

2015-05-21

Before Eternity: An Adventure Game Inspired by Sufi Mysticism

Seyed Siavash Mortazavi Ravari
Worcester Polytechnic Institute

Follow this and additional works at: <https://digitalcommons.wpi.edu/etd-theses>

Repository Citation

Mortazavi Ravari, Seyed Siavash, "Before Eternity: An Adventure Game Inspired by Sufi Mysticism" (2015). *Masters Theses (All Theses, All Years)*. 832.
<https://digitalcommons.wpi.edu/etd-theses/832>

This thesis is brought to you for free and open access by [Digital WPI](#). It has been accepted for inclusion in Masters Theses (All Theses, All Years) by an authorized administrator of Digital WPI. For more information, please contact wpi-etd@wpi.edu.

Before Eternity:

An Adventure Game Inspired by Sufi Mysticism

by

Seyed Siavash Mortazavi Ravari

A Graduate Project

Submitted to the faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirement for the

Degree of Master of Science

in

Interactive Media and Game Development

May 2015

Approved:

Professor Brian Moriarty, Advisor

Professor Emmanuel Agu, Reader

Professor Joshua Rosenstock, Reader

ABSTRACT

Before Eternity is a short 3D adventure game that addresses the purpose of our earthly lives, inspired by the Sufi poet Rumi. To support its mystical theme, the design employs impressionistic elements and symbolic activities which deliberately defy many conventions of traditional adventure games. This report explains the design and implementation of the game, as well as its technical and production aspects.

ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor, Brian Moriarty, for enduring my stubborn personality for more than two years. He gave me invaluable insight and advice, and he had been very patient with me. Without any doubt, my weekly meetings with him were the best experience I had at WPI as a student. He also spent many extra hours to edit and enhance this report, which was a huge favor. Thank you, Professor.

I would next like to thank my committee, Emmanuel Agu and Joshua Rosenstock, for all the feedback they have given to this project.

Then, I would like to thank Bashir Faramarzi for composing and playing the music for this game. I believe his music is one of the strong points of *Before Eternity*.

Finally, I would like to thank these people for helping my putting this project together, I really appreciate their kindness.

Suzanne DelPrete	for	Posing for Ashton animation reference
Piyali Mazumdar	for	Posing for Victor character reference
James Thibeault	for	Editing and enhancing all English passages
Ales Jug	for	Posing for William character reference
Andrew Cote	for	Posing for Joseph character reference
Romisa Rohani	for	Posing for Beth character reference
Milad Zabeti	for	Giving architectural art tips
Ron Friedman	for	Giving Ashton run animation tips
Jean Moreno	for	Providing water particle system for free
Shwan Tavakol	for	Playing Kamancheh (musical instrument)
All play testers	For	Play testing <i>Before Eternity</i>

Contents

ABSTRACT	i
ACKNOWLEDGEMENTS	i
INTRODUCTION	1
SECTION A: BUSINESS, MANAGEMENT & PRODUCTION	3
BUSINESS MODEL	3
Creative Director vs. Producer	8
Management	9
SECTION B: IMPLEMENTATION	12
Game Design	12
Inspiration	13
Premise	15
Mechanics	18
Design Elements	20
Integrating Aspects of Real Life	21
Symbolic Inventory Items	23
Visiting Characters Repeatedly	24
Before Eternity as an Art Game	24
UX Design in Before Eternity	25

Audio	26
Background Music.....	26
Sound Effects	30
Graphics.....	31
Creative Process.....	31
Sky	34
NPC Characters	34
Ashton Animation	36
Environmental Objects	39
Inventory Items.....	40
Technology	41
Source Control	42
OOP	43
World Rotation.....	44
Collision Detection	44
Animations	45
Music Controller.....	46
Random-Font Text Creator	47
Procedures	48

Conclusion	49
Works Cited.....	52
Appendices.....	53
Appendix A: Artwork References	53
Appendix B: Text Messages.....	55
Appendix C: Import Character Procedures	57
Appendix D: Questionnaire	63
Appendix E: Game Walkthrough	64
Appendix F: Screenshots	66

INTRODUCTION

Before Eternity is inspired by a passage from the book *Fihi ma Fihi*¹ by Rumi, the Persian poet and Sufi master.

The Persian original reads as follows:

یکی گفت که این جا چیزی فراموش کرده ام. [خداوندگار] فرمود که در عالم یک چیزست که آن فراموش کردنی نیست. اگر جمله چیزها را فراموش کنی و آن را فراموش نکنی باک نیست، و اگر جمله را به جای آری و یاد داری و فراموش نکنی و آن را فراموش کنی، هیچ نکرده باشی. هم چنانک پادشاهی تو را به ده فرستاد برای کاری معین. تو رفتی و صد کار دیگر گزاردی. چون آن کار را که برای آن رفته بودی نگزاردی، چنان است که هیچ نگزاردی. پس آدمی در این عالم برای کاری آمده است و مقصود آن است. چون آن نمی گزارد، پس هیچ نکرده باشد.

My own English translation:

Someone said: "There is something I have forgotten here." [Master] remarked "There is one thing in the whole world that should not be forgotten. If you forget everything else and do not forget that one, there is no concern. But, if you remember and accomplish all other things, except that particular one, you will have done nothing! It would be like a king sends you to a village on a particular mission. If you go and do one hundred other things there, as long as you do not conclude the mission you have been sent for, it will be like you have done nothing! Human beings have come to this world for a specific task, as well, and that is solely the purpose. If they do not fulfill it, they will have done nothing!"

¹ Literally means "What is inside, is inside.", which is a metaphor for being self-descriptive or self-proofed. In another poem, Rumi writes "The sun is the proof for the sun. If you are looking for a proof, just do not turn your face away!" (آفتاب آمد دلیل آفتاب گر دلالت باید از وی رخ متاب)

The goal of the game is to get players thinking about Rumi's message on our mission in this short life. All design elements try to support this mystic and mysterious theme. It is not designed simply for amusement, but seeks to leave a thoughtful impression on the player.

To accomplish its goal, *Before Eternity* does not follow many of the conventions of traditional adventure games. For example, most adventures require players to offer particular inventory items to specific game characters to progress through the story. *Before Eternity* lets players offer any item to anyone, with some probability that the offer will be accepted or rejected, much as in real life. The design also intentionally refuses to notify players when they have put the game in an unwinnable state by doing something wrong or failing to do something right. Instead, the game's brief duration encourages players to try it repeatedly, "dying" over and over again until the optimum ending is reached.

What is the answer? Who knows the answer? These are the questions *Before Eternity* hopes to pose to players.

This report explains the design and implementation of the game, as well as its business model and production methods. Section A talks about business, management, and production. Section B explains the implementation of the game. Section C provides citations, sources and various appendixes.

SECTION A: BUSINESS, MANAGEMENT & PRODUCTION

BUSINESS MODEL

In their book *Business Model Generation*, Osterwalder and Pigneur suggest a “canvas” approach for visualizing and analyzing various aspects of a business plan. (16) Based on this approach, a canvas for *Beyond Eternity* might include the following elements:

Value Proposition

Before Eternity is a game that will be released for major desktop and tablet platforms.

Customer Segments

Before Eternity is inspired by Sufi mysticism, and hopes to reach people curious about the purpose of our earthly lives. As a result, in the early stage of design, the game’s target demographic was defined as “mature people with an interest in spirituality.” Some hardcore game players may be found in this demographic, but the game does not pander to them.

Fullerton’s *Game Design Workshop* identifies a number of common player types. (104) Based on her taxonomy, *Before Eternity* is expected to be most appealing to Explorer, Artist and Storyteller players.

Customer Relationships

Player contact will be mediated through a website and a dedicated Facebook page.

Distribution channels

When complete, *Before Eternity* will be published in online markets such as Apple's App Store, Google Play, Valve's Steam, and Microsoft's Windows Store. Because the gameplay depends on reading text, which is difficult on small screens, the game will be targeted at tablets, laptops and desktops instead of phones. There will be no physical distribution (e.g., CD-ROM or DVD).

Revenue Streams

As the game is very short, and not replayable once completed successfully, potential buyers are unlikely to be willing to pay much for it. Nevertheless, some people are willing to support the creators of "art games" such as *Beyond Eternity*. So the game will either be published for free, with a mechanism to receive donations from fans and supporters, or it will be published for a very low price in the markets previously noted.

Key Activities

Game implementation has been the main activity in the project. Developing a website and a hub for players will be a major post-production activity.

Key Partnership

The game was mostly a solo production, with background music and instrumental sound effects created with Bashir Faramarzi. No partnerships with any company or organization were involved, aside from WPI.

Key Resources

Components of the game were created with various software applications and hardware devices. The tables below list these resources.

HARDWARE DEVICES

Device	Purpose
HP Envy15 PC Laptop CPU: Intel i7-2670QM @ 2.20GHz Memory: 8.0 GB DDR3 @ 1600 MHz HDD: 256GB SSD Graphic Card: AMD Radeon 6600M OS: Windows 8.1 Pro 64-bits	Development computer device
External Mouse and Keyboard	Productivity
2nd Display: HP S2031, 20-Inch, 1600 x 900 resolution	Productivity
Wacom Intuos5 Touch Medium	Digital Illustration and painting
HTC Cellphone	Taking reference photos
Zoom H1 Handy Portable Digital Recorder	Recording environmental sound effects

Except for the digital recorder borrowed from WPI's Academic Technology Center, all other hardware devices are personal items.

SOFTWARE APPLICATIONS

Software Application	Purpose
Unity 4.x (Updated several times during the production of the game. The last version used is 4.6.3f1)	Game Engine
Visual Studio 2013 Pro (Through Microsoft's Dreamspark program)	Editing C# Codes
Corel Graphics Suite X7	Digital illustration
Corel Painter 2015	Digital Painting
Blender	Creating the ring
Microsoft Office 2013	Productivity, Production and Management
Adobe Audition CC	Sound Editing

Section B of this document provides more details on how these applications were used to create *Before Eternity*.

Cost Structure

As the game was developed sporadically over the course of two years, the exact duration of production is unknown, but may be assumed to have required six man-months of effort by one person. Based on this, the following table explains the cost structure.

Item	Cost
Labor 25 weeks x 40 hours x \$30	\$30000
Corel Graphics Suite X7 (Educational License)	\$81
Corel Painter 2015 (Educational License)	\$75
Adobe Audition CC (One Month Subscription)	\$30
Wacom Intuos5 Touch Medium	\$350
Total	\$30,536

This table shows only project-specific expenses. For example, the laptop used for development was owned previously, and office space expense is not included, because the game was mostly developed at home.

