UML Diagram By
Sanchay Singh | 2023478

ScreenAdapter

Game

FreeTypeFontGenerator

SpriteBatch

ShapeRenderer

BitmapFont

Texture

Music

<<extends>>

<<extends>>

HomeScreen
- game: *Main*
- viewHeight: *int*
- viewWidth: *int*
- stage: *Stage*
- playTexture: *Texture*
- playButton: *ImageButton*
- quitTexture: *Texture*
- quitButton: *ImageButton*

+ HomeScreen(game: Main)
+ show(): *void*
+ render(delta: float): *void*
+ resize(width: int, height: int): *void*
+ hide(): *void*

Main (Game)
- batch: *SpriteBatch*
- shapeRenderer: *ShapeRenderer*
- font: *BitmapFont*
- smallFont: *BitmapFont*
- mediumFont: *BitmapFont*
- music: *Music*
- background: *Texture*

+ create(): *void*
+ dispose(): *void*

Stage

ImageButton

Skin

MainScreen
- game: *Main*
- viewHeight: *int*
- viewWidth: *int*
- stage: *Stage*
- backBtnTexture: *Texture*
- lvlBtnTexture: *Texture*
- lockTexture: *Texture*
- backBtn: *ImageButton*
- lvlBtn1: *ImageButton*
- lvlBtn2: *ImageButton*
- lvlBtn3: *ImageButton*
- lvlBtn4: *ImageButton*
- lvlBtn5: *ImageButton*

+ MainScreen(game: Main)
+ show(): *void*
+ render(delta: float): *void*
+ resize(width: int, height: int): *void* + hide():
void
- shakeButton(button: ImageButton): *void*
- drawCenteredText(font: BitmapFont, text:
String, button: ImageButton): *void*
- drawLockOverlay(button: ImageButton): *void*

SettingsScreen
- game: *Main*
- viewHeight: *int*
- viewWidth: *int*
- stage: *Stage*
- settingsBgTexture: *Texture*
- boardTexture: *Texture*
- crossBtnTexture: *Texture*
- soundTexture: *Texture*
- musicTexture: *Texture*
- languageTexture: *Texture*
- creditsTexture: *Texture*
- termsPrivacyTexture: *Texture*
- crossBtn: *ImageButton*
- soundBtn: *ImageButton*
- musicBtn: *ImageButton*
- languageBtn: *ImageButton*
- creditsBtn: *ImageButton*
- termsPrivacyBtn: *ImageButton*
- boardWidth: *float*
- boardHeight: *float*

+ SettingsScreen(game: Main)
+ show(): void
+ render(delta: float): void
+ createButton(texture: Texture, x:
float, y: float, size: float):
ImageButton
+ resize(width: int, height: int): void
+ hide(): void

Bird
- type: *String*
- size: *float*
+ texture: *Texture*
+ x: *float*
+ y: *float*
+ height: *int*
+ width: *int*

+ Bird(type: String, size: float, texture:
Texture, x: float, y: float, width: int, height: int)

LevelScreen
- game: *Main*
- viewHeight: *int*
- viewWidth: *int*
- stage: *Stage*
- pauseStage: *Stage*
- gameOverStage: *Stage*
- pauseTexture: *Texture*
- pauseButton: *ImageButton*
- settingsTexture: *Texture*
- settingsButton: *ImageButton*
- isPaused: *boolean*
- isGameOver: *boolean*
- score: *int*
- redTexture: *Texture*
- birds: *ArrayList<Bird>*
- slingShotTexture: *Texture*
- slingShot: *SlingShot*
- floor: *Texture*
- wall: *Texture*
- block: *Texture*
- materials: *ArrayList<Material>*
- pigTexture: *Texture*
- pigs: *ArrayList<Pig>*

+ LevelScreen(game: Main)
+ createGameOverScreen(): void +
createPauseScreen(): void
+ show(): void
+ render(delta: float): void
+ resize(width: int, height: int): void + hide(): void

Red
+ Red(texture: Texture, x: float, y:
float, width: int, height: int)

<<extends>>

Pig
+ size: *float*
+ hp: *int*
+ x: *float*
+ y: *float*
+ texture: *Texture*
+ width: *int*
+ height: *int*

+ Pig(size: float, hp: int, texture: Texture, x: float, y: float, width: int, height: int)

SlingShot
- texture: *Texture*
- x: *float*
- y: *float*
- width: *int*
- height: *int*

+ SlingShot(texture: Texture, x:
float, y: float, width: int, height: int)

AveragePig
+ AveragePig(texture: Texture, x: float, y: float, width: int,
height: int)

<<extends>>

Material
- name: *String*
+ texture: *Texture*
+ x: *float*
+ y: *float*
+ width: *int*
+ height: *int*

+ Material(name: String, texture: Texture, x: float, y: float, width: int, height: int)

<<extends>>

Wood
+ Wood(texture: Texture, x: float, y: float, width: int, height: int)