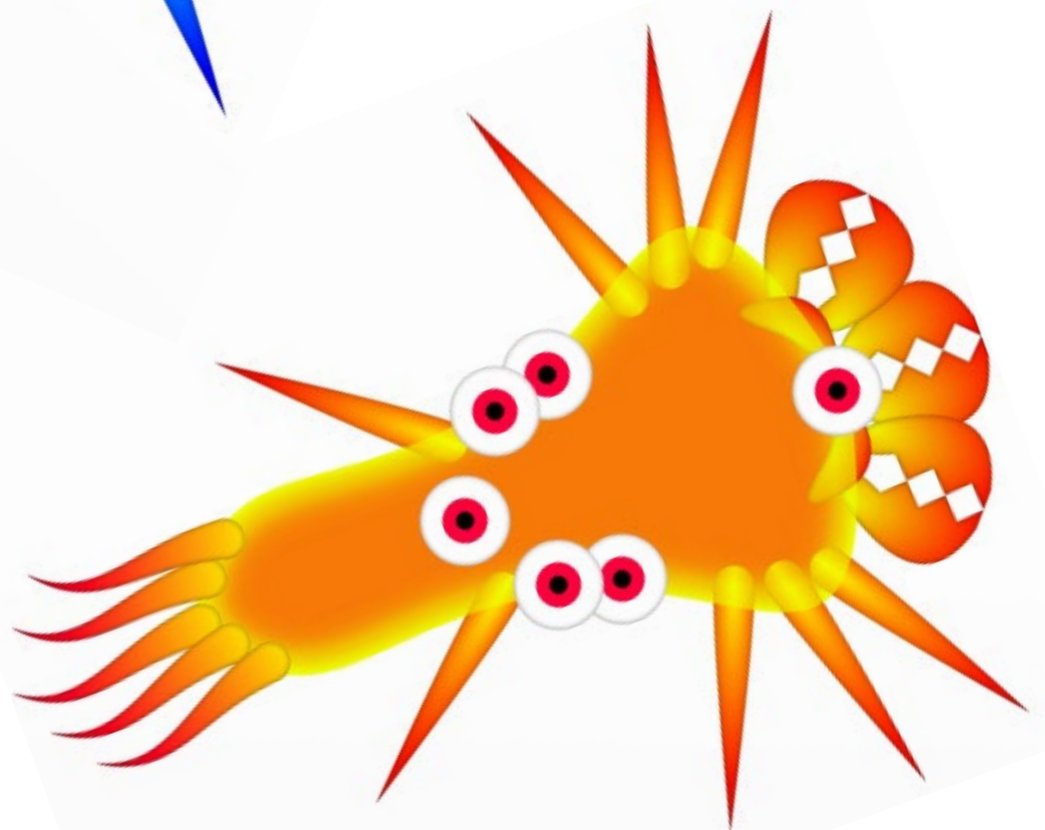
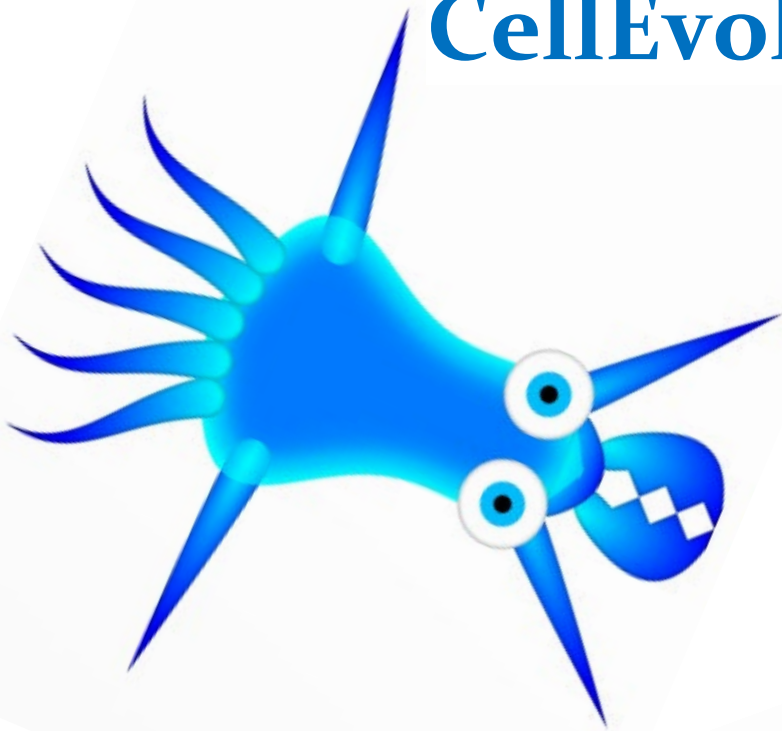


Project 2017 Bezier_Curve CellEvolution

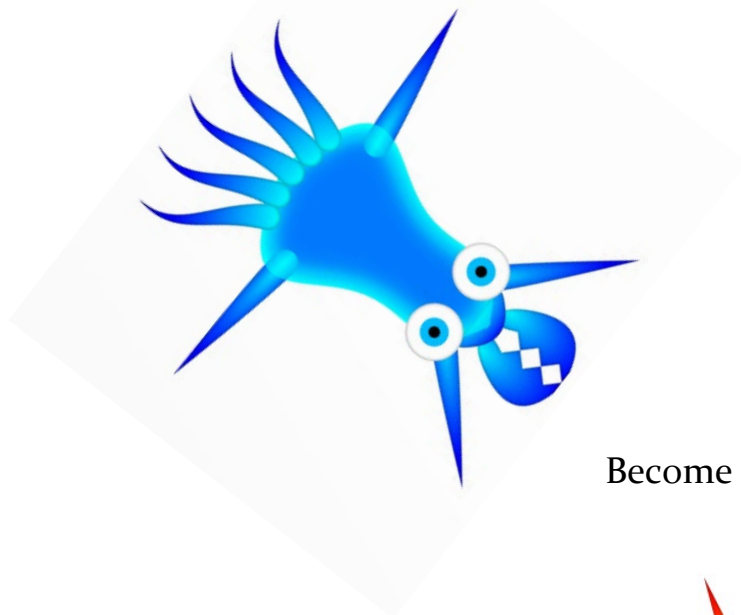


By Patipan Wongklaew 5931035321

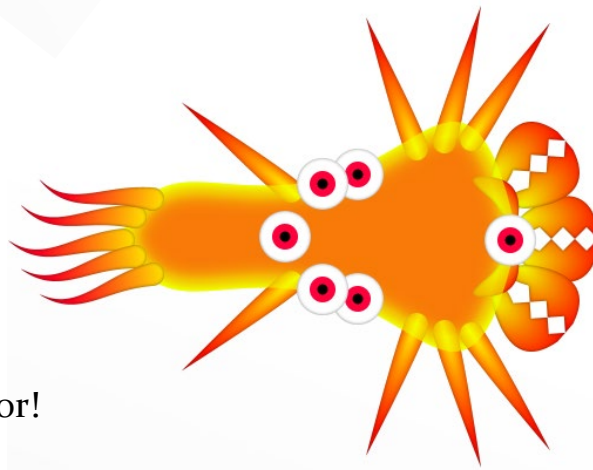
Project 2017 Bezier_Curve CellEvolution

Brief introduction about CellEvolution

Welcome to CellEvolution, our game allow you to be a simple micro-organism, your only goal is to compete against any other micro-organism and become dominant species of the world by using nature most powerful tool which is of course! EVOLUTION!



Become a cute little micro-organism



Or a ferocious predator!

REMEMBER EVOLUTION ALWAYS WINS!

User Manual

Main menu



Figure 1 main menu

This is main menu, when you die or wanted to evolve you'll have to come to this menu, here you have 3 options.

1 Start the game – this will instantly bring you to the survival of the fittest war, here you'll be test by the other microbes to find out how can your species survive the cruel world!

2 Evolve – as the name suggest, this is where you evolve, introduce new deadly cell part to your species, customize the look of your species, don't be fooled be how cute them can be!

