```
(VkInstance instance, VkSurface
                    KHR *surface)
                  + VkExtent2D GetExtent()
                  + bool HasWindowResized()
                  + void ResetWindowResizedFlag()
                  + GLFWwindow * GetGLFWwindow
                    () const
                  - void Init()
                  - static void FrameBufferResized
                    Callback(GLFWwindow *window,
                     int width, int height)
                               -mWindow
             VRE::VRE Device
    + const bool mEnableValidation
      Layers

    VkPhysicalDeviceProperties

       mProperties

    VkInstance mInstance

    VkDebugUtilsMessengerEXT

       mDebugMessenger

    VkPhysicalDevice mPhysical

      Device
    - VkCommandPool mCommandPool
    - VkDevice mVkDevice
    - VkSurfaceKHR mVkSurface
    - VkQueue mGraphicsQueue
    - VkQueue mPresentQueue
    const std::vector
      const char * > mValidation
      Layers
    - const std::vector<
       const char * > mDeviceExtensions
    + VRE_Device(VRE_Window
       &window)
    + ~VRE_Device()
    + VRE_Device(const VRE
      _Device &)=delete
    + VRE_Device & operator
      =(const VRE_Device &)
      =delete
    + VRE_Device(VRE_Device
       &&)=delete
    + VRE Device & operator
      =(VRE_Device &&)=delete
                                               -mWindow
    + VkCommandPool GetCommand
      Pool()
    + VkDevice GetVkDevice()
    + VkPhysicalDevice GetPhysical
      Device()
    + VkSurfaceKHR Surface()
      and 12 more...

    void CreateInstance()

    - void SetupDebugMessenger()

    void CreateSurface()

    - void PickPhysicalDevice()
    - void CreateLogicalDevice()
    - void CreateCommandPool()
    - bool IsDeviceSuitable
      (VkPhysicalDevice device)
    std::vector< const</li>
      char * > GetRequiredExtensions()
    - bool CheckValidationLayer
      Support()
    - QueueFamilyIndices
       FindQueueFamilies(VkPhysical
      Device device)
    - void PopulateDebugMessenger
      CreateInfo(VkDebugUtilsMessenger
      CreateInfoEXT &createInfo)
    - void HasGflwRequiredInstance
      Extensions()

    bool CheckDeviceExtension

      Support(VkPhysicalDevice
       device)
    - SwapChainSupportDetails
       QuerySwapChainSupport
      (VkPhysicalDevice device)
                                    -mDevice
                                      VRE::VRE_Renderer
                            - std::unique_ptr< VRE
                              _SwapChain > mSwapChain
                            std::vector< VkCommandBuffer</li>
                              > mCommandBuffers

    uint32_t mCurrentImageIndex

    int mCurrentFrameIndex

                            - bool mDrawStarted
                            + VRE_Renderer(VRE_Window
                              &window, VRE_Device &device)
                            + ~VRE Renderer()
                            + VRE Renderer(const
                              VRE_Renderer &)=delete
                           + VRE_Renderer & operator
                              =(const VRE_Renderer &)
                              =delete
                           + VkCommandBuffer BeginDraw()
               +mDevice
                            + void EndDraw()
                           + void BeginSwapChainRender
                              Pass(VkCommandBuffer commandBuffer)
                            + void EndSwapChainRenderPass
                              (VkCommandBuffer commandBuffer)

    bool IsDrawInProgress

                              () const
                           + VkRenderPass GetSwapChain
                             RenderPass() const
                           + float GetAspectRatio
                              () const
                           + VkCommandBuffer GetCurrent
                             CommandBuffer() const
                           + int GetFrameIndex()
                              const
                            void CreateCommandBuffers()

    void RecreateSwapChain()

                            void FreeCommandBuffers()
                           +mRenderer
    VRE::VRE_SharedContext
+ std::shared_ptr< VRE
  _DescriptorSetLayout
  > mGlobalDescSetLayout
+ std::vector< std::unique
  _ptr< VRE_DescriptorPool
  > > mObjectDescPools
+ std::vector< VkDescriptor
  Set > mSceneDescriptorSets
+ std::vector< VRE PointLight
  > mPointLights
+ VRE_GameObject::GameObjects
  Map * mGameObjMap
+ VRE SharedContext()
                 -mSceneContext
  VRE::VRE_LightRenderSystem
```

std::unique\_ptr< VRE \_Pipeline > mPipeline

Layout

&)=delete

+ void Init()

float dt)
+ void Render()

VkPipelineLayout mPipeline

(VRE\_SharedContext \*sceneContext)

+ VRE\_LightRenderSystem

+ VRE\_LightRenderSystem & operator=(const VRE

+ void Update(UBO &ubo,

 void CreatePipelineLayout (VkDescriptorSetLayout descSetLayout)
 void CreatePipeline

(VkRenderPass renderPass)

+ ~VRE\_LightRenderSystem()+ VRE\_LightRenderSystem

(const VRE\_LightRenderSystem

\_LightRenderSystem &)=delete

VRE::VRE\_Window

bool mFrameBufferResizedconst std::string mWindowName

GLFWwindow \* mWindow+ VRE\_Window(int width, int height, std::string name)

+ VRE\_Window(const VRE \_Window &)=delete+ VRE\_Window & operator =(const VRE\_Window &)

+ void CreateWindowSurface

+ ~VRE\_Window()

+ bool ShouldClose()

=delete

int mWidthint mHeight