

# Explorando a Realidade Virtual em Museus da Cultura Latino-Americana: Uma Imersão na Cultura Colombiana

Autor: William Ramirez Ruiz

2023

## 1 Especificação do programa

### 1.1 Escopo

O escopo do projeto abrange o desenvolvimento de um software de Realidade Virtual para Museus Culturais Latino-Americanos (SRVMCLA), com foco na exploração dos aspectos culturais da Colômbia. O programa permitirá que os visitantes dos museus vivenciem uma experiência imersiva e interativa por meio de ambientes virtuais, onde poderão explorar exposições virtuais, interagir com objetos culturais e aprender mais sobre a rica herança colombiana.

O escopo inclui, mas não se limita a:

- Desenvolvimento de software: Projeto e implementação do SRVMCLA. Integração de tecnologias de Realidade Virtual, como renderização em tempo real, rastreamento de movimento e interatividade.
- Ambientes virtuais: Criação de ambientes virtuais que representam diferentes aspectos da cultura colombiana, como arte, música, dança e história.
- Ambientes virtuais: Design e modelagem 3D dos ambientes virtuais, considerando aspectos estéticos e de usabilidade.
- Interatividade e recursos educacionais: Implementação de interatividade nos ambientes virtuais, permitindo que os visitantes interajam com objetos culturais, acessem informações adicionais e participem de atividades interativas. Desenvolvimento de recursos educacionais, como textos informativos, vídeos explicativos e áreas interativas, para fornecer conhecimento sobre a cultura colombiana.
- Testes e avaliação: Realização de testes para garantir a funcionalidade, usabilidade e eficácia do programa. Os testes serão conduzidos localmente pelo programador, visando identificar e corrigir possíveis problemas e garantir que o software atenda aos requisitos estabelecidos. Essa

abordagem foi adotada devido a limitações de recursos e disponibilidade de usuários em um estágio inicial do projeto.

- Documentação:

Elaboração de documentação completa do software, incluindo manuais do usuário e do desenvolvedor, guias de instalação e requisitos de sistema.

Os usuários principais do software são os visitantes do museu que desejam explorar e experimentar a cultura colombiana por meio da Realidade Virtual. Esses usuários podem ser tanto locais quanto turistas interessados em aprender sobre a cultura e o patrimônio latino-americano.

## **1.2 Requisitos funcionais**

- RF1: Exibição de ambientes virtuais.
- RF2: Navegação interativa.
- RF3: Interatividade com objetos virtuais.
- RF4: Recursos educacionais.
- RF5: Áudio e/ou guias de áudio.
- RF6: Integração de mídia.
- RF7: Personalização das configurações.
- RF8: Compatibilidade de hardware.
- RF9: Integração de feedback dos visitantes.

## **1.3 Requisitos não funcionais**

- RNF1: Desempenho.
- RNF2: Usabilidade.
- RNF3: Confiabilidade.
- RNF4: Compatibilidade.
- RNF5: Segurança.
- RNF6: Desenvolvimento sustentável.
- RNF7: Performance gráfica.
- RNF8: Disponibilidade.

## **1.4 Especificação dos requisitos**

### **1.5 Requisitos Funcionais**

#### **1.5.1 Ambientes Virtuais**

- O software deve permitir a criação e exibição de ambientes virtuais que representem aspectos da cultura colombiana, como exposições de arte, locais históricos e eventos culturais.
- Os ambientes virtuais devem ser imersivos, proporcionando uma experiência realista e interativa para os visitantes.

#### **1.5.2 Navegação e Interatividade**

- Os visitantes devem ser capazes de navegar pelos ambientes virtuais de forma intuitiva e interativa, utilizando dispositivos de entrada compatíveis com a tecnologia de realidade virtual.
- Deve ser possível interagir com objetos virtuais, como selecionar, ampliar, rotacionar e obter informações adicionais sobre os itens culturais.

#### **1.5.3 Recursos Educacionais**

- O software deve fornecer recursos educacionais para enriquecer a experiência dos visitantes, como informações detalhadas sobre os elementos exibidos, contexto histórico e cultural, bem como curiosidades relevantes.
- Deve ser possível apresentar vídeos, áudios e imagens relacionados aos elementos culturais, para aprofundar o conhecimento dos visitantes.

### **1.6 Requisitos Não Funcionais**

#### **1.6.1 Desempenho**

- O software deve ter um desempenho adequado, garantindo tempos de resposta rápidos e fluidez na navegação pelos ambientes virtuais.
- A renderização gráfica dos elementos virtuais deve ser realizada de forma eficiente, proporcionando uma experiência visual imersiva e de alta qualidade.

#### **1.6.2 Usabilidade**

- A interface do usuário deve ser intuitiva, de fácil utilização e acessível para usuários com diferentes níveis de habilidade.
- O software deve fornecer feedback claro e orientações adequadas para ajudar os visitantes a interagir e explorar os ambientes virtuais.

### 1.6.3 Segurança

- O software deve adotar medidas de segurança adequadas para proteger os dados dos usuários, garantindo a privacidade das informações pessoais.