3 Exit – glad you out game, feel free to comeback any time!

Evolve menu

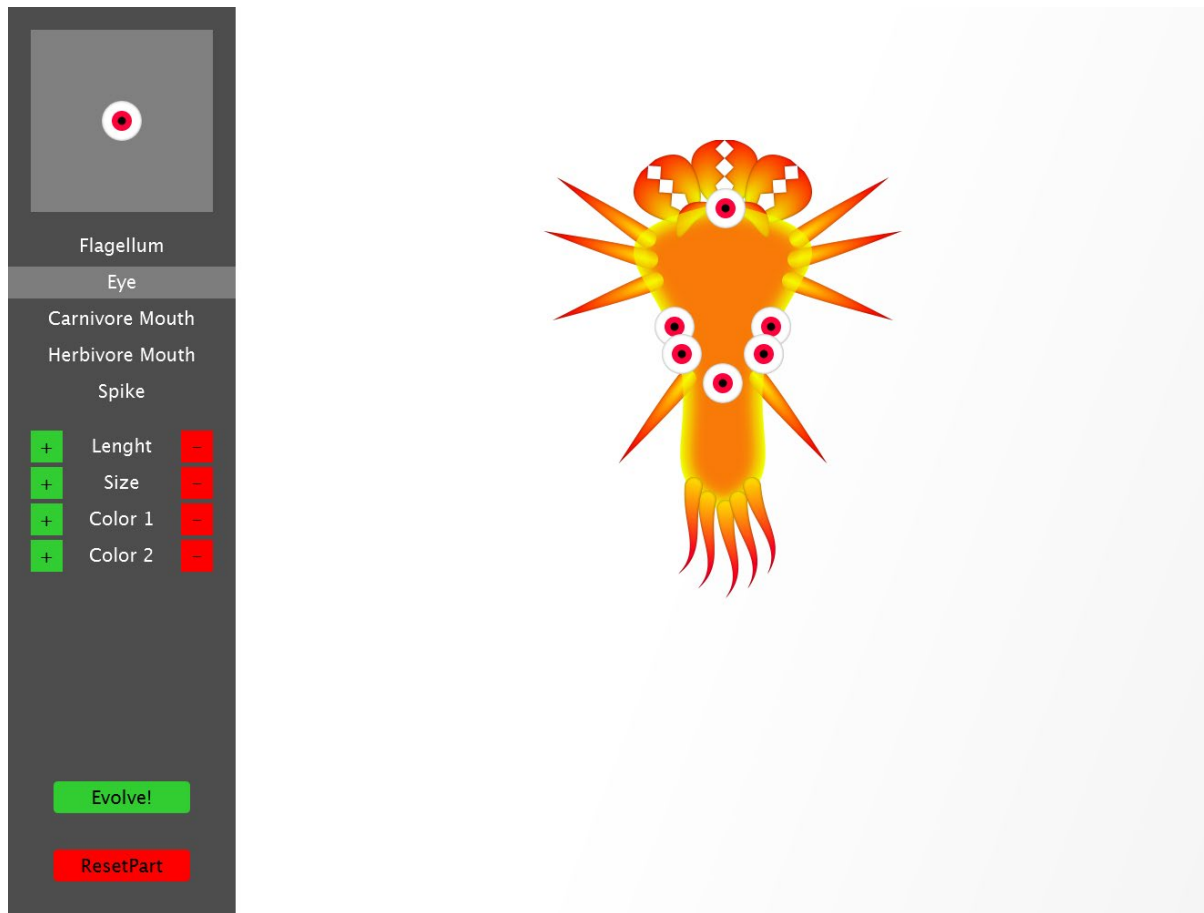


Figure 2 evolving menu

On the left is your control panel where you can add, remove, change size, change color etc.

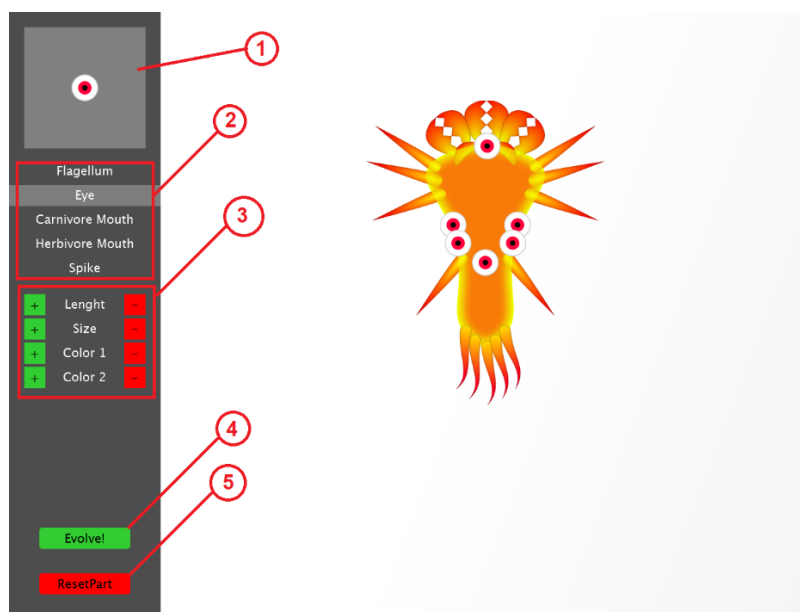


Figure 3 left panel

Detailed from figure 3:

1. This is part preview, it'll preview the part which you are selected preview the color, size and shape.
2. This is part selector, click any part's name here to select the part to preview it.
3. This is where you customize, changing both length and width of your microbe or color of your microbe here, also this can be use to change color of the selected part too.
4. When you are done and wanted to continue on your adventure click here to finish this evolution breakthrough.
5. With this button it allow you to reset all the mess you created to the selected part.

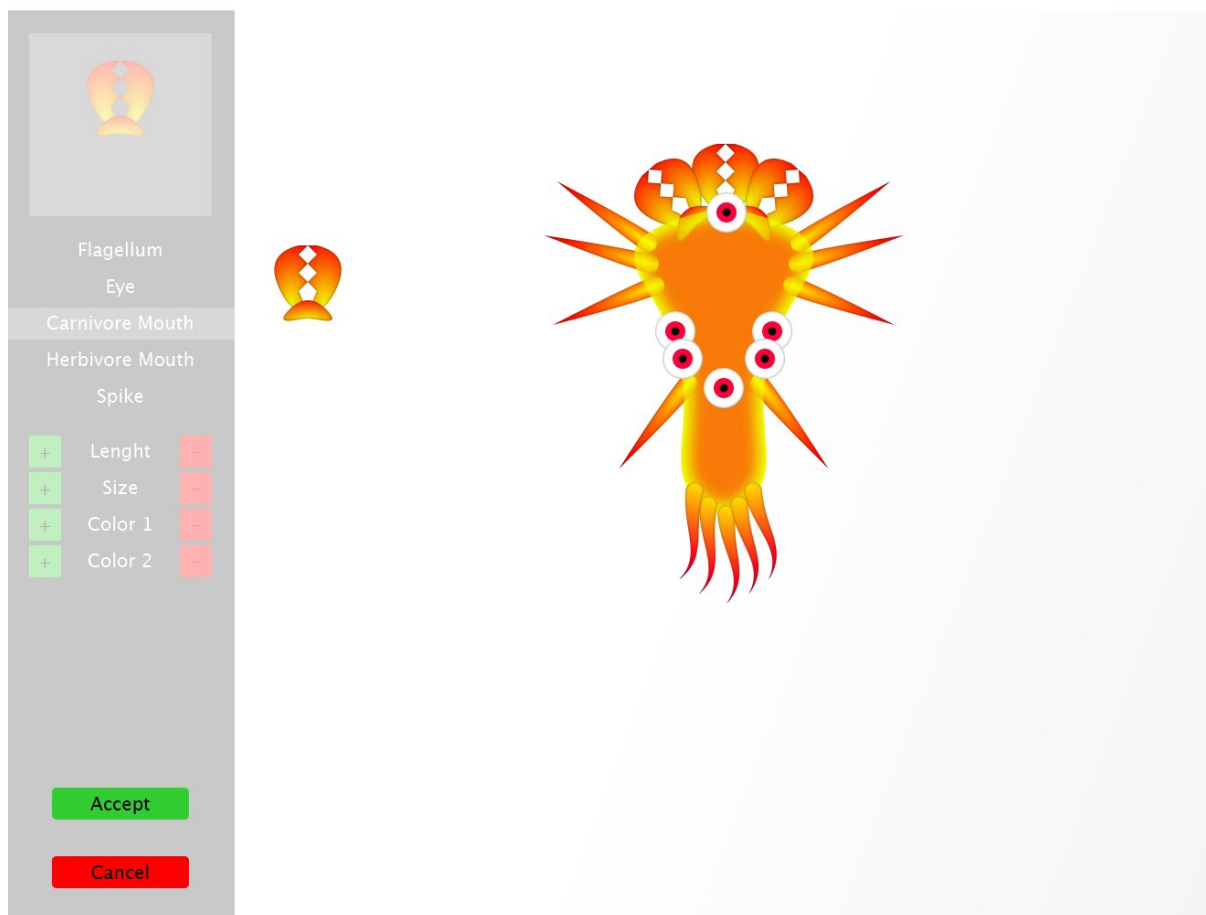


Figure 4 part set up

When hold on part previewer(number 1 in figure 3) you can drag the part to your microbe body. As shown in figure 4. You also can adjust the angle by scroll your mouse after you place part in the desire position click accept to continue or cancel ortherwise.

