

# CSS 450

## Final Project Proposal

Group: Dmitry Ryzhkov, Josh Belcourt, Kim Walker

### <GAMENAME>

#### OVERVIEW

<GAMENAME> 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 game requires a different amount of tokens to play. The main goal of <GAMENAME> is to accumulate as many tokens as possible. While each game costs tokens, tokens are rewarded when the player satisfies specific “mini-game” goals.

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 token cost and rewards are outlined in **Table 2**. The Player can view the scope window to get a better approximation where the ball may go

Table 1: Game 1 token overview

Initial Cost	Lose 3 tokens
1 Attempt	Win 6 tokens
2 Attempts	Win 3 tokens
3 Attempts	Win 1 token

after throwing the ball, however this view moves forcing the Player to be patient on when to throw the ball. The closer to the “sweet spot” on the milk bottles increases the chance the bottles all fall down. The Player has a limited amount of time to throw the ball at the milk bottles. The Player will forfeit the attempt if the timer expires.

Game 2 is a “duck hunt” style game where the Player attempts to hit moving targets on the screen using bean bags. The objects move across the screen and the player must use the scope to aim the bean bag. The moving objects move left to right (or vice-versa) with up and down movement. The object may randomly switch which direction it

**Table 2: Game 2 token overview**

Initial Cost	Lose 4 tokens
10 hits	Win 20 tokens
7 hits	Win 10 tokens
5 hits	Win 4 tokens
3 hits	Win 1 token

is traveling (left/right). Token costs and rewards are outlined in **Table 2**. The Player has 10 chances to hit 10 objects. The Player is rewarded a specific number of tokens based on the total number of objects hit.

## **FUNCTIONAL SPEC REQUIREMENTS**

- Hierarchy/Transformation
  - 1) The Player moves around the ball/bag with the mouse to aim
  - 2) The angle of the throw (over-the-top to sidearm) by changing the angle of the throw in the control area. Changing the angle changes the hand on the main screen to rotate.
- Primitive Attributes
  - 1) Ball, bag, objects textured and alpha blended
- 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 bag
  - 3) Ball/Bag
    - React to collision with bottle/object
    - Moves toward bottle/objects
    - Moves before being “thrown”
- Pre-defined animation
  - 1) Pressing LMB causes ball to “fly” towards bottle/object
    - direction defined before throwing
  - 2) Game 2 includes program-controlled animated objects
- World-coordinate window
  - 1) as user moves hand, scope window changes to approximate aim at distant objects
- Multiple-views
  - 1) main window - game window displays throwing object/hand
  - 2) second window - “scope” window shows close-up view of distant object to increase accuracy of throw; can be zoomed in or out
- Semantics
  - 1) User attempting to collect as many tokens as possible

- 2) User-selectable difficulties incorporate different variables (external variables, speeds, movement randomization)
- Audio
  - 1) Background music
  - 2) Collision sound between ball/bag and moving objects
  - 3) Collision sound between milk bottles
- Suspend/Resume/Reset
  - 1) Pause game
    - Stops timer in Game 1
    - Stops movement of hand
    - Stops movement of moving objects in Game 2
  - 2) Reset game
    - Resets game back to starting amount of tokens
- UI Friendliness
  - 1) Throwing hand controlled by mouse
  - 2) Throw initiated by LMB
  - 3) Angle of throw controlled by RMB drag

## **LAYOUT/DESIGN**

See attached drawings...

## **OBJECT INTERACTIONS**

- Ball/Bag - 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/bag - 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

## **VIEW/CONTROLLER PAIRS**

- COMING SOON...