

LICENCIATURA EM CIÊNCIAS DA COMPUTAÇÃO

CRIAÇÃO DE UMA APLICAÇÃO EM REALIDADE AUMENTADA

Carlos Beiramar Marco Silva
a84628 a86030

29 de junho de 2022

Orientadores:
Pedro Rangel Henriques
Lázaro Lima

Agradecimentos

Gostaríamos de começar por agradecer a todas as pessoas que contribuíram para que a realização deste trabalho fosse possível. Aos professores Pedro Rangel Henriques e Lázaro Lima, agradecer pela orientação e pela disponibilidade que nos facultaram durante o desenvolvimento deste projeto.

Resumo

O presente trabalho tem como objetivo introduzir o Pensamento Computacional em crianças e jovens para que, mais tarde, estejam habilitadas a resolver tarefas de Programação com maior facilidade e eficácia. Para isto, iremos criar uma aplicação desenvolvida em *Unity* com apoio do *Vuforia* e, para a implementação das scripts, foi usado *C#*.

Conteúdo

1	Introdução	3
1.1	Contextualização	3
1.2	Estrutura do Relatório	4
2	Análise e Especificação	5
2.1	Pensamento Computacional	5
2.2	Realidade Virtual	7
2.2.1	Cronologia da Realidade Virtual	7
2.3	Realidade Aumentada	11
2.3.1	Tipos de Realidade Aumentada	11
2.3.2	RA baseada em marcadores	12
2.3.3	RA sem marcadores	13
2.3.4	Realidade Mista	17
2.4	Especificação do Requisitos	19
3	Conceção da Proposta	20
3.1	Proposta de recursos de aprendizagem	20
3.2	Funcionamento da Aplicação	22
3.2.1	Targets	22
3.2.2	Objetos para cada target	25
3.2.3	Receitas	27
4	Desenvolvimento do sistema	28
4.1	Alternativas, Decisões e Problemas de Implementação	28
5	Apresentação do sistema	33
5.1	Testes realizados e Resultados	33
6	Conclusão	39

Lista de Figuras

2.1	Conceitos do Pensamento Computacional	6
2.2	Sensorama desenvolvido por Morton Heilig	8
2.3	Head-Mounted-Display	9
2.4	Cronologia da realidade virtual	10
2.5	RA baseada em marcadores	12
2.6	RA sem marcadores	13
2.7	Pokemon Go - RA baseado na localização	14
2.8	Pokemon Go - RA baseado em projeções	15
2.9	RA baseado em sobreposição	16
2.10	RA baseado no contorno	17
2.11	Diagrama de Milgram	18
3.1	Targets para representar as frutas.	22
3.2	Targets para representar os utensílios.	23
3.3	Targets para representar os ingredientes.	23
3.4	Target para representar o resultado final	24
3.5	Objects 3D que representam as frutas.	25
3.6	Objects 3D que representam os utensílios.	25
3.7	Objects 3D que representam os ingredientes.	26
5.1	Leitura da fruta	33
5.2	Leitura do utensílio	34
5.3	Leitura do primeiro ingrediente	34
5.4	Leitura do segundo ingrediente	35
5.5	Leitura do resultado obtido	36
5.6	Erro na escolha de fruta	37
5.7	Erro na escolha de utensílio	37
5.8	Erro na escolha do ingrediente	38

Capítulo 1

Introdução

1.1 Contextualização

O presente trabalho foi realizado como projeto final do curso de Ciências da Computação na Universidade do Minho e tem como objetivo a realização de uma aplicação que permita introduzir o Pensamento Computacional a crianças e a jovens para prepará-los para o futuro, tudo isto através da criação de uma aplicação em realidade aumentada. Entre inúmeras propostas, a escolha deste tema foi uma decisão unânime e deliberada por todos os membros deste grupo e houveram dois motivos que se destacaram para a tomada desta decisão.

O primeiro motivo foi pelo desafio de implementar uma aplicação em realidade aumentada, foi o que mais motivou todos os membros do grupo, não só para a escolha deste tema, como também para a realização deste projeto. O segundo motivo recaiu para a introdução do Pensamento Computacional nas crianças, algo que nós, membros do grupo e alunos de cursos relacionados com a informática, vemos como algo bastante importante e necessário.