Game screen

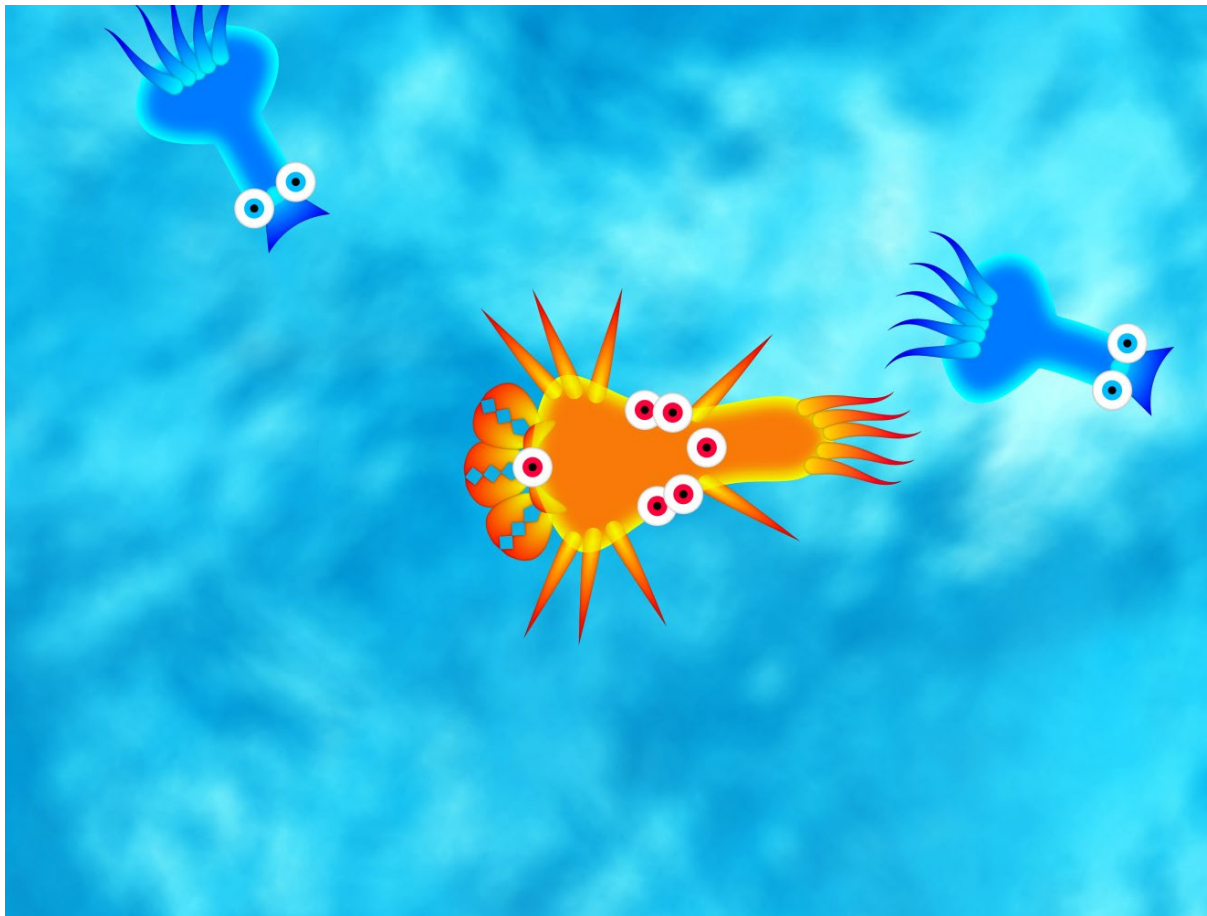


Figure 5 game screen

This is the place where you will have to struggle and compete with the other species to survive and become the most powerful creature!

Your microbe is at center of the screen. To control your microbe, you simply press arrow up, down to swim forward or backward, and you can press arrow left or right to change the direction you are heading

When you are hurt, your health bar will appear to indicate your health this is also apply to another species as well. (see figure 6 for visualization)

