Software Prototype Contribution Summary

Group Name : Delta

Below the script listing is an additional summary of work contributed to the project. As members contributed a great deal of time to Delta Dungeon in their own ways, the code listing and lines written doesn't feel to be an accurate representation of the overall effort provided by team members. As such, please be sure to read the additional details below.

Member	Filename	Summary	Lines	Included?
Abhishek Rajbhandari	BoardManager	A work-in-progress template for RoomGeneration	80	No
	LevelRoomLayout	A work-in-progress template for RoomGeneration	117	Yes
	RoomGenerationDuplicate	Builds rooms with enemies and destructible walls	420	Yes
	LevelGenManual_Abhi	Child of LevelGenManualManager. Contains a unique set of 2D int arrays for levels. Set up and ready to be implemented in the next game update.	360	No
Eric Pacheco	CameraController	The Camera Controller sets the camera to focus on the player and follows the player around the game. The code checks to see if the player is alive. The code also checks if the camera is in the right position so that is it constantly tracking the player.	97	Yes
	CameraPerlinShake	This allows the camera to shake smoothly on different kinds of impacts and actions taken between the different objects[4]. Implements GenericShake but removes rapid movements between positions.	27	No

	GenericShake	This allows the camera to move rapidly from left to right and right to left on different impacts made between objects	27	No
	LevelGenManual_Eric	This is an extension of LevelGenManualManager.cs. This creates a level map from a 2d int array and sets the vertical and horizontal length while adding grids to the level lists.These are used for each of the game developer to test and modify their own scripts and see how they interface on the standard construction of the game.	360	No
	TestingCameraController	Precursor to CameraController	7	No
Jackson Wise	DestroyByContact	Destroys any object that it touches. Used as a "killing floor" below the lava.	1	Yes
	DestroyByTime	Destroys object after set amount of time. Used primarily to remove particle effects (explosions) after they've played.	2	Yes
	EnemyManager	Parent class for enemies. Contains health, movement speed, death functions.	85	Yes
	GameManager	Central hub for all game type managers. Handles team logo, main menu and level generation.	495	Yes
	Hazard	Parent class for objects that can harm Player. Used for traps and explosive enemies	102	Yes

	ItemCreator	Instantiates weapon pickups of the specified types.	69	Yes
	Boss_HS	Controls the creation, raising/lowering and rotation of all five boss spheres.	306	Yes
	Boss_HS_Manager	Manages Main Boss Sphere - health, vocals, transition between attack stages	172	Yes
	Boss_HS_RoomTeleporter	Creates boss stage and teleports player to it.	6	Yes
	MovementWave	Makes projectile move in a sinusoidal manner while travelling forward	37	Yes
	PauseMenu	Pauses game. Includes code for restarting and quitting the game.	125	Yes
	Pickups_GunEffects	Sets the randomized weapon types for a weapon pickup.		
	PlayerManager	Keeps track of public variables used for Player conditional checking. Includes Movement, attacking ability, hit points/life of Player.	146	Yes
	Projectile	Projectile is used for shooting projectiles. Includes speed, damage, firing location, death audio and animation.	126	Yes
	Proj_Grenade	Child of projectile. Explodes on contact with enemies, walls and the ground.	49	Yes
	Proj_Mine	Child of projectile. Explodes on contact with enemies and walls. Sticks to the ground.	106	Yes
	Proj_PlasmaOrb	Child of projectile. Dies on contact with boundary walls. Dies after 4 hits with enemies or destructible walls.	38	Yes
	TrackingMine	Script which makes explosive mine follow and detonate/damage when close to player. Trigger is player walking through sphere collider.	67	Yes