### 1.6.4 Compatibilidade

- O software deve ser compatível com dispositivos de realidade virtual populares, como Oculus Rift, HTC Vive, para garantir que os visitantes possam acessar a experiência com os equipamentos disponíveis.

### 1.6.5 Documentação

- Deve ser fornecida documentação completa, incluindo manuais de usuário e guias de instalação, para ajudar os visitantes a utilizar o software adequadamente.
- A documentação deve ser clara, concisa e fácil de entender, abordando todos os aspectos relevantes do uso do programa.

## 2 Projeto do programa

### 2.1 Modelo de programação

Nossa arquitetura do software de Realidade Virtual para Museus seguirá o padrão MVP (Model-View-Presenter), uma arquitetura de software que separa as responsabilidades e a lógica em três componentes principais: Modelo (Model), Visualização (View) e Apresentador (Presenter), ver figura 1.

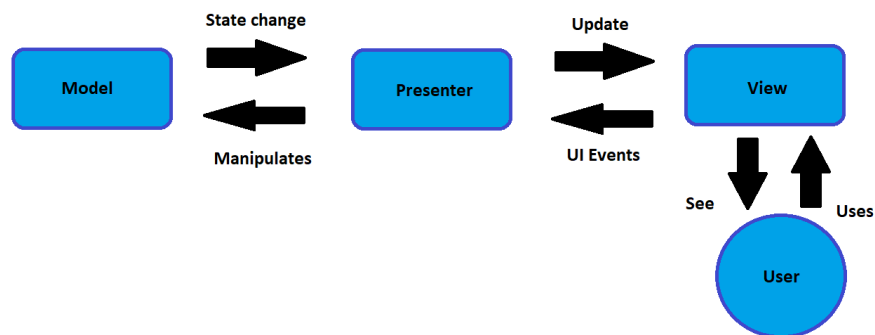


Figure 1: Arquitetura de modelo.

- Modelo

O Modelo representa os dados e a lógica de negócios do programa. Ele será responsável por gerenciar o acesso aos dados do banco de dados, como

informações sobre os elementos culturais e recursos associados. O Modelo vai ser implementado como classes  $C++$  que lidam com a recuperação e manipulação dos dados, fornecendo métodos para buscar informações específicas, atualizar registros, etc.

- Visualização

A Visualização é responsável pela apresentação da interface do usuário e pela exibição dos ambientes virtuais e elementos culturais. Ela será responsável pela renderização gráfica e pela interação com o usuário. A Visualização pode ser implementada usando os recursos do Unity, como objetos `GameObjects`, scripts de controle de cena e elementos de interface do usuário (UI) para exibir informações, menus e controles de navegação.

- Apresentador

O Apresentador atua como intermediário entre o Modelo e a Visualização. Ele lida com a lógica de negócios, processa as interações do usuário e atualiza a Visualização com base nos dados fornecidos pelo Modelo. O Apresentador é implementado como scripts  $C++$  que respondem a eventos de entrada do usuário, como cliques ou movimentos, interagem com o Modelo para recuperar informações e atualizam a Visualização de acordo.

