

Proposta de Trabalho de Conclusão de Curso

Alunas: Clara Yuki Sano e Júlia Melo Teixeira dos Santos

Orientador: Carlos Hitoshi Morimoto

Bacharelado em Ciência da Computação

Instituto de Matemática e Estatística (IME) - Universidade de São Paulo (USP)

Resumo

Este trabalho propõe o desenvolvimento de uma aplicação que utiliza Realidade Aumentada (AR) para o ensino introdutório de lógica de programação em Python. Seu funcionamento consiste na manipulação de blocos físicos pelo usuário que serão reconhecidos e associados a trechos de código Python pelo software, resultando na simulação do código final quando o arranjo dos blocos estiver correto. Apesar de inspirado na linguagem Scratch, este trabalho se destaca ao empregar código em uma linguagem de programação real e interação física, visando promover uma experiência pedagógica mais imersiva e envolvente. A aplicação será implementada utilizando a engine de jogos Unity, sua framework AR Foundation e a biblioteca Google ARCore e será analisada através de testes para avaliar sua usabilidade e seu impacto na aprendizagem. O trabalho será arrematado com uma reflexão sobre o impacto do uso de Realidade Estendida (XR) no processo de ensino e aprendizagem de programação.

Escopo

O escopo principal deste trabalho é o desenvolvimento de uma aplicação voltada para o ensino introdutório de lógica de programação em Python utilizando recursos de Realidade Aumentada. O público-alvo proposto são estudantes de ensino médio e estudantes de graduação que têm pouca ou nenhuma experiência prévia com produção de código e podem ser beneficiados pela abordagem lúdica.

A partir dos resultados obtidos com a aplicação, o trabalho também propõe uma reflexão crítica sobre impacto que a incorporação de tecnologias de Realidade Estendida (XR), termo abrangente para as Realidades Virtual, (VR), Mista (MR) e Aumentada (AR), tem no processo de aprendizado de programação.

Objetivo

O objetivo do trabalho é a criação de uma aplicação que efetivamente integra os planos físico e virtual em uma interface interativa híbrida de experimentação da lógica de programação e de construção de código em Python para iniciantes. A produção do software de qualidade não está restrita a sua implementação utilizando a framework *Unity AR Foundation* e a biblioteca *Google ARCore*, mas também depende da construção de uma linguagem de interação que conecta ambos os planos visando constituir uma experiência

pedagógica imersiva e contínua, sem que a manipulação dos elementos físicos pareça desconexa da exibição dos resultados virtuais.

Métodos

O desenvolvimento deste trabalho será conduzido em etapas que abrangem desde o estudo teórico sobre tópicos de Realidade Aumentada até a realização de testes finais com usuários, com a redação da monografia sendo feita de forma paralela a todas as fases. Cada etapa proposta segue uma lógica que prioriza, inicialmente, a definição clara dos objetivos, das estruturas e conceituais e das diretrizes técnicas necessárias para que a fase de implementação correspondente ocorra da forma mais sólida possível. Esse cuidado metodológico busca garantir coerência entre as intenções pedagógicas do projeto e as soluções técnicas adotadas.

A escolha das tecnologias a serem utilizadas no desenvolvimento da aplicação considerou critérios como compatibilidade, suporte da comunidade, curva de aprendizado e disponibilidade de recursos voltados a AR. A biblioteca *Google ARCore* foi selecionada como base da aplicação devido ao seu amplo suporte a diferentes dispositivos, documentação abrangente e recursos robustos para o desenvolvimento de experiências em AR. Tendo a framework base definida, foi realizada uma análise comparativa entre duas plataformas de desenvolvimento que a suportam: *Android Studio* e *Unity*. A plataforma *Unity* foi escolhida por apresentar uma interface mais acessível a desenvolvedores iniciantes dentro do campo de Realidade Aumentada, além de permitir uma integração direta com o framework *AR Foundation*, que pode ser utilizado em conjunto com o *ARCore*. O *AR Foundation* fornece uma camada extra de abstração multiplataforma que reúne um conjunto extenso de ferramentas para a construção de experiências imersivas, sendo um acréscimo bastante interessante para o desenvolvimento. Essa integração entre *Unity*, *AR Foundation* e *ARCore* será capaz de proporcionar um ecossistema mais completo e eficiente para a construção da aplicação proposta, facilitando a implementação de funcionalidades interativas baseadas na manipulação de objetos físicos e sua correspondência com estruturas de código em ambiente virtual.