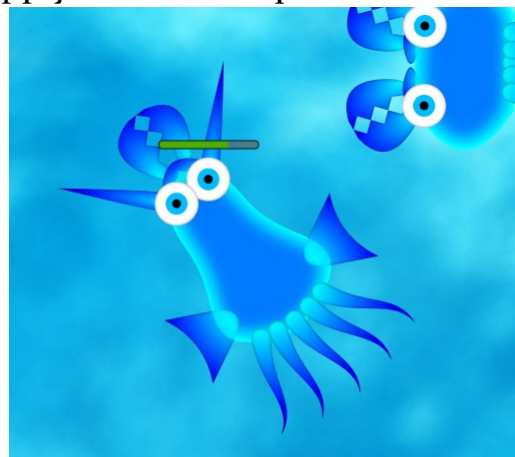


Figure 6 health bar

Implementation Detail

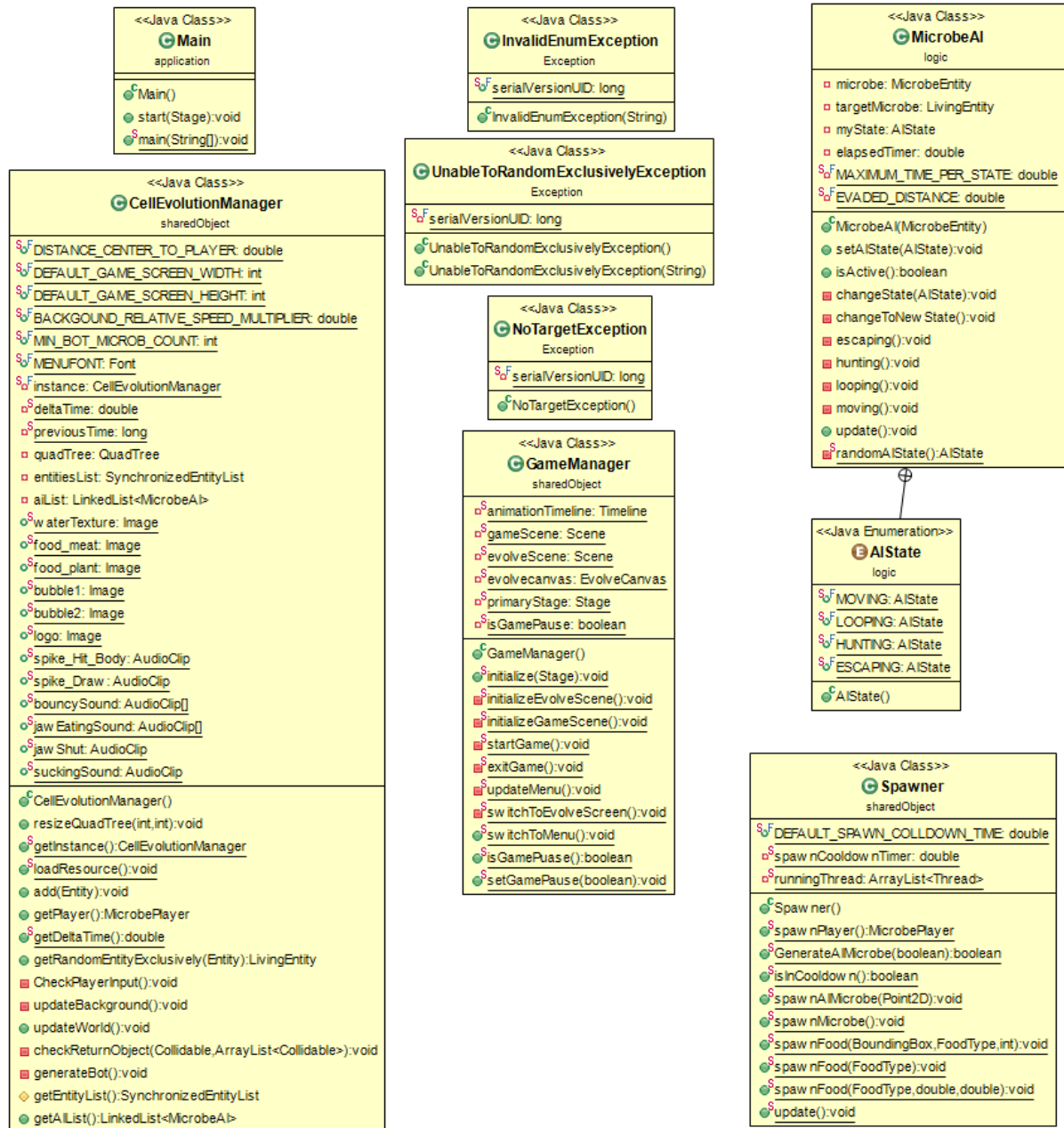


Figure 7 uml 1

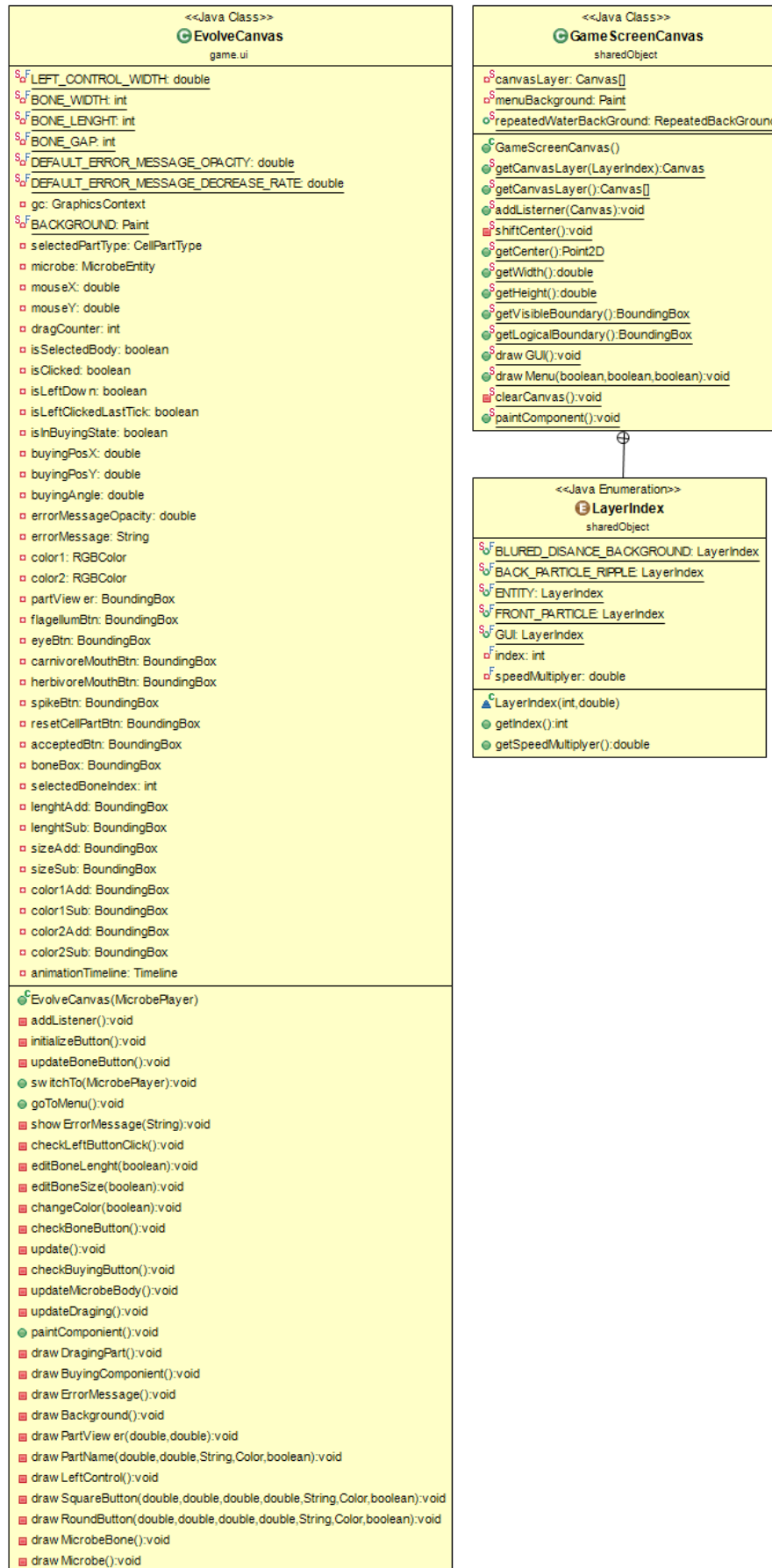


Figure 8 uml 2

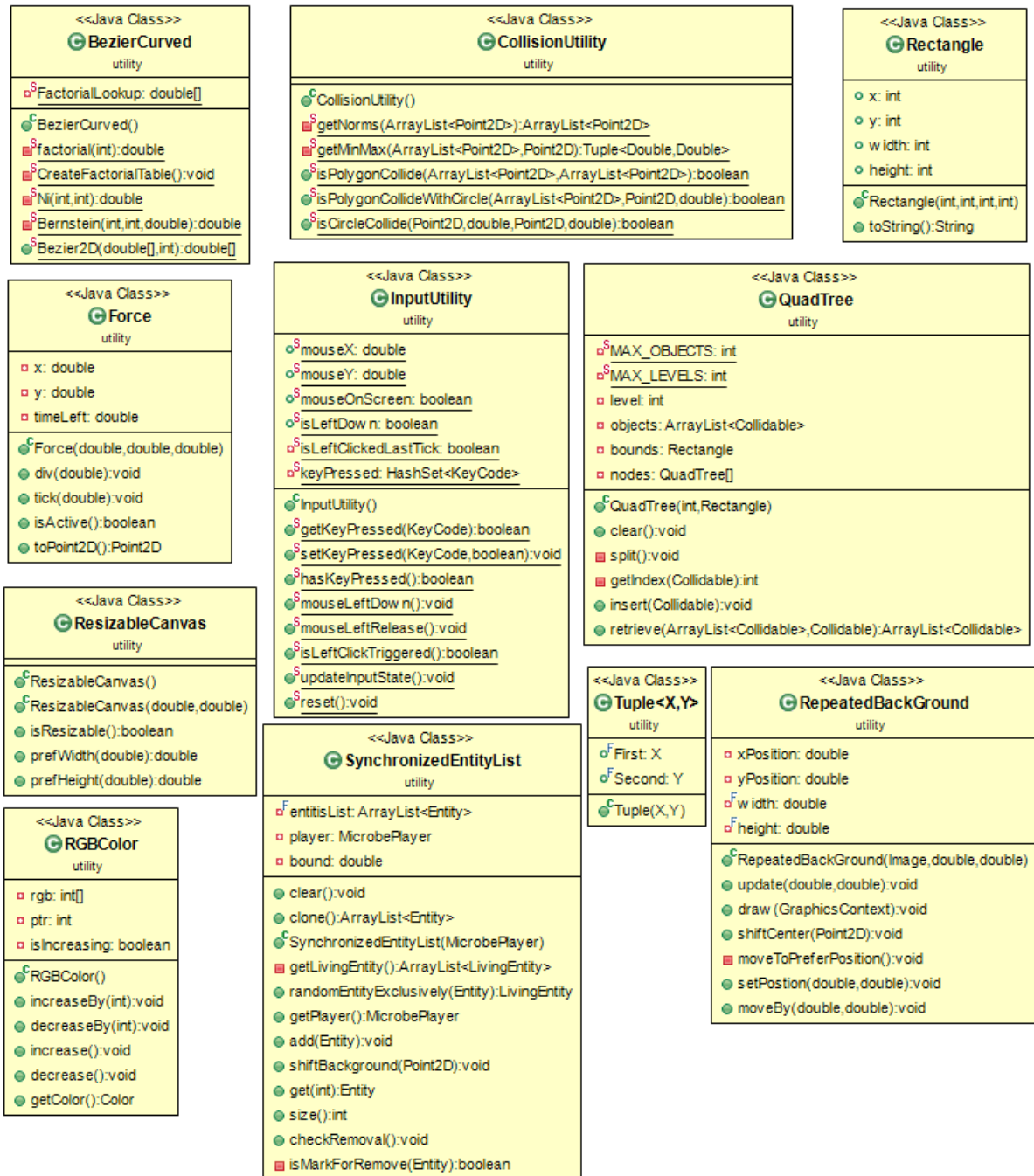


Figure 9 uml 3

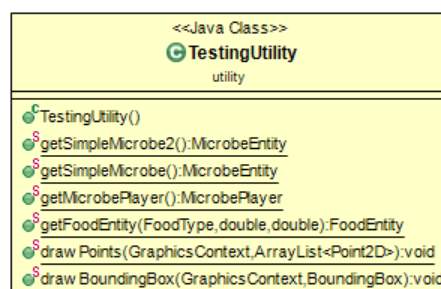


Figure 10 uml 4

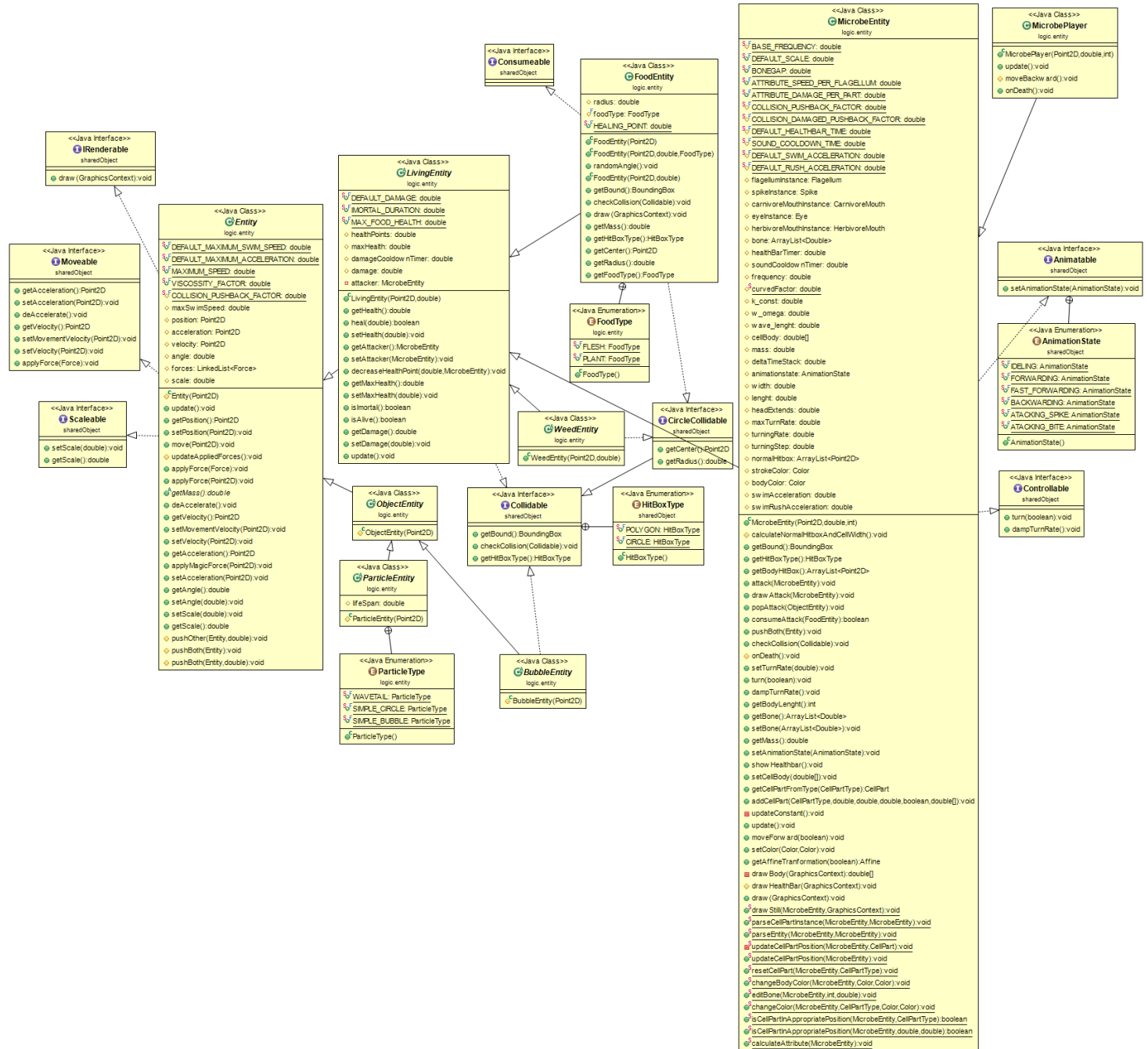


Figure 11 uml 5

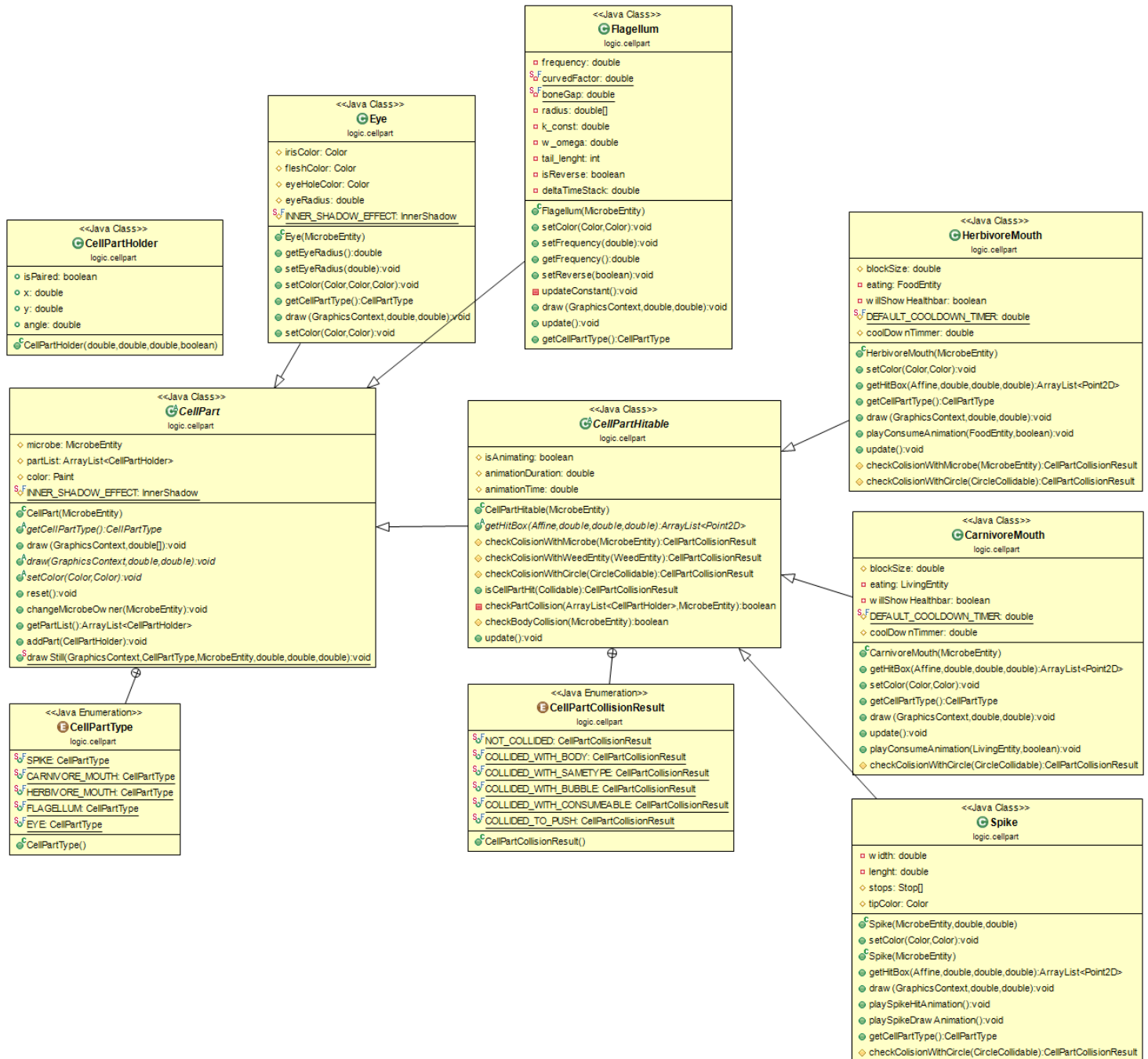


Figure 12 - uml 6

1. Package utility

1.1 Class BeizerCurved

1.1.1 Field

- <u>double[] FactorialLookup</u>	
-----------------------------------	--

1.1.2 Method

static	
- <u>double factorial(int n)</u>	
- <u>void CreateFactorialTable()</u>	
- <u>double Ni(int n, int i)</u>	
- <u>double Bernstein(int n, int i, double t)</u>	
- <u>double[] Bezier2D(double[] b, int cpts)</u>	

1.2 Class CollisionUtility

1.2.1 Method

- <u>ArrayList<Point2D></u> <u>getNorms(ArrayList<Point2D> points)</u>	
- <u>Tuple<Double, Double></u> <u>getMinMax(ArrayList<Point2D> points, Point2D axis)</u>	
+ <u>boolean isPolygonCollide(ArrayList<Point2D> poly1, ArrayList<Point2D> poly2)</u>	
+ <u>boolean isPolygonCollideWithCircle(ArrayList<Point2D> poly1, Point2D center, double radius)</u>	
