

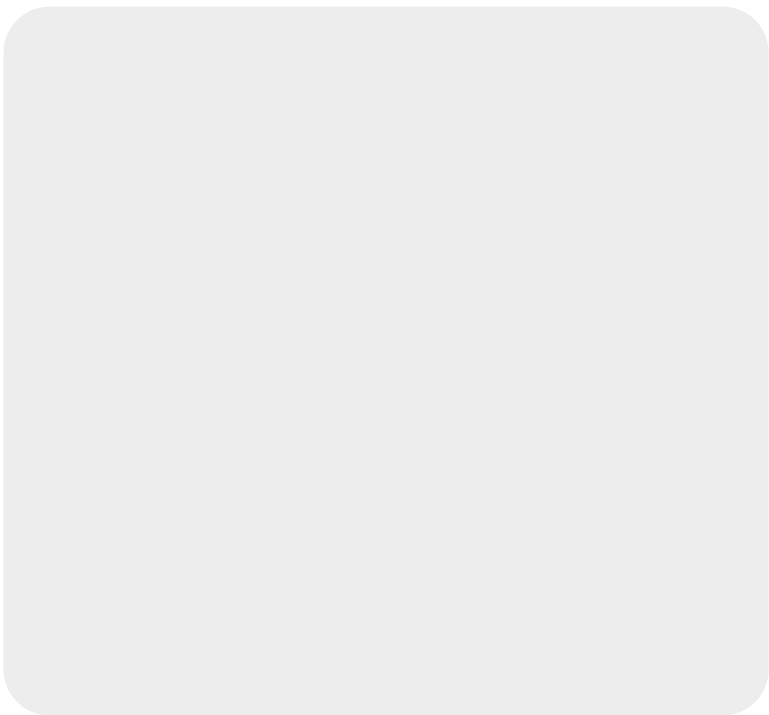
CART253 Project 2: Proposal

TITLE: *Madeleine*

(could be subjected to change)

Angel Cella Cenerini
Student ID: 40180087

01 - Describing Project



Concept

The project is inspired by the concept of letting the user experiment within an immersive environment largely morphed by lights and sounds; specifically, certain lights corresponding to specific sounds.

The entire project revolves around the user collecting Script Shreds and Tracks/Instruments/Pitch(to be decided) Sounds in order to compose a melody. The latter will be necessary to unlock the reading key to the entire narrative.

Structure

The design is structured in 7 levels (corresponding to the 7 musical notes); however, the number may decrease to 5 if the work-load proves to be excessive within the given amount of time.

The ideal outcome would be to develop levels intellectually engaging; puzzle-like challenges for the user to overcome. Each level, if completed successfully, will unlock one of the collectable items.

Achieving (or not) such items will determine which of the 3 (might become 2) possible endings the program will resolve to.

*More on the different endings in **02 - Discussing Programming Techniques**.

