Team 8 – Project Pitch

Game Title: UBZ: University of Bozos and Zombies

Story:
As the sole member of the UBC Zombie Survival Club (AMS Approval Pending), Bozo is a socially awkward horror super-fan who spent his 3 years in university obsessively honing his zombie survival skills and acing his immunology classes. Despite the ridicule of his peers, he was convinced that the apocalypse would soon occur, and would stockpile supplies in the clubroom (an old unsupervised janitor's closet in the biological sciences building). Against all odds, his suspicions were proven correct when a deadly and infectious virus descended upon the campus, turning the vast majority of students to zombies. The authorities have forcibly quarantined the campus, abandoning the survivors that remain within the school, but not if Bozo has anything to say about it. He sets out to explore the campus and seek out the source of the virus, uncover a secret conspiracy, and develop a cure using his immunology knowledge.

Major levels are designed after various familiar buildings on campus: Buchanan, AMS Nest, ICICS, Engineering, etc.

Core Game Design Elements:

1. Rendering: OpenGL renders different locations across the UBC campus, capturing the fast-paced side-scrolling platformer style. The environment showcases the hustle of an escaping student, approaching zombies, and a series of moving and static obstacles.
2. Assets:
    a. Geometry: Iconic UBC buildings, lecture halls, and outdoor areas provide the setting for platforming challenges.
    b. Sprites:
        i. Main character: Jeff Bozo
        ii. Student NPCs
        iii. Various types of zombies (slow, fast, projectile, exploding, jumping cat zombie)
        iv. Throwable objects (textbooks, coffee cups, students, etc.)
        v. Moving platforms (escalators, maintenance carts)
        vi. Obstacles (overturned tables, wet floor signs, piled-up chairs)
        vii. Walls (classroom doors, brick walls, lockers)
3. Audio:
    a. A high-tempo soundtrack
    b. Zombie moans
    c. Shouting students
    d. Environmental noises (alarm bells, slamming doors)
4. 2D geometry manipulation:
    a. Platforms vary in height and motion (some static, others moving up/down or side-to-side).

    b. Obstacles like overturned tables or wet floor signs require precise jumps or swift maneuvers. Walls can temporarily block progress, requiring players to find alternate paths or wait for the right moment. Ladders/elevators to climb up different floors
    c. Throwables have differing arcs and impact zones.
5. Gameplay logic/AI:
    a. Zombies maintain a relentless pursuit. Their pace adjusts based on the player's distance, ensuring constant tension.
    b. NPCs, when thrown, can slow or briefly halt zombies. Some NPCs (like the campus jock) have a bigger impact than others.
    c. Zombies may occasionally show varied behaviors, such as jumping or dodging thrown objects.
6. Physics:
    a. Momentum is vital. Rapid changes in direction or jumping can momentarily reduce Bozo's speed.
    b. Knockback effects from thrown NPCs depend on their size and the velocity of the throw.
    c. Gravity consistently affects jumps, falls, and throw trajectories. Moving platforms and obstacles introduce additional challenges.
7. Randomness/surprise component
    a. Airborne virus causes random students to turn into zombies
    b. Throwable objects (e.g., textbooks, coffee cups) randomly generated in the environment
    c. Hidden traps that will surprise the player

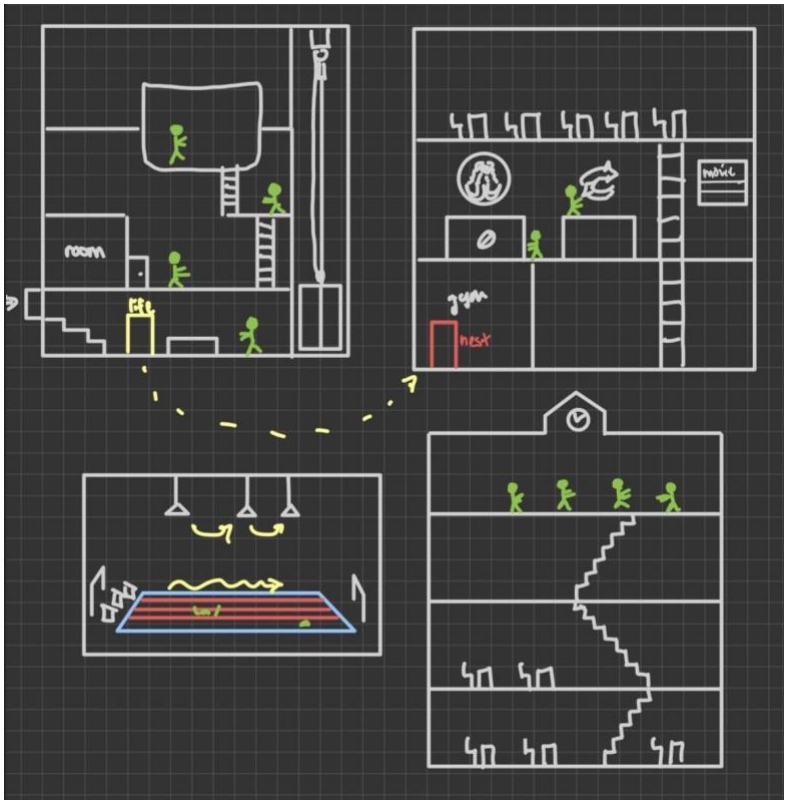
Player Goals:

1. Sprint forward while skillfully navigating the different and ever-evolving platforms and obstacles.
2. Save and recruit NPCs to help you in your quest - or throw them as a sacrifice to the zombies in order to buy you time to escape.
3. Master the moving platforms and timed challenges to maintain momentum and evade capture.
4. Arrive at the classroom sanctuary, marking the end of each level.
5. Earn points based on level completion time, distracted zombies, and effective use of obstacles.

Concept Art (on next page):

1. Start screen:

2. Sample level images:



3. Sprites: