AP Computer Science Final Project - README

Mecha Wars

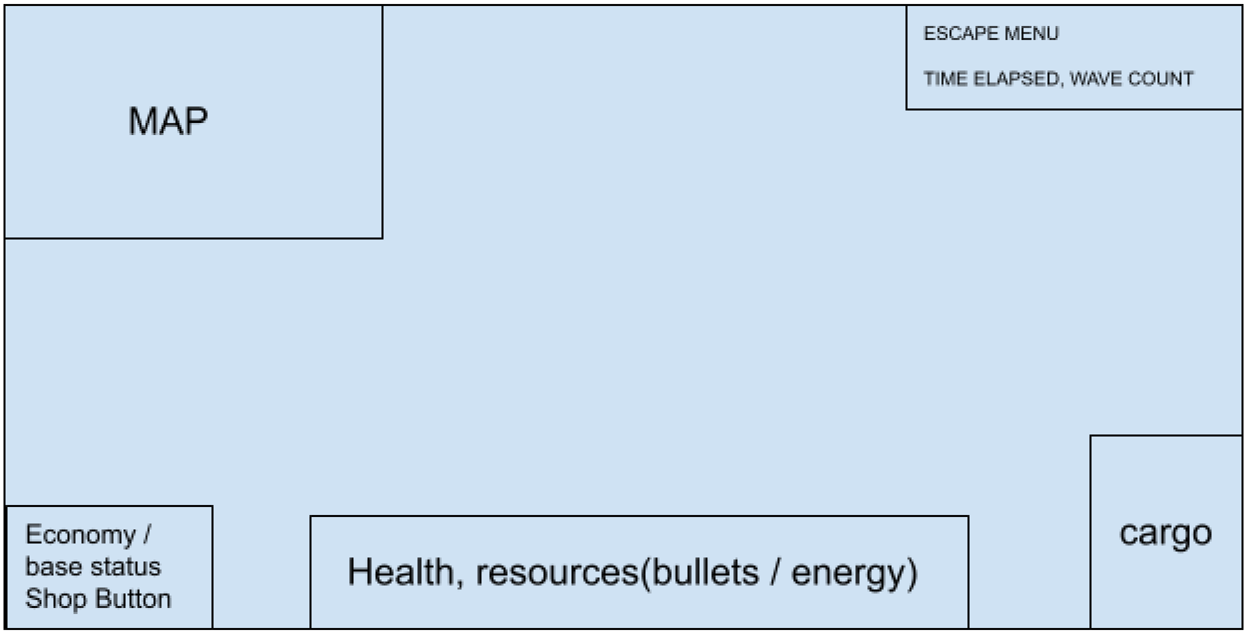
Authors: Harinandan Kotamsetti , Tanay Bartwal , Alexander Yue
Revision: 4/15/2022

Introduction:

Mecha Wars is a 2d video game developed with Java using the processing library. This game is a real time strategy action game. This program has a fusion of old arcade styled survival games with the more newer tower based games. Mecha Wars solves the problem of newer survival based games not having much customization. The game is a survival mode where the goal of the game is to protect the base from invading waves of enemies for the longest time possible. Mecha Wars is unique compared to other tower based games due to its ability to continue to customize units after being deployed. Other programs, like Clash of Clans only allows you to place troops but you can’t customize them after being deployed. Your base is a building at the heart of your area. You control the “hero” unit of your army, from which you can commission constructions of new army units including both defense and offensive units. The hero is a mech. Our game allows the user to choose between 3 different types of mechs each with unique abilities. Mecha Wars’ primary target users are people looking for a modern survival game with customization. Mecha Wars allows people with different play styles to play how they like. Mecha Wars has a mobility component which allows a specific Vanguard (one of the mechs) to move around the map faster. Mecha Wars also has a mech called Stelwart which has more health but has slower mobility. Finally, Mecha Wars offers Melner a mech which gives the best of both worlds and has moderate tankiness and moderate mobility.

Over time, your base becomes attacked by enemies which over time increase in basic stats. You can upgrade your army’s offensive and defensive abilities by using the shop, where credits earned from downing enemies are used to purchase upgrades for your army. Credits are earned after each wave of enemies are cleared. There are 3 different types of enemy waves. Small, Medium, and Large. More credits are given for bigger waves. The game ends once your base succumbs to the enemy attacks, practically, your health of your base reaches 0 at this end game phase.