+ <u>boolean isCircleCollide(Point2D center1, double radius1, Point2D center2, double radius2)</u>	

1.3 Class Force

1.3.1 Field

- double x	
- double y	
- double timeLeft	

1.3.2 Constructor

+ Force(double x, double y, double timeLeft)	
--	--

1.3.3 Method

+ void div(double mass)	
+ void tick(double tickTime)	
+ boolean isActive()	
+ Point2D toPoint2D()	

1.4 Class InputUtility

1.4.1 Field

<u>+double mouseX</u>	
<u>+double mouseY</u>	
<u>+boolean mouseOnScreen</u>	
<u>+boolean isLeftDown</u>	
<u>-boolean isLeftClickedLastTick</u>	
<u>-HashSet<KeyCode> keyPressed</u>	

1.4.2 Constructor

<u>+boolean getKeyPressed(KeyCode keycode)</u>	
<u>+void setKeyPressed(KeyCode keycode,boolean pressed)</u>	

1.4.3 Method

<u>+boolean hasKeyPressed()</u>	
<u>+void mouseLeftDown()</u>	
<u>+void mouseLeftRelease()</u>	
<u>+boolean isLeftClickTriggered()</u>	
<u>+void updateInputState()</u>	
<u>+void reset</u>	

1.5 Class QuadTree

1.5.1 Field

<u>-int MAX_OBJECTS</u>	
<u>-int MAX_LEVELS</u>	
<u>-int level</u>	
<u>-ArrayList<Collidable> objects</u>	
<u>-Rectangle bounds</u>	
<u>-QuadTree[] nodes</u>	

