CS426 Game project proposal

Team name: Make More Engines, Inc.
Team members: Jake Scott, Nina Garcia
Game name: Two Generic Princesses
Proposed work:

1. Provide an overview of your game (give a short description about the game, i.e., game plot, objective, etc.):

   A princess and her manservant fight through a dungeon to reach their kidnapped princess friend. They must fight using basic attacks, which are common to both players, and special moves, which are controlled with either physical or musical input to the Kinect.

2. What are the characters and their resources (e.g., health points, number of characters, weapons, etc.) in the game? What are the behaviors of the characters and how do they interact with each other and their resources?

   a. Fighter princess – attacks using physical gestures to execute combos such as slashing, roundhouse kicking, and chokeholds.
   b. Mage princess – attacks by playing strings of notes on a piano. Different notes will produce different special effects such as freezing enemies, creating walls, and planting bombs.
   c. Enemies – susceptible to different types of attacks. They attack the players who are navigating the dungeon and opening puzzle doors.

3. What types of conflicts do you have in the game?

   When a player is solving a puzzle, the other player must fight off monsters that come to stop them, creating a sense of urgency in solving the puzzle.

4. Provide sketches/drawings to show how your game will be played. (Attach to this page)

5. What is the main language you will use to implement the game?
   C++

6. What tools/libraries you will use?
   We tried SDL and a 2D version of our game during the Global Game Jam. We will use OpenGL for the semester.

7. What types of user interface will you provide/use?

   Players will control the characters with the Kinect. One player stands in front of the Kinect and controls the fighter princess. One player controls the mage princess with a MIDI instrument hooked up to the PC. The HUD will show what special combos the users are executing, and a tutorial on the attacks will be integrated into the early levels.

8. What are the milestones you plan to have? Please give a short description and an expected finish time each for milestone.

   a. finish Kinect interface to read and save poses (Feb 1)
   b. design magic and physical attack systems (Feb 1)
   c. design enemies and weaknesses (Feb 1)
   d. figure out capabilities of MIDI keyboard (Feb 1)
   e. integrate Xbox controller into design (Feb 15)
   f. finish level editor interface (Feb 15)
   g. implement each character's basic attacks (Feb 15)
   h. design physical puzzles (Feb 1, Mar 1)
   i. design magical puzzles (Feb 1, Mar 1)
   j. implement magic attack HUD (Mar 15)
   k. implement enemy AI (Mar 15)
   l. implement physical attacks (Feb 15, Mar 15, Apr 1)
   m. implement magical attacks (Feb 15, Mar 15, Apr 1)
   n. implement enemy types (Mar 1, Apr 15)
   o. implement physical attack HUD (Apr 1)
   p. expand audio capability to account for octaves (Apr 1)
   q. implement puzzles (Mar 15, Apr 15, May 1)

9. What difficulties/challenges do you foresee? How do you plan to address them?

   Physical gestures: certain groups of poses are hard for the Kinect to read. We either have to avoid those poses or add extra code to sort them out. We must give the user some appropriate feedback about his physical attacks as they are happening.

   Magical attacks: we must design around the musical capabilities of the average player. We limit the note palette to a 5-note pentatonic scale so that most spells are aurally pleasing even if the player mashes keys.

   Targeting: players can't aim attacks with the control stick, since they will only have one hand available. Targeting can be automatic and smart (lowest HP), automatic and randomized, or controlled some other way.

10. How would you divide the tasks among your team members?
    Nina – sound assets, audio input, battle mechanics
    Jake – art assets, physical gestures, rendering engine

11. Why do you think your game is interesting/exciting?
    It gets players moving, making music, and solving problems. (The game was originally designed with an educational purpose in mind – to teach pitch recognition.)

12. Do you model your game after an existing game? If so, what is it?
    Magicka has a spell-casting system where one combines elements such as fire, ice, and lightning to produce unique effects. The character in our game casts spells with notes instead of elements.