Godot side:

* Create AudioManager to play sounds. Can be child of scripts in game\_tree and have singleton instance access pattern (class\_name AudioManager + static var instance: AudioManager & ready(): instance = self) making use of it like this: AudioManager.instance.play\_sound\_foo().
* Add audio assets.
* Add graphical assets. They can be child of CanvasLayer in game\_tree. Accompanying script is probably good to make as well to control visibilities and tween single assets.
* Add all texts and choices to dialogue script from story
* Style the dialogue with RichTextLabel BBCode styling. (see <https://raw.githubusercontent.com/godotengine/godot-demo-projects/4.2-31d1c0c/gui/rich_text_bbcode/screenshots/rich_text_bbcode.webp>)
* Add finished typewriting sounds to DialogueUiManager (look for comment # todo audio manager should/could play a randomized typewriter sound). Playing text sounds follow audio pattern for example “0101” where every other character plays a sound (var play\_sound\_pattern exists and is “1010” by default).
* Limit typewriting sounds to certain situations: maybe just the email, or change sound afterwards to different one
* Add support for continuing text on screen instead of clearing the text and starting writing new one. Not top priority.
* When near playable build tomorrow change text skip style to speed up instead of skip instantly (now at skip to make testing fast).
* Add (inbe)tweens to values in game to make it juicy.
* Make game builds (maybe feedback build + final build)
* Add some font that is used.
* Make sure text is not scrollable and no scroll bars appear.
* Add option to go back to main menu and restart game. Low priority.
* More?

Other category

* More?

Game Design

* Decide when to use quantum scramble text effect (see also if it works with bbcode tags)  
  Maybe in the memories that are unlocked through the quantum echoes?
* Decide whether to use fade to blacks or not and where
* Design how the echo based visual effects and basic visual effects work and put it to GDD, to be implemented by programmers.
* More?

Testing

* Some branch finishable to end? (Showcase branch)
* All branches finishable
* Make sure there’s no major bugs when playing, report bugs