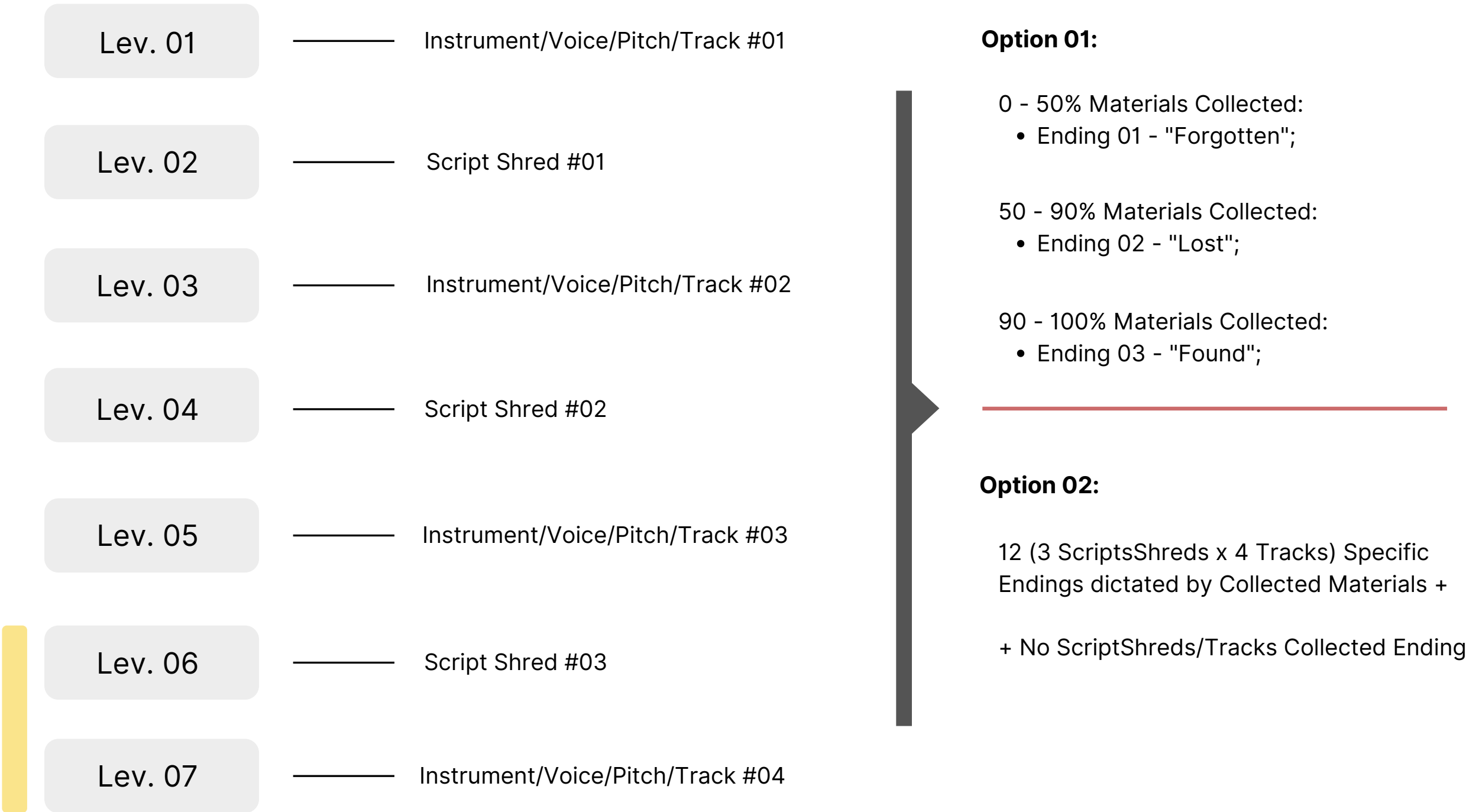
The graphics will aim to be abstract and immediate in order to better create a suspended atmosphere.

The user might be illustrated as a distinct entity, or merely addressed by the program, without ever acquiring a specific shape. This aspect is yet to be decided.

The title is inspired by Marcel Proust's "petites madeleines" (*À la recherche du temps perdu*), for the melody, if correctly attained, will bring to the surface a long-forgotten memory (this would correspond to the "Found" Ending).

Project Structure

Highlighted sections might be modified or removed completely



02 - Discussing Programming Techniques

General Plan

The strategy for this specific program is to build different levels for two main reasons:

1. to break a substantial program into organized sections. The work-load should thus result better distributed; it also permits greater creative freedom.
2. to allow the coding of each level to experiment with different aspects/topics addressed during the course. The entire project should feel connected and consistent, yet creative and different through each level.

Addressing Course Topics

Admittedly, the single levels have not been planned in detail yet, although there are certain programming "techniques" which will most definitely feature within the program:

- Conditionals;
- User Input (both via mouse and keyboard);
- Timer(s);
- Arrays;
- OOP;
- Inheritance;
- p5.sound;

As one would expect, mostly every topic covered during the course will be exploited within the development of the project.

*Including Variables and Functions in the list seemed pointless, for of course these will be invaluable when creating the program.

