Beat King

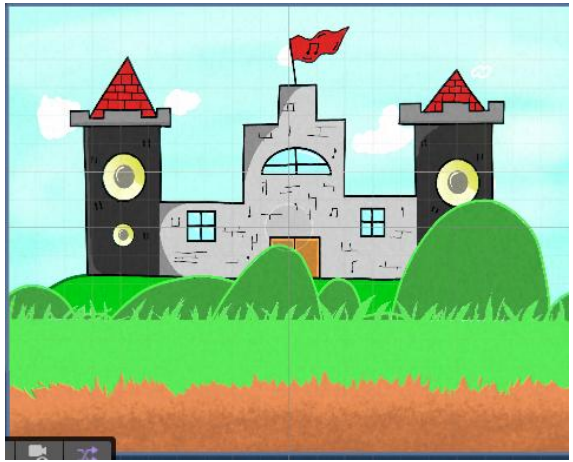
A wicked cool project by Kaelan Bartlett and Michael Ventricelli

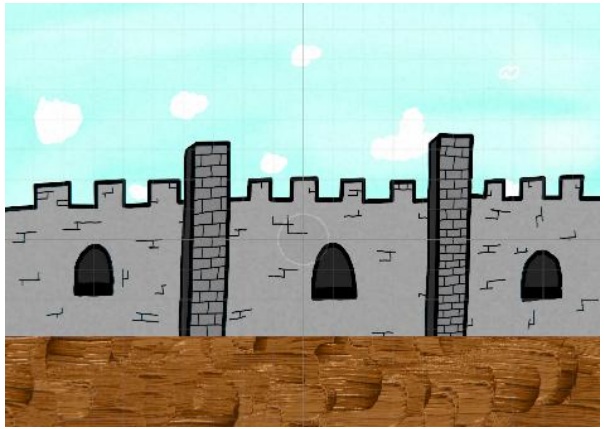
Game Overview

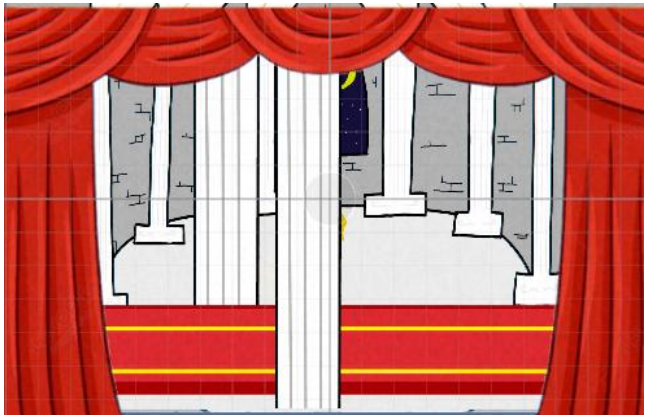
Beat King is a one key rhythm game set in a musical medieval world. You are a lone knight who sets out on a quest to beat their enemies after witnessing the death of your father. To progress through each level, you must click an attack key whenever the enemy aligns with your crosshair, which aligns with the melody of each level’s song.

Development Highlights

Three unique environments, each featuring custom assets and music.



