accessibility options and considerations that we hope will help as many people as possible enjoy it.

If you’d like to playtest Factory Reset, click this link: [Appendix B]. We’d greatly appreciate it!

Please note that we will not be offering payment for this.
Appendix L: Sample Twitter Outreach Post

We’re looking for playtesters! If you would like to playtest our game visit [WEBSITE LINK] and follow the steps.

Thank you for helping us make our game the best it can be!
Appendix M: README

FACTORY RESET

A game by Christian Adler, Raymond Dolan, Elizabeth Kirschner, and Henry Stadolnik. This was created as part of our Major Qualifying Project at Worcester Polytechnic Institute.

Discarded with a factory facility's scrap, the robot FyED-OR awakens with a single directive: return to their pre-programmed home. Wielding their magnetic limbs, they must push and pull their way through puzzles in the facilities of an enigmatic cyberpunk Soviet city.

RUNNING THE GAME:
Download the zip file from google drive.
Unzip the file without re-ordering its contents.
Run the .exe file for Factory-Reset-0.5.exe.

PLAYTESTING SURVEY:
If you have the time, please take a few minutes after playing to take our playtesting survey!
For more details and the survey link, please visit https://users.wpi.edu/~rfdolan/Factory-Reset/playtest_screening_public.html

CONTROLS (Controller)
Left joystick : Walk
Right joystick : Look
Cross / A : Pull self to polarized surface / detach from surface
L2 / Left Trigger : Set hands to negative polarity / disable hands polarity
R2 / Right Trigger : Set hands to positive polarity / disable hands polarity
O / B : Interact
(These are remappable!)

CONTROLS (Keyboard/Mouse)
WASD : Walk
Mouse : Look
Spacebar : Pull self to polarized surface / detach from surface
Left Mouse : Set hands to negative polarity / disable hands polarity
Right Mouse : Set hands to positive polarity / disable hands polarity
F : Interact
(These are remappable!)

IF YOU GET LAG:
If the game is laggy when running, try setting it to a lower resolution through the Graphics settings menu.
Some computers run the game fine, but others may experience heavy lag on higher graphics settings. It is recommended to have a dedicated graphics card to play this game.
Please note that we are aware of an issue where lower graphics settings may turn the environment darker in full screen mode - running the game in windowed mode instead should fix this.