Creative Director vs. Producer

One of the hardest things about being an independent developer is the need to perform two opposing roles: creative director and producer. There were many times in the development cycle that I became tempted to work on a particular task or add a feature, but the producer in me refused to go along, either because there were higher priority tasks to do, or it seemed too time-consuming and not worth the learning cost.

Management

The SCRUM project management technique usually requires a production team. Despite this, I used a SCRUM board (hosted by Trello.com) to organize, break down, and prioritize tasks, and to minimize time management issues.

No Early Game Prototype

Conventional game industry practice encourages the creation of an early functional prototype, which can then be iteratively improved through playtesting. Early in the development of *Beyond Eternity* I decided not to do this, for several reasons:

1. In order to create an early functional prototype, I would need to create art placeholders, and then later replace them with polished assets. Since most of art assets were 2D sprites which needed to be positioned very precisely in the game, the process of replacing them would be likely to cause redundancy in work. For example, the size, scale, alignment, and position of the buildings in the game world is pixel-scale accurate, and replacing a placeholder with a file of incorrect dimensions would require me to redo the entire process of sizing, scaling, aligning and positioning. This is a common issue in team-produced games, but as soloist, the extra work would likely cause delays in production. I might have tried creating accurate placeholders in the first place, but the process of deciding on object size and position involves both technical and aesthetic considerations that are difficult to determine early in development.

2. At the beginning of the project, my art skills were not as well developed. If I had decided to focus on the technical part in early phases of the production, I could end up being unable to create art assets of the required quality in a later phase. Therefore, considering the number of hours I could allocate to developing this project, I dedicated a certain number of hours to produce graphic assets.
3. I found using a digital tablet for a few days in a row very overwhelming and upsetting for my eyes and wrist. Also, working a few months just on programming, then a few month on producing art assets could be mentally exhausting and boring. So, I had to interlace different types of tasks in my daily schedule. By keeping my hands on all types of tasks during the production of *Before Eternity*, my brain remained sharp about where to locate a particular piece of code, or how to remake a particular art asset. This was also a time saver for me.
4. McLaughlin, Pollice and West's book *Head First Object-Oriented Analysis and Design* suggests that developers give higher priority to tasks that remove risks from a project. (338) The authors identify three types of tasks that introduce risk:
 - a. Tasks with unknown definitions.
 - b. Tasks with unknown means of implementation.
 - c. Tasks dependent on other tasks.

From the beginning of this project, I realized I had no idea how to create the main character's animation, a significant aspect of the game. Making progress in other parts of the game and leaving the project with this uncertainty would put the project at major risk. Therefore, I gave this task a very high priority, and decided to create and import

this animation into the game at an early stage. The process of creating this asset is described in Section B in this report.

A major mechanic of the game, which involves revolving around a ring-shaped game space and visiting various characters, was another high priority task, and I implemented this feature early in production as well.

5. *Before Eternity* hopes to evoke an emotional reaction from the player, and this could not be achieved unless the player was exposed to the finished product. The game starts with showing an unfinished sentence on the screen, and if the player performs all the desired actions, they will see the second half of the sentence. For me, it was very important to have this feature alongside all other features of the game, including the music and sound effects, to achieve the artistic goals. I believed that exhibiting a rough prototype of the game to playtesters would be like displaying an unfinished painting to an audience and asking for their opinion.

I am very aware of the importance of testing the game for quality assurances, enhancing performance, removing bugs and typos, or even making subtle changes to maximize the impact, but, this playtesting could not change the shape of *Before Eternity*.

SECTION B: IMPLEMENTATION

Game Design

Before Eternity is about an anonymous avatar of ambiguous gender who traverses a ring-shaped world seven times, and at the end encounters their own tomb.

The mechanical goal of the game is to reach the tomb with a lit candle in the character's chest. To accomplish this, the player must perform certain tasks, manipulating objects and interacting with other characters in a particular order. A complete walkthrough of the game is provided in Appendix E.

The game's aesthetic goal is to invoke questions in the player's mind: *Where are we from? Why are we here? Where are we going?*

Lens #11 in Schell's *The Art of Game Design* (Figure 1) suggests that game designers should be inspired by everything in the world. (59)

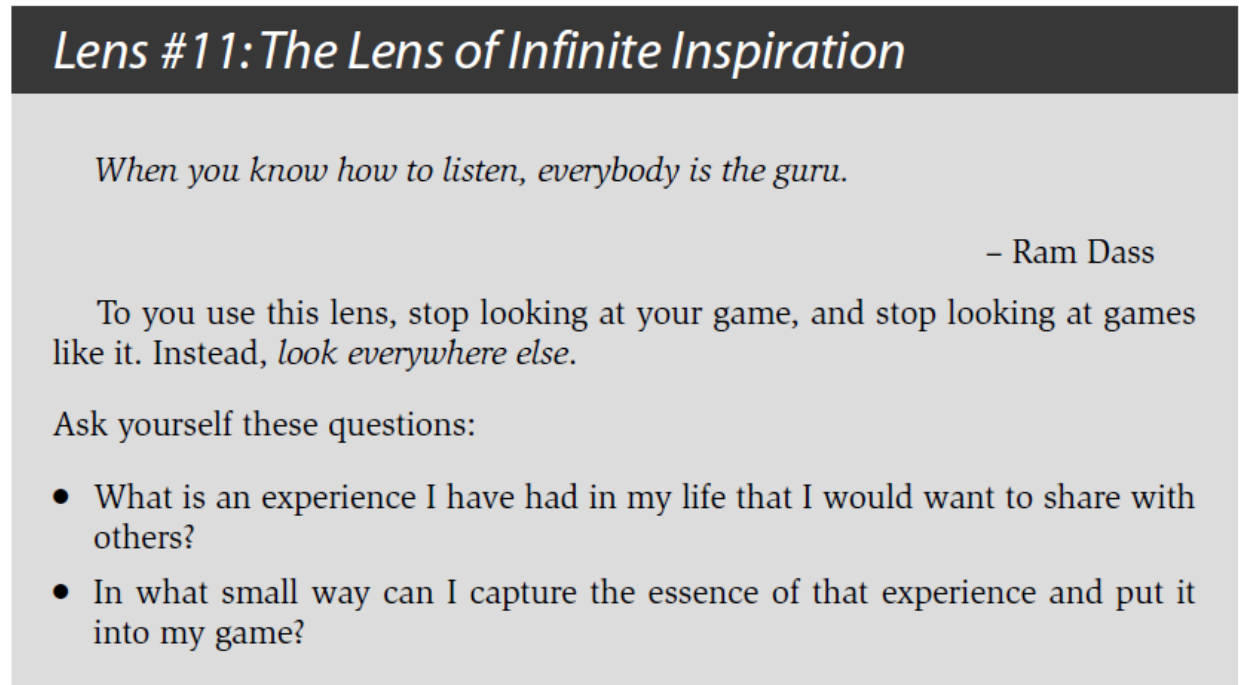


Figure 1. Lens #11 from Schell's *The Art of Game Design*.

Before Eternity is an example of applying this principle to game design.

Initially, I planned to work on a comic-strip style game as my master's project. But in summer 2013, I read an article on the properties of the Möbius strip on Wikipedia.com² that inspired me to make a game based on a looping topography.

About the same time, I had an interesting conversation with my project advisor, Professor Moriarty, about a passage from Rumi's *Fihi ma Fihi*. In chapter 4, the poet suggests

² http://en.wikipedia.org/wiki/M%C3%B6bius_strip

that people are sent to this world for a specific purpose, and that failing to accomplish that “one thing” is like accomplishing nothing at all. Rumi never explains what the “one thing” is, leaving it up to the reader to seek the answer.

Before Eternity, symbolically, offers the same goal for the game. The “one thing” in *Before Eternity* is having a lit candle when a player reaches their tombstone. If they finish the game in this condition, their mission is accomplished and they see the passage from *Fihi ma Fihi*. Otherwise the screen fades out, and the game returns to the main menu and welcome screen, ready to be tried again.

In order to experience *Before Eternity*, the player is not required to be familiar with Persian culture, Sufism, or Rumi. When a player reaches the optimum ending, the passage from Rumi’s book is revealed to the player, just to reinforce the impression.

An Iranian player might be more familiar with *Before Eternity*’s music and some of the game’s art style, especially the buildings. However, the game is not designed for any particular culture, religion or nationality, so anyone can have the same experience by playing *Before Eternity*.³

To keep the game accessible to an international audience, no Persian translations are provided for the text messages throughout the game.

³ In fact, it would currently be very difficult for Iranians to obtain this game electronically. Iranian individuals do not have access to online payment systems, and because of sanctions, any access of financial resources through an IP from Iran is likely to be blocked.

Premise

Before Eternity unfolds in a ring-shaped world. The player navigates an avatar around the ring for seven rounds, interacting with characters and objects along the way. Figures 2 and 3 show how the virtual camera is located behind the player avatar, and what the player sees from the camera's perspective.

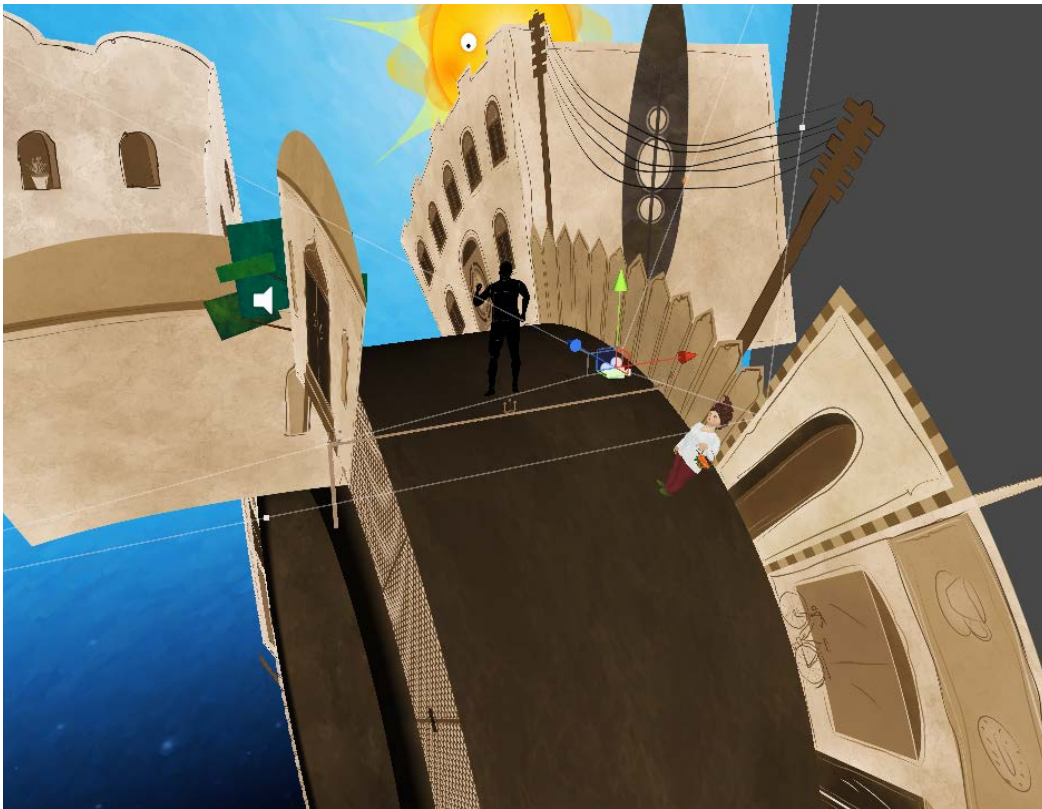


Figure 2. Scene architecture and camera position.



