OREGON GAME
PROJECT CHALLENGE

Achievements Guide

2019-2020

13th Annual Oregon Game Project Challenge

www.ogpc.info

Introduction

Games will be evaluated at the main event by a panel of judges consisting of community volunteers from university gamers to experienced industry professionals. Judges will assess games within five categories: Programming, Game Design, Art and Assets, Theme and Story, and Management. These categories are important to any game, and we encourage teams to work to meet as many achievements as possible.

Games are scored at the main event via earning achievements. Each achievement has a point value which contributes to one of the five scoring categories.

Standard Achievements can only be earned once. For multi-point achievements, teams earn the highest tier based on their work. For example:

Achievement	Description
Multilevel Madness	Have more than one level
	Have more than three levels

In this example, a team would get the tier that they qualify for based on the amount of work they chose to put into this achievement.

Choice of game style could mean that fewer points are available. For example, a text adventure game could do well in Theme and Story, but would score poorly in Art and Assets. In a similar fashion, a Unity game with default assets could score well in Game Design but would also suffer in Art and Assets.

Extra Credit achievements are scored in a different manner than the standard achievements. The first one you get is worth the most points and subsequent achievements are worth diminishing point values. Feel free to get them all if you have the time but don't worry too much if you can't, there are far more points to get in the Standard Achievements.

High scoring games in single categories or across all categories will receive awards as the judges see fit. Other awards are given based on judges' or people's choice. Teams compete only within their own middle school or high school division.

Programming

Standard Achievements

Git Gud

Use source control - or back up multiple versions of your source code.

Source control is a critical part of keeping track of your work, how it looks over time, and knowing who made each change. Almost every professional development group uses it to keep track of their code. Solutions like Dropbox, Google Drive, and OneDrive are an acceptable starting point and ensure that your game will survive losing that one flash drive! For advanced teams, Git, Mercurial, or TFS are more professional grade source control solutions. There is lots of documentation on these options so feel free to give them a try!

Squash 'em Flat

Describe a logic bug (not a typo!) you found and how you fixed it.

A bug describes a mistake in the logic of your code. It’s more than just a typo. If you forget a semicolon and your code stops compiling, just add a semicolon and move on. A logic bug is specifically when you expected a variable to be in a certain state, and it wasn’t. These bugs often require the programmer to make changes to their code: adding a branch of logic to handle a specific case, making sure your math is correct, or even just making sure that you’re not overstepping the end of an array. The changes required to fix a logic bug are usually more complicated than a simple typo.

Smooth Operator

Game does not crash, lock up, or experience any game-breaking bugs during judging.

A well-built game should perform well, even when being watched! If the demonstrator messes up and doesn’t do something right, that is not a bug. Glitching through walls, breaking the physics engine, failing to pick up an item on keypress, or causing the game to stop responding/crash are all to be avoided!

Code It Forward

Code is clean, well-documented, and variable/function names represent their purpose.

Clean code should be easy to read with simple names and easy-to-follow logic. This also includes comments, tabbing, semicolons, and functions that are as simple as possible (no monster functions that do everything!). Example:

```
class Actor {
	constructor() {
		// Health will be overwritten when extending
		this.health = 1;
		this.alive = true;
	}

	// Base takeDamage, will be overwritten when extending
	// If health is too low, actor will die and be removed from scene
	takeDamage(damage) {
		this.health -= damage;
		if (this.health <= 0)
			this.alive = false;
	}
}
```

Circle of Life

Game has a time-based system (real time or game time) such that the world changes around the player (weather, time of day, plant growth, tides, different missions).

One way to vary gameplay, introduce some interesting programming challenges, and encourage advanced planning is through an in-game day/night cycle, weather pattern, or other similar time-based event system. Many popular games make use of rain, sunlight, sandstorms, snow, fog, and other such atmospheric cycles. Think of games such as Skyrim with its weather and day/night cycles or Pokémon games such as Sapphire or Omega Sapphire that use a tidal system (Shoal Cave) and use time to track berry growth and Eevee evolutions. Some games such as Final Fantasy Tactics use time to change the missions you can accept, and which cities get attacked by challenging clans. While it would be tempting to simply swap assets for darker ones on a given map, we're looking for a real-time change (since this is a programming achievement!). Think about how a level or map could change, what events could be introduced, or which items can be picked up depending on in-game time passing.

R1, L2, Down, L1, R1

Use a console or hidden menu to change or view the game state for the sake of testing (Default console commands are not allowed!)

Simply having a console is not enough (if you’re using Unity, Unreal, or something similar that already has a console, you’ll still have to do work to get this achievement). You must show custom functionality in the console that is specific to your game (again, turning off collision because your engine exposes that does not count, show us your code!). Have a cheat code to jump to different levels or time of day, have a code to give the player a pickup, or have a cheat to allow super jumps would all be great examples. You can also use this console or hidden menu system to jump to specific things you want to show the judges.