	WeaponsManager	Sets player gun components and fires player guns	490	Yes
	LevelGenManualManager	Creates base stage with floors, boundary walls, lava shader. Calls upon RoomGenerator	1092	Yes
	LevelGenManual_Jackson	Child of LevelGenManualManager. Contains a unique set of 2D int arrays for levels. Set up and ready to be implemented in the next game update.	360	No
	_CONSTANTS	Constant static ints for the weapon designations. Discarded because static ints are public and can be altered.	6	No
	_DEBUG	Static bools for testing conditions. Discarded because static bools are public and can be altered.	4	No
	TestEnemy	Precursor to EnemyManager.	73	No
	TestingBulletDamage	Precursor to Projectile.	60	No
	TestingPlayerSphereController	Allowed simple horizontal and vertical movement.	7	No
	TestingPlayerWeapon	Precursor to WeaponsManager.	23	No
	TestingRoomSizeGenerator	Used to make test rooms and give a feel for how large Unity Units are in-game.	136	No
Joshua Peckham (** Majority of my work didn't require coding**)	Fight	Controls melee attacks for Unity-Chan.	48	Yes
	Respawn	Respawns player at set location.	9	No
	LevelGenManual_Joshua	Child of LevelGenManualManager. Contains a unique set of 2D int arrays for levels. Set up and ready to be implemented in the next game update.	360	No
	Character Development	● Developed Unity chan ○ 4 different models, each replaced the next with	0	Yes

		improvements towards the end goal.
		
		● Animation and transition between the different animations
		   ○ 3 in-depth blend tree animations
		   ○ Adapted fighting capabilities		

Steven DeMartini	DeltaMusicManager	Stops old music track and plays new music track passed to it.	4	No
	DJ	Sends new music track to DeltaMusicManager.	4	No
	SoundManager	Import script designed to play audio clips.	58	No
	LevelGenManual_Steven	Child of LevelGenManualManager. Contains a unique set of 2D int arrays for levels. Set up and ready to be implemented in the next game update.	360	No

All members:

Documentation:

- All members contributed a great deal of effort in preparing the game documentation. Groups members would typically meet up for two hours each day before class, and on several

Abhi

Scripts:

- RoomGeneration
    - Switch statement and empty methods written by Jackson, variables and methods logic

written by Abhi.

      - A large amount of work went here, as this script alone was just under 500 lines. This script was essentially the heart of the game's level generation.

Eric

Enemies:

- Imported and set up turrets which functioned (were able to fire at a target) (ran out of time and didn't get a chance to add to the main scene).

Sprites:

- Created dozens of sprites to be used for particle system special effects.

Special Effects (complete Particle Systems):

- Projectile
- PS_Explosion_Teleport 1
- PS_Healing
- PS_OrbElectric
- PS_teleport

Special Effects (individual components of Particle Systems):

- Projectile:
      - partilces
      - partilces (1)

- PS_Explosion_Teleport 1:
      - GroundMark
      - GroundSmoke
      - GroudSmoke (1)
      - BlackSmoke
      - ExplosionSmoke
      - impact
      - GroundBeam
      - VerticleImpact
      - RoundSmoke
      - particles
      - BeamLight

- PS_Healing

- Base
        - HealingIcon
        - particles
        - vertical Cuts

_ PS_OrbElectric
        - Cirlce
        - Electric Beam
        - particles
        - smokeOutside

_PS_teleport
        - GroundSymbol
        - GroundSymbol (1)
        - circle
        - VerticalCircle
        - Smoke
        - particles

Joshua

Character Development:

- Bone structure manipulation, animation, motion capture implementation, user control

Steven

Other:

- Most of Steven's work involved being a secretary cheerleader and he contributed greatly to the game documentation (SRS, SDS)
- Provided many game ideas and team encouragement

Jackson

Scripts:

Scripts written by Jackson were used as a foundation for several aspects of the game, though the work done by the rest of the team provided the framework which was built upon the foundation. Although the script listing appears heavily weighted towards Jackson creating more scripts, the contributions by the entire team were roughly equal, and the unity of team participation was necessary to create a functioning and exciting prototype of Delta Dungeon.