## 2.2 Arquitetura e projeto

### 2.2.1 UML: VR museum

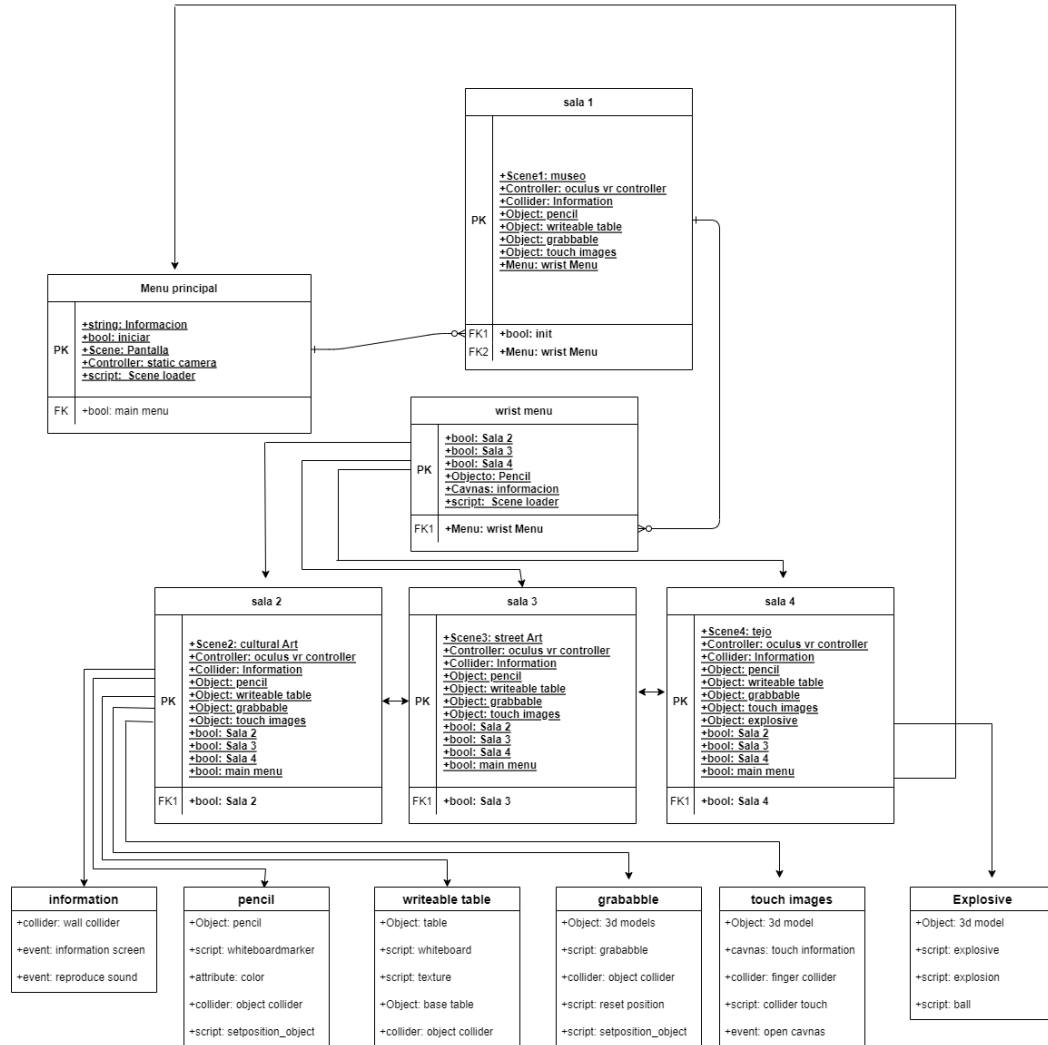


Figure 2: UML do projeto SRVMCLA.

### 2.2.2 Diagrama de sequencia

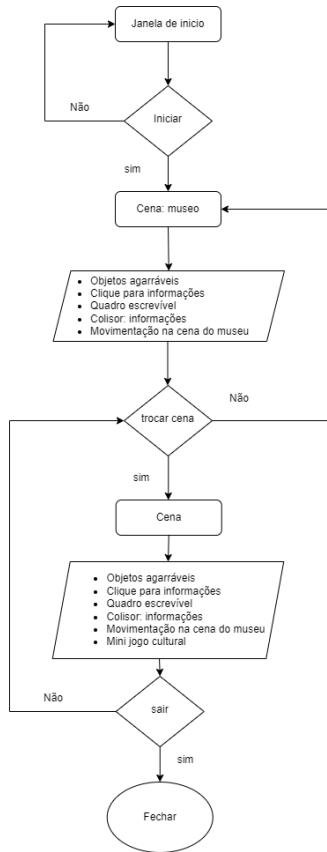


Figure 3: Diagrama de fluxo do projeto SRVMCLA.

O diagrama de fluxo mostra de forma geral como o usuário irá navegar no museu virtual, ver Figura 3. Como pode ser observado, percorreremos cada cena a partir de um menu e o programa será cíclico, o que significa que poderemos revisitar outras exposições. Para sair, será necessário apenas selecionar essa opção. De forma geral, cada sala seguirá a mesma lógica, com exceção da sala principal e das outras três salas, que terão interações únicas.

### 2.2.3 Modelamiento: Objetos

- Modelagem 3D: Para criar objetos 3D, são utilizadas ferramentas de modelagem 3D, como Blender, Maya ou 3ds Max. Essas ferramentas permitem criar geometria, aplicar texturas, definir materiais e estabelecer propriedades visuais dos objetos.

- Importação no Unity: Uma vez que os objetos 3D são criados e exportados em um formato compatível, como FBX ou OBJ, eles são importados no Unity. O Unity é um motor de jogo multiplataforma que suporta a importação de modelos 3D e fornece ferramentas para sua gestão.
- GameObjects: No Unity, os objetos 3D são representados como GameObjects, que são contêineres para componentes. Os GameObjects podem ser criados e colocados na cena e podem ter transformações (posição, rotação e escala) aplicadas a eles.
- Componentes: Os componentes são peças funcionais que são anexadas aos GameObjects para definir seu comportamento e características. No caso de objetos 3D, alguns componentes comuns são: MeshRenderer (renderização de malha), Collider (colisões físicas), Rigidbody (simulação de física), Animator (animações), entre outros. Esses componentes permitem definir a aparência visual, interações físicas e comportamento dos objetos no ambiente de jogo.
- Scripts e programação: O Unity permite estender a funcionalidade dos GameObjects por meio da escrita de scripts em linguagens como C++ ou JavaScript. Esses scripts são anexados como componentes aos GameObjects e são usados para controlar seu comportamento, como a interação com o jogador, a lógica do jogo ou a manipulação de propriedades.
- Hierarquia de GameObjects: Os GameObjects podem ser organizados em uma hierarquia para estabelecer relações parentais e fazer com que os objetos sigam transformações relativas. Isso permite criar estruturas mais complexas e manter uma organização clara dos objetos na cena.

Para o nosso caso, os scripts serão em C++ e serão destinados a efeitos de ações, como menus suspensos, relógio e interações, como pegar, redefinir posição e interagir com o ambiente.