Project summary - Blobby adventure

This platformer expanded beyond a basic p5.js exercise into a miniature systems project. The colourful and joyful design was made in collaboration with my youngest nephew.

Key extensions implemented:

Added all 3 extensions cited in the assignment

1. Added sounds effects and looping music
2. Used the factory pattern to create platforms
3. Created enemies in the form of worms that when crushed makes the character loose a life

Most complex / difficult parts:

There are a lot of different parts and algorithms in this codebase that i found difficult to implement.

1. By far the most difficult feature to implement was the algorithm to procedurally generate all the world entities using procedural world resizing with density scaling.
2. The multi-layer parallax ecosystem (hills, clouds, three tree depths, mountains) tied to a noise-smoothed wind effect that gently moves trees and the other flora.
3. Platform generation with reachability & spacing constraints
4. Worm critter hazard with life penalty and splash effect
5. HUD elements

also ...

- Rejection-sampling platform & canyon placement without overlaps while preserving jump reach.
- Balancing decorative density (trees / grass / flowers) to avoid visual noise yet keep the world alive.
- Precise yet forgiving platform collision (vertical snap tolerance, drop-through).
- Maintaining deterministic feel while injecting subtle variation (per-worm parameters, coin pulse, particle fades) (took a long time to perfect it).

Skills learned / practiced:

As a senior developer the key concepts were already part of my skills, so i centered my effort to create more complex algorithms using math drom the computational math school module

- Procedural content generation heuristics (spacing, exclusion zones, probabilistic soft rejections) and performance guarding with bounded attempt loops.
- Lightweight physics integration (velocity + gravity) and stateful animation posing.
- Parallax layering & environmental motion for depth.