CPSC 386 Final Project, due Friday, 21 Oct 2018 (at 2355)

Your name John Margis

Repository `https://github.com/` margisj / CrossyRoad

Verify each of the following items and place a checkmark in the correct column. Each item incorrectly marked will incur a 5% penalty on the grade for this assignment.

Completed	**Not Completed**	**Crossy Road**
[✓ ❑]	❑	Have Crossy Road installed as an app on their mobile phone.
[✓ ❑]	❑	Game has startup screen with Crossy Road logo sliding in from the upper right at a down angle of 30 degrees.
[✓ ❑]	❑	Implemented the game's HUD (head's up display) showing the high score, current score (number of jumps), if this is a new high score, and coins collected.
[✓ ❑]	❑	Implemented at least four of the 3d actor assets in the game in MagicaVoxel (Chicken, Bunny, Duck, Fish, Frog, Pig, Robot, Sheep, Steer, Unicorn, Wizard, or custom actors). If you have custom actors, list them here: ____________, ____________, ____________, ____________. The rest can be developed collaboratively.
[✓ ❑]	❑	Implemented the safe grassy area 3d assets in MagicaVoxel, **including code to populate them w/trees and rocks**. Trees/rocks should allow the actor to pass.
[✓ ❑]	❑	Created highway obstacle (1-10 lanes) area 3d assets in MagicaVoxel—w/**code to populate them with cars/trucks, and control their movement**. Cars/trucks are timed to allow the actor to cross. Multi-lane roads must have lane markers.
❑	[✓ ❑]	Implemented railroad obstacle area 3d assets—1-2tracks—w/**code to populate them w/trains/crossing arms and trains, and control their movement.** When a train is coming: ring a bell, light crossing arms brightly, and flash left/right. Trains are 5-6 cars long, w/ leading, trailing, and interior cars. Trains should fit on the tracks. Multi-track trains should allow the actor to pass.
❑	[✓ ❑]	Implemented the water obstacle area 3d assets in MagicaVoxel—1-6 logs wide, randomly selected—with code to populate them with logs/lily pads, to control their movement, and control the waterfall's appearance. Logs/lily pads are timed to cross safely without being carried out of the water's range of safety. Actors carried into the waterfalls should cause the actor's death.
[✓ ❑]	❑	Imported all actor, safe area, obstacle and miscellaneous 3d assets into Unreal 4, and rotated and scaled them to their proper proportions.
[✓ ❑]	❑	Correctly implemented crouching and jumping with delay with Blueprints or in C++, so the actor crouches as long as the arrow key (left/right/up/down) keys

		are pressed, but jumps immediately when it is released.
[✓ ❑]	❑	Correctly implemented the actor’s movement through the game, so it jumps from spot to spot, turns to jump in the correct direction, makes jumping sounds (and occasional actor sounds), refuses to move into immovable obstacles (trees, rocks, train crossing arms), and dies when colliding with fatal objects.
❑	[✓ ❑]	Implemented the particle system for the splash of water when the actor falls in, and the explosion of the actor when it is run over, or runs into a train or oil truck. Correct colors should be emitted, according to actor and oil truck flames.
❑	[✓ ❑]	Implemented the eagle asset, imported it, and written the code that controls its swoop down and carry away of the actor if there is no forward movement for a short time, or if the actor moves backwards several times.
[✓ ❑]	❑	Implemented the dynamic generation/destruction code for allowing the level to be continuously populated as the actor moves forward in the world. Note: the actor cannot move backwards to areas that have already been destroyed.
[✓ ❑]	❑	Used Audacity to record the music and game sounds, and implemented them. Actor sounds (cluck, quack, flopping, ribbit, baa, neigh, boing, ____) when moving, and when dying (louder, more shrill versions of above). Horn sounds, bell for train crossing warning, swoosh when train goes by, eagle shrieking.
[✓ ❑]	❑	Others (at least three) have played your game and signed off on it as fun.
[✓ ❑]	❑	Project directory pushed to new GitHub repository listed above
[✓ ❑]	❑	Project directory has been pushed using a GitHub client, not by manually dragging-and-dropping files onto the GitHub web page.

Comments on your submission