1.5.2 Constructor

<u>+QuadTree(int pLevel, Rectangle pBounds)</u>	
---	--

1.5.3 Method

<u>+void clear()</u>	
<u>-void split()</u>	
<u>-int getIndex(Collidable pRect)</u>	
<u>+void insert(Collidable pRect)</u>	
<u>+ArrayList<Collidable> retrieve(ArrayList<Collidable> returnObjects, Collidable pRect)</u>	

1.6 Class Rectangle

1.6.1 Field

+ int x	
+ int y	
+ int width	
+ int height	

1.6.2 Constructor

+ Rectangle(int x, int y, int width, int height)	
--	--

1.6.3 Method

+ String toString()	
---------------------	--

1.7 Class RepeatedBackGround

1.7.1 Field

- double xPosition	
- double yPosition	
- final double width	
- final double height	

1.7.2 Constructor

+ RepeatedBackGround(Image image, double x, double y)	
---	--

1.7.3 Method

+ void update	
+ void draw(GraphicsContext gc)	
+ void shiftCenter(Point2D velocity)	
- void moveToPreferPosition()	
+ void setPosition(double x, double y)	
+ void moveBy(double x, double y)	

1.8 Class ResizableCanvas

1.8.1 Constructor

+ ResizableCanvas()	
+ ResizableCanvas(double width, double height)	

1.8.2 Method

+ boolean isResizable()	
+ double prefWidth(double height)	
+ double prefHeight(double width)	

1.9 Class RGBColor

1.9.1 Field

- int[] rgb	
- int ptr	
- boolean isIncreasing	

1.9.2 Constructor

+ RGBColor()	
--------------	--

1.9.3 Method

+ void increaseBy(int number)	
+ void decreaseBy(int number)	
+ void increase()	
+ void decrease()	
+ Color getColor()	

1.10 Class SynchronizedEntityList

1.10.1 Field

- final ArrayList<Entity> entitisList	
- MicrobePlayer player	
- double bound	
+ void clear()	
+ ArrayList<Entity> clone()	

1.10.2 Constructor

+ SynchronizedEntityList(MicrobePlayer player)	
--	--

1.10.3 Method

- ArrayList<LivingEntity> getLivingEntity()	
+ LivingEntity randomEntityExclusively(Entity exclude) throws UnableToRandomExclusivelyException	
+ MicrobePlayer getPlayer()	
+ synchronized void add(Entity en)	
+ void shiftBackground(Point2D velocity)	
+ Entity get(int index)	
+ int size()	
+ synchronized void checkRemoval()	
- boolean isMarkForRemove(Entity en)	

1.11 Class Tuple

1.11.1 Field

+ final X First	
+ final Y Second	

1.11.2 Constructor

+ Tuple(X x, Y y)	
-------------------	--

2. Package sharedObject

2.1 Interface Animatable

2.1.1 Field

+enum AnimationState{IDELING, FORWARDING, FAST_FORWARDING, BACKWARDING, ATTACKING_SPIKE, ATTACKING_BITE}	
---	--

2.2.1 Method

+void setAnimationState(AnimationState state)	
---	--

2.2 Class CellEvolutionManager

2.2.1 Field

+double <u>DISTANCE_CENTER_TO_PLAYER</u>	
+int <u>DEFAULT_GAME_SCREEN_WIDTH</u>	
+int <u>DEFAULT_GAME_SCREEN_HEIGHT</u>	
+double <u>BACKGROUND_RELATIVE_SPEED_MULTIPLIER</u>	
+int <u>MIN_BOT_MICROB_COUNT</u>	
+Font <u>MENUFONT</u>	
- <u>CellEvolutionManager instance</u>	
- <u>static double deltaTime</u>	
- <u>long previousTime</u>	
- QuadTree quadtree	
- SynchronizedEntityList entitiesList	
- LinkedList<MicrobeAI> aiList	
+ <u>Image waterTexture</u>	
+ <u>Image food_meat</u>	
+ <u>Image food_plant</u>	
+ <u>Image bubble1</u>	
+ <u>Image bubble2</u>	
+ <u>Image logo</u>	
+ <u>AudioClip spike_Hit_Body</u>	
+ <u>AudioClip spike_Draw</u>	
+ <u>AudioClip[] bouncySound</u>	
+ <u>AudioClip[] jawEatingSound</u>	
+ <u>AudioClip jawShut</u>	
+ <u>AudioClip suckingSound</u>	

2.2.2 Constructor

+ CellEvolutionManager()	
--------------------------	--

2.2.3 Method

+void resizeQuadTree(int width, int heigh)	
+ CellEvolutionManager getInstance()	
+void loadResource()	

+ void add(Entity entity)	
+ MicrobePlayer getPlayer()	
+ double getDeltaTime()	
+ LivingEntity getRandomEntityExclusively(Entity exclude)	
-void CheckPlayerInput()	
- void updateBackground()	
+ void updateWorld()	
-void checkReturnObject(Collidable checkingentity, ArrayList<Collidable> returnObjects)	
-void generateBot()	
# SynchronizedEntityList getEntityList()	
+ LinkedList<MicrobeAI> getAIList()	

2.3 Interface CircleCollidable extends Collidable

+ Point2D getCenter()	
+ double getRadius()	

2.4 Interface Collidable

+ BoundingBox getBound()	
+ void checkCollision(Collidable other)	
+ HitBoxType getHitBoxType()	
+ enum HitBoxType	

2.5 Interface Controllable

+ void turn(boolean left)	
+ void dampTurnRate()	

2.6 Class GameManager

2.6.1 Field

- <u>Timeline animationTimeline</u>	
- <u>Scene gameScene</u>	
- <u>Scene evolveScene</u>	
- <u>EvolveCanvas evolvecanvas</u>	
- <u>Stage primaryStage</u>	
- <u>boolean isGamePause</u>	

2.6.2 Method

+ <u>void initialize(Stage primaryStage)</u>	
- <u>void initializeEvolveScene()</u>	
- <u>void initializeGameScene()</u>	
- <u>void startGame()</u>	
- <u>void exitGame()</u>	

- void <u>updateMenu()</u>	
- void <u>switchToEvolveScreen()</u>	
+ void <u>switchToMenu()</u>	
+ boolean <u>isGamePause()</u>	
+ void <u>setGamePause(boolean isGamePause)</u>	

2.7 Class GameScreenCanvas

2.7.1 Enum LayerIndex

2.7.1.1 Field

- int <u>index</u>	
- double <u>speedMultiplier</u>	

2.7.1.2 Constructor

LayerIndex(int index, double speedMultiplier)	
---	--

2.7.1.3 Method

+ int <u>getIndex()</u>	
+ double <u>getSpeedMultiplier()</u>	

2.7.2 Field

- Canvas[] <u>canvasLayer</u>	
- Paint <u>menuBackground</u>	

2.7.3 Method

+ Canvas <u>getCanvasLayer(LayerIndex layer)</u>	
+ Canvas[] <u>getCanvasLayer()</u>	
+ void <u>addListener(Canvas cv)</u>	
- void <u>shiftCenter()</u>	
+ Point2D <u>getCenter()</u>	
+ double <u>getWidth()</u>	
+ double <u>getHeight()</u>	
+ BoundingBox <u>getVisibleBoundary()</u>	
+ BoundingBox <u>getLogicalBoundary()</u>	
+ void <u>drawGUI()</u>	
+ void <u>drawMenu(boolean isActiveButton1, boolean isActiveButton2, boolean isActiveButton3)</u>	
- void <u>clearCanvas()</u>	
+ void <u>paintComponent()</u>	

2.8 Interface IRenderable

+ void <u>draw(GraphicsContext gc)</u>	
--	--

2.9 Interface Moveable

+ Point2D getAcceleration()	
+ void setAcceleration(Point2D acceleration)	
+ void deAccelerate()	
+ Point2D getVelocity()	
+ void setMovementVelocity(Point2D velocity)	
+ void setVelocity(Point2D velocity)	
+ void applyForce(Force force)	

2.10 Interface Scaleable

+ void setScale(double scale)	
+ double getScale()	

2.11 Class Spawner

2.11.1 Field

+ double <u>DEFAULT SPAWN COLLDOWN TIME</u>	
- double spawnCooldownTimer	
- LinkedList<Entity> spawningQueue	
- ArrayList<Thread> runningThread	

2.11.2 Method

+ MicrobePlayer spawnPlayer()	
+ boolean GenerateAIMicrobe(boolean <u>isBypassCoolDown</u>)	
+ boolean isInCooldown()	
+ synchronized void spawnAIMicrobe(Point2D <u>position</u>)	
+ void spawnFood(BoundingBox bound, <u>FoodType type, int amount</u>)	
+ void spawnFood(FoodType type	
+ static void spawnFood(FoodType type, double x, <u>double y</u>)	
+ void update()	

3 Package logic

3.1 Class MicrobeAI

3.1.1 Field

- MicrobeEntity microbe	
- LivingEntity targetMicrobe	
- AIState myState	
- double elapsedTimer	
- double MAXIMUM_TIME_PER_STATE	
- double EVADED_DISTANCE	