Instructions:


MAP
ESCAPE MENU
TIME ELAPSED, WAVE COUNT
Economy / base status Shop Button
Health, resources(bullets / energy)
cargo

GAMEPLAY:

In the main mode of the game (single player survival), you will be controlling a mecha bot with the WASD keys. The mouse will be used to fire different types of ammunition, one for left and another for right. The different types of ammunition depends on the mecha bot you choose. The health of each unit on the screen will be indicated with a small health bar on the top of each of the units. The health of your own mecha bot will be in the top left

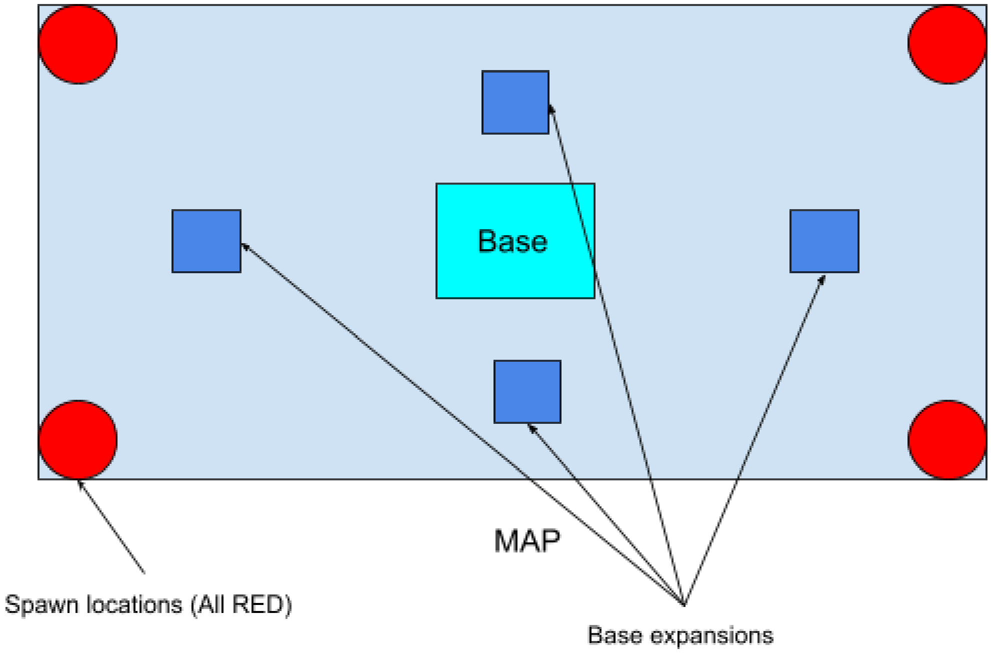
There will be a shop section that can be accessed. The shop menu is a different screen, which can be accessed with a button on the main gameplay screen. Once the shop button is clicked, the screen will feature a list of units and upgrades you can buy along with the current credit count.

WAVE MANAGEMENT:

Enemies will approach and try to take over the base in waves. There are three types of enemies.

- Small
- Medium
- Large

For every wave, there is a 60 second preparation time (buy units and upgrades from the buy menu in this preparation phase). Once the 60 second preparation time is passed, you will enter a state where the wave approaches your base to attack your units.



<u>Features List (THE ONLY SECTION THAT CANNOT CHANGE LATER)</u>:

Must-have Features:

[These are features that we agree you will *definitely* have by the project due date. A good final project would have all of these completed. At least 5 are required. Each feature should be fully described (at least a few full sentences for each)]

- Map system:

  At least 2 premade maps for game design. There will be a minimap in the bottom left corner of the screen, which will show what is on the map at this time.
- Different Mech models:

  Different mechs can be used, all of which have different stats and abilities, each having advantages and disadvantages.
- Army unit system:

  Automatic attacking units of different types. Army units include both defensive units which don't move and movable units like tanks. Units should not contact each other.
- Premade spawn locations at base.

  Base systems, each with its health and default defensive units.
- Shop system:

  Get credit for kills, use credits to make units (increase army size/strength).

Shop will contain purchasable unit constructions and main unit upgrades.

- Waves of enemies

Want-to-have Features:

[These are features that you would like to have by the project due date, but you're unsure whether you'll hit all of them. A good final project would have perhaps half of these completed. At least 5 are required. Again, fully describe each.]

