So far, this document is just me (Magnus) thinking out loud. I will apply a more rigorous structure and format in the future. Also, this is where I will be taking notes for the eventual QA workshop + associated texts.

* How to test games in a general sense? Typical methods, models, criteria for evaluation.
* What are we testing for? Usability? User experience? What criteria is used to define a successful test?
* When and how often?
  + I need a better understanding of the technical workflow and version control to comfortably navigate different versions of the game.
* The significance of sound: including audio in a game presents certain risks and challenges. Moreover, making audio a primary mechanic produces more challenges yet: will we need special gear to better test the game (eg, headphones)? Should we test in different settings (noisy, quiet) as part of QA?
  + Similarly, should the musical intelligence of players be considered? If we have a puzzle that asks players to repeat a chord progression we risk alienating those unable to for whatever reason. Is that a problem? It is a consideration.
* A perspective: we are testing to Assure the Quality of what we are making. We are not necessarily making a game for everyone or a game to be played under every possible condition. As long as “most” players can have the experience intended by us, we are doing the right thing.
* The big C: corona will almost certainly limit our ability to test things with people physically. A problem or an opportunity? We will need to make sure that working versions of the game can be produced and distributed reliably. In other words, Lullaby will probably be running on many different systems way ahead of release - and Martin/Hanna getting their hands on the program.
* Feedback: how, when, what, where, and why? A Google Forms? Interviews? Going by the above, what kind of feedback are we looking for? Qualitative interviews or systems diagnostics?
* My role in all of this: I should probably create some kind of workflow (or ingrain QA more visibly in the development process).

Workshop:

Relations to QA:

* Lead Designers needs to explain intended experience. Main person to relay findings to.
* Work with Producer to schedule testing.
* Tech Lead decides who is in charge of the bug list.

Bug testing:

* Work with Tech Lead to figure out a testing schedule:
  + Option 1: Set day to test newest build.
  + Option 2: Tech asks for specific tests.
* Findings are compiled in a bug list.
  + Good descriptions!
  + Accessible to Producer.

The bug list:

* Should contain:
  + Status (closed/investigating/open)
  + Severity (must fix/should fix/shippable/won’t fix)
  + Reporter - who found the bug
  + Assignee - who will squash it
  + Description of bugs
  + Other comments
* Alex: bug tracking software is overkill for this project.

The UX testing process:

1. Identify what needs testing.
2. Prepare testing plan.
3. Recruit participants.
4. Conduct testing.
5. Analyze findings.
6. Summarize findings in report/action plan.
7. Present report/action plan to relevant parties.

The testing plan:

Describe in detail to identify foreseeable problems.

* Purpose! Why are we even testing?
* Roles (who will act as loggers, moderators, guides, etc)
* Prelude: make sure the build works and everything is set up.
* Method: how do we record the test? What are they doing and what are we asking?

Approaches to playtest:

* Standard playtest:
  + Participants play a distinct segment of the game (a demo).
  + Participants are then interviewed/given questionnaires.
  + Useful when testing the collective experience of the game.
* UX style playtest:
  + Participants given tasks within a game (“jump across this pit”).
  + Questions given between each task/at the end.
  + Useful when testing a few specific things.
* Asking broad questions (“what did you think?”) is good for testing the game holistically.
* Make sure you get consent to record interviews - transcription takes time!
* Used quotes in the report are taken on good faith.
* UX tasks: specific tasks. Can you equip a sword? Can you go to a different level?

Three kinds of interviews:

* Unstructured:
  + Essentially a conversation.
* Semi-structured:
  + Suggested default: open-ended questions with follow-ups.
* Structured:
  + Close-ended questions. When doing these, you might as well do a questionnaire.

Recruiting participants:

* Grab as many different people as you can.
* Don’t just go for gamers.
* Provide an incentive where possible. Be creative!
  + Offer to put people in the credits as “play testers”.
* Don’t use Discord to interview people. Zoom is better and faster.
* Rule of 5 suggests that after five testers you will have the majority of your data.
* Thematic analysis: look for common problems, patterns of behavior.
  + Don’t make testers self-conscious.

Compile findings:

* Look for themes in notes and recordings.
  + “All players had issues with this.”
  + “This one player did something very unusual.”
* Turn findings into a written document. Keep it short.
* Participant quotes gives your findings credibility - taken on good faith.
* Suggest changes, depending on team structure.

Test compatibility of controllers/people

BUG LIST

NAMES OF PARTICIPANTS/FORUMS OF PARTICIPANTS/TYPES OF PARTICIPANTS

CRITICAL STEPS FOR TESTING

PLAN FOR TESTING - WHAT TESTS AND WHEN?