3.1.2 Constructor

+ MicrobeAI(MicrobeEntity microbe)	
------------------------------------	--

3.1.3 Method

+ void setAIState(AIState state)	
+ boolean isActive()	
- void changeState(AIState state)	
- void changeToNewState()	
- void escaping() throws NoTargetException	
- void hunting() throws NoTargetException	
- void looping()	
- void moving()	
+ void update()	
- AIState randomAIState()	

3.1.4 Enum AIState

+ enum AIState{MOVING, LOOPING, HUNTING, ESCAPING; }	
--	--

4 Package logic.entity

4.1 Abstract Class Entity implements IRenderable,Moveable,Scaleable

4.1.1 Field

+ double <u>DEFAULT_MAXIMUM_SWIM_SPEED</u>	
+ double <u>DEFAULT_MAXIMUM_ACCELERATION</u>	
+ double <u>MAXIMUM_SPEED</u>	
+ double <u>VISCOSSITY_FACTOR</u>	
# double <u>COLLISION_PUSHBACK_FACTOR</u>	
# double maxSwimSpeed	
# Point2D position	
# Point2D acceleration	
# Point2D velocity	
# double angle	
# LinkedList<Force> forces	
# double scale	

4.1.2 Constructor

# Entity(Point2D position)	
----------------------------	--

4.1.3 Method

+ void update()	
+ Point2D getPosition()	
+ void setPosition(Point2D position)	
+ void move(Point2D velocity)	
# void updateAppliedForces()	
+ void applyForce(Point2D force)	
+ double getMass()	
+ void deAccelerate()	
+ Point2D getVelocity()	
+ void setMovementVelocity(Point2D velocity)	
+ void setVelocity(Point2D velocity)	
+ Point2D getAcceleration()	
+ void applyMagicForce(Point2D force)	
+ void setAcceleration(Point2D acceleration)	
+ double getAngle()	
+ void setAngle(double angle)	
+ void setScale(double scale)	
+ double getScale()	
# void pushOther(Entity other, double Factor)	
# void pushBoth(Entity other)	
# void pushBoth(Entity other, double Factor)	

4.2 Class FoodEntity extends LivingEntity implements Consumeable, CircleCollidable

4.2.1 Field

# double radius	
# FoodType <u>foodType</u>	
+ double <u>HEALING_POINT</u>	

4.2.2 Constructor

+ FoodEntity(Point2D position, double maxHealth, FoodType type)	
+ FoodEntity(Point2D position, double maxHealth)	

4.2.3 Method

+ void randomAngle()	
+ BoundingBox getBound()	
+ void checkCollision(Collidable other)	
+ void draw(GraphicsContext gc)	
+ double getMass()	
+ HitBoxType getHitBoxType()	
+ Point2D getCenter()	
+ double getRadius()	
+ FoodType getFoodType()	
+ enum FoodType	

4.3 Abstract Class LivingEntity extends Entity implements Collidable

4.3.1 Field

+ double <u>DEFAULT_DAMAGE</u>	
+ double <u>IMORTAL_DURATION</u>	
+ double <u>MAX_FOOD_HEALTH</u>	
# double healthPoints	
# double maxHealth	
# double damageCooldownTimer	
# double damage	
-MicrobeEntity attacker	

4.3.2 Constructor

+ LivingEntity(Point2D position, double maxHealth)	
--	--

4.3.3 Method

+ double getHealth()	
+ boolean heal(double amount)	
+ void setHealth(double health)	
+ MicrobeEntity getAttacker()	
+ void setAttacker(MicrobeEntity attacker)	

+ void decreaseHealthPoint(double dmg, MicrobeEntity attacker)	
+ double getMaxHealth()	
+ void setMaxHealth(double maxHealth)	
+ boolean isImmortal()	
+ boolean isAlive()	
+ double getDamage()	
+ void setDamage(double damage)	
+ void update()	

4.4 Class MicrobeEntity extends LivingEntity implements Animatable, Controllable

4.4.1 Field

# double <u>BASE_FREQUENCY</u>	
+ double <u>DEFAULT_SCALE</u>	
+ double BONEGAP	
+ double <u>ATTRIBUTE_SPEED_PER_FLAGELLUM</u>	
+ double <u>ATTRIBUTE_DAMAGE_PER_PART</u>	
# double <u>COLLISION_PUSHBACK_FACTOR</u>	
# double <u>COLLISION_DAMAGED_PUSHBACK_FACTOR</u>	
# double <u>DEFAULT_HEALTHBAR_TIME</u>	
# double <u>SOUND_COOLDOWN_TIME</u>	
# double <u>DEFAULT_SWIM_ACCELERATION</u>	
# double <u>DEFAULT_RUSH_ACCELERATION</u>	
# Flagellum flagellumInstance	
# Spike spikeInstancer	
# CarnivoreMouth carnivoreMouthInstance	
# Eye eyeInstance	
# HerbivoreMouth herbivoreMouthInstance	
# ArrayList<Double> bone	
# double healthBarTimer	
# double soundCooldownTimer	
# double frequency	
# double curvedFactor	
# double k_const	
# double w_omega	
# double wave_lenght	
# double[] cellBody	
# double mass	
# double deltaTimeStack	
# AnimationState animationstate	
# double width	
# double length	
# double headExtends	
# double maxTurnRate	
# double turningRate	
# double turningStep	
# ArrayList<Point2D> normalHitbox	
# Color strokeColor	

# Color bodycolor	
# double swimAcceleration	
# double swimRushAcceleration	

4.4.2 Constructor

+ MicrobeEntity(Point2D position, double maxHealth, int boneLenght)	
---	--

4.4.3 Method

# void calculateNormalHitboxAndCellWidth()	
+ BoundingBox getBound()	
+ HitBoxType getHitBoxType()	
+ ArrayList<Point2D> getBodyHitBox()	
+ void attack(MicrobeEntity other)	
+ void drawAttack(MicrobeEntity other)	
+ void popAttack(ObjectEntity bubble)	
+ boolean consumeAttack(FoodEntity other)	
+ void pushBoth(Entity other)	
+ void checkCollision(Collidable other)	
# void onDeath()	
+ void setTurnRate(double rate)	
+ void turn(boolean isLeft)	
+ void dampTurnRate()	
+ int getBodyLenght()	
+ ArrayList<Double> getBone()	
+ void setBone(ArrayList<Double> bone)	
+ double getMass()	
+ void setAnimationState(AnimationState newState)	
+ void showHealthbar()	
+ void setCellBody(double[] cellBody)	
+ CellPart getCellPartFromType(CellPartType partType)	
+ void addCellPart(CellPartType partType, double x, double y, double angle, boolean isPaired, double... args)	
-void updateConstant()	
+ void update()	
+ void moveForward(boolean isFast)	
+ void setColor(Color strokeColor, Color bodyBaseColor)	
+ Affine getAffineTranformation(boolean isReverse)	
- double[] drawBody(GraphicsContext gc)	
# void drawHealthBar(GraphicsContext gc)	
+ void draw(GraphicsContext gc)	
+ void drawStill(MicrobeEntity entity, GraphicsContext gc)	
+ void parseCellPartInstance(MicrobeEntity lhs, MicrobeEntity rhs)	

<u>+void parseEntity(MicrobeEntity lhs, MicrobeEntity rhs)</u>	
<u>-void updateCellPartPosition(MicrobeEntity microbe, CellPart instance)</u>	
<u>+ void updateCellPartPosition(MicrobeEntity microbe)</u>	
<u>+void resetCellPart(MicrobeEntity microbe, CellPartType partType)</u>	
<u>+void changeBodyColor(MicrobeEntity microbe, Color color1, Color color2)</u>	
<u>+void editBone(MicrobeEntity microbe, int boneIndex, double changeAmount)</u>	
<u>+void changeColor(MicrobeEntity microbe, CellPartType partType, Color color1, Color color2)</u>	
<u>+ boolean isCellPartInAppropriatePosition(MicrobeEntity microbe, CellPartType partType)</u>	
<u>+ boolean isCellPartInAppropriatePosition(MicrobeEntity microbe, double x, double y)</u>	
<u>+void calculateAttribute(MicrobeEntity entity)</u>	

4.5 Class MicrobePlayer extends MicrobeEntity

4.5.1 Constructor

<u>+ MicrobePlayer(Point2D position, double healthPoints, int boneLenght)</u>	
---	--

4.5.2 Method

<u>+ void update()</u>	
<u># void moveBackward()</u>	
<u>+ void onDeath()</u>	

5 Package logic.cellpart

5.1 Class CarnivoreMouth extends CellPartHitable

5.1.1 Field

# double blockSize	
- LivingEntity eating	
- boolean willShowHealthbar	
# double DEFAULT_COOLDOWN_TIMER	
# double coolDownTimmer	