Leaderboard

Include a save system, high score table, or some kind of persistent, local record of previous plays of the game.

This is pretty simple in concept: save to a file, use localStorage, or do whatever you need to do to make sure that a relaunch of your game can load in and show data from a previous play. There is no requirement of format of the save, or how many saves, or where you save to, just make sure that the game can access data from a previous play.

Cut to the Chase

Show a screenshot of an early build of the game running with placeholder sprites. (Programmer art)

This is also pretty simple: press the Print Screen button at some point early on so you can show the judges how your game looked at that point. There’s no specific requirement for what is shown as long as it’s your game (we don’t need to see code or anything like that, just the game is fine). Remember, we are specifically looking for placeholder art so we don't want to see lots of finished work! It's ok to start programming before all of the art is done!

Order Up

Include a start menu with play and quit, as well as either options or credits. All of these must be functional.

When the game launches, it should show this menu before play starts. The menu can be in game, a separate scene, its own control, whatever you like as long as there’s a buffer between launching the game and playing it. The play and quit options should start/pause and exit the game. The options or credits should include information about your team and your game, maybe a link to your TMS entry or to a blog, developer bios, or whatever you like.

Stop! Hammer Time!

Have a pause menu with game settings (such as volume, gamma, difficulty). These also must be functional.

This has to function as a separate menu from the start menu. When the game launches, it should go to the start menu, when you pause, it should go to the pause menu. There are no requirements about which specific settings we want to see as long as there are settings. Things like volume, gamma, difficulty, graphics, or key bindings, would all be fine. There are also no requirements on the number of settings to have implemented. A pause menu with the option to resume game, change volume, and quit the game would suffice but we encourage more settings.

License to Win

Release your source code online under an open source license (i.e. GPL, MIT, CC0).

Releasing your code under an open source license is pretty simple. If you're using something like Git, make your repository public and add the license file (it's ok to steal this from somewhere, that's how the pro's do it). If you're using something that doesn't use individual source files, add the license either as a header in the file, or as a standalone file next to your game files in a .zip folder.

Extra Credit Achievements

Handyman

Use an advanced game engine, library, or other external toolkit.

There are a plethora of game engines out there but we recommend Unity, Unreal, or Cry Engine. Libraries such as Three.js, Pygame/Pyglet, or Phaser.io are also encouraged if libraries are easier than a full engine. While there is certainly merit in making an engine from scratch or using something like Scratch, there are also a large number of very powerful engines that are used by industry professionals all around the world.

Method to My Madness

Diagram some aspect of the game with a professional diagram (state/data flow, UML, activity diagram).

Understanding the data flow, UI flow, activity flow, or states your game has are important for keeping track of your game as it grows. They can also be used to help bring another person onto your team without needing to verbally describe everything. UML (Unified Modeling Language) is a good place to start with this one as it contains well-documented descriptions of many types of charts and diagrams you can make for your game.

You're All in It Together

Game uses a server to track high scores or other statistics globally (cloud saves or accounts would work too).

This is similar to **Leaderboard**, but the saved data must go on a server somehow. Integrating with Google Drive, a database, or even a simple Node.js server that keeps track of stuff in an array are all valid methods. The server/database does not have to be persistent, we just need to see that data is being passed off to a foreign location where it will be held until requested in the same session. Don’t worry if the server happens to be running on your local machine. It’s very common to have both client and server running on the same developer station when debugging.

The Far Lands

Game features procedural content generation (like Minecraft's world generation, or a player model generator).

This can be achieved in several different ways: Minecraft is a great example of procedural content generation in level design, clouds can be generated procedurally, shrubbery or other plants could be generated, music could be generated, textures could be generated, the map could be generated, and pretty much anything else in the game. The big thing is, something has to come out of the generator that is unique, and it has to be an asset (or assets) that can be placed in the world and viewed by the player. Something like random player stats does enhance replayability, but is not procedural *content* generation.

Easter Fairy

Program an "Easter egg" into the game.

This can be anything from a flight simulator as Excel once had, to a room to show off unused assets, to something else fun or surprising that you want to include but maybe doesn't fit with the rest of the game. It could be a reference to a favorite game, meme, or song. Find a creative way to include something fun and unexpected in your game that you might not otherwise get to include!

Game Design

Standard Achievements

Top-Down

- Write a game design document.
- Keep your requirements document updated to match your final game.

For the first achievement, your design document should include a description of the game, some information on hardware requirements (PC/Ti84+/Xbox), development software (MonoDevelop/Visual Studio), and user interface to name a few (first person/third person). Also, game style, target audience, and main plot points. There are formal standards for design documents (IEEE to be specific) but we won't be enforcing anything specific. As long as you can show us that you had a document that had some design elements decided on before you started working you'll get points. If you're lost, there is lots of help online about what a requirements document should look like!