Figure 3. Player's perspective of scene from camera position.

In his article "Premise: The Key to Interactive Storytelling," Stephen Schafer writes: "From a psychological standpoint, a player that wins a well-designed game will also win by association with the protagonist, purpose, and premise of the game." (5) *Before Eternity* attempts to emotionally influence the player by placing them in the symbolic predicament of the avatar, trapped in a seemingly endless loop.

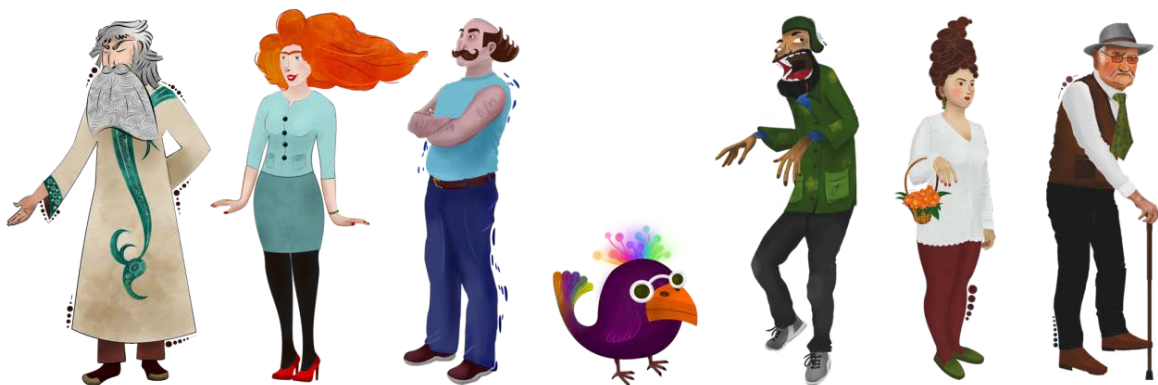


Figure 4. NPCs (left to right): Victor, Emily, Sebastian, Rira, Joseph, Beth and William.

Ashton, the player's avatar⁴, meets seven non-playable characters (NPCs) on each circuit of the world: Victor, Emily, Sebastian, William, Joseph, Beth, and Rira (a bird), shown in Figure 4.

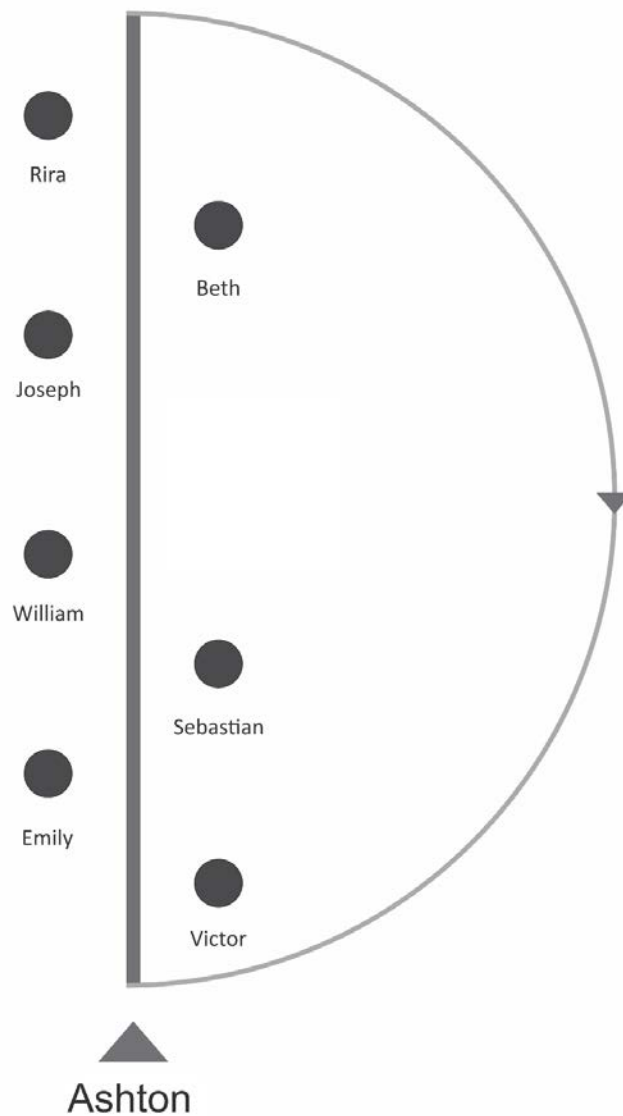


Figure 5. Order of NPC encounters.

Figure 5 is a diagram of the closed loop, in which Ashton encounters the main NPCs.

⁴ This character was initially named Adam, but after making the decision of having a morphing unisex avatar for the playable avatar, I changed the name to Ashton, which is a unisex name.

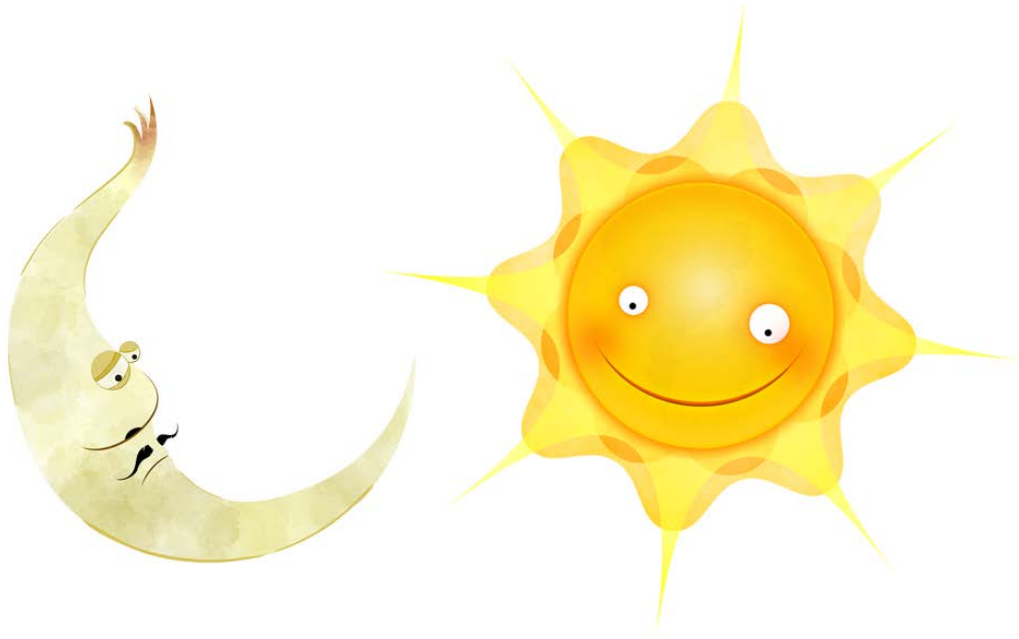


Figure 6. Moonak and Hoor.

In the sixth round of the game, player meets two additional characters, Moonak (the moon) and Hoor (the sun), shown in Figure 6.

[Mechanics](#)

The player moves forward along the ring by holding down the space bar. Releasing the space bar immediately stops the movement. Backwards movement is not allowed, and no NPC interaction is allowed while the playable character is moving.

When the player releases the space bar, the inventory appears as a panel on the bottom of the screen.

The mouse is used to select characters and objects of interest. Responsive objects are highlighted in an obvious way when the cursor hovers over them. Clicking on a highlighted object triggers a scripted response, such as a character dialogue and/or a change in object state.

Some interactions add or remove items from the avatar's inventory. Inventory items are indicated by screen icons, and can be "applied" to responsive characters or objects by dragging them with the mouse. Appendix E contains a complete list of interactions required to reach the optimum ending.



Figure 7. Inventory bar with three objects: wick, block of wax and mug.

Weinschenk (65) notes that while people are interacting with something, there are three types of effort or "loads" that can be expected:

- Cognitive (including memory)
- Visual
- Motor

These loads are the physical and psychological actions that people do in relation to particular environments, products, or services. Weinschenk adds:

The best example of purposely increasing loads is gaming. A game is an interface where one or more of the loads have been intentionally increased to provide challenge. Some

games have high cognitive loads because you have to figure out what's going on. Some have high visual loads, where you have to find things on the screen. Some have high motor loads, where you have to use the keyboards or a separate device to move the cursor or shoot the villains. (67)

By Weinschenk's definition, *Before Eternity* applies cognitive loads on its players, and in order to keep them focused on analyzing the events and discerning the spiritual import of the game, it tries to minimize motor load.

Design Elements

Lens #9 in Schell (Figure 8) suggests that a game should have a theme, and that everything in the game should reinforce its theme. (53)

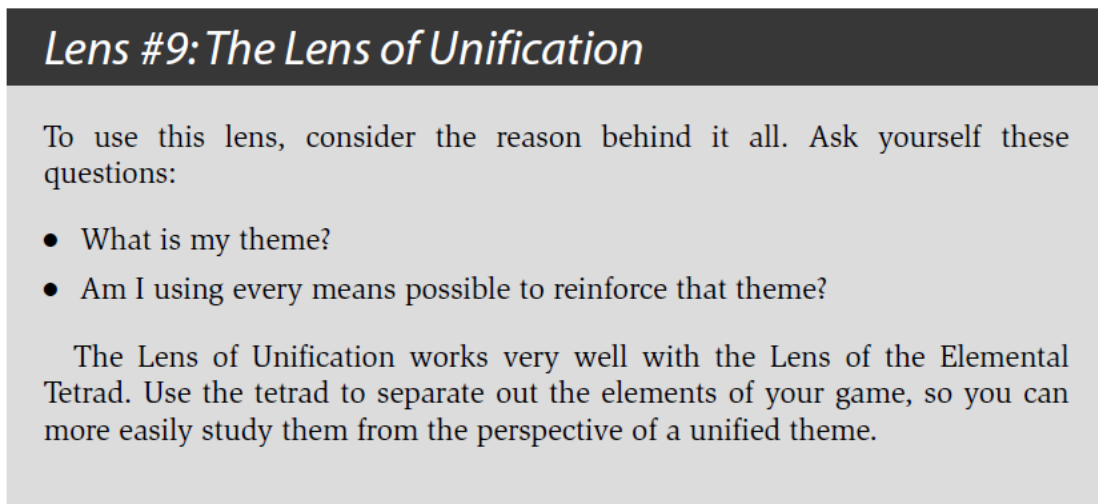


Figure 8. Lens #9 from Schell's *The Art of Game Design*.