Level Structure

The current plan is to build a structure in which the user may or may not surpass a specific level, yet would still be able to move onto the next one. Only if successful, they obtain one of the collectable items; the user needs to achieve those in order to play the final melody. To make this possible, an idea would be to set timers on all levels; this way it would be guaranteed that the user does in fact move forward in the program, and each challenge becomes significant when determining the ending.

However, this outline may change. The program might envision merely one ending; in this given scenario, the user has all of the time available to eventually surpass each level and obtain the total of items. This outcome would probably result less interesting and engaging to the user, however, it is a valid backup plan.

Option 01:

- 0 - 50% Materials Collected:
 - Ending 01 - "Forgotten";
- 50 - 90% Materials Collected:
 - Ending 02 - "Lost";
- 90 - 100% Materials Collected:
 - Ending 03 - "Found";

Option 02:

- 12 (3 ScriptsShreds x 4 Tracks)
Specific Endings dictated by
Collected Materials +
 - + No ScriptShreds/Tracks Collected
Ending
- "Forgotten", "Lost", or "Found" Ending

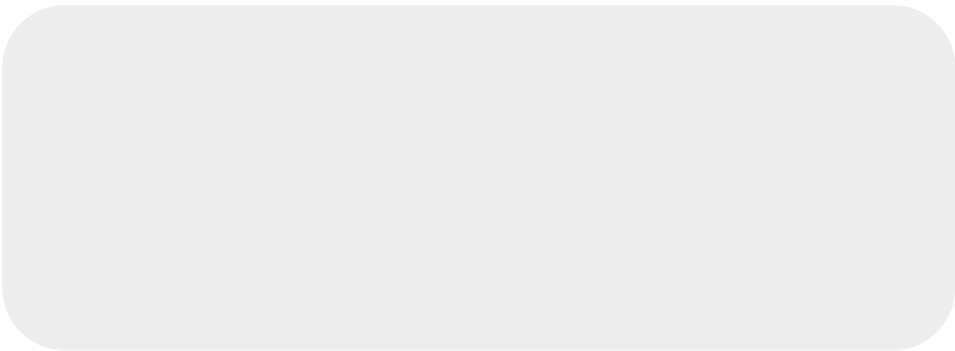
Endings Structure

When directing the program to choose a certain ending, there are two possible strategies available:

Option 01 - The program will keep count of the items collected, without bothering to pay attention to the single identity of said items.
eg: whether we obtain a Script Shred or a Track/Instrument, the program will consider them as the same entity and will add one value to the list.

Option 02 - The program does not only keep count of the items, it is also concerned with their individual identity. Consequently, the nature of the distinct object will matter in defining the ending.
eg: If the user obtains a Script Shred, they could potentially play that section of the melody, yet would require at least one Track/Voice/Instrument to produce any sounds.
This outcome would imply that the program contains all of the possible conclusions creatable while engaging with the project. A most interesting and credible structure, although much more time consuming.

03- Assessing Details



Consulting Examples

Naturally, each program constructed as an Assignment, explained during the Activities Videos, or TMI examples will result extremely valuable when setting up this project.