As you work on your game, you'll probably find that things will change: requirements will be replaced or removed, your story will develop, and you'll probably miss some deadlines. To get the second tier of this achievement, show the judges which aspects of your design document changed and talk about what they were before.

Eightfold

Show off a paper or other low-fidelity prototype of your game (a picture of it is okay).

This was also covered at Game Jam. For those who didn't go, make a quick-and-dirty version of your game that covers the core elements and mechanics to make sure everything plays together nicely and is fun. This should be done before development starts but can also be done as development is happening as a means of testing something out before spending the time to implement it.

Think Tank

Show a diagram of at least one element of the game logic (e.g. decision tree, player progression, or flowchart)

Again, there are formal diagrams that can be used (UML has a few) or you can be a little more informal. We're just looking for proof that you thought through your game a bit and have planned how important aspects will work. Showing with a diagram how the players actions impact their progression, using a diagram to show AI logic, or even something like multiplayer networking. If there’s a process required for something to work, a diagram can be used to show how everything connects.

Escalator

The game starts simple and introduces new features or mechanics gradually.

Most great games start the player off with a few simple features/mechanics, allow the player to get used to them, and then add more until the player has full control of every mechanic. In some games (think Skyrim or Deus Ex), the features/mechanics are introduced via the skill tree method. In some games (think Minecraft or Magicka) the features/mechanics are all available at the beginning of the game, but the players aren’t necessarily aware of them all and need to play for a while before they learn everything. Either method may be employed, just make sure that players aren’t overwhelmed when they first start playing your game!

Metamorphose

Show how your game changed after a design review. (Screenshot, documents, or meeting notes)

If your team is keeping up to date with your design document, you should be able to recognize when you start to move away from your original goal. If this is an intentional change, update your design document to get the Adaptive Design achievement. If this is not an intentional change from your original plan, you need to take a look at what you were originally trying to accomplish and make changes to your game. When you have to make changes, show a screenshot from before and after the change to show how you made your game fit your original idea better.

S Rank

The game periodically gives feedback to the player based on their performance and actions.

There are a few ways to get this achievement and they’re all perfectly valid. The easiest to think about is a HUD (a health bar, map, selected ability, or sprint bar would all be examples of things you might find on a HUD) rendered on top of the game. Another option would be to adjust saturation levels as the player takes damage, gets tired/hungry/thirsty, or runs out of air, or remove saturation from the screen to indicate that there’s a problem. Another method would be to have someone speaking to you over a comms device, have an NPC follow you around and talk, even put indicators on the player character themselves. We’re just looking to see that the player is given some indication of what’s going on in the game regardless of your specific implementation.

Groundhog Day

Game allows the player to have a notably different experience on each replay.

Everyone loves a game they can come back to and play again and again. Whether your game allows the player to take a different path, to solve puzzles in a different way, or uses procedural content generation to put the player in a completely new world, if the player is able to do different things from one game to the next, you'll get the points for this one.

Bribery

- Get people outside the team to playtest your game.
- Explain a change you made to your game because of feedback from a playtester.

For the first tier, it's important to remember that playtesting is an important part of any game creation process. It allows your team to verify that the game you designed is fun, engaging, and communicates what you were trying to communicate. A good playtesting session should show you areas that need improvement, things you've done well, and any additions you maybe need and haven't thought of. It's a great way to get some feedback on your game before you have a panel of judges looking over your work and giving you a score! To get this achievement, show some photos or notes from the playtest.

To get the second-tier achievement, you must explain to/show the judges what you learned from your playtest session and what you did to address what you learned. Was there anything you found that was really confusing? Was there anything you found that people really liked? Were there any bugs you only found when having other people play your game?

Taking Aim

Describe the game's target audience and what decisions were made to make the game appeal to them.

Every single game has a target audience: that is, the people the game was designed for. A few examples would be "Single moms from 20-40 years old", "retired coal miners from Alabama", or "11-18 year-olds like us."   To make your game appeal to whatever audience you have chosen, you will probably have to make some decisions about what is appropriate for those players. Many adults are going to enjoy games that make them think, little kids will often enjoy puzzle games, and most retired coal miners probably want simple mobile games. There's no right or wrong here, just make a convincing argument for your audience and related choices.

Stage Fright

- Have at least two different game levels or environments.
- Have at least four different game levels or environments.

Level design can be one of the more time-consuming aspects of creating a game. Whether purely visual (think of the backgrounds in Mario Kart, they don't change the race, but they make it interesting), or more heavily linked to game play (think biomes or dimensions in Minecraft) crafting multiple levels with love takes time and effort. Even open world games such as GTA or Spore have to put thought into designing all the aspects of the varied environments one can find in their respective gameplay.

The main idea with Stage Fright is that your game has more than one setting for players to explore which helps keep your players engaged through the whole experience of your game. Your varied environments could be independent levels, separate areas in one level, or the same area in a level but different textures and/or models. Guacamelee day/night levels are a good example of using the same level but still having a varied environment.