1.2 Estrutura do Relatório

Este relatório encontra-se dividido em 5 capítulos:

- No primeiro capítulo, apresenta-se uma contextualização do trabalho e a estrutura do relatório.
- No segundo capítulo, encontra-se uma análise ao trabalho proposto e especificados os requisitos da aplicação.
- No terceiro capítulo, são expostas as partes do *backend* e a interface da plataforma, explicando a lógica das decisões tomadas.
- No quarto capítulo, são discutidas decisões e são apresentados alguns testes realizados.
- No quinto e último capítulo, apresenta-se uma revisão e análise final ao trabalho realizado, expondo o que foi feito durante o semestre e o que poderá ser feito no futuro para melhorar a plataforma.

Capítulo 2

Análise e Especificação

2.1 Pensamento Computacional

O Pensamento Computacional permite aos programadores pegar num problema complexo, perceber que tipo de problema é e desenvolver soluções possíveis para esse problema. Posteriormente, estas soluções poderão ser apresentadas de uma forma que um computador possa compreender. Existem inúmeros grupos de competências que influenciam a forma como o ser humano aborda um problema e tenta encontrar soluções para o mesmo.

De todos estes grupos, há alguns que se destacam quando o termo programação está presente, sendo eles:

- **Decomposição**

Corresponde à habilidade de dividir um problema complexo em partes mais pequenas. Assim, quando se trabalha com pequenos fragmentos de um problema, torna-se mais fácil descobrir a sua solução. A decomposição, para além de tudo isto, contribui para uma diminuição da ansiedade e do medo de encarar os desafios por partes dos estudantes pois, os alunos conseguem encontrar as respostas de cada parte do problema com mais confiança e rapidez.

- **Abstração**

É a capacidade de manter o foco em processos relevantes em vez de priorizar os pequenos detalhes, com o objetivo de criar uma solução abrangente que possa ser válida para todos os problemas. Aqui é possível desenvolver uma análise mais crítica em relação ao problema abordado visto que se filtra os dados mais relevantes para a resolução das questões.

- **Reconhecimento de padrões**

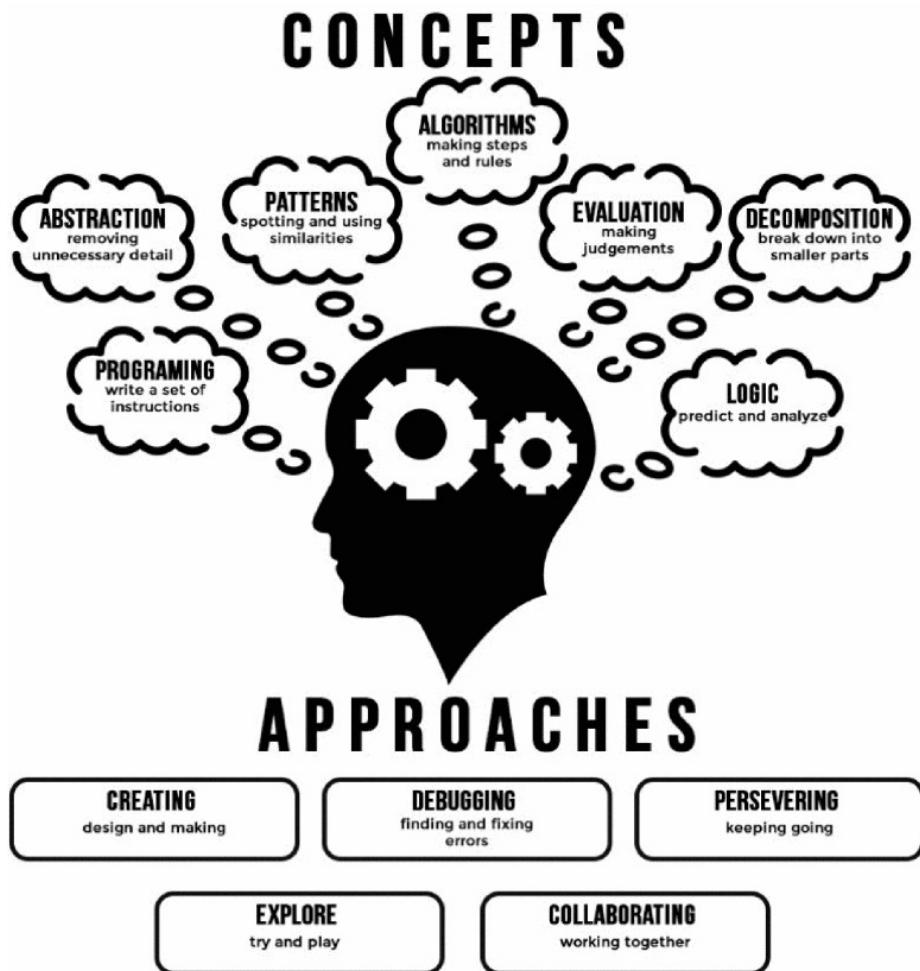


Figura 2.1: Conceitos do Pensamento Computacional

Para identificar tendências de comportamento é necessário observar atentamente a questão, reconhecendo padrões e similaridades. Permite o desenvolvimento de soluções para problemas comuns de forma inovadora.

- **Pensamento algorítmico**

O pensamento algorítmico corresponde à criação de passos e soluções para alcançar um objetivo específico para qualquer problema.

2.2 Realidade Virtual

Uma das muitas formas de o ser humano se expressar ao longo do tempo era com representações da realidade, sendo elas desenhos, figuras, pinturas... A introdução do computador acabou por potencializar tudo isso permitindo gerar ambientes tridimensionais interativos em tempo real através de realidade virtual.

A interação do utilizador com o ambiente virtual é um dos aspectos importantes da interface e está relacionada com a capacidade do computador detetar as ações do usuário e reagir instantaneamente. A grande vantagem deste conceito é que permite que as habilidades e os conhecimentos intuitivos do utilizador possam ser utilizados para a manipulação dos objetos não virtuais.

Assim, como o nome indica, a realidade virtual permite ao utilizador interagir com situações imaginárias envolvendo objetos virtuais sendo eles estáticos ou em movimento. Permite também reproduzir com fidelidade ambientes da vida real de forma a que o utilizador possa experienciar esses ambientes. [1]

2.2.1 Cronologia da Realidade Virtual

Ao longo da evolução da realidade virtual houveram algumas datas mais marcantes que permitiram ao ser humano uma evolução acentuada e uma expansão do seu conhecimento.

Em **1952** surgiu a primeira máquina de realidade virtual criada pelo cinematógrafo Morton Heilig. Aplicando o seu conhecimento em cinema e com a ajuda do seu parceiro, desenvolveu o *Sensorama*. Neste máquina era possível jogar um jogo onde se conduziu uma moto nas cidades de *Brooklyn*, o jogador conseguia sentir o vento na sua cara, as vibrações do banco da mota e, uma visão 3D e o cheiro da cidade. Acabou por ser patentiada 10 anos mais tarde em **1962**.



Figura 2.2: Sensorama desenvolvido por Morton Heilig

Cinco anos mais tarde, em **1968**, Ivan Sutherland, informático dos Estados Unidos, desenvolveu o primeiro *Head-Mounted-Display*, *The Sword of Damocles*, um dispositivo *display* que se usava na cabeça como um capacete ou parte de um capete e que tinha uma ótica, caso fosse monocular, ou duas óticas caso

fosse binocular. O desenvolvimento deste **HMD** tinha como principal objetivo facilitar o trabalho de pilotos de aviões, engenheiros e até ajudar na medicina.