5.1.2 Constructor

+ CarnivoreMouth(MicrobeEntity microbe)	
---	--

5.1.3 Method

+ ArrayList<Point2D> getHitBox(Affine baseTransformation, double angle, double x, double y)	
+ void setColor(Color baseColor, Color innerGlowColor)	
+ CellPartType getCellPartType()	
+ void draw(GraphicsContext gc, double posX, double posY)	
+ void update()	
+ void playConsumeAnimation(LivingEntity food, boolean willShowHealthbar)	
# CellPartCollisionResult checkColisionWithCircle(CircleCollidable other)	

5.2 Class CellPart

5.2.1 Field

# MicrobeEntity microbe	
# ArrayList<CellPartHolder> partList	
# Paint color	
# InnerShadow <u>INNER_SHADOW_EFFECT</u>	

5.2.2 Constructor

+ CellPart(MicrobeEntity microbe)	
-----------------------------------	--

5.2.3 Method

+ CellPartType getCellPartType()	
+ void draw(GraphicsContext gc, double[] xShift)	
+ void draw(GraphicsContext gc, double posX, double posY)	
+ void setColor(Color color1, Color color2)	
+ void reset()	
+ void changeMicrobeOwner(MicrobeEntity microbe)	
+ ArrayList<CellPartHolder> getPartList()	

+ void addPart(CellPartHolder part)	
+ enum CellPartType	
+ void drawStill(GraphicsContext gc, CellPartType partType, MicrobeEntity microbe, double x, double y, double angle)	

5.3 Class CellPartHitable extends CellPart

5.3.1 Field

# boolean isAnimating = false	
# double animationDuration	
# double animationTime	

5.3.2 Constructor

+ CellPartHitable(MicrobeEntity microbe)	
--	--

5.3.3 Method

# CellPartCollisionResult checkColisionWithMicrobe(MicrobeEntity other)	
# CellPartCollisionResult checkColisionWithWeedEntity(WeedEntity other)	
# protected CellPartCollisionResult checkColisionWithCircle(CircleCollidable other)	
+ CellPartCollisionResult isCellPartHit(Collidable other)	
- boolean checkPartCollision(ArrayList<CellPartHolder> otherPartList, MicrobeEntity other)	
# boolean checkBodyCollision(MicrobeEntity other)	
+ void update()	
+ enum CellPartCollisionResult	

5.4 Class CellPartHolder

5.4.1 Field

+ boolean isPaired	
+ double x	
+ double y	
+ double angle	

5.4.2 Constructor

+ CellPartHolder(double x, double y, double angle, boolean isPaired)	
--	--

5.5 Class Eye extends CellPart

5.5.1 Field

# Color irisColor	
# Color fleshColor	
# Color eyeHoleColor	
# double eyeRadius	
# InnerShadow <u>INNER_SHADOW_EFFECT</u>	

5.5.2 Constructor

+ Eye(MicrobeEntity microbe)	
------------------------------	--

5.5.3 Method

+ double getEyeRadius()	
+ void setEyeRadius(double eyeRadius)	
+ void setColor(Color irisColor, Color fleshColor, Color eyeHoleColor)	
+ CellPartType getCellPartType()	
+ void draw(GraphicsContext gc, double posX, double posY)	
+ public void setColor(Color color1, Color color2)	

5.6 Class Flagellum extends CellPart

5.6.1 Field

-double frequency	
-double <u>curvedFactor</u>	
-double <u>boneGap</u>	
-double[] radius	
-double k_const	
-double w_omega	
-int tail_lenght	
-boolean isReverse	
-double deltaTimeStack	

5.6.2 Constructor

+ Flagellum(MicrobeEntity microbe)	
------------------------------------	--

5.6.3 Method

+ void setColor(Color baseColor, Color glowColor)	
+ void setFrequency(double frequency)	
+ double getFrequency()	
+ void setReverse(boolean isReverse)	
-void updateConstant()	
+ void draw(GraphicsContext gc, double posX, double posY)	
+ void update()	
+ CellPartType getCellPartType()	

5.7 Class HerbivoreMouth extends CellPartHitable

5.7.1 Field

# double blockSize	
-FoodEntity eating	
-boolean willShowHealthbar	
# double <u>DEFAULT_COOLDOWN_TIMER</u>	
# double coolDownTimmer	

5.7.2 Constructor

+ HerbivoreMouth(MicrobeEntity microbe)	
---	--

5.7.3 Method

+ void setColor(Color baseColor, Color innerGlowColor)	
+ ArrayList<Point2D> getHitBox(Affine baseTransformation, double angle, double x, double y)	
+ CellPartType getCellPartType()	
+ void draw(GraphicsContext gc, double posX, double posY)	
+ void playConsumeAnimation(FoodEntity food, boolean willShowHealthbar)	
+ void update()	
# CellPartCollisionResult checkColisionWithMicrobe(MicrobeEntity other)	
# CellPartCollisionResult checkColisionWithCircle(CircleCollidable other)	

5.8 Class Spike extends CellPartHitable

5.8.1 Field

- double width	
- double length	
# Stop[] stops	
# Color tipColor	

5.8.2 Constructor

+ Spike(MicrobeEntity microbe, double width, double length)	
+ Spike(MicrobeEntity microbe)	

5.8.3 Method

+ void setColor(Color dipColor, Color tipColor)	
+ ArrayList<Point2D> getHitBox(Affine baseTransformation, double angle, double x, double y)	
+ void draw(GraphicsContext gc, double x, double y)	
+ void playSpikeHitAnimation()	

+ void playSpikeDrawAnimation()	
+ CellPartType getCellPartType()	
# CellPartCollisionResult checkCollisionWithCircle(CircleCollidable other)	

6. Package game.ui

6.1 Class EvolveCanvas extends Canvas

6.1.1 Field

- double LEFT_CONTROL_WIDTH	
- int BONE_WIDTH	
- int BONE_LENGTH	
- int BONE_GAP	
- double DEFAULT_ERROR_MESSAGE_OPACITY	
- double DEFAULT_ERROR_MESSAGE_DECREASE_RATE	
- Paint BACKGROUND	
- GraphicsContext gc	
- CellPartType selectedPartType	
- MicrobeEntity microbe	
- double mouseX	
- double mouseY	
- int dragCounter	
- boolean isSelectedBody	
- boolean isClicked	
- boolean isLeftDown	
- boolean isLeftClickedLastTick	
- boolean isInBuyingState	
- double buyingPosX	
- double buyingPosY	
- double buyingAngle	
- double errorMessageOpacity	
- String errorMessage	
- RGBColor color1	
- RGBColor color2	
- BoundingBox partViewer	
- BoundingBox flagellumBtn	
- BoundingBox eyeBtn	
- BoundingBox carnivoreMouthBtn	
- BoundingBox herbivoreMouthBtn	
- BoundingBox spikeBtn	
- BoundingBox resetCellPartBtn	
- BoundingBox acceptedBtn	
- BoundingBox boneBox	
- int selectedBoneIndex	
- BoundingBox lengthAdd	
- BoundingBox lengthSub	
- BoundingBox sizeAdd	
- BoundingBox sizeSub	
- BoundingBox color1Add	

- BoundingBox color1Sub	
- BoundingBox color2Add	
- BoundingBox color2Sub	
- Timeline animationTimeline	

6.1.2 Constructor

+ EvolveCanvas(MicrobePlayer player)	
--------------------------------------	--

6.2.2 Method

- void addListener()	
- void initializeButton()	
- void updateBoneButton()	
+ void switchTo(MicrobePlayer player)	
+ void goToMenu()	
- void showErrorMessage(String message)	
- void checkLeftButtonClick()	
- void editBoneLenght(boolean isIncrease)	
- void editBoneSize(boolean isIncrease)	
- void changeColor(boolean isBase)	
- void checkBoneButton()	
- void update()	
- void checkBuyingButton()	
- void updateMicrobeBody()	
- void updateDraging()	
- void paintComponient()	
- void drawDragingPart()	
- void drawBuyingComponient()	
- void drawErrorMessage()	
- void drawBackground()	
- void drawPartViewer(double x, double y)	
- void drawPartName(double x, double y, String text, Color color, boolean isHilight)	
- void drawLeftControl()	
- void drawSquareButton(double x, double y, double width, double height, String text, Color color, boolean isClicked)	
- void drawRoundButton(double x, double y, double width, double height, String text, Color color, boolean isClicked)	
- void drawMicrobeBone()	
- void drawMicrobe()	

7. Package exception

7.1 Class InvalidEnumException extends java.lang.RuntimeException

7.1.1 Field

+ long serialVersionUID	
-------------------------	--

7.1.2 Constructor

+ InvalidEnumException(String argo)	
-------------------------------------	--

7.2 Class NoTargetException extends Exception

7.2.1 Field

+ long serialVersionUID	
-------------------------	--

7.3 Class UnableToRandomExclusivelyException extends Exception

7.3.1 Field

+ long serialVersionUID	
-------------------------	--

7.3.2 Constructor

+ InvalidEnumException(String argo)	
+ UnableToRandomExclusivelyException()	

8. Package application

8.1 Class Main extends Application

8.1.1 Methode

+ void start(Stage primaryStage)	
+ void main(String[] args)	