Stage Fright is fairly straight forward: to qualify for the first level, make at least two levels or environments, and for the second level, make at least four. Remember, this is not a programming achievement. Showing the design and planning that went into a level that the programmers didn't have time to implement is ok!

Extra Credit Achievements

Cha Cha Slide

The game communicates new mechanics and challenges to the player via a tutorial or similar method.

To ensure that your players know how to play your game, a tutorial or other similar method of showing the mechanics should open the game. Games like Civilization or Hearthstone have a tutorial that guides the users through exactly what to click on, where to place things, and what everything does. Some games take a more laid back approach and simply tell the player what to do and then prevent them from progressing until they complete the task without the nitty gritty guide. Either method would be fine as long as the player is walked through the various mechanics of the game.

42 Entertainment

Use a physical object as part of your game (flashdrive, QR code, etc.) that contains necessary information to complete a challenge.

There are several games that have done this such as I Love Bees, Year Zero, and Why So Serious? In these games, players had to use flashdrives, cellphones, cakes, pay phones, and all sorts of real-world devices and objects to progress in the game. For this achievement, hide a required piece of information (code, puzzle solution, description of item to pick up) on a physical item (flashdrive, QR code, SD card, even a floppy disk or something you make up) that you scatter around the main event (make sure you know where they are so you can pick them up after the event!). Be sure to have one of these items on hand so you can show the judges how this works. Also, label the physical object with your team name and table number (assigned at the main event) so that people know where to go when they find one of your objects.

It's Over 9000

Include multiple difficulty levels for players to select from.

Have you ever played a game and decided to raise or lower the difficulty of the game while you were playing? Many games have multiple levels of difficulties which can do anything from change the number of enemies or health, to change the time limit, to modifying any number of other stats or game variables. Teams can use a slider or discrete levels and can modify any number of variables or other in-game means to adjust difficulty.

Strategy

Demonstrate how the player can use multiple approaches to solve a given problem.

To get this achievement, you must demonstrate that your game allows players the freedom to pick a playstyle or to use different skills to complete the same challenge. Think of a dungeon in Skyrim. You can fight your way through the guards using whichever method of fighting you've leveled, you can lockpick your way into the building and bypass everyone, or you can use stealth to sneak past everyone. These options in gameplay make the game more fun and allow people to play the game the way they want.

With Style

Create a style guide for your game of at least one page. Should detail elements such as visual style, mood, programming standards, and voice.

A style guide is an important part of any major project. This will encompass many aspects of your game including programming standards, mood/voice, and artistic styles. The basic principle of a style guide is to allow a new team member to have a single document for reference on "how we do things". If the new member is trying to program, there should be information on naming conventions and where to put curly braces. This will probably require most members on your team to write a little bit about what they're doing! If you need more information on style guides, Valve has a character style guide from Dota 2 available: http://bit.ly/2hCTtgz. Note that this example does not cover anything other than character design, so we would be looking for a more broad document than this, but it should give a pretty good idea of the types of things to include.

Art and Assets

Standard Achievements

Maker's Touch

- At least four of the game's graphical assets were made by the team
- At least ten of the game's graphical assets were made by the team.

Graphical assets include everything from character sprites to foliage models, particle effects to skybox/backgrounds, and even UI elements such as buttons. Editing an asset made by someone else does not count, but, for example, making a texture for someone else's model will count toward this achievement (but not the other person’s model). Remember, any assets not created by your team must be cited!

What a Concept!

- Show off concept art you made for your game
- Post art online starting at conception and ending at production (Tumblr, Deviantart, blog, Cartrdge, TMS)

For the first part of the achievement you only need to bring your concept art to the main event and display it in some way. If some of your art wasn't implemented feel free to bring it anyway, as long as you have some art that did make it into the game.

The second part, you must post art online *in addition to* displaying your concept art at the event. This can be via Tumblr, DeviantArt, or a similar platform dedicated to your art and the progression of your designs, or it can just be posts on the blog your team may have made for the production of the game in general. Either is fine, but be sure to post not just your original sketches but also works in progress and finished art.

Firebrand

Both the team and the game have logos (there need to be two distinct logos).

This is pretty self-explanatory. Your logo can be stylized typeface with some kind of art included, or you can make the whole thing from scratch. As long as you as you have one for your team and one for your game you're golden. Hint: Posting these on TMS is a good idea and will count towards other achievements.

Font of Dreams

Game does not contain any system fonts (looking at you Papyrus, Comic Sans, and Times New Roman).

Default fonts include Arial, Times New Roman, Comic Sans, Papyrus, or pretty much anything you'll find in Word's font menu. You can use default fonts where appropriate (in a copyright statement or your credits for example) but having them in your menu or in the actual gameplay is a no-go.

All that Glitters

Enhance the game with decorative visual effects (particle effects, atmospheric effects).