With this advice in mind, I decided to reinforce *Beyond Eternity's* mystical theme by several means.

Strange Visuals

Since the game happens in an abstract setting, I tried to make the visual elements appear strange and unusual by using elements of Middle Eastern architecture, curious-looking trees and flowers, and exaggerated character designs.

Exotic Music

The layered background music is performed on traditional Persian instruments, adding to the exotic atmosphere of the game.

Ashton's Animation

The morphing, gender-changing visualization of the playable character, Ashton, not only is strange looking, but also suggests that Ashton might be anyone, a symbolic representation of all human beings.

Integrating Aspects of Real Life

Since *Before Eternity* is about the spiritual purpose of life, I decided to mimic or simulate certain aspects of real life to the extent possible. Some of the resulting design decisions run against the conventions of traditional adventure games.

Time Passage

The game tries to reinforce the concept of time passage. Since our perception of time is relative, I tried to incorporate randomness into several time-related events.

Days and Nights

Except for level 6, in which the player should interact with the moon and sun in a particular order, the sky alternates between days and night after a random period of time between 30 and 50 seconds.

The Clock House

On top of a building in the game, there is a clock with hands which move at a different randomly-chosen speed on each circuit around the ring.

Forward-Only World

The forward-only motion mechanic in the game suggests that “You cannot go back and undo what you have done already.” In *Before Eternity*, running around the ring and reaching your tombstone at the end is like moving along the timeline of your life.

The Bell Tolling

The rhythm for the background music of the game is a distant bell, which tolls continuously from the beginning of the game to the end, further reinforcing the idea of the relentlessness of time.

William’s Death

In level 5 of the game, the NPC William dies. His image is replaced with a tombstone.

Inventory Mechanism

In conventional adventure games, it is common to allow only specific items to be offered to particular characters in the game. For *Before Eternity*, I decided to make it possible for the player to offer any inventory item to any character. This decision was made to mimic what is possible in real world.

Although this mechanic risks confusing experienced gamers, the very short duration of gameplay, small number and short life cycle of inventory items, and explicit hints help to make this design less frustrating for players. Moreover, by “losing” a few times, the player will realize that they cannot “lawn mow” the game. “Lawn mowing” (a term introduced to me by Professor Moriarty) occurs when a player begins applying every inventory object to every possible objects/character in the game, in order to discover what works.

Symbolic Inventory Items

If the player does everything correctly up to the fifth round, they can retrieve four objects from Ashton’s chest:

- A diploma scroll: the symbol of our education or our vanity for scientific achievements.
- A few coins: the symbol of our materialistic attachments.
- A string of beads: the symbol of our vanity for ethical and religious achievements.
- A ring: the symbol of our attachments to earthly love.

To achieve the optimum ending, these objects must be donated to any character along the way. This is the only way Ashton can earn the candle installed in her/his chest.

Visiting Characters Repeatedly

The characters in *Before Eternity* are the people that the player “lives” with during their short life in the game. The closed-loop topography makes it possible to encounter each character several times, as often happens in real life. Sometimes the player knows what a particular character will expect in the next level, but sometimes unexpected things happen, such as the death of William. Characters may give the player an item or “talk” by displaying a message, or they may not. In this case a message box with the text “blah blah” is shown.

Before Eternity as an Art Game

Before Eternity is a game designed to emphasize art, and the design goals are based on Sufi literature. Its main intention is to trigger a process in which the player attempts to interpret the symbols in the game and think about the text passage from Rumi’s book, *Fihi ma Fihi*. This passage is displayed to the player only at the optimum ending, which is achieved by having a lit candle when their tomb is reached.

Beside the emphasis on art, by displaying pieces of literature in English and Persian language and considering the philosophy of life, *Before Eternity* also has educational purposes, and might therefore be categorized as a “serious” game.

UX Design in Before Eternity

In the development of *Before Eternity*, in order to help the player to focus on the story, rather than the mechanics of the game, I have conducted some UX design:

1. The player can only start an interaction when they are not running.
2. When the player releases the space bar and Ashton stops running, the inventory pops up automatically, to avoid an extra interaction for showing the inventory.
3. Many adventure games, such as *Broken Age*, require the player to drag items from inventory to the point of interest, but in *Before Eternity*, clicking an item in inventory “picks up” the object, attaching it to the mouse cursor until you click on a point of interest to use the item, or on another place to return it to the inventory.
4. The characters that can receive items are obviously highlighted, if the player hovers an inventory item on top of them. This makes it clear who is capable of receive an item.
5. The text messages spoken by characters are short. Back-and-forth dialog between Ashton and the NPCs has been avoided.
6. Inventory items have short life cycles. Each item that is added to the inventory is intended to be used in the same or the next round of the game.

Audio

In *Before Eternity* there are two kinds of audio assets:

- Background music
- Sound effects

Sound effects of the game are either instrumental or environmental.

Background Music

The background music for *Before Eternity* is composed and played by Bashir Faramarzi. Bashir lives in the US, and we spent time talking over phone or Skype to develop the music. I believe using Persian musical instruments and having a Persian-themed composition have enhanced the mysterious theme of the game.

Here is a list of criteria I asked Bashir to follow while composing the music:

1. The background music should have a slow tempo.
2. It should be looped.
 - a. Not a seamless loop, such as MIDI loops
 - b. No breaks or unpleasant jumps at the boundaries of the loop.
3. It should be layered and delivered in separate synchronized files, in a way that I can add and remove tracks individually.
4. Since a complete round of running around the world is less than one minute, the background music should not be longer than a minute as well. This guarantees that the player will hear a whole loop, at least once per round.
5. The beat for the music should be maintained by a distant bell tolling.

- a. It makes the music more mystic and mysterious.
 - b. It represents the passage of time.
 - c. When the player reaches their tomb at the end of round seven, they see the actual bell tolling on top of their tombstone.
6. From level 2 to 4, three layers should be added to the music, one at a time.
 7. From round 5 to 7, the previously added layers should be removed from each level in reverse order, one at a time.



Figure 9. Setar, kamancheh and daf.

Setar: <http://anbari.persiangig.com/neynava/images/setar.jpg>

Kamancheh: <http://www.nimarahmati.com/Multimedia/5.jpg>

Daf: http://www.ahangac.com/modules/portfolio/dataimages/IMG_935457-2AE9A5-2CED55-D8E101-9691F4-2034F7.jpg

After the first round of brainstorming with Bashir, we decided to use a setar, santour and daf as our musical instruments, but later we replaced the santour with a kamancheh (Figure 9).

This table shows which layers are played in each round:

Level	Music Layers
1	Bell
2	Bell + Setar
3	Bell + Setar + Kamancheh
4	Bell + Setar + Kamancheh + Daf
5	Bell + Setar + Kamancheh
6	Bell + Setar
7	Bell

In order to have these layers synchronized in the game, the layers of music are added to Unity's main camera as "Audio Sources" from the beginning (Figure 10). The instrumental mix is then controlled by increasing or decreasing the volume of particular layers as desired.

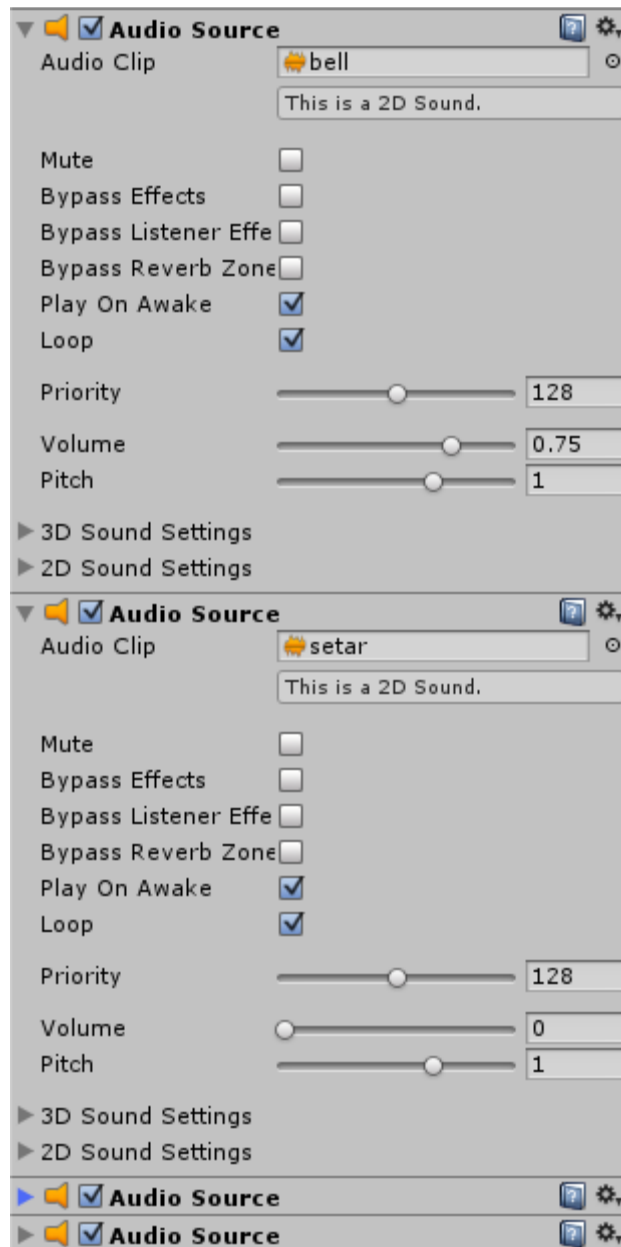


Figure 10. Audio layers in Unity game engine.

Sound Effects

There are two types of sound effect in *Before Eternity*. Physical actions in the game that can produce real sounds have environmental sound effects.

