Universidade Presbiteriana Mackenzie
Eduardo Veit Ferrão RA: 10389961
Leonardo Pinheiro de Souza RA: 10388961
Lucas Paulo da Rocha RA: 10391076
Luiz Octavio Tassinari Saraiva RA: 10374379

Faça uma pesquisa sobre APIs gráficas e elabore um texto descrevendo os resultados da pesquisa.

O seu texto deve conter:

- **Uma breve descrição da API gráfica que você selecionou para a pesquisa;**
- **Como a pipeline é documentada pelos desenvolvedores da API;**
- **Quais linguagens de shading (shaders) são suportadas pela API;**
- **Um exemplo de código que demonstra o uso da API (pode ser um "Hello, World!" gráfico – renderizar um triângulo na tela);**
- **Um exemplo de código de shader suportado pela API;**
- **A descrição de um exemplo de aplicação que usa a API.**

A API Vulkan é uma interface de programação de aplicativos (API) gráfica que se destaca pelo seu foco em comandos de baixo nível, projetada para minimizar o overhead da CPU e otimizar a distribuição do processamento entre múltiplos núcleos. Desenvolvida para substituir a API OpenGL, a Vulkan é particularmente benéfica para desenvolvedores de jogos ao facilitar a mitigação de gargalos de processador, graças à sua robusta implementação de funcionalidades multi-threading.

A pipeline da Vulkan é meticulosamente documentada, e sua gestão é feita por meio de um objeto monolítico, que encapsula a descrição de todos os estágios dos shaders. Este design permite uma otimização eficaz dos shaders, pois elimina a necessidade de validação de input durante a montagem da pipeline. As linguagens de shader suportadas pela Vulkan incluem HLSL, GLSL e SPIR-V, proporcionando aos desenvolvedores uma variedade de opções para a programação gráfica.

Abaixo segue um exemplo de código em C++ para desenhar um triângulo e um código que cria um layout de Shader no Vulkan.

```
#include <vulkan/vulkan.h>
#include <iostream>
#include <stdexcept>
#include <cstdlib>

class HelloTriangleApplication {
public:
    void run() {
        initVulkan();
        mainLoop();
        cleanup();
    }

private:
    void initVulkan() {

    }

    void mainLoop() {

    }

    void cleanup() {

    }
};

int main() {
    HelloTriangleApplication app;

    try {
        app.run();
    } catch (const std::exception& e) {
        std::cerr << e.what() << std::endl;
        return EXIT_FAILURE;
    }

    return EXIT_SUCCESS;
}
```

```
// CODELAB: hellovk.h
void HelloVK::createDescriptorSetLayout() {
  VkDescriptorSetLayoutBinding uboLayoutBinding{};
  uboLayoutBinding.binding = 0;
  uboLayoutBinding.descriptorType = VK_DESCRIPTOR_TYPE_UNIFORM_BUFFER;
  uboLayoutBinding.descriptorCount = 1;
  uboLayoutBinding.stageFlags = VK_SHADER_STAGE_VERTEX_BIT;
  uboLayoutBinding.pImmutableSamplers = nullptr;

  VkDescriptorSetLayoutCreateInfo layoutInfo{};
  layoutInfo.sType = VK_STRUCTURE_TYPE_DESCRIPTOR_SET_LAYOUT_CREATE_INFO;
  layoutInfo.bindingCount = 1;
  layoutInfo.pBindings = &uboLayoutBinding;

  VK_CHECK(vkCreateDescriptorSetLayout(device, &layoutInfo, nullptr,
                                       &descriptorSetLayout));
}
```

A descrição de um exemplo de aplicação que usa a API.
Red Dead Redemption 2 utiliza a API gráfica Vulkan para renderização no PC. Vulkan é uma API gráfica de baixo nível que oferece maior controle sobre o hardware, permitindo aos desenvolvedores otimizar o desempenho e aproveitar ao máximo os recursos disponíveis, por isso a escolha em utilizar em um jogo de mundo aberto como RDR2.
Além disso, a escolha da Vulkan pode ser vantajosa devido à sua capacidade de distribuir melhor a carga de trabalho entre os núcleos da CPU e os recursos de GPU, resultando em melhor desempenho (especialmente em hardware moderno).

Fontes:

- https://www.adrenaline.com.br/amd/entenda-o-que-e-a-vulkan-e-porque-ela-pode-mudar-o-mercado-de-games-para-pc/
- https://docs.vulkan.org/spec/latest/chapters/pipelines.html
- https://docs.vulkan.org/spec/latest/appendices/spirvenv.html#spirvenv-extensions
- https://www.vulkan.org/tools
- https://docs.vulkan.org/tutorial/latest/03_Drawing_a_triangle/00_Setup/00_Base_code.html
- https://www.shacknews.com/article/114834/should-you-choose-vulkan-or-directx-12-in-red-dead-redemption-2
- https://developer.android.com/codelabs/beginning-vulkan-on-android?hl=pt_br#5