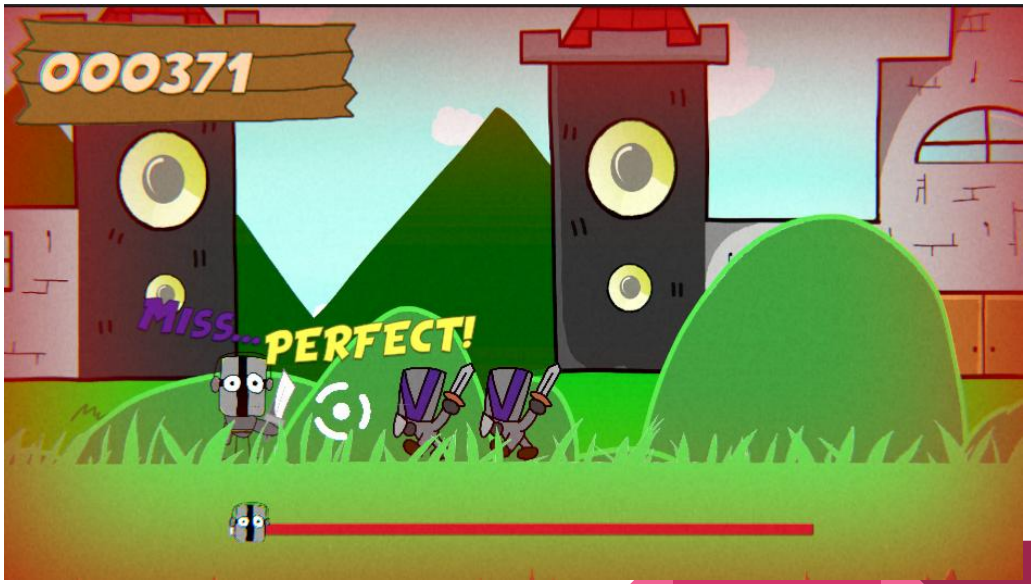




Development Highlights Cont.

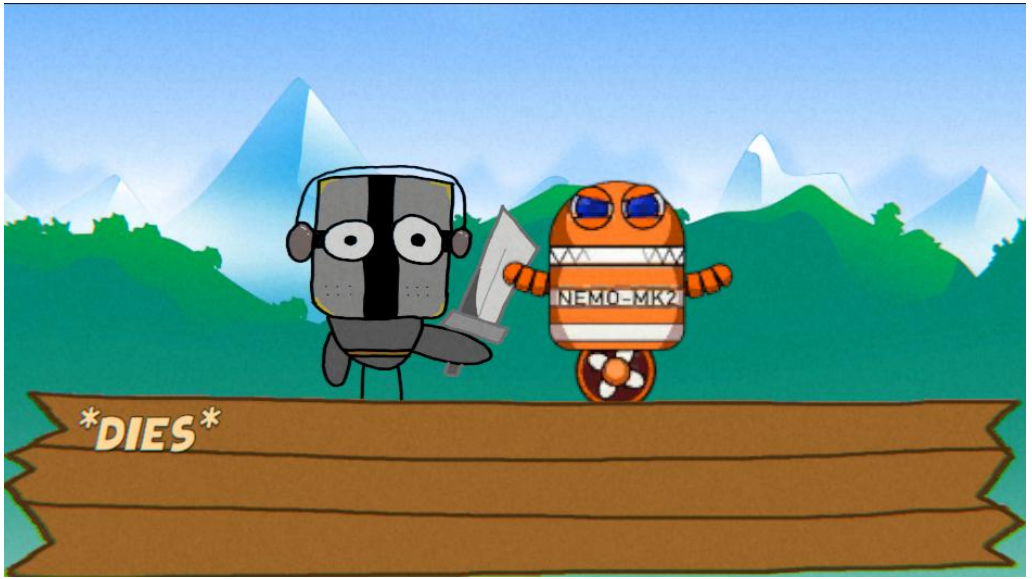
Detailed and animated UI, with lots of visual feedback.

- Score and health trackers
- Hit indicators
- Low health vignette
- Beat synced visuals



Development Highlights Cont.

Modular cutscene and song systems utilizing Scriptable Objects.



Progress & Verification - Initial Proposal

- Three fully unique levels
- Cutscenes to explain the story
- Rhythm based gameplay
- Visual Synchronization

