Landon Vance

Battlefield Strategy Design:

The goal of this project is to create a turn based strategy game that rewards players for coordinated unit placement, movement, and attacks. Through the use of flanking, varying terrain, and unit types this game should encourage players to develop unique, responsive, and exciting strategies. Design of this game should focus on unit interaction, and be split according to three basic categories: Unit Properties, Unit Movement, and Unit Attack.

Overarching Design:

-Made in GameMaker

-2D overhead view

-Grid map pattern

-Many different “unit” types

-Many different “terrain” types

Basic Units:

-Archer: Medium damage, high range unit

-Calvary: High damage, mobile unit

-Heavy Infantry: High-armor, low mobility unit

-Light Infantry: All around unit

-Mage: Ranged, unarmored unit

Unit Properties:

In order for units to interact with each-other it is necessary for them to share certain basic attributes. To ensure that the units “feel” distinct, however, it is also necessary that each unit have varying magnitudes of these basic properties, as well as have some special properties that make them unique.

Basic Properties:

-Health

-Armor

-Attack Damage

-Attack Range

-Movement Range

Special Properties:

-Archer:

-Extra High-Ground advantage

-More vulnerable to flanking

-Calvary:

-Extra flanking damage.

-Severely inhibited by rough terrain

-Heavy Infantry:

-Less vulnerable to flanking

-Gain armor for adjacent (allied) heavy infantry

- Light Infantry:

-Less affected by rough terrain

-Mage:

-Attack ignores armor

-Gains extra attack for adjacent (allied) mages

Terrain Properties

-Terrain Types:

-Mountains:

-Armor bonus

-Bonus attack damage vs. units not on mountains

-Slows movement

-Hills:

-Bonus attack damage vs. units not on hills or mountains

-Slows movement

-River:

-Armor reduction

-Stops movement

-Swamp

-Massive armor reduction

-Stops movement

Unit Movement:

One of the major goals of this game is to reward intelligent unit positioning, it follows that the way different units move should affect this positioning. Unit movement is intuitive within the grid pattern. Units may move either up, down, left, or right. Each of these movements costs one move-range. Terrain interaction with movement should make each map (or room in game-Maker Speak) feel like a new challenge. Hills and Mountains should cost more “movement units” to travel over. Because of this mountains will represent a significant barrier to army movement while providing a tactical advantage (as explained in the Unit Attack section). River and swamp terrain should bring a unit to a halt (that is moving onto a river or swamp square should consume the rest of a unit's movement for that turn). By doing this water effectively segments battlefields, making crossing rivers very dangerous, and holding river crossings crucial. In order to prevent abuse of good defensive terrain position unit “stacking” is prevented; that is only one unit may occupy a grid space.

Unit Attack:

Attacking always occurs after a unit's movement (if the unit chooses to move at all). Attacking ends a unit's turn. The attack phase of the game hinges upon unit placement and terrain interaction. The actual attack calculations are standard and quite simple (in pseudo-code):

VictimUnit.health -= AttackingUnit.attack – VictimUnit.armor

Any effect placement or terrain has on the attack phase stems from attack and armor manipulation. A few examples:

-Heavy Infantry being attacked will take less damage if there are adjacent allied Heavy Infantry units because the heavy infantry will have extra armor.

-An archer attacking from a mountain will do extra damage because of the mountain attack damage bonus.

-A Calvary unit will deal more damage to an archer's flank because the Calvary deals extra damage while flanking and the archer has less armor because it is getting flanked.

Extra Customization:

Additional choices should be provided in order to diversify the tools at player's disposal (and keep the game interesting). These choices include a choice between additional unit abilities and/or special units.

Additional Special Abilities:

-Archer:

-Extra attack range

-Calvary:

-Can use “leftover” movement to retreat after attacking

-Heavy Infantry:

-Deals damage back when attacked.

-Light Infantry:

-Immune to terrain disadvantages

-Mage

-Ability to heal a nearby unit

or

-Ability to lower a nearby enemy unit armor

Additional Units:

-Ranger: Enhanced Archer

-First-Sword: Enhanced Infantry

-High-Mage: Enhanced Mage

-Commander: Grants bonus stats to all nearby units.

Additional design decisions:

Game Modes:

In order to further diversify game play multiple game modes should be provided. These different modes should encourage and reward different styles of play.

-Last Man Standing Mode: Victory is gained by destroying all of the opponents units.

-Capture the Flag Mode: Victory is achieved by destroying one particular unit (unit instance)

User interface:

The user interface should use a mix of keyboard and mouse input, and should be minimalistic and intuitive.

Class Design:

Class design in this game is intuitive, if a little different because of Game-Maker Language. The GameMakerLanguage version of classes are objects. Every specific unit-object (Archer, Calvary, etc...) inherits from a parent object simply named 'Unit'. Any shared functionality should reside within the parent Unit code. Every terrain type should inherit from a parent object simply named 'Terrain'. As with the units, any shared functionality should reside in the code of the parent object.

Story:

Traditionally, 2D games do not include intricate story lines (if they include story at all), and this game will be no exception to that rule. Some introduction/tutorial should by given stating that the game is about the strategic clashing of armies, but a story-line in the conventional sense will not be included.