Materials

Graphics

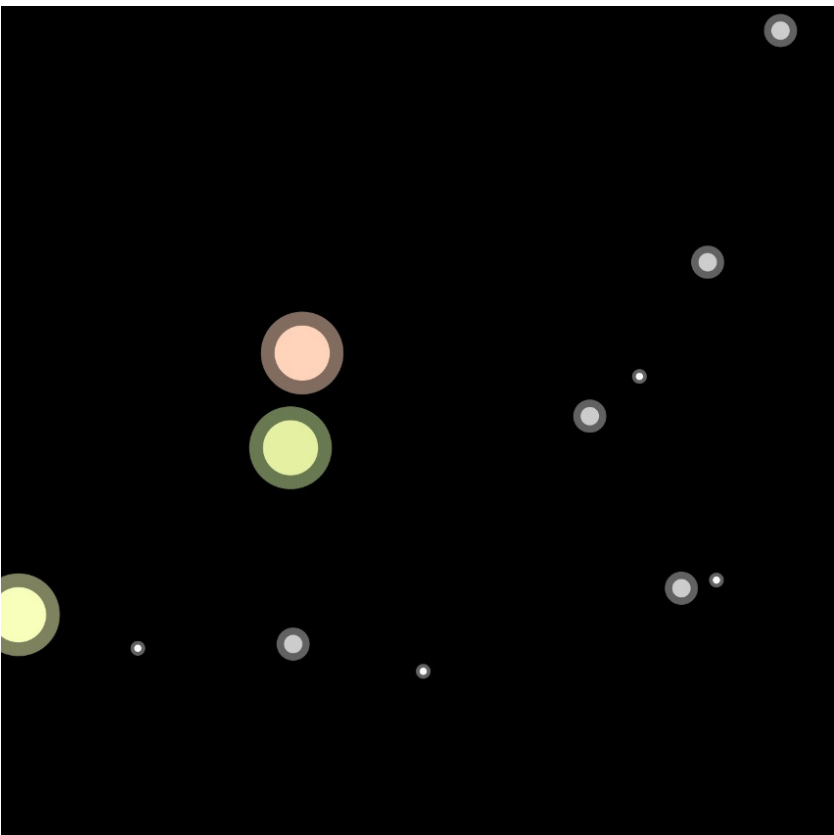
All graphics will be original works (or plain geometry, courtesy of p5). As mentioned, the style will aim to be quite minimalistic.

Graphic inspiration template



Credit:
https://www.reddit.com/r/wallpaper/comments/hyf1k5/oc_floating_spheres_of_light_4k_uhd_3840_x_2160/

Atmosphere/Style
illustrated in Prototype
(may change)



Sounds

Considering that the entire project is inspired by the interaction of sound and light, the aspiration is to compose an original soundtrack/melody specifically for this project.

FL Studio would be the most palpable candidate for such task.

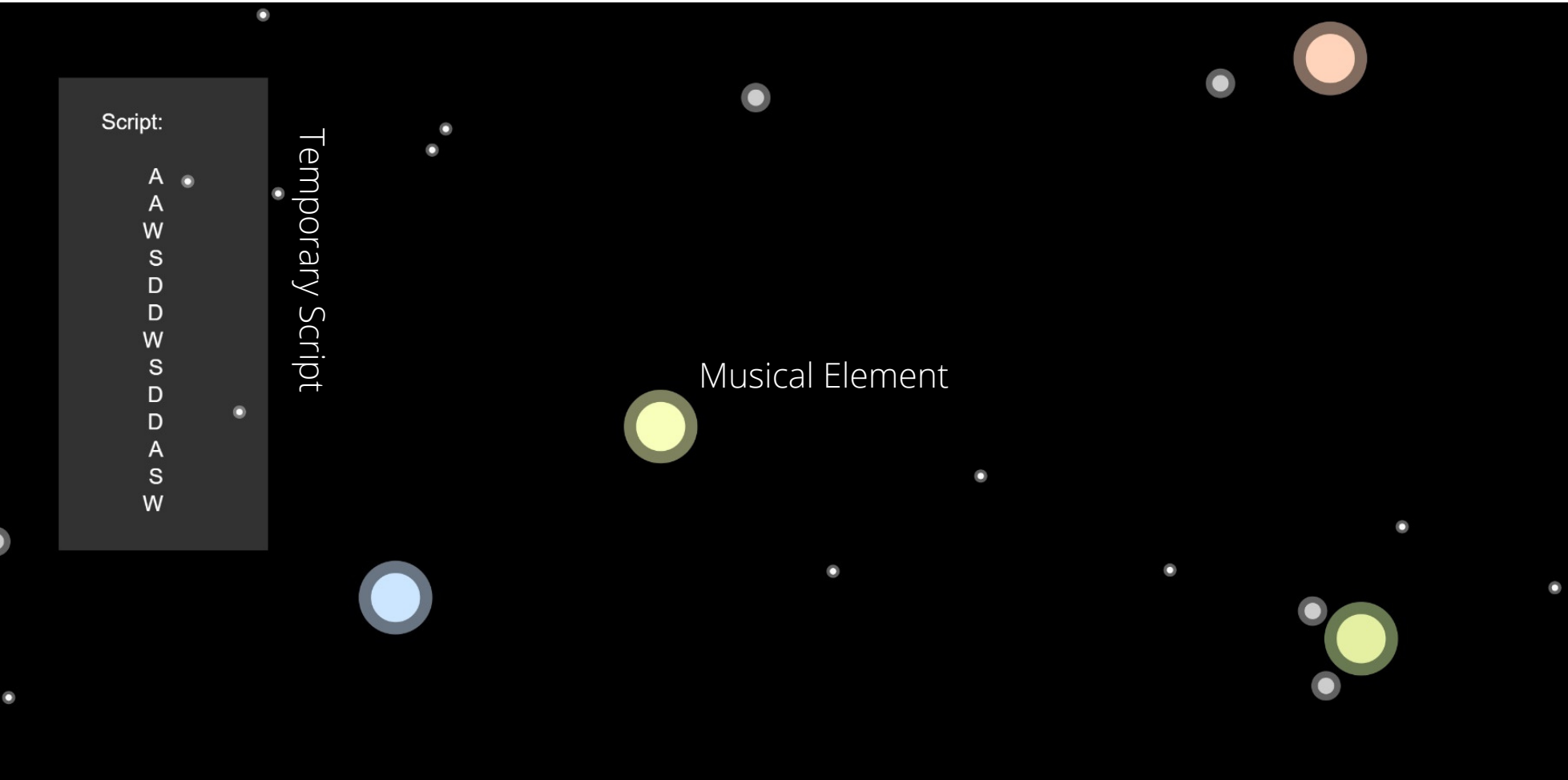
Needless to say, the musical aspect of the project is secondary to the actual programming; therefore, in the eventuality that the design turns out to be slightly too ambitious for the assigned time period, an existing soundtrack will be used (and appropriately credited, of course)instead.