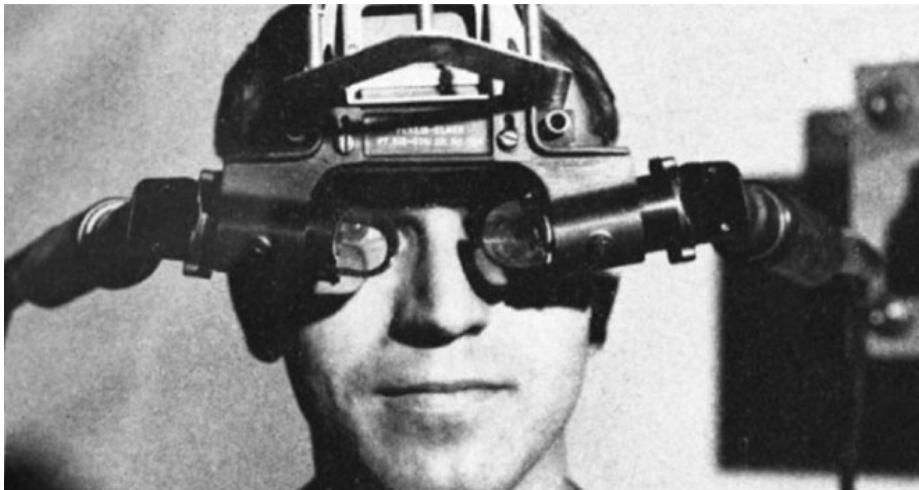


Figura 2.3: Head-Mounted-Display

A room within which the computer can control the existence of matter. [8]
Entre **1977** e **1982** foi criada a primeira luva digital pela **Universidade de Illinois**, projeto denominado de "Sayre Glove". Uma das primeiras aplicações no mundo real desta invenção foi no filme *Star Wars*.

Em **1980** o termo Realidade Virtual foi patenteado por Jaron Lanier, artista e cientista da computação que conseguiu convergir dois conceitos antagónicos em um novo e vibrante conceito capaz de captar a essência dessa tecnologia: a procura pela fusão do real com o virtual.

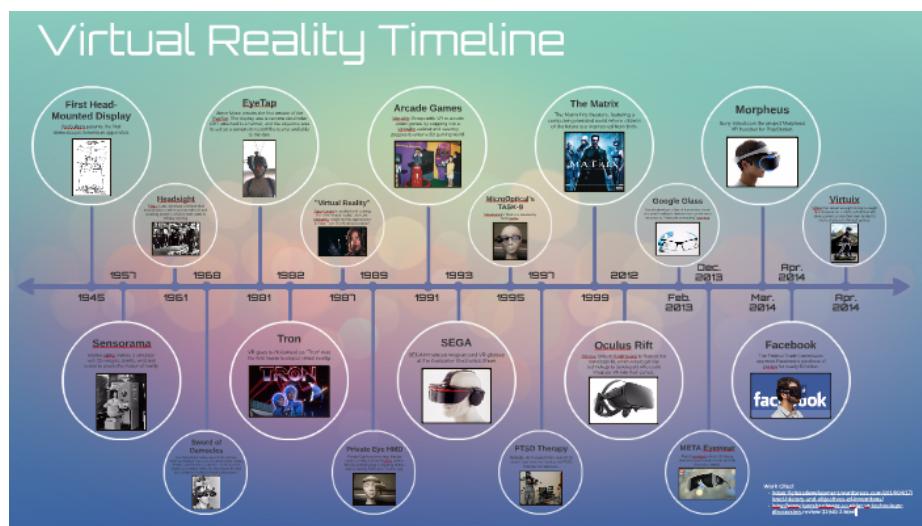


Figura 2.4: Cronologia da realidade virtual

2.3 Realidade Aumentada

A realidade aumentada beneficiou-se do avanço da realidade virtual e do desenvolvimento dos computadores de forma a permitir que fosse possível sobrepor elementos virtuais à nossa visão da realidade.

Ao contrário da realidade virtual, que transporta o utilizador para o ambiente virtual, a realidade aumentada mantém o utilizador no seu ambiente físico e transporta o ambiente virtual para o espaço do utilizador, permitindo a interação com o mundo virtual de forma mais natural e sem necessidade de adaptação. [1]

Existem vários conceitos e aspectos importantes que estão relacionados com a realidade aumentada, conceitos esses que irão ser abordados de seguida.

2.3.1 Tipos de Realidade Aumentada

Existem diferentes conceitos de realidade aumentada e cada um desses conceitos é adequado para uma determinada utilização, no entanto todos os conceitos têm características em comum.[1] Assim, a principal diferença é entre:

- Realidade aumentada baseada em marcadores.
- Realidade aumentada sem marcadores.

2.3.2 RA baseada em marcadores

As aplicações desenvolvidas com realidade aumentada baseada em marcadores utilizam imagens alvo (*targets*) para posicionar objetos num determinado espaço. Estes *targets* informam a aplicação onde serão colocados os objetos 3D dentro do campo de visão do utilizador.[5]



Figura 2.5: RA baseada em marcadores

2.3.3 RA sem marcadores

A realidade aumentada sem o uso de marcadores permite que objetos virtuais 3D sejam posicionados no ambiente de imagem real ao examinar as características presentes nos dados em tempo real. Este tipo de orientação baseia-se no hardware de qualquer *smartphone*, seja a câmara, GPS ou acelerômetro, entre outros, enquanto o software de realidade aumentada completa o trabalho.