- Defense Movement Mechanism:

  More advanced defensive units, each of the units have a smarter system in terms of targeting appropriate targets.
- Pets with different abilities:

  Purchasable (high price) pet units with unique abilities, with higher properties than other units.
- Intelligence factor: The NPCs are more intelligent with their attacks making the difficulty of destroying the base more difficult. Coordinated waves and priority targeting of certain types of units.
- Pool command system:

  System to select multiple units and command them at the same time, rather than making them units.
- More buildings to expand base and construction capabilities.

Stretch Features:

[These are features that we agree a *fully complete version of this program would have, but that you probably will not have time to implement*. A good final project does not necessarily need to have any of these completed at all. At least 3 are required. Again, fully describe each.]

- Have a story for the game
- Implement a PvP game mode, using networking to communicate between two computers to play the game.
- Multiple players are able to play on each side in a team based game mode
- Multiplayer mode where multiple people are able to play on one side or against each other.

Class List:

[All attributes including health, damage, and other values will be made proportional to each other, actual values vary ingame]

- Unit [all objects that have basic properties like health, energy, damage stats]

- NPC [Auto units, which are simply used as part of army] [AI for Unity game developers: How to emulate real-world senses in your NPC agent behavior]
    - Mobile Units [automatic units which attack] (want to have)
        - Soldier bot.
            - High mobility and small in size.
            - Low damage (⅓ damage of tanks)
            - Cheap (5x cheaper than tanks)
        - Tanks.
            - Low mobility and large in size.
            - High damage
    - Defense Units [stationary automatic units which defend]
        - Cannon artillery (high damage, single target, high speed)
        - Missile launcher (missile projectile, aoe(stretch feature), multiple missiles are launched at the same time)
        - Mortar (single aoe strike, huge range, but blind spot when enemies are closer)
- Main units [units which have more diverse settings and]
    - Controllable
    - Player [superclass containing common properties including stats and attack moves]
        - Mech type 1: *Vanguard*
            - Attributes include:
                - High mobility (speed, dashes,)
                - Low health
                - Moderate damage stats
            - Weaponry:
                - Low damage machine gun. High fire frequency.
                    - Shoots “bullet” projectiles.
        - Mech type 2: *Stelwart*
            - Attributes include:
                - Tankiness (health, shields)
                - High energy and ammunition storage.
                - Low damage stats, but access to damage over time and mines.
            - Weaponry:
                - Fire, damage over time.
        - Mecha type 3: Melner
            - Attributes include:

          - Moderate mobility and tankiness (blend of 1 and 2).
          - High raw damage output using ammunition.
        - Weaponry:
          - Slow fire frequency, high damage.
            - Shoots “bullet” projectiles.
    - Pets (stretch)
- Weapon projectiles
  - *Bullet [goes in straight line, less damage]*
  - Missile [follows target, aoe]
- Shop
  - A shop system, coded as it extends from screen.
  - Contains purchasable:
    - Units
      - Adds units that can be placed on field.
        - 1x tank.
          - Less mobile unit with high base health.
          - Upgrades (see shop section) will improve damage and health of this unit.
        - 1x soldier unit
          - More mobile unit with high movement speed.
          - Upgrades (see shop section) will improve damage and health of this unit.
    - Upgrades.
      - Improves characteristics of main mecha bot, and all units on field.
        - +1 level for tank
          - For every level, damage increases.
          - For every third level, health increases.
          - Movement speed and all other attributes are unchanged.
        - +1 level for soldier
          - For every level, damage increases.
          - For every third level, health increases.
          - Movement speed and all other attributes are unchanged.
- Map system
  - **Field objects**
    - Array that will contain all objects that are currently placed on the field.

- Draw method will cycle through all objects in the field (enemies and allied units including the base), render their operations, and display them in the processing window.
- GameUnit
- Inanimate

- Screens
    - GameScreen
    - MenuScreen
    - Screen
    - ScreenSwitcher
    - SelectScreen
    - ShopScreen
- HomeBase
- DrawingSurface

Credits:

- Project Contributors:
    - Alex Y:
        - Added additional classes on UML to project file
        - Added shop menu button to game screen, upgrade/buy buttons
        - New javadocs
        - Shooting mechanics
        - Base health
        - Finalized UML
    - Harinandan K:
        - Preliminary project setup.
        - High level class structure.
        - GameScreen, SelectScreen
    - Tanay B.
        - Created Mechs (different models)
        - Enemy waves
        - Worked on Projectile and Missile
        - Completed UML Diagram
        - Added missing class files