

FroggerModel
- <u>CAR_START_Y</u> : int - <u>INDEX_INCREMENT</u> : int - <u>MAX_LEVELS</u> : int - <u>OFFSCREEN_LEFT</u> : int - <u>OFFSCREEN_RIGHT</u> : int - <u>PANE_WIDTH</u> : int - <u>RIVER_START_Y</u> : int - currentLevel : int - gameMode : int
+ FroggerModel() + generateCarPaths() : Road[] + generateLilyPads() : LilyPad[] + generateWaterObjectPaths() : River[] + levelUp() : boolean + resetLevels() + setGameMode( mode : int )

Frog
- <u>STARTING_X_POS</u> : int - <u>STARTING_Y_POS</u> : int - <u>fileName</u> : String - <u>height</u> : String - <u>width</u> : double - isOnWaterObject : boolean
+ Frog() + <u>getHeight()</u> : double + <u>getWidth()</u> : double + getIsOnWaterObject() : boolean + restartFrog() + setXTranslation( Xtranslate : double ) + setIsOnWaterObjectFalse() + setIsOnWaterObjectTrue()

MovingObject
- <u>HEIGHT</u> : double - gameMode : int - pathTransition : PathTransition - thePath : Path
+ MovingObject( fileName : String, startX : int, startY : int, endX : int, gameMode : int) - createPath( startX : int, endX : int) - createPathTransition( startX : int, endX : int) + <u>getThePath()</u> : Path + moveFroggerObject() + setWidth( width : int )

FroggerMain
- mediaPlayer : MediaPlayer - theController : FroggerController - theMainMenu : FroggerMainMenu - theModel : FroggerModel - theView : FroggerView
+ FroggerMain() + handle ( event : KeyEvent ) + init () + <u>main ( args : String [ ] )</u> + start ( primaryStage : Stage )

FroggerMainMenu
- LOADTIME : Integer - beginner : RadioButton - exit : Button - expert : RadioButton - levelGroup : ToggleGroup - root : StackPane - start : Button

FroggerController
- <u>MAX_NUM_SCORES</u> : int - <u>SCORE_INCREMENT</u> : int - <u>STEP_SIZE_LR</u> : double - <u>STEP_SIZE_UD</u> : double - carCollisionsTask : CarCollisionsTask - controlledByKeys : boolean - frogIndex : int - gameOver : boolean - highScores : HighScores - lilyPadCollisionsTask : LilyPadCollisionsTask - <u>maxFrogIndex</u> : int - maxScore : int - <u>minFrogIndex</u> : int - numLives : int - ridingWaterObjectTask : RideTheWaterObjectTask - score : int - sleepTimeFactor : int - theMainMenu : FroggerMainMenu - theModel : FroggerModel - theMoveCarTask : MoveCarsTask - theMoveWaterObjectsTask : MoveWaterObjectsTask - theView : FroggerView - waterObjectCollisionsTask : WaterObjectCollisionTask
+ FroggerController( theView : FroggerView, theModel : FroggerModel, theMainMenu : FroggerMainMenu ) - adjustScore( i : int ) + checkBottomBound() : boolean + checkCarCollisions() + checkLeftBound() : boolean + checkLilyCollisions() + checkRightBound() : boolean + checkTopBound() : boolean + checkWaterObjectCollision() + endGame() + getControlledByKeys() : boolean + getMaxScore() : int + getNumLives() : int + isGameOver() : boolean + removeLife() + restartFrogIndex() + setControlledByKeysFalse() + setGameOver( gameover : boolean ) + startTheCars( delay : int ) + startTheWaterObjects ( delay : int ) + stopRidingWaterObjectTask() + updateFrogDownPosition() + updateFrogLeftPosition() + updateFrogRightPosition() + updateFrogUpPosition()

CarCollisionsTask :: Task <Integer>
- cars : MovingObject[ ] - stopTask : boolean
+ CarCollisionsTask( cars : MovingObject [ ] ) + call() : Integer + stopTask()

RideTheWaterObjectTask :: Task <Integer>
- theObject : MovingObject - isPaused : boolean
+ RideTheWaterObjectTask( theObj : MovingObject ) + call() : Integer + stopTask() + resumeTask() + pauseTask()

HighScores
------------

FroggerView
- <u>NUM_LIVES</u> : int - carGroup : Group - lilyPadGroup : Group - root : Pane - safeFrogs : ArrayList<Frog> - score : SimpleIntegerProperty - theFrog : Frog - theLilyPads : LilyPad[] - theLives : ArrayList<ImageView> - theModel : FroggerModel - theRivers : River[] - theRoads : Road[] - waterGroup : Group
+ FroggerView(theModel : FroggerModel) + addFrog() - addLilyPads() - addLives() + addPaths() + addScore() - clearFrogs() + endGame( finalScoer : int, scores : ArrayList<Integer>, froggerController : FroggerController) + <u>getNum_LIVES()</u> : int + getRootNode() : Pane + getRootXMax() : int + getRootXMin() : int + getRootYMax() : int + getRootYMin() : int + getTheFrog() : Frog + getTheLilyPads() : LilyPad[] + getTheRivers() : River[] + getTheRoads() : Road[] + launchNewFrog() + levelUp() + removeNextLife() +restartGame( froggerController : FroggerController) + updateScores( score: int )

LilyPadCollisionsTask :: Task <Integer>
- lilyPads : LilyPad [ ] - onPad : boolean
+ LilyPadCollisionsTask( lilypads : LilyPad [ ] ) + call() : Integer

WaterObjectsCollisionsTask :: Task <Integer>
- waterPath : River - stopTask : boolean
+ WaterObjectsCollisionsTask( path : River ) + call() : Integer + stopTask()

MoveCarsTask :: Task <Integer>
- cars : MovingObject[ ] - baseDelay : int
+ MoveCarsTask( cars : MovingObject [ ], delay : int) + call() : Integer

MoveWaterObjectsTask :: Task <Integer>
- waterObjects : MovingObject[ ] - baseDelay : int
+ MoveWaterObjectsTask( waterObjects : MovingObject [ ], delay : int) + call() : Integer