Em relação à interface e funcionamento da aplicação, houve uma inspiração na lógica visual presente em linguagens visuais e educativas como o Scratch, mas com uma abordagem diferente imbuída em Realidade Aumentada: ao invés de blocos digitais manipulados via tela sensível ao toque, os usuários irão interagir com blocos físicos, semelhantes a peças de Lego, que representam comandos e estruturas fundamentais da linguagem Python. Esses blocos serão reconhecidos pela aplicação por meio da câmera do dispositivo e mapeados dinamicamente para trechos de código. A união física dos blocos corresponderá à construção sequencial de um programa em Python, promovendo uma associação direta e tátil entre lógica computacional e estrutura de código.

O cenário de uso proposto para a aplicação envolve um dispositivo celular com sistema Android, acoplado a um visor do tipo Google Cardboard, que permite a imersão do usuário no ambiente de Realidade Aumentada com menor custo e maior acessibilidade. A câmera do próprio celular será utilizada para capturar e reconhecer os blocos físicos, enquanto a interface em AR será sobreposta ao ambiente real visível pelo visor.

Ao final da montagem e encaixe de blocos, o código resultante será executado pela aplicação e seus efeitos serão simulados visualmente no ambiente de AR, possibilitando aos usuários observar o comportamento do programa em tempo real, dentro do espaço físico ao seu redor. Essa estratégia visa facilitar a compreensão de conceitos abstratos por meio de uma linguagem de interação lúdica, concreta e intuitiva, promovendo o engajamento e a retenção de conhecimento por parte dos estudantes iniciantes.

Forma de análise

A análise da performance da aplicação será majoritariamente qualitativa. Serão realizados testes com usuários iniciantes no estudo da computação para que comportamentos que afetados pelo uso do software na absorção e entendimento do conteúdo de lógica de programação possam ser observados e avaliados. São eles:

- Engajamento e interesse do usuário em utilizar a aplicação e completar o código;
- Facilidade com que usuários compreendem a linguagem de interação com a aplicação, relacionando arranjo físico dos blocos com construção de código e conceitos abstratos de programação;
- Contribuição que a simulação do código final oferece ao entendimento da lógica de programação pelo usuário.

Adicionalmente, serão observadas algumas métricas quantitativas relacionadas a esses comportamentos, como o tempo e tentativas necessárias para que os usuários consigam arrastar os blocos corretamente, formando um código executável em Python.

Bibliografia

Documentação e outros recursos didáticos sobre as frameworks e plataformas que serão utilizadas no projeto:

- Documentação [Google ARCore](#)
- Documentação [AR Foundation](#)
- [Getting started with Unity](#)
- [Unity for beginners Brackeys course](#)

Recursos de estudo relacionados a AR e Programação Visual:

- [Augmented Reality for Everyone - Full Course](#)
- [Characterizing Visual Programming Approaches for End-User Developers: A Systematic Review](#)

Cronograma das principais atividades

Atividade	Duração estimada
Estudo teórico sobre XR, interação físico-virtual e ensino de programação	1 mês

Instalação e familiarização de Unity com framework <i>AR Foundation</i> e biblioteca <i>Google ARCore</i>	2 semanas
Definição de regras e convenções de interação do usuário	2 semanas
Prototipação da interface e realização de testes de interface	3 semanas
Implementação de detecção de blocos	3 semanas
Implementação de detecção de interação do usuário	3 semanas
Implementação de mapeamento de objeto físico (blocos) para virtual (código em Python)	1 mês
Definição de maneira de execução de código Python	2 semanas
Implementação de execução de código Python	3 semanas
Definição de maneira de simulação de código Python em AR	2 semanas
Implementação de simulação de código Python em AR	1 mês
Realização de testes com usuário e análise dos resultados	1 mês