- Plant the seeds
- Clean the chest
- Fill the bowl with water
- Water the seeds
- Pick the hive
- Pick the rose
- Open the chest's lock
- Empty the chest (remove 4 items)
- Install the candle
- Light the candle
- Candle dies

I recorded the raw sounds with a Zoom H1 digital recorder (Figure 11).

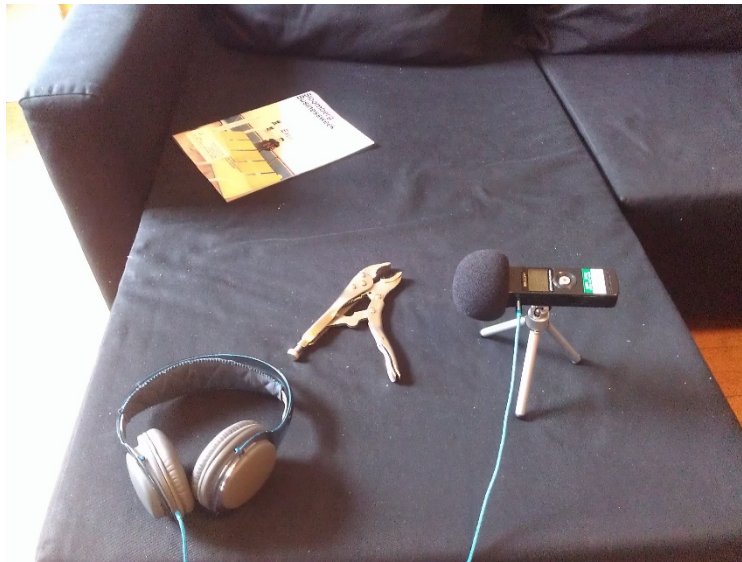


Figure 11. Recording “open the chest lock” effect with a pair of pliers.

For abstract actions, Bashir produced some instrumental sound files. These files are used for four purposes:

1. Adding and removing items to and from inventory.
2. A short melody when Rira (the bird) sings poems.
3. Menu items mouse over
4. Dialog starter

In order to have a more abstract look and feel for the game, I decided not to include voice messages for the pieces of texts that are displayed for each character during gameplay.

Graphics

Before Eternity consists of 2D sprites rendered in a 3D environment. The only 3D objects in the game are the path ring, and the fountain base. The ring is a Unity cylinder mesh primitive. I found that adding a particle system for the fountain's water on top of a 2D sprite did not look good, so I also used a cylinder mesh as the base for the fountain.

The following chapters describe the creative process and procedures to produce 2D art assets for the game.

Creative Process

Before Eternity's illustrations were usually started with a sketch on paper, or occasionally from a reference photo on the web (see Appendix A). For most of the pieces, I created a vector drawing in Corel Draw. Corel Draw allows an artist to assign vector objects to different layers and export the file in .psd (Photoshop) format. These files would then be brought into PhotoPaint and/or Painter to adjust, color, paint, and polish the illustration.

After completing an illustration file, I usually set it as my desktop background for a few days. This let me look at it from different angles and distances for a fairly long period. If I found things I didn't like about the illustration, I would go back and make changes until it satisfied me.

Sometimes I created assets that I realized I did not like at all. One example of this is the trees, which were inspired by the shape of feather. I created four generations of trees until I liked the look and feel of them (Figures 12-15).



Figure 12. Trees, first generation.



Figure 13. Trees, second generation.



Figure 14. Trees, third generation.

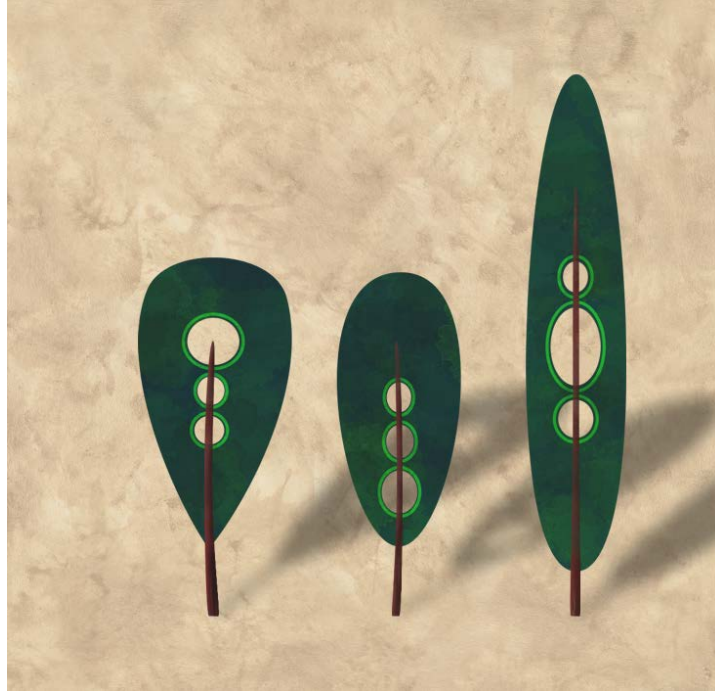


Figure 15. Trees, fourth and final generation.

Sky

The sky background is a 3700 x 6500 pixel image, which scrolls up and down to create the change from day to night. The file must be large because the sky sprite is far from the camera, located behind the ring. Moonak and Hoor, the moon and sun characters, are located on the sky sprite.

NPC Characters

There are six human NPCs, plus a bird (Rira). For all characters except Emily, I took photos to have references for body gesture. Emily's body gesture is based on a photo reference from the Internet. The faces of Joseph and William are also based on images from the Internet; the rest were created from personal sketches. (See Appendix A for all art sources.)

A collection of 3D models for a character, including a head with a bun, a face with closed eyes, a torso in a white dress, a hand, a leg in a red stocking, and a basket of flowers.

Ashton Animation

Ashton is the playable character of the game. Initially I wanted to have two different characters for the playable character, one girl and one boy, and change their presentation at the beginning of each round change to demonstrate their aging. However, this idea required fourteen loops of animation, seven for each character, which seemed excessive. Instead, I decided to visualize the playable character as a morphing and sex-changing avatar. I also decided to create this animation in silhouette to make it more abstract, and also to achieve a better look and feel for the transitions between frames.

To implement this, I started looking for four animation models, two men and two women. The goal was to ask them to run in front of a green screen, extract frames and paint on them, then interlace the frames.

At first, I recorded myself running in front of a green screen. Then a female IMGD student volunteered to run for me as a reference. But when I extracted the frames from her video, I realized that having different running styles and body shapes, together with the lack of exact synchronization between runners, would make this approach impractical.

Finally, I decided to create vector graphics for the frames based on my own running reference, then manipulate them to produce the second male character, and two female characters. This approach guaranteed that the posture of all four characters in a particular frame number would be the same. I originally wanted to have eight frames for each character, but since eight (the number of frames) is divisible by four (the number of characters), I realized that all my loops would start and end with the same character. Therefore I created four loops of

running animation with nine frames for each character, then interlaced them in a man-woman-man-woman fashion. At the end I had an animation of 36 frames in which each loop of running starts with a significant character (Figure 17).

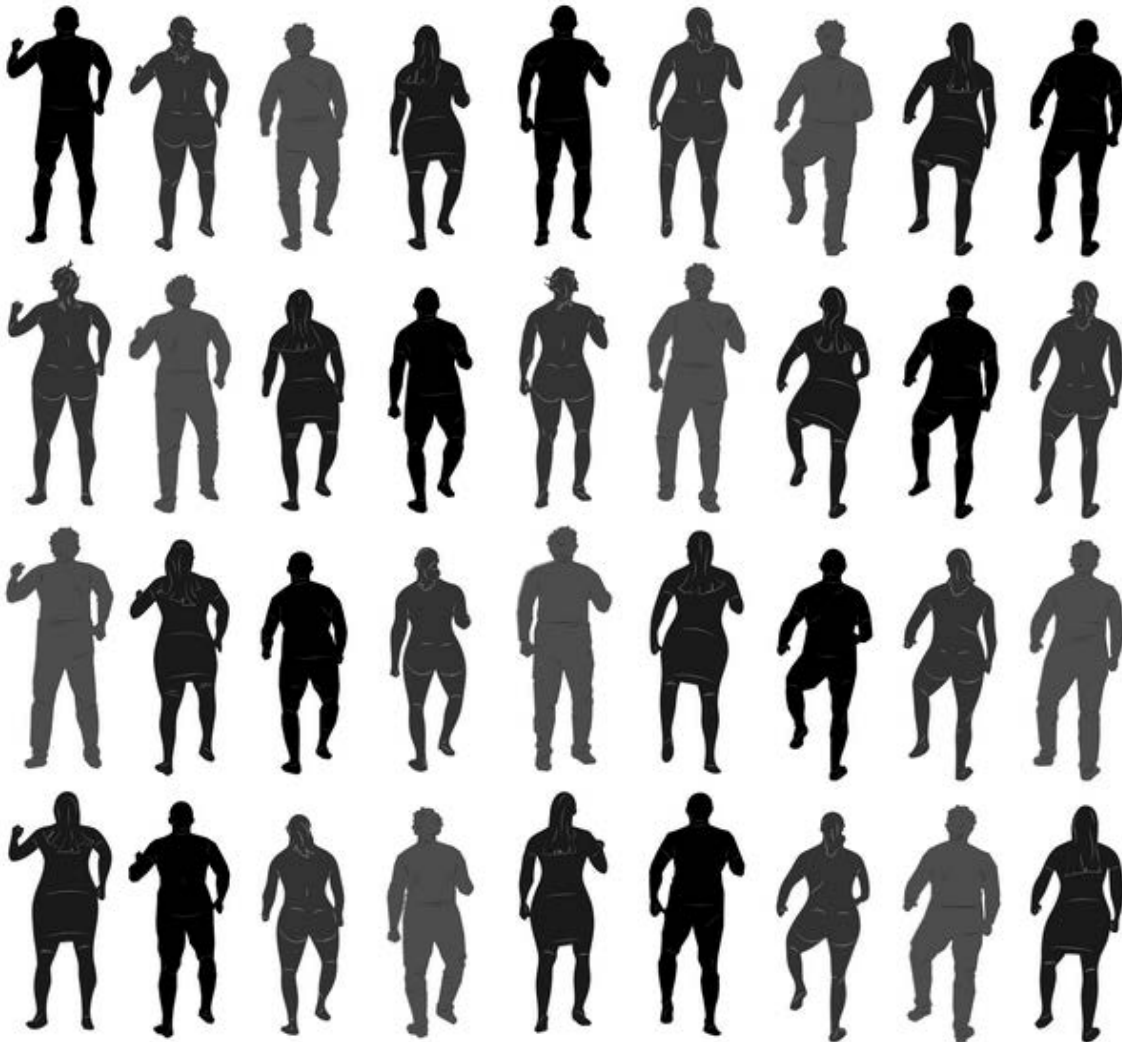


Figure 17. Ashton animation sprite sheet.

