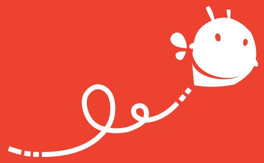
FilamentGames™

www.filamentgames.com

☎ (608) 251-0477



Playtesting

What Is Playtesting?

Making software--any software, whether it's an app on your phone to help you manage your finances or a game for the Playstation 4--is hard. Making good software that doesn't drive your users nuts is harder because you have to pay attention to something called usability. People have brains, and brains have limits in how much they remember, what they notice, and how long they can pay attention; software is usable if it is designed in such a way that it conforms to those limits. Usable software is intuitive and effective for users to accomplish whatever task is at hand; unusable software is clunky, overly complicated, and confusing.

*Digital games need to be **usable**, but they also need to be **playable.***

Digital games need to be usable, but they also need to be playable. It's not enough for a player to easily navigate from one menu to another and interact with buttons that accomplish what players think they'll accomplish; instead, games need to be engaging and fun on top of being usable. No one ever played Morrowind for its fabulous user interface (UI). (Though I can think of a few otherwise-fun games that are rendered unplayable because of their terrible UI.)

Game studios employ a technique called playtests. Playtests allow developers to test game usability--where are players getting stuck or frustrated?--in addition to finding out whether the game is fun to play. In Filament's case, we also use playtests to measure the efficacy of our games as learning tools.

This all sounds very complicated, but conducting playtests is fairly straightforward if you know the questions you're trying to answer and have a plan in place.

Questions and Playtest Goals

The most important part of conducting a useful playtest is knowing what you're playtesting for. Don't ever go in just to "see what happens." You'll see things happen. Whether what you see will make any sort of positive impact on what you do depends on how you're focusing your efforts and your players' efforts. This is not to say that exploratory playtests aren't useful; however, even when you try to see the game "in the wild," you still want to have particular questions in mind.

To determine the goals of a playtest, think about what questions you most need to have answered. Early on, it's useful to conduct playtests of paper prototypes to confirm the core game loop works and is fun, or that wireframes of menus are intuitive. You might also see what the target audience thinks about concept art and style. At Alpha, you might test a rough digital version of the core loop for engagement and comprehensibility. At Beta, you might see whether players think the different project pieces are coming together to form a coherent whole.

Testing goals and questions can be abstract or concrete, subjective or objective, and will vary widely depending on the phase of the project. Abstract questions are best answered through discussion following the test; concrete questions are best answered with observation and specific metrics such as time-on-task. In any case, specific questions are generally more useful than vague ones.

HERE ARE SOME EXAMPLES:

ABSTRACT

- What aspects of the game do/don't players enjoy or find interesting? Why?/Why not?
- Are players confused by anything in the game? How can the game help players understand better?
- How do players react to the overall aesthetic of the game? The art, the characters, the sound, etc?

CONCRETE

- From the start menu, can players begin the main game? How long does this take?
- Can players navigate from the game to the menu to adjust voice-over volume? What is their menu path?
- On level five, about how long does it take for players to realize they can adjust the weight of the catapult's frame? Do they? How many attempts do they make on level five before they change the frame?

Whatever you do, make sure that the goals you set for the playtest are actionable for the stage of development you're in. Early on (during paper prototyping or alpha) it makes sense to test the core game loop, because you can iterate and revise the core mechanics without having to throw away a lot of work. It makes less sense for you to test whether the core game loop works and is fun if you're in beta; by then, trying to change core components of the game will cost a lot more.

Playtest Planning and Implementation

Just as you don't want to playtest "just to see what happens," you don't want to go into a playtest without any sort of plan or protocol in place. A protocol is more than a to-do list. If done well, a protocol can help you coordinate and run a playtest with few (if any) hiccups.

