Working Breakdown Agreement

Team Members

- Alexander Authors will be referred to as aaut
- Joeseph Terry Ward will be referred to as jwar
- Jared Neil Tucker will be referred to as jtuc

Assignment Break Down

- Req1: Let it Grow (jtuc)
- Req2: Jump up Super Star (aaut)
- Req3: Enemies (aaut)
- Req4: Magical Items (jtuc)
- Req5: Trading (jwar)
- Req6: Monologue (jwar)
- Req7: Reset Game (jwar)

REQ1: Let it grow!

Person(s) responsible: jtuc will be responsible for the class diagrams and package,

ASSIGNMENT 2 EDIT: jtuc is responsible for implementing this requirement in code.

Scenario: We want to enrich the world with some plants

Some classes we will need to implement are:

- Sprout()
    - it has a 10% chance of spawning Goomba on its position in every turn.
    - it takes 10 turns to grow into a small tree/Sapling.
    - possible implmentation:

```
public class Sprout extends Tree {
    // inherit attributes such as:
    /*
    int turnsAlive;
    */
    // location?
    Location sproutsLocation;
    // countTurns: int

    public Sprout(int x, int y) {
        sproutsLocation = new Location(x, y);
    }
    public Goomba spawnsGoomba() {
        // create Goomba
        return Goomba();
    }

}
```

- Sapling()
    - it has a 10% chance to drop coins on its location ($20) in every turn.
    - It takes another 10 turns to grow into a tall tree/Mature.
- Mature()
    - It has a 15% chance to spawn Koopa in every turn.
    - If an actor stands on it, it cannot spawn Koopa.
    - For every 5 turns, it can grow a new sprout in one of the fertile squares randomly.
        - if there is no fertile square, it will stop spawning sprouts.
        - fertile squares are currently only Dirt.
    - It has 20% chance to wither and die in every turn.
        - Becomes Dirt
- The generated coins should stay in the same position even if the tree grows.

REQ2: Jump Up! Super Star!

Person(s) responsible: aaut will be responsible for this

ASSIGNMENT 2 EDIT: aaut is responsible for implementing this requirement in code.

Scenario: A Super Mario game will not be complete without having a jump feature.

When standing next to high ground, actor can jump on it.

- going up requires a jump
- going down does not require a jump
- jumps have success rates and fall damage consequences.

Implementation notes:

- could implement some enums to assign and calculate fall damage e.g.

```
enum FloorType {
    GROUND,
    SPROUT,
    WALL,
    SAPLING,
    MATURE
}

enum FallDamage {
    LOW,
    MEDIUM,
    HIGH
}
```

REQ3: Enemies

Person(s) responsible: aaut is responsible for this.

ASSIGNMENT 2 EDIT: aaut is responsible for implementing this requirement in code.

The game in its current stage implements two enemies:

- Goomba:
  - Starts with 20hp
  - attack is 50% hit rate, 10dmg
  - every turn has 10% chance to suicide (removed from map).
- Koopa (Koopa Troopa?):
  - starts with 100hp
  - when defeated, not removed from map.
  - stay on ground (D), can only be broken by wrench.
  - Wrench: 80% hit rate 50dmg
  - destroyed shell -> super mushroom

Key Points:

- Once enemy is engaged in a fight (player attacks enemy or enemy attack player). it will folow the player.
- unconscious enemy must be removed from map (hp = 0?).
- Enemies cannot enter floor.

REQ4: Magical Items

Person(s) repsonsible for this: jtuc will be responsible for this.

ASSIGNMENT 2 EDIT: jtuc is responsible for implementing this requirement in code.

Two magical items:

- Super Mushroom:
  - denoted with (^)
  - if consumed:
    - increase max hp by 50
    - the display character evolves to the uppercase letter
    - can jump with 100% success rate
- Power Star:

    - denoted with (*)
    - will disappear after 10 turns regardless of where it is
    - if consumed:
        - heals 200hp
        - makes consumer invincible
        - if there are more Power Stars in inventory, then turn their timers back to 10 turns.
        - allows player to walk to higher grounds (does not need to jump)
        - for every destryed ground, drops coint ($5)
        - immunity -> all enemies attack become 0dmg
        - when active, a successful attack on an enemy kills it instantly.

REQ5: Trading

Person(s) responsible: jward will be responsible for this.

ASSIGNMENT 2 EDIT: jwar is responsible for implementing this requirement in code.

Scenario:

The coin ($) currency is the player uses to trade for items.

- coin has integer value
- coins spawn randomly from Sapling()
- collected coins traded with Toad

Trade Options:

- Super Mushroom: $400
- Power Star: $600
- Wrench: $200

Toad (O) is a friendly actor.

NOTE:

- Items bought from the store are not droppable
- If not sufficient money, print("You do not have enough coins!")

REQ6: Monologue

Person(s) responsible: jward will be responisible for this.

ASSIGNMENT 2 EDIT: jwar is responsible for implementing this requirement in code.

Scenario:

You will be able to speak with Toad (O).

The Toad will say 1 of 4 sentences randomly:

- "You might need a wrench to smash Koopa's hard shells."

- "You better get back to finding the Power Stars."
- "The Princess is depending on you! You are our only hope."
- "Being imprisoned in these walls can drive a fungus crazy 😦"

NOTE:

Once the player holds a Wrench, he won't' say the first sentence. When the Power Star effect is active, the second monologue must not be printed in the console. Other than these two scenarios, he will randomly pick a monologue above.

REQ7: Reset Game

Person(s) responsible: jward will be responsible for this.

ASSIGNMENT 2 EDIT: jwar is responsible for implementing this requirement in code.

Scenario: If game becomes too overwhelming becomes hard to walk around so need to reset.

- need to be able to reset game at anytime.

Consequences of resetting:

- trees have 50% chance to become dirt.
- all enemies are killed.
- player status goes back to normal (not super mushroom or power start)
- player health reset to full
- remove all coins from ground
- Super Mushrooms/Power Stars may stay.

WBA based off Packages

- ground (jtuc)
  - contains:
    - plants
  - relates to:
    - items
    - actor
    - enemies
- actor (aaut & jwar)
  - contains:
    - << abstract >> Entity --> Toad()
    - << abstract>> Enemy --> Goomba(), Koopa()
    - << abstact >> Actory <-- Player()
    - Location()
  - relates to:
    - << abstract >> Ground
    - item

        - wallet
- item (jtuc)
    - contains:
        - wrench
        - Super Mushroom
        - Power Star
        - Coin
    - relates to
        - << interface >> isConsumable()
        - << abstract >> WeaponItem()
        - << abstract >> Actor
        - Ground
        - wallet
- wallet (jwar)
    - contains:
        - Coin
    - relates to:
        - Entity Toad
        - Player
- reset (jwar)
    - contains:
        - << interface >> Resettable
        - Reset
        - Reset Manager
    - relates to:
        - Ground
        - Enemies
        - Player
        - Item

Sign Here

I Joseph Ward agree

Jared Tucker adheres to these guidelines

I Alexander Authors agree to this wba

Reviews and Reviewers

Deliverables due Monday 2nd of May, 2022

Deliverable	Reviewer	Date Reviewed
Alexander Authors	Jared Tucker	02/05/2022

Deliverable	Reviewer	Date Reviewed
Joseph Ward	Alexander Authors	02/05/2022
Jared Tucker	Joseph Ward	02/05/2022