Progress & Verification - Reality

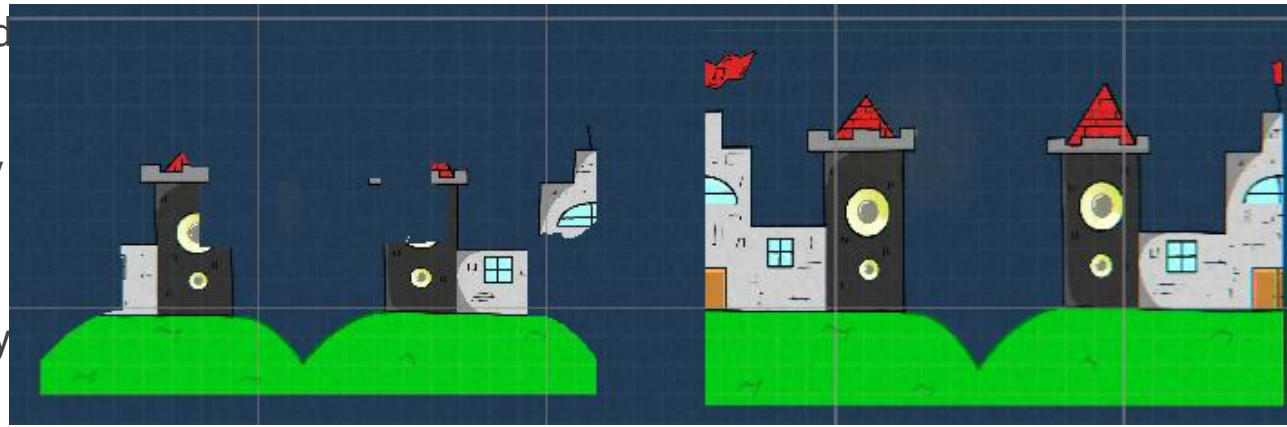
Visual Synchronization was cut due to a lack of necessity. Input delay was not noticeable enough to warrant the time investment of creating a dedicated setting to correct it.

Cutscenes were also not given as much refinement as desired. This occurred simply due to a lack of time, and most of the focus being dedicated towards the gameplay.

Challenges and Solutions - Challenges

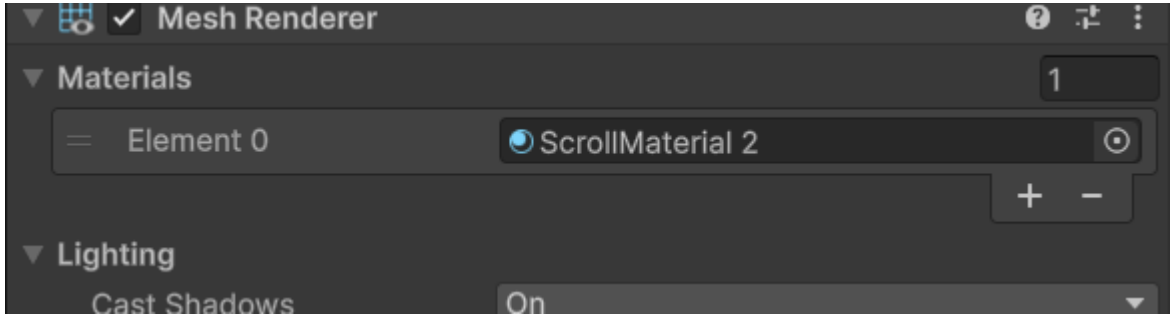
The biggest challenge we faced was implementing parallax layers. Sprites with transparency render incorrectly when given an offset via Materials.

Another challenge was severely underestimating how much time creating assets takes.