Have you ever sent something up in a satisfying puff of smoke, or had sparkles float around after you've cast a spell? These are the kind of effects we're looking for. It should be said, however, that any kind of gore effects not only won't count toward this achievement, but could disqualify your entry!

Tracklist

- Include a soundtrack
- Include a soundtrack with three or more tracks

For the first level, your game just needs a soundtrack of some kind. Go find a cool song to play on the menu or during game play to liven things up a bit!

For the second rank, your soundtrack needs at least three tracks. You don’t have to create them, but as always, make sure everything is licensed and cited properly.

Beethoven

- At least one track in the soundtrack is composed by the team.
- At least three tracks in the soundtrack are composed by team.

For the first rank, you need to write at least one song in your soundtrack. There can be other songs included from other sources, but at least one must have been created by a team member.

For the second rank, your soundtrack needs at least three tracks created by your team. There can again be other tracks included that were not made by your team, but at least three must be original compositions.

Foley

- Game has a variety of sound effects
- Game features a variety of sound effects made by team

To get Foley level one, you need to have a few different sound effects to spice up your game. Footsteps, dings and bells for checkpoints or finding important objects, explosions, birdsong, you get the drift. Does not need to be made by you.

To make it to the next level, get out there and stomp, whistle, sneeze, laugh, and record your way to extra points.

Poster Child

Create a full-size promo poster for your game (18x24 inches+, like movie poster, not poster board, school appropriate)

Though we encourage blog posts and trailers in other achievements, this is about making a single image to promote your game. Think of the iconic Star Wars posters or other movie posters as you see them outside a movie theater. It should probably include a character from the game, it might have a catch phrase, it should have the name of your game, and should also have the names of the people that made it. You don't have to use in-game footage, many professional grade posters use drawn and stylized images that might not ever be present in the actual game. Make something to get people excited!

Colors of the Wind

Make and use your own color palette and explain your choice

A generic color palette is just a group of colors you choose, but to get this achievement you need to do more than that. Choose a type of color palette (Monochrome, Analogous, Complementary for example) and explain your choice. If you decided not to go with a typical color palette, explain why you chose the colors you did. Displaying your palette along with your concept art is a good idea.

Citation Needed

Have a credits page in the game that cites where all external resources came from.

As per OGPC rules, all assets/resources that are not made by the team must be cited. That could be music, models, textures, fonts, sound effects, etc. For this achievement, create an in-game credits screen that shows these citations as well as lists the people on your team and what each of them did. It is extremely important that teams do not plagiarize or steal work that is not theirs. Some resources may carry licenses of their own that require including that license or restricting use. Respect the licenses of everything and put whatever is needed on a credits screen to ensure copyright and licensing compliance.

Extra Credit Achievements

Master Craftsman

All of the game's graphical assets were made by the team.

Whether sprites, 3D models, animations, particle effects, skyboxes/backgrounds, or even UI elements such as buttons or checkboxes, everything must be made by the team. Keeping a consistent graphical style is an important part of a game’s atmosphere and ensuring a consistent user experience. The only way to ensure that everything blends together perfectly is to make it all yourself. “All graphical assets” means **all**, so make sure everything in the game was created by a team member.

Steamboat Willie

Game features animated graphics in multiple places

Animated graphics refers to objects and/or characters with multiple frames. Just moving a sprite is not sufficient. It could be a tree swaying back and forth in the background, a walking animation, a bird flapping across the sky, something growing or hopping, or twirling. They can be simple, as long as they have more than one frame.

Dimensional Shift

Make and use 3D models in your game (or 2D sprites generated from your 3D models)

The first part should be straightforward. Make your own 3D models and use them in your game. The second part just means that you've decided to model something, but since your game is 2D you create a sprite from your model and use it in your game. This can be useful, but we'll let you figure that out on your own.

Drums of War

Audio changes dynamically based on gameplay (e.g. music cues respond to gameplay or mood affects music)

Have you ever noticed how the music in lots of games gets really intense when you start a boss fight? That, but not just boss fights. Your audio should respond to, or be indicative of, something that is happening or is going to happen in your game. You can make your player nervous, excited, relaxed, or many other emotions depending on how you use sound in your game.

Talkie

Use voiceovers recorded by the team (for dialogue, exposition, narration, etc).

Let's hear those beautiful voices! Lend a little realism to your characters, give your game some back story, or make witty comments on how your player is progressing. There are lots of games that have not only given their character voices, but made very effective use of a narrator (Bastion is one great example). If you’d rather your main character not talk, give voice to a secondary character, it could even be a parrot! Give your game a voice!

Theme and Story

Standard Achievements

Theme Hospital

- Story, art, and assets are somewhat connected to the theme.
- Story, art, and assets are all completely connected to the theme.