Figura 2.6: RA sem marcadores

Assim, existem quatro categorias de RA sem marcadores:

- RA baseado na localização;
- RA baseado em projeções;
- RA baseado na sobreposição;
- RA baseado no contorno.

RA baseado na localização

Esta tecnologia utiliza a localização e os sensores de um dispositivo inteligente para posicionar o objeto virtual no local ou ponto de interesse desejado.[6] O

exemplo mais representativo deste tipo de realidade aumentada é o jogo Pokémon Go, que utiliza realidade aumentada baseado na localização, dando imediatamente vida ao ambiente do utilizador.



Figura 2.7: Pokemon Go - RA baseado na localizaçāo

RA baseado em projeções

A realidade virtual baseada em projetos centra-se na renderização de objetos virtuais 3D dentro de espaço físico do utilizador. Permite ao utilizador mover-se livremente no ambiente de uma área específica, onde se coloca um projetor fixo e uma câmara de localização.[4] A maior utilidade desta tecnologia é criar ilusões sobre a profundidade, posição e orientação de um objeto, projetando luz artificial sobre superfícies planas reais.

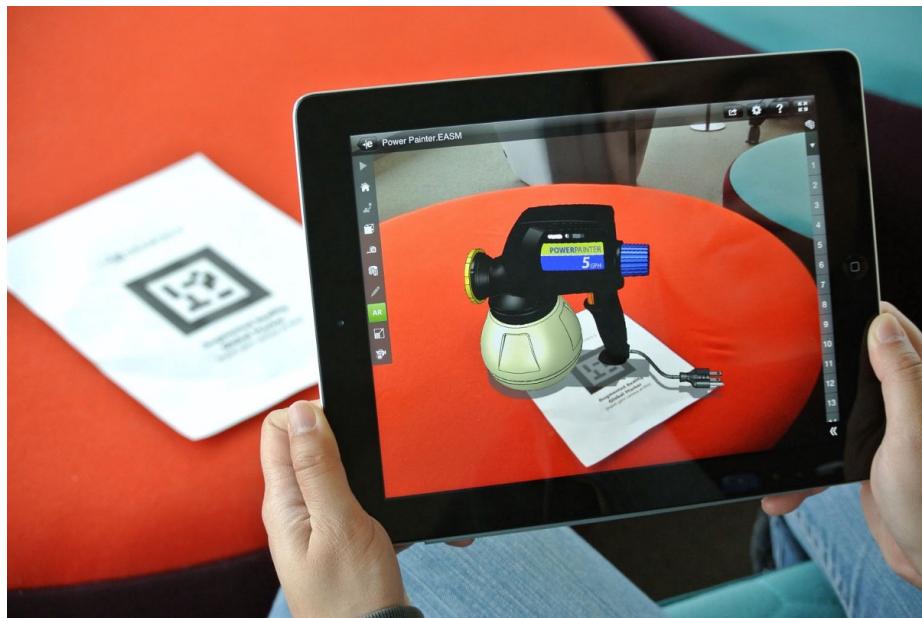


Figura 2.8: Pokemon Go - RA baseado em projeções

RA baseado na sobreposição

Este tipo de realidade aumentada é utilizado para substituir a visão original de um objeto por uma imagem virtual atualizada desse objeto para o olho humano.[7]



Figura 2.9: RA baseado em sobreposição

RA baseado no contorno

Esta tecnologia, com apoio de câmaras especiais, é utilizada essencialmente para os olhos humanos delinearem objetos específicos com linhas para facilitar certas situações como por exemplo, pode ser utilizado para sistemas de navegação automóvel para permitir uma condução segura em situações de baixa visibilidade.[2]