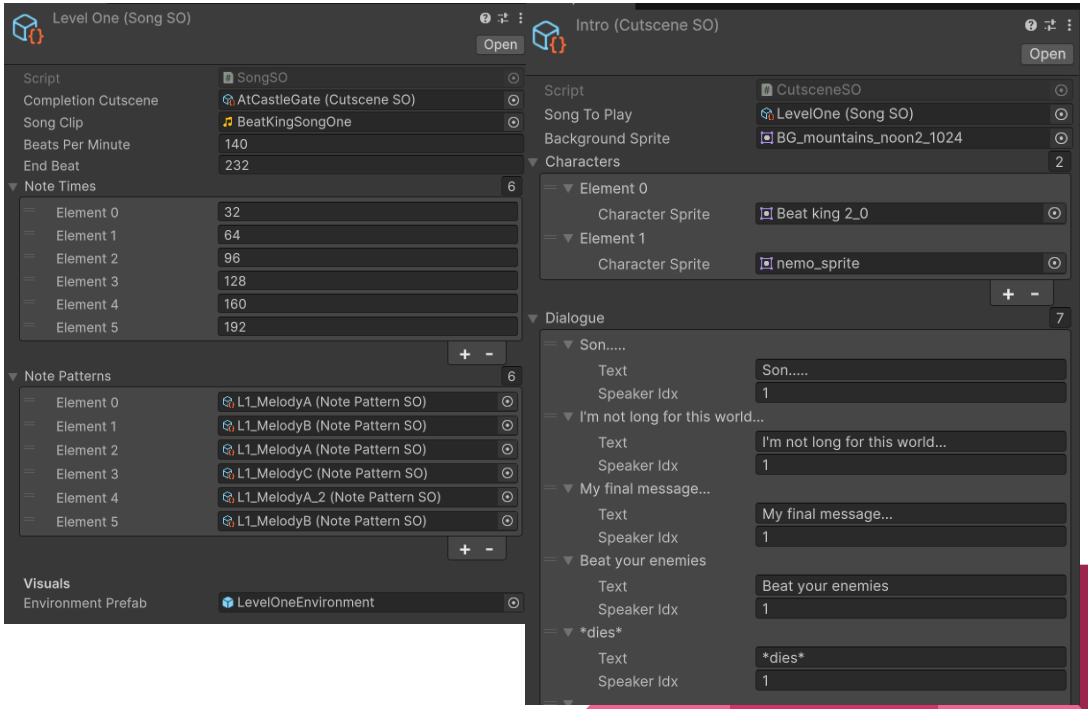
Challenges and Solutions - Solutions

The solution was to replace the sprite renderer with a mesh renderer, using a quad as the mesh. This also results in drawing now being sorted by Z, technically meaning Beat King is 3D.



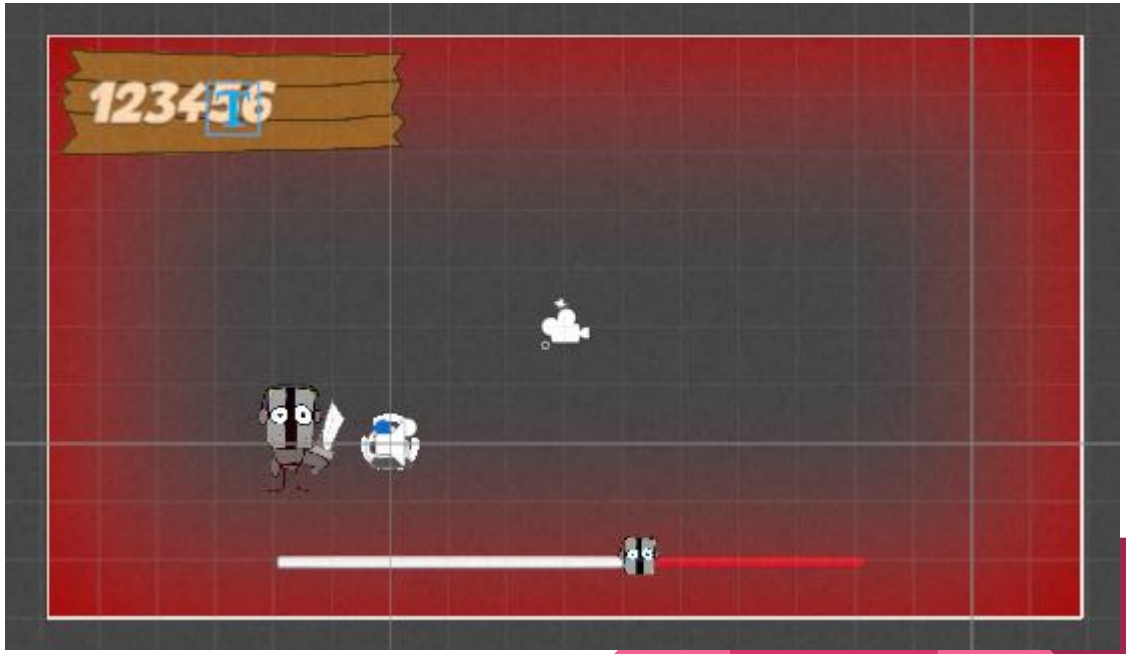
Visual Documentation

Songs and cutscenes utilize Scriptable Objects to make creating new songs and cutscenes faster.



Visual Documentation - Gameplay

The gameplay scene only contains content relevant to all levels. The environment is spawned up upon loading the scene depending on the contents of SongSO.



Technical Deliverables

Game Link: https://helodity-m.github.io/Fog/
GitHub Link: https://github.com/Helodity-M/Fog

Portfolios:

- Michael:
- Kaelan: https://helodity.xyz/portfolio