buildingmanager

Accomodation
+Accomodation()

LectureHall
+LectureHall()

Library
+Library()

Course
+Course()

FoodZone
+FoodZone()

Recreational
+Recreational()

Nature
+Nature()

<<enumeration>> BuildingTypes
Accomodation
LectureHall
Library
Course
FoodZone
Recreational
Nature

<<Abstract>> Building
-placedTexure : Texture {readOnly}
-collisionTexture : Texture {readOnly}
-draggingTexture : Texture {readOnly}
-isSelected : boolean
-isPlaced : boolean
-width : int {readOnly}
-height : int {readOnly}
-type : BuildingTypes

+Building(placedTexture : Texture, collisionTexture : Texture, draggingTexture : Texture, width : int, height : int, type : BuildingType)
-updatePosition(mousePos : Vector2)
+handleDragging(mousePos : Vector2, colliding : boolean)
+selectBuilding()
+placeBuilding()
-incrimentCount()
-setDraggingTexture(collision : boolean)
+clampPosition()
+getDistanceFrom(building : Building) : int
+ getCentre() : Vector2
+getCol() : int
+getRow() : int
+getIsSelected() : boolean
+getIsPlaced() : boolean
+getBuildingWidth() : int
+getBuildingHeight() : int
+getBuildingType() : BuildingTypes
+dispose()

BuildingManager
-buildings : Array<Building>
-currentBuilding : Building
-currentlySelecting : boolean

+BuildingManager()
+addBuilding(building : Building)
+input(mousePos : Vector2, clicked : boolean, backspacePressed : boolean)
-removeBuilding()
-handlePlacing()
+handleSelection(type : BuildingTypes)
-handleType(type : BuildingTypes)
-handleDragging(mousePos : Vector2)
-isColliding(building : Building) : boolean
+draw(spriteBatch : SpriteBatch)
+getCurrentlySelecting() : boolean
+dispose()

1   0..*

mapmanager

Path
+Path(x : int, y : int, width : int, height : int)

Paths
+createPaths()

0..*   1

<<enumeration>> Status
Building
Path
Unoccupied

Grid
-rows : int {readOnly}
-cols : int {readOnly}
-gridColor : Color {readOnly}
-cellSize : float {readOnly}
-grid : Status[][]
-shapeRenderer : ShapeRenderer

+Grid()
-updateGrid(building : Building)
+draw(viewport : StretchViewport)
-setupRenderer(viewport : StretchViewport)
-drawHorizontalLines()
-drawVerticalLines()
+dipose()

Map
-width : int {readOnly}
-height : int {readOnly}
-grid : Grid {readOnly}
-buildings : BuildingManager
+collidableSprites : Array<Sprite>
-spriteBatch : SpriteBatch
-viewport : StretchViewport

+Map(game : Main)
+getBuildingManager() : BuildingManager
+input()
+draw()
-drawSetup()
-drawBackground()
-drawPath()
+resize(width : int, height : int)
+pause()
+dispose()

GUImanager

Timer
-elapsedTime : float
-maxTime : float {readOnly}
-reachedMaxTime : boolean

+Timer()
+update()
+getElapsedTime() : float
+hasReachedMaxTime() : boolean

BuildingMenu
-stage : Stage
-buildings : BuildingManager
-menuBar : Image
-skin : Skin {readOnly}
-accomodationCount : int
-lectureHallCount : int
-libraryCount : int
-courseCount : int
-foodZoneCount : int
-recreationalCount : int
-natureCount : int
+buildingCounts : int[]
-countLabels : Array<Label>

+BuildingMenu(stage : Stage, buildings : BuildingManager)
+createBuildingMenu()
-createImageButtons()
-createImageButton(type : BuildingTypes, buttonGap : int)
-setupImageButton(index : int, buttonGap : int)
-addImageButtonClick(button : ImageButton, type : BuildingTypes, index : int)
-setupCountLabel(index : int, button : ImageButton)
-updateCountLabel(index : int)
-createCountLabels()
-createMenuBar()
+draw()
+dispose()

GameMenu
-stage : Stage
-skin : Skin {readOnly}
-buildingMenu : BuildingMenu
-timer : Timer
-timerLabel : Label
-isPaused : boolean

+GameMenu(game : Main, timer : Timer, buildings : BuildingManager)
+activate()
-createMenu()
-createTimerLabel()
-updateTimerLabel()
+pause()
+resume()
+input()
+draw()
+getPaused() : boolean
+dipose()

GameScreen
-game : Main
-viewport : StretchViewport
-timer : Timer
-menu : GameMenu
-isPaused : boolean = false
-hasEnded : boolean = false
-map : Map

+GameScreen(game : Main)
+show()
+render(delta : float)
-input()
-update()
-draw()
+resize(width : int, height : int)
+pause()
+resume()
+hide()
+dispose()

StartScreen
-game : Main
- viewport : StretchViewport
-stage : Stage
-startButton : ImageButton

+StartScreen(game : Main)
+show()
-addStartButton()
-setupStartButton()
-addStartButtonClick()
-drawBackground()
+render(delta : float)
+resize(width : int, height : int)
+pause()
+resume()
+hide()
+dispose()

EndScreen
-game : Main
-viewport : StretchViewport
-stage : Stage
-scoreLabel : Label
-skin : Skin {readOnly}
-score : int

+EndScreen(game : Main, score : int)
-createScoreLabel()
+show()
-drawBackground()
+render(delta : float)
+resize(width : int, height : int)
+pause()
+resume()
+hide()
+dispose()