This is a very important achievement! The theme ought to be evident in every aspect of your game from large too small. Game objects, dialog, sprites/models, music, and level design all need to be consistent with one another and relate to the theme. A game that has some element of the theme, but isn't consistent or fully connected to the theme would only qualify for the first tier. This is not to say that your game can't have aspects that vary from the theme, just be sure that everything is consistent within your game's universe. You can even go crazy here, we encourage students to find creative interpretations of the theme, just be sure that everything is integrated and consistent. The game should be truly integrated with the theme across setting, story, music, and visuals. If you can show how the theme is used in all aspects of your game, you'll get the higher tier.

Storytime

Game communicates a story of some sort (No Tetris).

Everyone loves a good story. Pick something that sounds interesting to your team and include the story in your game. You'll again have to ensure that the story is evident in all aspects of your game, but as long as there is a story and it's integrated into gameplay, we don't care what the story is. Ninjas in space, subterranean farming, or the tales of Boaty McBoatface, have fun and be creative!

Exposition

Game environments contain information about the story (collectible notes, logs, meaningful setpieces).

Ever had to collect all the pictures to learn the backstory of a particular character? Or maybe you've wandered around a dark forest collecting notes and avoiding the most slender of men? For this achievement, you should have items in your game that mean something and contribute to the story. It can be notes, vases, statues, debris, or any other item as long as it adds to the story and requires the player to find them.

Detail-Oriented

Story is revealed progressively via cutscenes or meaningful player/NPC interactions.

Rather than putting all of your story at the beginning or end of your game, players should be led through the story as they play. Cut scenes, NPC dialog, an internal monolog, radio chatter, or even forced dialog trees are great examples of ways the story can be told progressively. Don't mix up this achievement with **Exposition**. Having elements to find is great, but for this achievement we want the player to be presented with the story smoothly blended into the player experience.

Ludonarrative

Theme is expressed through gameplay (the player's actions must reflect the theme).

This is similar to **Theme Hospital**, but focuses more on player actions. Does the player’s choice to sneak past the guards say something about the theme? Maybe the player can explore different parts of the map that relate to the theme, maybe an item or use of an item connects to the theme, or maybe you come up with a new game mechanic that relates to the theme. The theme should be evident in all aspects of your game and the player’s actions and decisions are not exempt. Make sure everything the player does is related to the theme!

Far Far Away

- The story takes place in a well-defined setting.
- The story takes place in multiple, well-defined settings that give meaning to, and enhance the plot.

The setting of a game is very important (especially when the theme could be interpreted as a setting) and should be something your team spends some time on. The players should be able to easily determine where they’re currently playing, whether underground, in space, on a boat, or wherever your game takes place. For the first rank, simply make sure that whatever the setting is for your game, it is clear to the player through the entire story arc.

For the second tier, the story needs to make use of multiple settings. Incorporating different environments into the story can be a bit challenging sometimes, but simply using varying backgrounds like Mario Kart would not qualify for this rank. We’re really looking for something where the story makes use of a new environment. Maybe the player needs to sneak past someone, so you wait until night and crawl past under the cover of darkness. Simply allowing the character to enter a house, like in Pokémon, for the sake of talking to someone or gathering generic resources would not count.

E for Everyone

Both gameplay and story are family-friendly. (It can still explore mature themes, however explicit material cannot be judged.)

This is one of the most important achievements. We cannot allow teams to demo games that show blood, gore, sexually explicit material, politically charged content, or anything else that would warrant anything above an E10+ rating for your game. If your game contains explicit material, we cannot allow you to show your game, art, paper prototype, or whatever else you bring that contains explicit material. You can still show off your code, talk through your design process, and other kid-friendly materials, but you will lose out on a lot of points. If your game is determined to contain explicit material, an OGPC official will have to ask you to put away the offending material. For more information, see the Rating subsection of the The Game and Theme section of the Competition Manual.

NOTE: A game may contain mature themes (death, desperation, abandonment, loneliness) while still being appropriate.

E10+ Allows:

- Cartoon, fantasy or mild violence
- Mild language
- Minimal suggestive themes

Not allowed are:

- Blood or realistic violence
- Gambling
- Strong language
- Sexual content

Exoskeleton

Create a detailed outline for your game’s story.

This could be in your design document or another document you’ve already made, but should be an overview of your story including major plot points and other key decisions. There’s no strict length or word count, as long as you’ve covered everything important and can walk a judge through your story, you’ve done everything you need to do.

Curly Braces

Story has a clear beginning, middle, and end (archetype cycles such as the hero's journey).

Every good story starts somewhere, guides you along as the characters grow and the story is revealed, and ends somewhere else. The who, what, when, where, and how of your story are completely up to you. Just make sure your story starts somewhere, has some progression, and ends somewhere else. Story writing is a very creative process, and we don't want to impose too much on the creative process. If you can argue your beginning, middle, and end and show real change to a judge you'll be fine.

Supporting Role

Story features a secondary or supporting character.