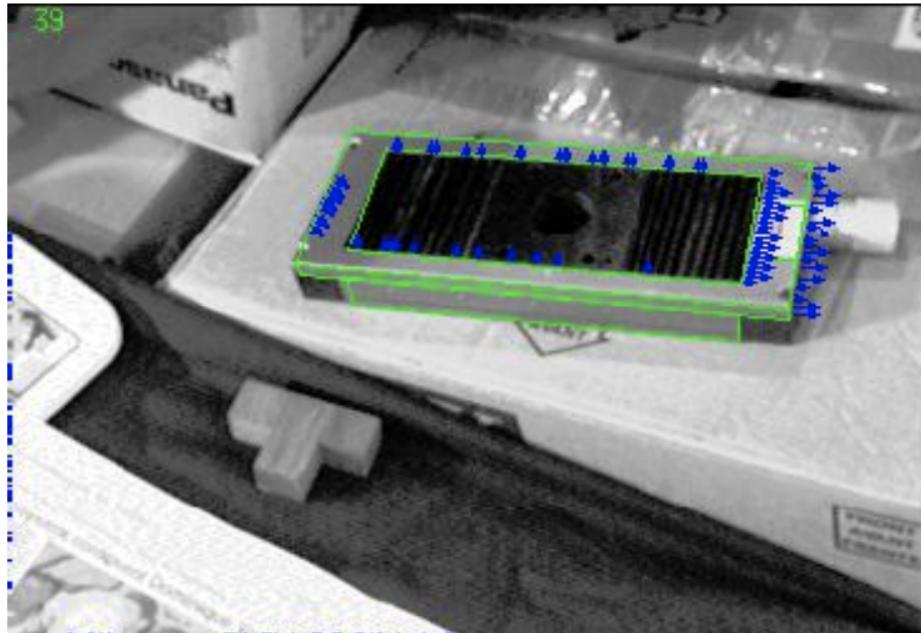


Figura 2.10: RA baseado no contorno

2.3.4 Realidade Mista

Within this [reality-virtuality] framework it is straightforward to define a generic Mixed Reality (MR) environment as one in which real world and virtual world objects are presented together within a single display. [3]

A realidade mista pode ser definida como a sobreposição de objetos virtuais gerados por computador com o ambiente físico, com o apoio de algum dispositivo tecnológico, em tempo real. O presente projeto é a definição de isso mesmo, onde é utilizada a *webcam* do *smartphone* de forma a ser possível visualizar objetos tridimensionais.

Assim, ao misturar cenas reais com virtuais, a realidade mista vai para além

da capacidade da realidade aumentada de reproduzir o mundo real. A realidade mista junta elementos virtuais ao ambiente real ou leva elementos reais ao ambiente virtual.

Para ajudar a entender este conceito de realidade mista, Milgram criou um diagrama em 1994, diagrama de Milgram.

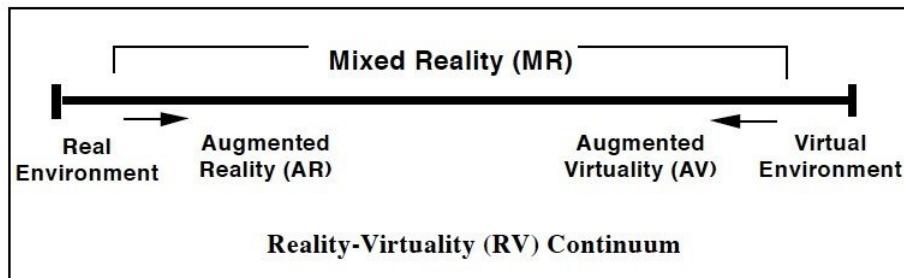


Figura 2.11: Diagrama de Milgram

A realidade aumentada e a virtualidade aumentada são casos particulares da realidade mista, no entanto o conceito realidade aumentada é usado de uma forma muito mais abrangente.

A realidade aumentada usa técnicas computacionais que geram, posicionam e mostram objetos virtuais integrados ao cenário real, enquanto a virtualidade aumentada usa técnicas computacionais para capturar elementos reais e reconstruí-los, como objetos virtuais realistas, colocando-os dentro de mundos virtuais e permitindo a sua interação com o ambiente.

Uma forma de sintetizar as diferenças entre realidade virtual e realidade aumentada é a seguinte:

- A realidade virtual trabalha unicamente com o mundo virtual, transfere o utilizador para o ambiente virtual.
- Realidade aumentada possui um mecanismo para combinar o mundo real com o mundo virtual, mantém o senso de presença do utilizador no mundo real.

