

# CSS 450 - Final Project

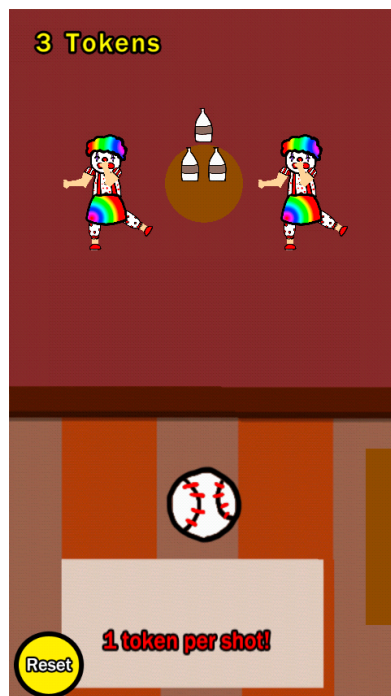
Group: Dmitry Ryzhkov, Josh Belcourt, Kim Walker

## CARNIVAL!

### OVERVIEW

CARNIVAL! is a first person “shooter” where the Player tries to win at various “mini-games” based on common carnival games. The Player begins a new game with a certain amount of tokens and each throw of the ball uses 1 token. The main goal of CARNIVAL! is to succeed at each game.

Game 1 is a game where three **Milk Bottles** are setup and the player attempts to knock them all down on a single throw. The Player gets three chances to knock over 3 milk bottles before it's game over. The closer to the “sweet spot” on the milk bottles increases the chance the bottles all fall down.

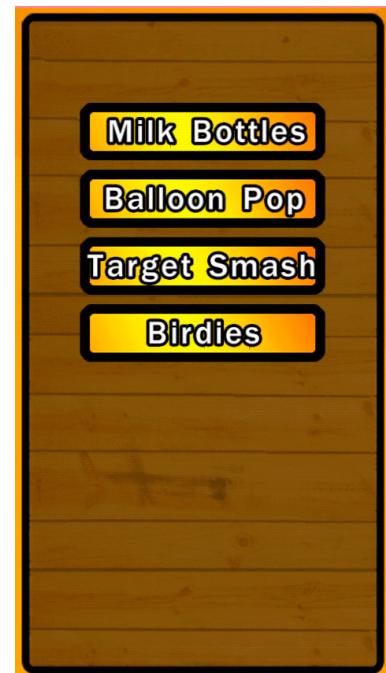


In Game 2, **Balloon Pop**, there are 10 targets which are rotated in a constantly moving circle. The player gets 10 tokens/chances to hit the 10 moving objects. A “hit” is only true when the ball is right over the balloon before it disappears. A balloon will not pop if the ball just passes over it.

Game 3, **Target Smash**, is a very basic game where the player attempts to hit both moving targets that are being moved around the screen. The player gets three chances to hit the two moving targets, and wins if the targets are hit before running out of tokens.

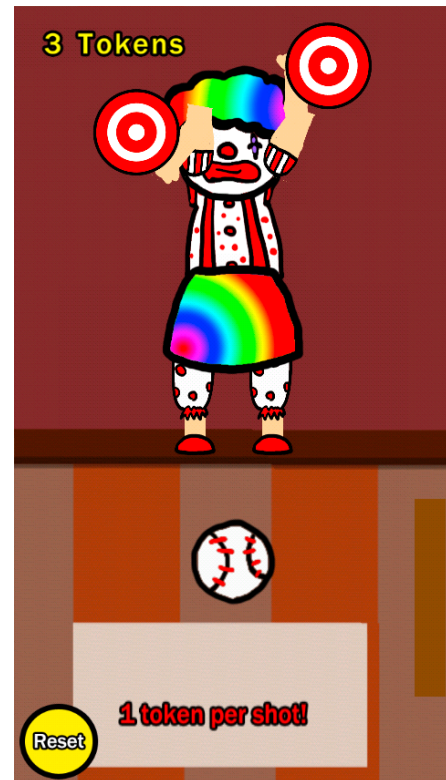
**Birdies**, Game 4 is a “duck hunt” style game where the Player attempts to hit moving targets on the screen using the baseball. The moving objects move left to right (or vice-versa) with up and down movement. The objects randomly appear on the the screen from the left or right

side. The Player has 20 chances to hit 10 objects. besides hitting all 10 targets, a player goal can be to hit all targets in 10 shots.



## FUNCTIONAL SPEC REQUIREMENTS

- Hierarchy/Transformation
  - 1) The player throws the ball by pulling back from the ball and releasing
  - 2) The distance from the ball changes the power of the throw
- Primitive Attributes
  - 1) Balls, objects textured and alpha blended
  - 2) Circles, rectangles subclassed for birdies and bottles
- Complexity: number of transforms
  - 1) Milk bottles
    - React to collision with ball
    - React to collision with other milk bottles
  - 2) Moving objects
    - React to collision with ball
  - 3) Ball
    - React to collision with bottle/object
    - Moves toward bottle/objects
    - Moves before being “thrown”
  - 4) Clowns
    - multi-joint movement
    - arms, hands, legs
- Pre-defined animation
  - 1) Pressing LMB causes ball to “fly” towards bottle/object
    - direction defined before throwing
    - power/speed defined before releasing
  - 2) All mini-games include program-controlled animated objects
- World-coordinate window
  - 1) Main screen for game
- Multiple-views
  - 1) main window - game window displays throwing object/hand
  - 2) second window - zooming via “pinch-to-zoom” to get closeup of clown and other objects in world view
- Semantics
  - 1) User attempting use as few throws as possible
  - 2) User-selectable difficulties based on the game incorporate different variables (external variables, speeds, movement randomization)
- Audio
  - 1) Background music
  - 2) Collision sound between ball/ and moving objects
  - 3) Collision sound between milk bottles
- Suspend/Resume/Reset
  - 1) Pause game
    - Phone lock button pauses action



- 2) Reset game
  - Back button returns user to main screen or phone home
  - Reset button resets game back to starting # of tokens
- UI Friendliness
  - 1) Ball movement controlled by finger
  - 2) Throw initiated by pulling back finger and then releasing
  - 3) Speed of throw controlled by distance from ball of finger

## **OBJECT INTERACTIONS**

- Ball - Milk bottle
  - Ball hitting bottle(s) causes bottle(s) to move away from ball
  - collision interaction based on point of collision
- Milk bottle - Milk bottle
  - Milk bottle collision causes milk bottles to change position and current velocity
  - direction/speed based on angle of collision and speed of attacking bottle
- Ball - moving object
  - Ball hitting moving objects causes moving object to spin and then disappear
  - Ball moves reacts to object collision and changes direction based on angle of attack
- Ball - Balloon
  - Ball hitting (when descended) removes balloon
- Ball - Target
  - Ball hitting target removes target

## **VIEW/CONTROLLER PAIRS**

- View: Main Window
  - Controllers
    - Mouse Events
      - Finger
        - releases ball/bag
      - Finger Drag
        - changes power of throw
    - Timer
      - Update simulation (model)
    - Controls
      - Automated object movement (clowns, heads, balloons)
- View: pinch-to-zoom
  - Controllers
    - Finger pinch-to-zoom
      - zooms into objects onto screen to get better look