After you've determined that you want to playtest your game and what questions you need the playtest to answer, you'll need to recruit 'target audience' playtesters. In Filament's case, most of the time we need to playtest games intended for use by students in classrooms; hence, over the years we've developed a network of teachers who are enthusiastic about technology and who are willing to let us visit their classrooms to test games with students. We've also partnered with youth organizations and libraries in order to test games that are built for less formal learning environments.

Once you've found playtesters and arranged a date and time, you need to make some decisions regarding how your test will run. As a matter of course, you'll establish your ***technology needs***, your ideal ***test setup***, and a ***test guide***.

TECHNOLOGY NEEDS:

A key consideration here is making sure the game works on site in advance of the test.

If you're testing a digital game, you need to make sure that you'll have appropriate testing devices on hand for the playtest. If you're working with a school, sometimes they'll have access to devices--Chromebooks, PCs, tablets--that can be used if all playtesters need to do is go to a URL. Typically, however, you'll need to bring devices with a build of the game pre-installed. This helps save time and allows you to control a critical component of the overall testing environment.

A key consideration here is making sure the game works on site in advance of the test. This is less of an issue if your game doesn't require internet access to work (as in the case of a pre-installed app), but any kind of networking or internet features will absolutely need to be checked beforehand. It can be as simple as going to the test site a day before and launching the app on the local internet to make sure it works.

IDEAL TEST SETUP:

> *Think carefully about this setup because it may have consequences on the types of data you're able to collect.*

How do you want the playtest to run, overall? Are playtesters all together, playing side by side, chatting with each other as they play, while observers pay attention to the general flow? Or do playtesters get paired one-on-one with an observer and play away from other testers? In our experience, having playtesters play all together is a useful way to see how a game would function in the wild, but it's easier to pay attention to a specific play experience if you work with testers one-on-one. Think carefully about this setup because it may have consequences on the types of data you're able to collect.

TEST GUIDE: SEQUENCE OF EVENTS & SCRIPT

The test guide is the roadmap for the actual playtest. (See below for an example.) The guide can take many forms, but typically it will include a sequence of events, a breakdown of roles and responsibilities, and a script that includes setup and questions to ask.

SEQUENCE OF EVENTS:

- A checklist that spans the entire process--pre-test, test, and post-test--and makes roles and responsibilities clear.

PRE-TESTING PHASE:

- The preparatory work that must be done before a playtest is possible. Things you'll do during this phase include creating a live, testable build of your game, getting together your technology, testing out the build on-site (if possible), securing permission/ consent depending on the age group, and preparing any sort of presentation or out-of-game tutorials.

TESTING PHASE:

- The actual on-site test with real playtesters. This phase will make use of a script and includes setting the stage with playtesters, play time, observation, and interviews.

SCRIPT:

- Your tool for handling the testing phase. In it, you'll include introductory information--who you are, what you're testing—as well as specific instructions for getting the test started, and ending questions for wrapping up the playtest. The script can be a literal script that you read word-for-word, or it can be an outline similar to the sequence of events. What matters is that the script is a useful resource for running the test.

POST-TESTING PHASE:

- The last part of a playtest includes processing observations and information such that it can be acted upon. Detail is important: the more thorough you are when planning this sequence, the less likely there will be confusion and the more likely it is that you'll execute a successful playtest.

A Note about Notes

As you watch your playtesters play and listen to the things they say, you'll be tempted to write down everything. While recording as much information as possible might seem like the smartest thing to do, you might spend more time writing than observing and miss something important. Take just enough notes that you can refer back to them and remember what was happening. In my experience, often a useful note will include where/when in the game a thing happened, which specific interaction triggered the observation, and what happened. For example:

> ***level 5 begin; frame select interface; spent a lot of time clicking back and forth between options, said he didn't get why options mattered, didn't notice the tutorial text***

One last point: in your notes, try not to speculate the why of what you're seeing. Instead, stick to what you see or hear. You can prompt playtesters at the end for an explanation of what you saw (or you can ask in the moment, though that might ruin the flow of the test for the player).