Creating the idle animation for the character was much easier, because it is just a loop over the four characters of Ashton's animation standing still (Figure 18).

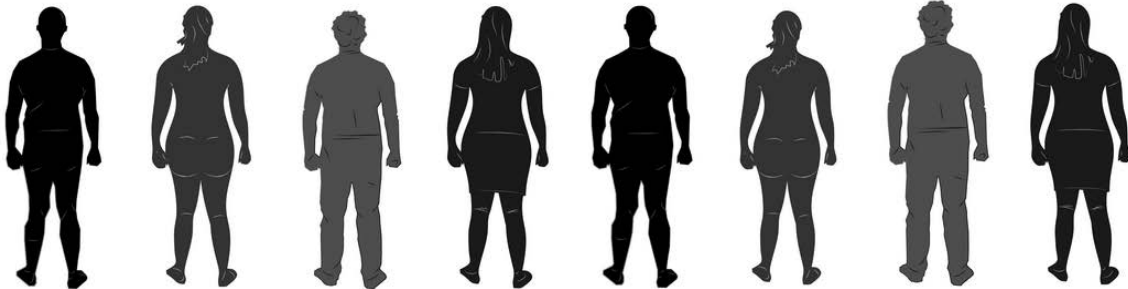


Figure 18. Idle animation sprite sheet.

Creating the animation for Ashton was the most challenging part of the production. I had never done anything like this before in my creative projects.

Environmental Objects

There are only a few environmental objects in *Before Eternity*. It is an option to add more pieces of environmental art to the game, before commercializing it, just to make it more beautiful.



Figure 19. Flower pot.

Flower Pot

The flower pot plays an important role in the game. Refer to the complete walkthrough (Appendix E) for details.



Figure 20. Fountain. Water particle source: <http://jeanmoreno.com/>.

Fountain

The fountain's water was created using a particle system by Jean Moreno (Figure 20).

Inventory Items

The icons representing inventory items were created from my own sketches, or based on reference photos. After creation, they were exported as 64x64 .png files with background transparency to the Unity engine. (Figure 21; see Appendix A for all reference art sources.)



Figure 21. Inventory icons.

Technology

Beyond Eternity was built with the Unity game engine, and coded by C# using the Visual Studio 2013 IDE. The game has a fairly simple technical profile, and the programming aspect of the production was straightforward. The following sections of this report explain the game's technical development.

Source Control

I used TortoiseSVN, a GUI application for the Apache Subversion source control system, for *Before Eternity's* production. Although there are tools such as AnkhSVN that allow working with Subversion directly from within Visual Studio, programming was not the only aspect of the production for me, so I did not use this option.

I researched a number of free source control hosting services, such as GitHub and BitBucket, but the large size of this project made these solutions inefficient. Putting the repository on a local hard drive, even if it an external HDD, is not a very safe option. I finally settled on using Microsoft's cloud-based OneDrive application for my source control system.

Currently, Microsoft OneDrive does not allow direct file access over HTTP or FTP, and as a result it cannot be used as a repository for Git or Subversion. But OneDrive offers an application that users can install on their computers to connect their cloud storage to their hard drive. By having the OneDrive app installed on my Windows machine, I was able to create a repository directly in a directory inside my OneDrive storage space, and since it was a physical folder on my HDD, OneDrive didn't complain. Then, when committing my project, Subversion updates the repository in my local drive's folder, and the OneDrive service that is watching that folder detects the changes and updates my storage on the cloud.

A second benefit of this mechanism is I can commit my code even when I do not have Internet connectivity. As soon as my computer gets connected, OneDrive starts the updating process to upload the new files on the cloud storage.

OOP

Before *Eternity's* source code takes advantage of object-oriented programming principles to make the code more efficient. Figure 22, created by Visual Studio, displays the hierarchy of character classes in the game.

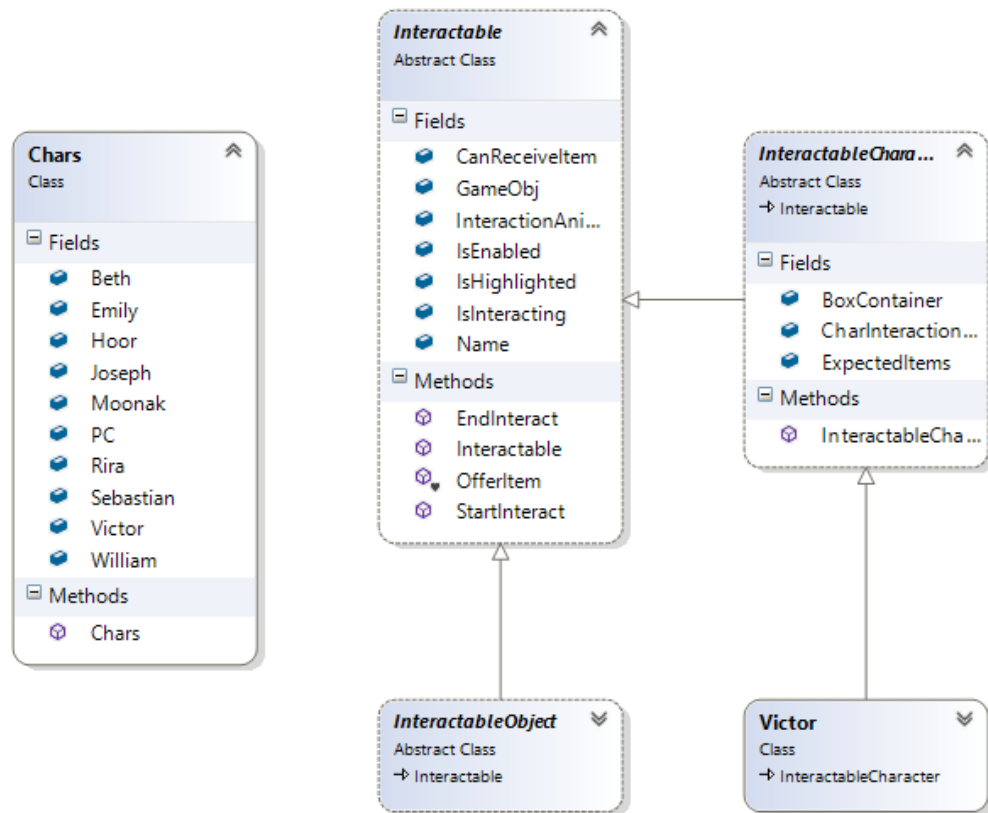


Figure 22. Character class hierarchy.

The code will eventually need post-production refactoring to remove redundancies and make it more readable.

World Rotation

When I started programming *Before Eternity*, I assumed that all the objects in the game would be fixed, and only the PC's avatar and camera would revolve around the ring. Later, I realized that many more objects needed to be fixed relative to the camera, and I would have to rotate them with the camera around the world as well. The exit screen, sky objects, main directional light, transparent game objects that reset other objects in the scene, and (most important) the mouse cursor object are all examples of this.

As a result, I decided to leave the camera, PC's avatar and other objects fixed, and instead, make the world and all its objects rotate around the ring's X-axis. What the player sees on the screen is an illusion that makes it appear that Ashton is moving around the world.

Collision Detection

Because my 2D sprites are not all located on the same plane and are moving in 3D space, Unity's new "Box Collider 2D" component could not be used. All objects that required collision detection needed a conventional 3D box collider.

Although there are not many real collisions between game objects in *Before Eternity*, Unity internally uses ray casting for mouse events, and ray casting requires collision detection. Because of this, all clickable objects in the game needed a box collider component. Figure 23 shows the green highlighted box colliders for inventory items.



Figure 23. Box colliders for inventory items.

Animations

Except for the PC and candle flame animations, which are based on sprite sheets, all animations in *Beyond Eternity* are dynamic and scripted in Unity. All human NPCs have two animations that are added to different layers of the same Unity Animator object (Figure 24). Each of these animations can be played in code by increasing or decreasing the weight of the layer. In my case, in order to add and remove the animation layer, I set its weight to 1 or 0.



Figure 24. Animation timeline for the Beth NPC.

Animation layers can be additive in Unity, which means I can set the weight for both character's talk and highlight to 1 and display them both at the same time.

For the cases in which animations must transition from one state to another, animation states are created. In many games this can be very complicated, but in *Beyond Eternity* there are only a few such instances, and they all have just two states. Figure 25 shows the sky's animation states, which includes a Boolean parameter (IsDay) used to trigger the transition.



Figure 25. Sky animation states with Boolean parameter.

Music Controller

At the beginning of rounds two, three and four, a new instrumental layer is added to the music. These are removed in reverse order at the beginning of rounds five through seven. This mechanism is all implemented in code.

A player may reach the start position of a round in the middle of a music loop, and the sudden activation of new layer might not sound pleasant. Therefore, the music controller of the game has a Unity FixedUpdate loop that can increase and decrease the volume of each layer over a given duration. When the player reaches the start point of the ring, based on which round it is, a parameter is sent to the music controller which adds and removes the desired layer by fading its layer in or out.

Random-Font Text Creator

While designing different aspects of the game, I decided to render the text messages in random fonts for each character. After creating the first message box in Corel PhotoPaint, in which I manually changed the font of each character, I realized this procedure would be extremely tedious. Therefore, I wrote a small external utility application that randomizes the font of characters for the given text, and then saves the output as a .png file with transparent background (Figure 26). This Windows app is written in C# and requires the .NET Framework to be installed to run.

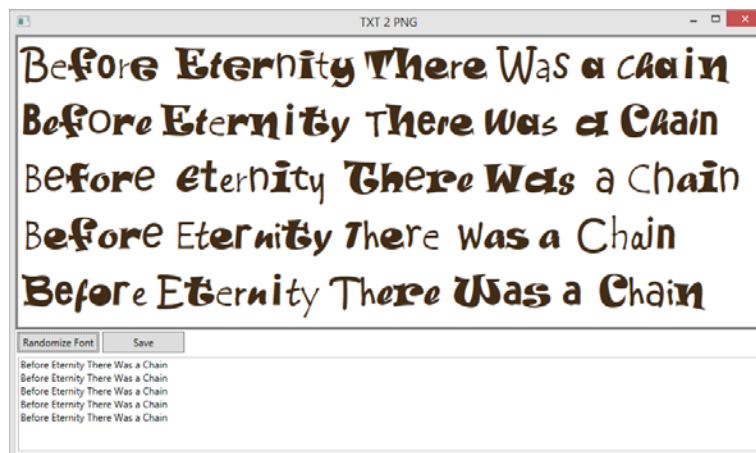


Figure 26. Sample output of random message generator.