Disclaimer

The final project may differ or evolve from the current description envisioned in this proposal. As briefly mentioned previously, two outlines have been set up for the future program: a more sophisticated and satisfying template, and a simpler, less intricate one. The aim is of course to successfully carry out the former; the latter has been established merely as a backup plan.

04- Discussing Prototype



Brief Description

The prototype provided explores the bare principles and mechanisms of a certain section featuring in the final project: playing the infamous melody. It will most likely run nearing the end of the project, and it will partly determine its outcome.

Of course, the template provides a rough sketch of a more polished and elaborated design for the final project.

Structure

The user will be directed by the Script on which key (currently *A*, *W*, *S*, and *D*) to press in order to play the melody correctly. Should the user fail to adhere to the right order, they will be brought back to the Title Screen; in the actual project, however, their success or failure will define the ending.

Each light illustrated belongs to a subclass. They gently float around the screen; their general behaviour is dictated by the Light superclass. The smaller, white lights function as mere background elements for a more immersive environment. The remaining 4 ones, however, are interactive. When pressing one of the aforementioned keys, the item will temporarily gain size (replicating a glowing effect of sorts), and return to their original state after 1 sec.

Script

The current Script is a mere rectangle containing the ordered list of keys to press. However, it would be ideal to experiment with its aesthetics and performance:

- **eg:** making the instruction appear as an icon (similarly to a button) and disappear to leave room for the following direction as soon as the user presses the desired key; or, also, creating a flowing text of sorts. These details are yet to be decided.

On the other hand though, the more technical functioning of the Script will roughly follow the same trend as shown in the prototype:

- through two arrays, one intended for the Correct Key Sequence (aka the Melody) and one for the keys pressed by the User, the program is able to store specific data and subsequently compare both in order to check the user's progress.

Sounds

The .mp3 files used for this prototype are not the intended ones for the final project, just a temporary and quick solution.

Source: freesound.org
Link: <https://freesound.org/people/Tristan/sounds/19459/>