After the Playtest

If you designed your high-level goals well, you should learn things during the playtest that can help you with development. If you're lucky, you might even accidentally uncover problems that you can fix.

> *It's important that you unpack the feedback you recorded as soon as possible after the playtest, otherwise you risk forgetting details.*

If possible, every attendant should record their notes in a document that's accessible for the entire team. Someone (usually the designer or project lead) should go through the feedback and arrange it in some logical sequence, either chronologically or thematically. It's important that you unpack the feedback you recorded as soon as possible after the playtest, otherwise you risk forgetting details. We've found that it's helpful to schedule a debrief meeting within 24 hours of the playtest so that the entire playtest team can come together, review their notes, and come up with ideas about ways to interpret and use the feedback.

FINAL TIPS:

- Emphasize with playtesters that you're assessing the game, not them. If they have trouble it's not because they're doing something wrong. In fact, if they have trouble, it's very helpful when they point it out!
- If playtesters ask for help, consider letting them struggle for a bit as you observe and ask questions. Watching what a player does to try to solve a problem will help you think about what the game needs to do better. Remember, you won't be there when users play the final product, so the game needs to stand on its own.
- When you introduce your game, don't provide more than basic information for context. You don't want players to go into the playtest with biases about the game.
- Encourage your playtesters to talk aloud as they play. While they might find it awkward at first, getting players to say their thoughts as they play will help you gain insight into a player's thought process during play.
- Sometimes, pairing players together on a single device and having them talk about the game as they play will provide insight akin to single player talk-alouds.
- During the playtest itself, it's usually better not to disrupt play by asking questions. Instead, take notes about what you're seeing, where players are getting stuck, what they're saying, that you can refer to later. Save your questions for the end.
- Following a playtest, when you ask playtesters questions about their experience with the game, it is often helpful to debrief as a group. Some of the most helpful or insightful information emerges when playtesters talk with each other about the game.
- During debriefs, you'll want to start with open-ended "easy" questions that everyone can have an immediate answer for; as the debrief continues, you can unpack the answers that people give for more detailed information. In our experience, some of the most valuable debrief questions to ask are:

  What was something you liked doing in the game?

  What was something that was hard in the game?

  Who got to the end of the game?

  What was something you'd like to change in the game?

  Was there anything you wanted to (or expected to be able to) do in the game, that you couldn't do?

  What advice would you give to somebody who wanted to be good at this game?

- You should plan on having at least two people at your playtests: one person to facilitate and run the show, another to focus on taking notes. Obviously you can have more than two people, but it's helpful to have one clear facilitator. It's essential that the playtest team get together to review their notes and process the playtest together.
- Depending on the age of your playtesters, you may need to secure school or parental permission. It's helpful to have a stock "letter home" that describes what your playtest will entail and what you'll do with the data you collect. Make sure to get these forms to playtesters well in advance of the testing date--it might take time for them to return to you.
- On a related note, collecting footage of the playtest can be helpful, but you absolutely must have permission from the playtesters or their parents/guardians.

TEST TITLE: BACKYARD ENGINEERS TUTORIAL PROTOTYPE 10/10

DATE(S): 10/10

TIME(S): (overall time off-site, individual class period times)

Overall off-site time: 8a - 12p

Class times (45 min each):

- 9:26a - 10:11a
- 10:14a - 10:59a

PLAYTEST PROCEDURE AND SCRIPT:

(testing objectives, sequence of events, what teachers and students will do, what observers will do)

OBJECTIVES:

- Using only the tutorial as designed and presented, are students able to understand the interface and use it to move or launch catapults?
- Using only the tutorial as designed and presented, are students able to understand how each catapult component affects their attributes and how that relates to solving each level?
- Do students find the new art style of 'enemy' characters to be appealing?
- What is missing from the tutorial that can scaffold players into fluency with the game?