Good stories often have other characters; sometimes it's an old friend or enemy, maybe a love interest, or a close family member, but there should be at least one other person in the story. For a supporting character, the story must spend some time on this character. They can't just be a quick blip on the radar, but they need to recur and have a decent place in the story. Some examples of supporting characters are Ron Swanson from Parks and Rec, the Dursleys in Harry Potter, or Faramir in The Lord of the Rings. These characters are not main characters, but they get a decent amount of development and dialog as well as playing an important part in the story.

To be clear, this is not an enemy type. This is a character in your story, they might be an enemy, they might be a companion, they might just be a narrative voice like in Bastion, but they need to have character development.

Extra Credit Achievements

Plot Thickener

Story features character development (change over time).

Either the main character or a secondary character must grow or change as the game progresses. This can be because of the addition of abilities as long as they're thematically relevant and fit in your overarching story but can also be shown entirely via the story and not expressed in gameplay. Any character that starts as one person and ends as another will get you this achievement.

Did the Research

Research the theme and write at least one page on your findings (should be at least 500 words with at least two sources).

Every major game that explores some concept of the real world will have to do research on the facts. Whether it's Assassin's Creed, Civilization, or Call of Duty, there will always need to be some work done before development to ensure that everything is historically accurate, physically accurate, and relevant to the chosen theme and story. The page should be at least 500 words with at least two sources. This is challenging, but very worthwhile for your game!

Guiding Light

Create a reference guide (manual) to the story, background, or characters present in the game of at least one page.

For example, you might have a picture, a table of stats, and the backstory listed for all of the main characters in one little booklet; you could also have a few paragraphs in a booklet that talk through the story; finally, you could show an image and give some quick facts about all of your environments/levels, again in a booklet format.

Word Smith

Write a short story based on, or related to, your game. (At least 1200 words typed or 4 pages of a graphic novel or comic).

Writing is an important part of games but sometimes plots, characters, or chapters of a story have to be removed from the game for various reasons. For this achievement, judges will be looking for a story written in the same world as your game. The characters don't have to be the same (so it could be a story about the main character's kids, etc.), the exact location can change (maybe the game takes place in Chicago but this story is in Tokyo), and the time period doesn’t have to be the same (could be a detailed backstory for an event or other plot point), just be sure that the story is in the same world/universe as your game.

Easel

Create a storyboard and written description of at least one scene or screen in the game.

A storyboard is another very important part of your design process and story development. A storyboard helps you plan how a scene looks, who moves where, and what parts of the story are revealed. The description should talk about what is happening, and why the scene is important. Like with many things, you can go online and find many examples of storyboards if you’re having trouble thinking of what to do.

Management

Standard Achievements

Clockwinding

Show off meeting minutes or other notes regarding team decisions.

Big projects often require lots of meetings. A good meeting starts with a list of things to accomplish (the agenda) and ends with notes to keep track of what actually happened (the minutes). If someone missed a meeting, they should be able to look at these minutes to catch up. Meeting minutes are also a great way to track the history of a project. For this achievement, have someone keep minutes (or better yet, rotate who takes notes) and bring these along for the judges to see. If your team doesn't keep formal minutes, you can also just bring any notes that show the discussions you've had about decisions for the game.

Resonance

Briefly describe how two or more team members worked together on a single task.

Teamwork is a beautiful thing! Think about times that your team members worked together on the same thing to get work done. It could be two programmers troubleshooting together, two artists collaborating on the same asset, or any other example of helping each other out.

Dissonance

Briefly describe how your team resolved a conflict or disagreement.

Even the best teams don't always agree on everything. Was there a disagreement over art or music style, which game engine to use, or the way the story unfolds? Did you flip a coin? Did you have a debate? Be prepared to talk about one of these disagreements and how you worked it out.

Exhibit A

Show off your game and team at your booth using visual aids such as a slideshow or poster board.

Your booth is just a table until you get setup. Teams should use their table to create an eye-catching display for their game. Stand out with a poster board with some details about your team and/or your game. Consider a digital photo frame with a slideshow of your team working through the season. Have some swag to give away, match things with your game universe, and have fun with your display!

13th Floor

Give the game's elevator pitch (convince people to buy your game in ~30 seconds).

In the business world, the best way to get the right people to listen to your idea is to be able to summarize your idea and communicate it quickly. Your elevator pitch is your quick summary of your game in thirty seconds or less. Imagine you are stuck in an elevator with a big-time game producer. This might be your only chance to make a great impression. Don't just try to talk about your game quickly, or you will sound rushed as you try to squeeze in every detail. Practice explaining your game with just the right amount of detail. "Space Duck is a side-scrolling exploration game where you have to find all the parts to repair a crashed space ship so the hero can return home." Don't try to explain every character or detail – there are other achievements for that.

Team Spirit

Bring team business cards, a banner, flyers, buttons, pamphlets or other promotional materials.

You don't have to spend much money to really bring the team spirit. You can paint or cut out a banner, you can design a brochure on the computer, and you can create nearly free business cards from sites like www.vistaprint.com. Big shows like PAX and E3 are full of handouts to get people to notice the games. Screenshots, descriptions, team information, and your logo are all good things to add to your items.