Currently, the GUI of the app is minimal and ugly, and the list of fonts that the app chooses from, along with the size and color of the output text, are hardcoded. But with improvements, I think it has the potential to be converted to a useful utility for game makers and graphic designers.

Procedures

During the development of *Before Eternity*, there were times when I had to repeat or redo exactly the same procedures that I had done before. Most of these procedures were relatively simple. But, the process of importing a character sprite sheet, placing it on the ring, and animating it was very complex and tedious.

In order to reduce code redundancy, I used a software development technique called “convention over configuration.” I wrote pieces of code that could trigger various animations for a potential character. Then, to import a new character to the game, I named the objects and animations with specific pre-defined names. This allowed the attached code to detect and play the correct animations without doing extra steps.

Since the procedure to import a new character to the game was so long, I wrote down the steps and for each character I followed it to have the new character imported. This procedure is described in Appendix C.

Conclusion

Although it's probably too early to judge the success of *Before Eternity*, so far all playtesters have replied to survey question #4 "Was the game successful in telling its story and giving you the impression?" with "Yes." (See Appendix D.) Nevertheless, I do not consider the current version of the game to be final, as there are many things I would like to add or improve. This section provides a brief overview of what I could do better and what I would like to add.

Being a stand-alone developer has pros and cons. On the creative side, I was happy that I am getting exactly what I want, but on the management or production side, having a partner would have definitely speeded up the production. I have realized that when I have a partner, I start to be more organized and efficient automatically, without a need to be pushed.

I think I could have also benefited from having an expert Unity programmer at my side. The code clearly needs some refactoring and optimization.

Although releasing the game on tablet devices has been a major business goal from the beginning, not having access to Android and iPad tablets prevented me from fully testing the game during development. The Unity game engine makes it fairly easy to deploy a game on various platforms, but I am concerned that running the game on an Android or iOS tablet will produce a significant drop in the frame rate. Although *Before Eternity* does not have many high-polygon objects, it does incorporate many large .png files with transparency, which might fill up a tablet's RAM quickly. Fixing this would involve reducing the size of all sprites in the game, which means I would have to reposition and align them again, one by one, a very tedious

process. Using a 'tablet-first' development approach might have avoided this potential problem.

There are many improvements that I would like to apply to *Before Eternity* at this stage. Some were discovered by watching players test the game, while others were suggested by the testers themselves:

- Make the goal of the game clearer at the beginning. Some players had problems locating the almost-hidden box on Ashton's body. Those who did often could not understand what they ought to do with it.
- Begin by showing the sky in sepia, and make it colorful only in the sixth round, to encourage interaction. Although there are a few direct hints, some players did not realize they could interact with the moon and sun.
- Add audiovisual clues for adding and removing items from inventory.
- Add audiovisual acknowledgements for major progress in the game, such as opening the chest, creating the candle, etc.
- Edit some of text passages. As an example Victor should say "A person should shed all the belongings first...", thus encouraging players to get rid of everything they are carrying in their chest.

I have also decided to reject some of the feedback I received. My process for accepting or rejecting feedback is to consider if it supports the aesthetic goals of the game. If it does, I try to incorporate it.

Some feedback that would make sense in a conventional adventure game did not meet this criteria. Here are two examples:

- “Add a progress indicator that tells me which level I’m on.” This is standard practice for a conventional game, but our real lives do not have an indicator that shows us how many more years we will live. However, I might consider a less explicit way to indicate the player’s progress to the player, perhaps by the changing amount of fruit on a particular tree.
- “Give a warning, such as a few seconds of flicker, before killing the candle in round seven. It is frustrating to lose the game because of a distraction, such as a phone call.” This suggestion also conflicts with the game’s philosophy. If you have something valuable in your life, a distraction may cause you lose it. *Before Eternity*’s short duration allows a player to reach the same point in less than nine minutes.

Works Cited

- Charsky, D. "From Edutainment to Serious Games: A Change in the Use of Game Characteristics." *Games and Culture*. 2010, Vol. 5: 177-198. Print.
- Fullerton, T. *Game Design Workshop: A Playcentric Approach to Creating Innovative Games*. (Third edition) A K Peters/CRC Press, 2014: 104. Print.
- McLaughlin Brett, Pollice Gary, West David. *Head First Object-Oriented Analysis and Design*. O'Reilly, 2006: 338, 349. Print.
- Osterwalder Alexander, Pigneur Yves. *Business Model Generation: A Handbook For Visionaries, Game Changers, And Challengers*. Wiley, 2010: 16-19. Print.
- Rūmī Jalāl ad-Dīn Muhammad, Foruzānfar Badi'ozzamān, Yazdāni Zeynab. *Fih'e ma Fih'e*. Attar, 2002: 127. Print.
- Schafer, S. "Premise: The Key to Interactive Storytelling." *Game Career Guide*, 2010. URL: http://www.gamecareerguide.com/features/357/premise_the_key_to_interactive_.php. Web.
- Schell, J. *The Art of Game Design: A Book of Lenses*. CRC Press, 2014: 53, 59. Print.
- Thackston, W.M. Jr. *Signs of the Unseen: The Discourses of Jalaluddin Rumi*. Shambhala, 1999: 15. Print.
- Weinschenk, S. *100 Things Every Designer Needs to Know About People*. New Riders, 2011: 65-67. Print.

Broken Age (Act 1). Double Fine Productions, 2014. Software.

Appendices

Appendix A: Artwork References

All URLs were in working in the time of writing the paper

Candle	https://theotherhubby.files.wordpress.com/2013/11/tumblr_m96a8dqkam1rdmkbl_o1_500.gif
Wax	http://curd-nerd.com/wp-content/uploads/2011/06/YellowCheeseWax.jpg
Bulb	http://figurasgeometricas.info/wp-content/uploads/2013/12/geometria-espacial.jpg
Coins	http://imperfectspirituality.com/wp-content/uploads/2010/09/Stack-of-coins1.jpg
Handkerchief	https://www.everplans.com/sites/default/files/styles/article_header_image/public/handkerchief-items-to-bring-main.jpg
Ring	http://4.bp.blogspot.com/-7zuC17spjZo/T-Pd_vJf-KI/AAAAAAAAAD5k/oBoQyibTe1Y/s320/Titanium+Wedding+Rings+for+Men2.jpg
Paper scroll	https://s-media-cache-ak0.pinimg.com/736x/08/7b/5d/087b5dfbef28cca562e39b866eb2e4e7.jpg
Emily	http://www.photoventure.com/wp-content/uploads/2013/01/amazing-photography-infographics-posing.jpg
Trees	https://s-media-cache-ak0.pinimg.com/736x/5f/35/f7/5f35f7be030af5a0247d508d3f8af8fc.jpg https://s-media-cache-ak0.pinimg.com/736x/4f/cb/f6/4fcbf6cb8eaec7a996650593d4ebc3a8.jpg https://s-media-cache-ak0.pinimg.com/736x/76/9a/2e/769a2e960a71f189e203c82215674639.jpg https://s-media-cache-ak0.pinimg.com/736x/83/f6/50/83f6508c3cc395320b48522c709cc562.jpg https://s-media-cache-ak0.pinimg.com/736x/8d/20/35/8d2035c2f776d301ba1953707a3f6fcb.jpg https://s-media-cache-ak0.pinimg.com/736x/72/70/c8/7270c881f26d93b92106fe19abbc6f32.jpg https://s-media-cache-ak0.pinimg.com/736x/11/4a/ba/114aba600e1d4d931a6567c9c0f1239a.jpg https://s-media-cache-ak0.pinimg.com/736x/0c/99/88/0c99880cb86cee21df00b55a8bc6a8d6.jpg https://s-media-cache-ak0.pinimg.com/736x/54/05/29/54052964128550a5f0f3f1c61ec77d51.jpg https://s-media-cache-ak0.pinimg.com/736x/dc/75/e9/dc75e967588b8510c89899242f262820.jpg

William's tombstone	http://image.made-in-china.com/2f0j00mMPaWnDqMjkC/Monument-Tombstone-11.jpg
William Face	http://juliacomedy.com/wp/wp-content/uploads/2012/02/angry-old-man.jpg
Candle flame	https://theotherhubby.files.wordpress.com/2013/11/tumblr_m96a8dqqam1rdmkblo1_500.gif
Environmental Art Assets	http://www.lulusgalena.com/images/Milas_Herbalist_Shop.jpg http://4.bp.blogspot.com/-J0YUNnjaOho/TcTOOrPWFHsI/AAAAAAAAApo/KMwKPJilFYA/s1600/20110321-herb+shop-20717.jpg http://i0.wp.com/www.kavehfarrokh.com/wp-content/uploads/2010/12/Badgir-kaveh-Farrokh-Yazd.jpg http://danamotor.ir/Pigeon_House_Meybod_Yazd.jpg http://media-cache-ec0.pinimg.com/736x/74/ec/7e/74ec7e66ff73626f94c896d26409e2c1.jpg http://i1.trekearth.com/photos/23824/46.jpg http://news.cheapflighthouse.co.uk/wp-content/uploads/2011/10/yazd-iran.jpg http://www.radiopayam.ir/media/cache/42/92/4292531340985d682bcc0289211c0f9b.jpg http://www.pix2fun.net/wp-content/uploads/%D9%85%D8%B9%D9%85%D8%A7%D8%B1%DB%8C-%DA%A9%D9%87%D9%86-%DB%8C%D8%B2%D8%AF-7.jpg http://www.bartarinha.ir/files/fa/news/1391/4/10/91514_537.jpg http://www.yazdfarda.com/UserFiles/Image/n/n_25668.jpg http://www.sar-resid.ir/wp-content/uploads/2012/03/dar2.jpg http://mw2.google.com/mw-panoramio/photos/medium/35498024.jpg http://mw2.google.com/mw-panoramio/photos/medium/17837300.jpg
Joseph's head	https://www.behance.net/gallery/Water-gun/2117870

Appendix B: Text Messages

This table lists all the text messages that are displayed for characters of the game. These text messages are still subject to change.

