Demo Game

Mention I thought a cool touch was changing the game icon for the window (top right) woo.

Talk about the map design. Tile-based, each level is define using different tile types.

Go battle. HP BARS! Mention we designed the gameplay using different game states (main menu, play screen, combat screen). This allows for efficient computing – because we only have to run update cycle for whichever screen we’re on.

Mention we were able to tap into the windows API and use message boxes (cool touch, no message boxes in XNA boo!)

battle system – rectangle collision detection into combatScene. First strike determined by a random generator initial seed generated when the game is first started and passed in whenever player enters CombatScene.  
  
battle delay implemented so game does not feel robotic instantaneous decreases in player health when player decides to attack.  
  
battle messages implemented through API boxes.

PHOTOSHOP JARGON.

Explain. Like explaining to an employer. The cool shit. The hard shit. Basically modify this doc.

ALGORITMS. PHYSICS. AI. MATH. MATRICIES\*\*\*\*\*\*\*\*\*\*

Artemis’ Diamond is a single player RPG game. The player plays as Artemis as he is on a quest to find the diamond that has been stolen from him. The evil enemies from the “OtherWorld” where everything is inverted and evil Artemis clones have stolen Artemis’ diamond and have given it to the leader of the “OtherWorld”, Bob the Blob. The goal of this game is to fight all the evil Artemis’ and destroy Bob the Blob to retrieve back the lost diamond.

The main controls of the game are the directional arrow keys, the enter button to select, and the escape key for battles. Artemis is moved by the directional arrow keys. He is able to collect health regeneration potions around maps and he walks up to the enemies to initiate a battle with them. Artemis’ goal on each map is to eliminate every enemy on the map in order to move on to the next level. Each “OtherWorld” enemy is uniquely and familiarly named. If Artemis fails to defeat the enemy, the game will end and the player will have to start over from the beginning. When Artemis defeats all enemies as well as Bob the Blob, Bob will release the diamond from hostage and Artemis will collect it to win the game.

The game itself is interesting because it is an RPG style game utilizing a tile map system similar to a classic Nintendo game. Each map is hard coded into the tilemap.cs class and each tile is defined to a specific tileID and rectangle on the tileset. The game has multiple scenes switching back and forth. There is a main menu, main game screen, battle screen, and a win/loss screen. \*\*\*\*\*\*\*HP BARS\*\*\*\*\*\*, \*\*\*\*\*\*\*\*MESSAGE BOXES\*\*\*\*\*\*, The battle screen and combat system is activated when the player intersects an enemy. This is checked by rectangular collision detection. The first strike is determined by a random generator that is initialized when the game is first started and is passed every time the battle screen starts with the combat system. It picks whether the enemy attacks first or the player attacks first. There is a battle delay implemented so the game does not have instantaneous decrease in player health when a player decides to attack and the enemy decides to attack right after.

Some difficult aspects were designing the levels with the tile system implemented. We would have to run every time we wanted to see changes in map. A level editor would be easier to see changes or by using an external level loading system using text files. There were a few bugs with the battle code with the enemy attacking twice and more after the battle was finished but these were resolved.