Dress for Success

Coordinate your team's dress on event day (custom t-shirts, matching colors, themed with your game, etc.).

You know what looks cool? A whole team in matching outfits! It can be jeans and team t-shirts, or light dress pants with a dark shirt. You could even all dress up like characters in your game. You decide on your team's look and make sure everyone matches.

Blogatog

- Have a development blog or social media page for your team (minimum one post a week for one month).
- Update your blog or social media page at least once a week for two months (the more the better!).

Your TMS page is great as your official download/details page, but you can really reach out to your fans with a blog, Facebook page, or other social media. Use your logos and pictures, and publish information about your team process. You could even use it to keep your meeting notes (minutes). Fans always love to see the process of how their favorite games are made.

For the second tier, don't just create a blog or social media page and abandon it! Go back to it to post a new status or other details every few days for a couple of months. Don't let down your fans!

Skynet

- Include team/game logos, team photos, and game screenshots on your TMS page
- Completely fill out your game page on TMS.

No game on Steam or Amazon would sell very well without all the details filled out. TMS has slots for team logo, game logo, team group photo, and game screenshots. Make sure you add logos/screenshots/images wherever there's space! Not only is it worth the achievement, our awards presentation gets generated directly from TMS, so you don't want to be the team with "NO PICTURE" next to its name!

For the second tier, you've got space for pictures on TMS, but you can also upload your game or link to it so people can play it. It's so much fun to be able to play everyone else's game, so make sure you give teams the chance to play yours. Fill out the game description and instructions, set the game engine and language, and make sure that you fill out everything. Use the "Edit Game Entry" in the upper-right of your game details page.

Hype Train

Make a banner/sidebar advertisement for your game and post it on social media.

Games make the most money when they get their message in front of as many people as possible. Make a banner ad or a sidebar ad to try to get people to click it and play your game (you don't actually have to upload it to an ad network unless you want to). To make an effective ad, try to think of a few images or words that really sum up the core aspects of your game and make sure to have the name of your game somewhere!

Chatterbox

Use an asynchronous communications tool such as email, slack, remind, etc. throughout the game making process to stay connected.

Communication as a team is vital to creating a positive team environment as well as a polished game. Having trouble figuring out what to work on next, send a message or email out to the group. Not sure how this bug is happening, or why Gimp isn't working, send out a message to everyone. Found a cute cat picture you want to share with everyone, make sure your whole team gets to see it! There are many free tools teams can use such as Slack, Remind, or even just email. Pick a tool and use it through the development cycle to stay connected with your team.

Extra Credit Achievements

Giving Back

Publish an online tutorial about a problem you solved during development.

Think of how many times you got help by searching online for an article, Q&A, or video. Now it's time for you to give back to the community! Think of something that was challenging to you during game development and make a video, article, or other online tutorial to help others. Be sure to have a good title, screenshots/video, and an explanation that will make sense for other people who have the same challenge.

Media Mogul

Create and submit a press release for your game online or in print (newsletter, newspaper, school bulletin, blog, website, etc.).

Contact your local newspaper, TV/radio station, or other media to get some buzz for your team and game. Everyone loves to read about students have fun doing something cool, so make yourselves famous! It might be hard to actually get coverage, so if your team can't get a media outlet to pick up your press release, have one made and ready to show off to the judges.

Final Cut

Make a trailer for your game.

Do you drool over videos for the newest games on Steam and YouTube? It's time to make your own! Capture some gameplay, add some titles, maybe even create a custom scene. Watch a few game trailers to see what you like about them, and then make something similar for your game. Some movie maker software even makes it easy to combine clips with music in different styles. Make sure to use music and voice clips that you have permission to use (same rules as with the game). Even better, use your own content!

Manifest

Create a physically crafted mascot for your game (plushy, sculpture, papercraft, etc.).

There are so many options to bring your game into the real world! Fold an origami character, use clay, stitch and stuff some fabric, or even 3D print one of your models. It's fun to see your game characters and items brought to life. Print, stitch, fold, glue, or whatever you need to do to create your items and make them part of your booth.

Taskmaster

Make milestones, break development milestones into smaller tasks, and assign time estimates to each one. (Trello/Jira/Asana/Excel)

Project management isn’t easy, but it's a crucial step in keeping a game on schedule and having it ready to compete with at the Main Event. Generally, there are fewer milestones than there are tasks: think of "complete all character animations" or "complete rough draft of story" as milestones and "make a pick-up animation" or "decide on a character arc" as smaller tasks. Assign time estimates for all of your smaller tasks, add them up to see how long you expect your milestones to take. Tools such as Trello or Jira (feel free to use whatever tool works best for your team) can make this process easier/faster and having everything in an electronic format makes it easier for everyone to be checking in even if they're working on tasks from home.