Level	Character	Asset Name	Message Text
1	Victor	en_vl1m1	A gold coin covered in mud is still gold. However, to have it revealed, the dirt should be wiped off first!
1	Sebastian	en_sl1m1	Bro! You might need some seeds down the road! Don't forget to water them!
1	William	en_wl1m1	Hey! Have you locked all your valuables? Everything you hold dear should be sealed tight in a box.
1	Joseph	en_jl1m1	Get it away from me! Wet eyes need a handkerchief! Take it! Mine are dry! Take it!
2	Victor	en_vl2m1	Tears of love can wash the grime away.
2	Emily	en_el2m1	I've been trying to sell this for weeks and no one's buying, you can have it!
2	William	en_wl2m1	I dreamt about my Laura again. Her memories always make my eyes wet! I miss her so much!
2	Joseph	en_jl2m1	I ate a candle once. For a while I was a jack-o-lantern!
2	Beth	en_bl2m1	There is, like, no way I'm going to tell you my secret!
3	Victor	en_vl3m1	Unhide it, but who can open it? Maybe, the least expected one!
3	Sebastian	en_sl3m1	I'm out of beeswax, I need some to style my mustache!
3	Joseph	en_jl3m1	Give a flower to a girl and she'll tell you everything. And they think I am the crazy one!
3	Beth	en_bl3m1	Between you and me, that bird Rira knows something! Shoo it away, and you'll see!
4	Victor	en_vl4m1	A person should shed his belongings first, then he will be able to treasure what is most valuable!
4	Sebastian	en_sl4m1	Thanks bro! I will extract some wax for my mo, and will give you some too!
4	Joseph	en_jl4m1	Did I tell you I ate a candle once? That day I could touch the skies!
4	Beth	en_bl4m1	This world is so dark! We need some candles! All I need is some wax, a wick, and some sort of container.

5	Victor	en_vl5m1	Well done, my son. Those who free themselves from attachments deserve a leading light in their hearts!
5	Emily	en_el5m1	Oh dear. My bamboo mug broke. Maybe you can find a use.
5	Sebastian	en_sl5m1	You can have the extra wax, dude. Time to impress the ladies...
5	Joseph	en_jl5m1	Flossing with a candle wick isn't good as floss. It's better as a wick!
5	Beth	en_bl5m2	Great, I'll make the candle, but did you expect me to set it up for you?
5	Beth	en_bl5m1	Good! But what about the other ingredients?
6	Victor	en_vl6m1	Yes, Beth gave me the candle. I will place it in your chest. But do not forget, for no earthly fire can light it!
6	Moonak	en_ml6m1	Hoor is my love, my light. Please ask her to fill this jar with her presence.
6	Hoor	en_hl6m1	Has Moonak confessed his love for me?
6	Hoor	en_hl6m2	I love Moonak too, give him this!
6	Moonak	en_ml6m2	Hoor has bathed me with joy! Let me light your candle with some of this!
7	Victor	en_vl7m1	Let this candle lead you through the darkness. You should not stop anymore! Run to the end!
	Adam	en_algm1	There is something hidden beneath the dirt! I can feel it!
	general	en_glgm1	Blah...
	general	en_gthm1	Thank you!
	general	en_gthm2	Oh!

Appendix C: Import Character Procedures

The following steps were used to import a new character into the game.

1. Slice character image into body parts and layout them in a .png file with transparent background. A max size of 1024x1024 is good for the file.
2. Import the .png file in the Unity, choose the Sprite Mode as Multiple, click the Sprite Editor button and
 - a. Define body parts regions. If the automatic slicing is not working, draw a rectangle around one body part manually, and it will be converted to a slice.
 - b. Choose the pivot points.
 - c. Finally rename them, then click apply. This will make a sub-folder and put the body parts in it.
3. Drag body parts on an empty spot of the game scene, and locate them in the correct positions.
4. Arrange them in a desired hierarchy, and name the root as CharacterSprite, in a way that lets the animations work properly. For example, beard sprite should be a child of head sprite, so when the head is tilting, the beard also tilts with its parent. If there is no single root available for body parts, make a new empty object and call it CharacterSprite, and move all sprite objects under it.
5. Arrange the "order in layer" parameter in the sprite rendered component. Every object that should be rendered on top of another object, should has a higher number for this field.

6. Grab them all and resize them to the desired size. (This will change each part scale parameters individually, therefor the parent container will have a (1,1,1) scale factor.
7. Rotate the world to the point that you want to place the character.
8. Duplicate CharacterName game object, call it CharacterNameBodyContainer (For example EmilyBodyContianer). Make it a child of Character. Grab the whole character sprite objects and put them under CharacterBodyContianer.
9. Copy the transform of the CharacterName to this object, then make it as the child of the parent CharacterName.
10. Tag the Character game object as CharLeft or CharRight, depending on the positioning of it.
11. Tag CharacterBodyContainer game object as HasLookAtCamAnim
12. Before adding the rotation and reset animations towards the camera, add the highlight and talk animation. These 2 animations should be added to CharacterSprite (because they're easier to make before the character is rotated).
 - a. But first, try to see if the rotation can work well on CharacterNameBodyContainer.

Sometimes due to some scaling/rotating issues, the rotation towards camera works very weird.
13. Select CharacterSprite in Hierarchy, open the Animations panel, and add a new animation called CharacterTalk to this object.

- a. Doing this, automatically adds the animator controller component called CharacterSprite to this game object. And a new file with the same name will be created in the same directory.
 - b. Change the animation fps to 30.
 - c. CharacterTalk animation length should be between 1 second and 2 seconds.
14. Animate body parts for the talk animation, and use curves to smooth them out.
15. In animation panel, create a new animation called CharacterHighlight. This will be a 2s animation that changes the color of character when the player drags items from inventory on them.
- a. At the starting and ending key frames, the color of Sprite Renderer should be white (255,255,255).
16. Select all body parts, add a key frame for the second 1, and select the color of Sprite Rendered component to (126,204,111), a light green color, and then revert it back to white for the key frame on 2s line. Smooth out the animation.
17. Open the CharacterSprite animator controller. Remove all existing states from the base layer. Make a new layer and call it CharacterTalk. Drag CharacterTalk animation from Project into this layer.
18. Set the Layer Weight to 0. Set the Blending to Additive.
19. Make a new layer, name it CharacterHighlight. Drag CharacterHighlight animation from Project to this layer.

20. Set the Layer Weight to 0. Set the Blending to Additive.

21. By doing these, highlight animation should work automatically, but for the talk, you have to write code.

a. To start the talk animation for this character, run this code whenever required:

```
this.InteractionAnimator.SetLayerWeight((int)Utility.Constants.AnimationLayers.CharacterTalk, 1);
```

b. To end the talk animation, run this code:

```
this.InteractionAnimator.SetLayerWeight((int)Utility.Constants.AnimationLayers.CharacterTalk, 0);
```

c. Since I am just changing the weight of the layer to start and stop the animations, and I have set the layer's blending mode to Additive, highlight and talk animation can run simultaneously on the character object.

22. Select CharacterBodyContainer and make 2 animations for it: CharacterRotate rotates the character towards the camera over 2 seconds, and CharacterReset reverses this animation. This will automatically create the CharacterBodyContainer animator component for this game object, and the CharacterBodyContainer.controller file will be automatically added to the same folder, where .anim files exist.

a. Make the CharacterRotate animation as 30 fps.

b. For CharacterRotate animation, make sure to set the environment camera to 2d, so while animating, you can only rotate it around the correct axis, and no other parameters are changed.

c. Smooth out the animation by curves.

d. Reset animation has only 1 frame and resets the character to its initial rotation.

e. For reset animation, go to 2nd frame and make a very slight change in a rotation property. It works like it is not animating anything between 2 frames, and it looks like the character is not moving at all.

23. Open the CharacterBodyContainer controller in the animator window. Make a Boolean parameter for the animation, call it IsLookingAtCam. When IsLookingAtCam is false, it change the animation state from rotate to reset, and when it is true, it changes the animation state from reset to rotate.

24. Make the CharacterReset as the default animation.

25. Click on CharacterRotate and CharacterReset animations in the Projects pane, and then uncheck the Loop Time option in the inspector.

26. Add a box collider to CharacterBodyContainer game object, with no trigger and a Rigidbody with no gravity. This will let the game object to collide with activation plane and reset plane.

27. Add CharacterInteractionController script component to CharacterBodyContainer

28. Add a new class file like this:

```
public class Character : InteractableCharacter
```

a. You can copy and paste an existing character class file like Victor.cs.

b. Don't forget to rename all fields, and remove all not-general logics.

29. Add a field for this newly created class to Chars class, and then add this code to Chars constructor:

```
public Character;
```

```
...
```

```
Character = new Character() { Name = "Character" };
```

30. In the file Game.cs, inside WireCharacterGameObjs() method, add a new case for this character.

Appendix D: Questionnaire

The following questionnaire was administered to players after testing *Before Eternity*:

- A. Please enter your age, gender and level of education.
- B. How often do you play video games?
- C. How familiar are you with the "Adventure" game genre?
 - 1. What is your first impression of *Before Eternity*?
 - 2. What is *Before Eternity* about? What is the game trying to tell you?
 - 3. Was the game successful in telling its story and giving you the impression?
 - 4. How would you describe *Before Eternity* to someone who has not played the game yet?
 - 5. What do you dislike in *Before Eternity*? How would you make it better? What is missing?

Appendix E: Game Walkthrough

1. Click on Ashton (playable character) body, and read the message.
2. Talk to Victor.
3. Talk to Sebastian. He gives some seeds.
4. Plant the seeds in the empty flower pot that is located after William.
5. Talk to Joseph. He gives you a handkerchief.
6. Talk to Victor.
7. Talk to Emily. She gives you an empty bowl.
8. Talk to William.
9. Give him the handkerchief.
10. William gives back the wet handkerchief.
11. Use the wet handkerchief on Ashton's body. A locked chest appears.
12. Use the empty bowl on the fountain to grab some water.
13. Use the full bowl on the flower pot to water the seeds.
14. Talk to Victor.
15. Talk to Sebastian. He needs some beeswax.
16. Pick a rose
17. Pick the hive from the tree.
18. Talk to Joseph.
19. Give the rose to Beth.
20. Click Rira a few times until the poem it is singing contains a golden key. Pick the key.
21. Use the key on Ashton's body. The chest gets open.

22. Click on the chest to retrieve the items.
23. Talk to Victor.
24. Donate all four items retrieved from the chest to the characters on your way. You can give any item to anyone you would like.
25. Give the hive to Sebastian.
26. Talk to Beth.
27. Talk to Victor.
28. Talk to Emily, she gives you a mug.
29. Talk to Sebastian, he gives you a block of wax.
30. Talk to Joseph, he gives you a wick.
31. Give the wick, the wax, and the mug to Beth.
32. Talk to Victor. He installs the candle in your chest.
33. Click on Moonak, he gives you an empty bulb.
34. When Hoor, the sun, appears, give the bulb to her. She fills it with light.
35. When Moonak appears again, give him the shiny bulb. He lights the candle for Ashton.
36. Talk to Victor.
37. Press the space bar and do not release it until the end screen appears.

Appendix F: Screenshots