SEQUENCE: PRE-TEST:

- FACILITATOR will provide Consent Form to TEACHER by 9/25 with a 10/8 due date.
- The Backyard Engineers test build URL will be made live (externally accessible) on Wednesday or as early as possible on Thursday. If possible, FACILITATOR will create a goo.gl with it.
- As soon as the link is live, FACILITATOR will send it to TEACHER so that she can confirm the build is accessible and playable on their network using a student login.
- On Thursday, FACILITATOR will create a presentation that introduces students to the game and that stands in as the tutorial. This stand-in is the prototype of the tutorial and FACILITATOR will 'pretend' to be the computer. He will not deviate from his presentation or answer questions.
- FACILITATOR will also create cheat sheets that will stand in for in-game help that students can refer to.
- FACILITATOR will create a google doc for all attendees to record their observations in.
- FACILITATOR will schedule a debrief meeting.

TEST:

- FACILITATOR will give a general introduction and set expectations for the playtest (2 min).
- FACILITATOR will give introductory and tutorial presentation. OBSERVER will pass out cheat sheets. Any questions will be recorded and referred to at the end, but no answers or additional help will be given unless absolutely necessary (5-7 min).
- Following presentation and question collection, FACILITATOR will write link on whiteboard and students will navigate to URL. As soon as they arrive, they begin playing.
- OBSERVER watches as students play, taking notes about what students are struggling with (25 min).
- At 10 minutes before bell, FACILITATOR will bring the group back together and ask debrief questions, OBSERVER and other staff take notes.

POST-TEST:

- Upon return, all attendees record their observations in a Google doc.
- During debrief meeting, OBSERVER will record significant findings as user stories.

LOCATION: SOME MIDDLE SCHOOL IN SOMEPLACE, WI

DIRECTIONS: (to school site, parking, office, classroom...)

Teacher indicated that we should ask visitor desk to be taken to the Field Room

Some Middle School
123 Street Street
Someplace, WI 12345

(123) 456-7890

TEACHER NAME(S): Mary P. Oppins, mpoppins@sms.k12.wi.us

PRINCIPAL NAME: W. Onka (has given permission and confirmed with superintendent)

STUDENTS: (number of students per class period, name of teacher per period)

Class 1: 30 seventh graders, five of whom have IEPs. One will likely need language support with cheat sheet.
Class 2: 27 eighth graders, three of whom have IEPs.

SCHOOL-PROVIDED EQUIPMENT:

TECH:

HP Chromebooks, two years old

NETWORKING:

school wireless

OTHER SUPPLIES:

n/a

DEVELOPER CHECKLIST:

EQUIPMENT: (computers, tablets, A/V)

n/a

NETWORKING: (hotspot? local server? offline?)

n/a

OTHER SUPPLIES:

Tutorial Presentation
Cheat Sheet
URL to game

CONSENT FORMS:

Yes

DATA COLLECTION FORMS: (assessments, surveys, observer data sheets, etc.)

just notebooks

TEST TITLE:

DATE(S): 10/10

TIME(S): (overall time off-site, individual class period times)

PLAYTEST PROCEDURE AND SCRIPT:

(testing objectives, sequence of events, what teachers and students will do, what observers will do)

OBJECTIVES:

SEQUENCE: PRE-TEST:

TEST:

POST-TEST:

LOCATION:

DIRECTIONS: (to school site, parking, office, classroom...)

TEACHER NAME(S):

PRINCIPAL NAME:

STUDENTS: (number of students per class period, name of teacher per period)

SCHOOL-PROVIDED EQUIPMENT:

TECH:

NETWORKING:

OTHER SUPPLIES:

DEVELOPER CHECKLIST:

EQUIPMENT: (computers, tablets, A/V)

NETWORKING: (hotspot? local server? offline?)

OTHER SUPPLIES:

CONSENT FORMS:

DATA COLLECTION FORMS: (assessments, surveys, observer data sheets, etc.)