The Magic Shepherd

Group 6

Theme and interpretation

“You only have one magic shepherd.”
One magic shepherd will be used to herd all the sheep to the barn.

Game idea in about 100 words

The game revolves around a magic shepherd who has to get his sheep to his barn. He has to do this by guiding them across his farm while trying to avoid anything that can harm the sheep. In the levels you can buy power-ups to make your job a little bit easier. These power-ups can also be farmed while playing the level and finding these things in certain spots.

Key components

- Computer Graphics:	13*	Matthijs
o 3D animated models	***	
▪ 3D models for the shepherd, sheep and enemies, these have to be animated because they are all a vital part of the game		
o Texture as input	**	
▪ Textures to be used as input for the levels using algorithms		
o Animated textures	*	
▪ Animated textures could be used in the playing field, e.g. a small pond		
o Audio mixer effects	*	
▪ Sounds for the sheep and certain movements		
o Play with lights and shadows	*	
▪ Try to get good shadows and a well-lit playing field		
o Start pause end screen	*	
▪ Start, pause and end screen for the game		
o High scores	*	
▪ High scores added for the amount of sheep and other things you have picked up or used		
o Options	*	
▪ Settings menu, mute sound, change game speed etc.		
o Credits	*	
▪ Description of us, the people who made the game and what our roles were		
o UI animations	*	
▪ Animations in the UI interface		

- Artificial Intelligence (AI) — 12* — Dylan
    - Huge amount of differently dumb enemies — ***
        - This is mainly for the sheep. They have a mind of their own but will try to group together
    - Path finding using own algorithm — ***
        - When the shepherd is not in reach of the sheep and the sheep have been standing still for a certain time, they will start to wander to another position. This will be accomplished by means of a customized algorithm.
    - Use a neural network — ****
        - We will use PSO (Particle Swarm Optimisation) to determine the position of sheep within the herd and the position of the entire herd.
    - Smart enemies you always lose from — **
        - Witches that fly around the map that will steal sheep from you and you can't get them back. These enemies either know where the sheep are or will learn.
- Web & Databases, Gama Analysis — 6* — Joris
    - Your own game analysis tool on your webserver — ****
        - We will look at what the player has done during the game. How much wheat he gathered and fed to the sheep, how long he took and how much sheep he saved.
          This is where we will analyse the playthrough data and where we will give an analysis in a visual form like graphs or /and a textual format. This analysis will have to happen automatically after the player has finished playing and after the data has been sent to the web server.
    - Collect and show high scores from webserver — **
        - We are going to determine a high score based on your performance in the game and this performance will be collected and shown from a webserver.
- Programming — 10* — Umbriël
    - Procedurally generated levels/weapons — **
        - The different levels could be procedurally generated and each time get a little harder.
    - Moving platforms — *

        - A couple of different moving platforms in the level on which the sheep will stay in place but can also move on their own.
    - Race against the clock *
        - Have some (or maybe all) levels have a count down and you'll have to get your sheep into the barn before time reaches zero.
    - FPS independent **
        - We will use deltaTime etc. to make sure the game runs at the same speed regardless of processor speed. This is done, because otherwise things like animations, sheep, and the shepherd will have a too high execution speed and will make the game unplayable.
    - Game speed can be changed by player *
        - The whole game will be either slowed down or faster.
    - Use Unity's triggers to trigger certain actions *
        - Triggers to pick up certain items
    - Use Unity's physics for all movement, collision etc. **
        - Unity's physics will be used for all the movement, collisions with the environment etc.

Total: 42*

Roles

Umbriël Post	- U.Post@student.tudelft.nl	- Lead Programmer
Dylan van Opdurp	- dylanvanopdurp@gmail.com	- Game Designer
Joris Jonk	- joris33@gmail.com	- Producer
Rick Oskam	- r.oskam@student.tudelft.nl	- World Builder
Matthijs Verrijzer	- M.G.P.Verrijzer@student.tudelft.nl	- Lead Artist

Time schedule

Week 1:

- Define the core project
- Start making the core elements

Week 2 – 4:

- Make prototypes and start testing these
- Try to get most elements together into a functioning game (doesn't have to be good as long as it somehow works)
- Make an early access game

Week 5 – 8:

- Work on the game by adding textures, particle effects etc.
- Make a beta game

Week 9:

- Finish the game by fixing minor bugs

GitHub page

https://github.com/Dylan231/MagicShepherd