```
attack() {
  this.increaseMana();
  this.updateAnimation(this.player, 'idle', 'attacking');
  var damage = this.calculateDamage(5, 10);
  this.enemy.health -= damage;

  if (this.isGameOver()) {
    return;
  }

  this.enemyAttacks();
},
```

THE GAME SEQUENCE

```
data() {
  return {
    player: {
      animation: 'idle',
      imagePath: require(`@/images/viking/profile.jpg`),
      health: 100,
      mana: 0,
    },
    enemy: {
      animation: 'idle',
      imagePath: require(`@/images/enemy/profile.jpg`),
      health: 100,
    },
    gameIsRunning: false,
    animationDelay: 300,
  }
},
```

```javascript
attack() {
  this.increaseMana();
  this.updateAnimation(this.play
  var damage = this.calculateDam
  this.enemy.health -= damage;

  if (this.isGameOver()) {
    return;
  }

  this.enemyAttacks();
},
```

```javascript
data() {
  return {
    player: {
      animation: 'idle',
      imagePath: require(`@/images/viking/
      health: 100,
      mana: 0,
    },
    enemy: {
      animation: 'idle',
      imagePath: require(`@/images/enemy/
      health: 100,
    },
    gameIsRunning: false,
    animationDelay: 300,
  }
},
```

```
attack() {
  this.increaseMana();
  this.updateAnimation(this
  var damage = this.calcula
  this.enemy.health -= dama

  if (this.isGameOver()) {
    return;
  }

  this.enemyAttacks();
},
```

```
isGameOver() {
  if (this.enemy.health <= 0) {
    this.updateAnimation(this.enemy, 'dead', 'dying');
    this.dialogTitle = 'You won!';
    this.showDialog = true;
    return true;

  } else if (this.player.health <= 0) {
    this.updateAnimation(this.player, 'dead', 'dying');
    this.dialogTitle = 'You lost!';
    this.showDialog = true;
    return true;
  }
  return false;
}
```