2.4 Especificação do Requisitos

Para o desenvolvimento deste projeto foi necessário escolher uma analogia na qual estivesse evidente uma mensagem relacionada com o Pensamento Computacional. Sendo assim, o objetivo da aplicação será mostrar ao utilizar que a partir de um input é possível obter diferentes outputs.

Assim, a analogia presente neste projeto é a de uma aplicação que permite seguir todos os passos uma receita específica e chegar a um resultado final.

Para concretizar a ideia desta analogia, no próximo capítulo, será feita a demonstração da aplicação em funcionamento tal como a relação que existe entre o este projeto e o pensamento computacional.

Capítulo 3

Conceção da Proposta

3.1 Proposta de recursos de aprendizagem

De modo a integrar o desenvolvimento de Pensamento Computacional e a conseguir que o utilizador se mantenha cativado, foi decidido criar o jogo que, mais tarde, acabaria por se tornar no *FruitCombo*. Porém, apenas criar um jogo não era o objetivo deste projeto, mas sim criar uma ferramenta que ajudasse a treinar o Pensamento Computacional de, maioritariamente, crianças de ensino primário, mas que pudesse ajudar público de todas as idades. Para tal, tivemos o cuidado de nos focar em desenvolver os grupos mencionados anteriormente.

- **Programação:** Na sua base, um programa é um conjunto de instruções que descrevem uma tarefa a ser realizada, para tal, usamos cartas (ou targets) de modo a introduzir estas instruções ao utilizador. Cada uma das cartas simboliza uma instrução do programa final, que, neste caso, é um programa um produto alimentar (bolo, compota ou sumo) através dos targets usados; instruções usadas para obter o produto final;
- **Abstração:** Usamos abstração de modo a mostrar que o Pensamento Computacional não é usado apenas para programar, isto é, criar um dos produtos alimentares possíveis requer que o utilizador use um algoritmo a partir das ferramentas disponibilizadas, decomponha o problema em mãos de forma a atingir o resultado pretendido, pense no próximo passo para chegar mais perto da solução, etc ... ou seja, estes processos não estão ligados apenas à programação mas sim ao dia-a-dia de todas as pessoas;
- **Reconhecimento de Padrões:** O reconhecimento de padrões está presente, por exemplo, nas pequenas figuras dentro dos targets. Estas figuras estão presentes para que, mesmo utilizadores que não consigam ler, possam utilizar esta aplicação;
- **Algoritmos:** De modo a criar qualquer um dos produtos alimentares possíveis, é necessário seguir que o utilizador siga um número predeterminado

de passos, esses passos alteram de modo a atingir uma solução diferente. Esse passos constituem um algoritmo;

- **Avaliação:** Antes de escolher um target o utilizador tem que avaliar se foi a melhor escolha de forma a conseguir alcançar o produto desejado, e, depois de fazer essa escolha, avaliar se não haveria escolha melhor;
- **Decomposição:** Tendo um objetivo em mente, neste caso escolhendo um dos produtos possíveis de criar, decompor a criação desse produto nos passos que constituem o algoritmo;
- **Lógica:** Previsão do target a ser usado dependendo do produto pretendido e de targets usados anteriormente, caso já tenham sido utilizados.

3.2 Funcionamento da Aplicação

Na implementação desta aplicação foi optado pelo grupo em conjunto com os 2 professores em desenvolver uma aplicação onde, a partir de o mesmo input, seria possível atingir várias outputs diferentes. Para demonstrar isto, a analogia escolhida foi:

- 1^a etapa: escolher uma peça de fruta.
- 2^a etapa: escolher um utensílio.
- 3^a etapa: escolher os ingredientes.
- 4^a etapa: observar o output obtido.

Cada um destes inputs é representado por um **qrcode** específico de forma a que sejam reconhecidos pela câmara do telemóvel e sejam associados a um e um só objeto.

3.2.1 Targets

Frutas

As frutas presentes na aplicação serão representadas pelos seguintes targets.



Figura 3.1: Targets para representar as frutas.

Utensílios

Os utensílios disponíveis na aplicação serão representados pelos seguintes targets.



Figura 3.2: Targets para representar os utensílios.

Ingredientes

Os ingredientes disponíveis na aplicação serão representados pelos seguintes targets.



Figura 3.3: Targets para representar os ingredientes.

Resultado

Para obter o resultado existe apenas um target que recolhe o output final.



Figura 3.4: Target para representar o resultado final

3.2.2 Objetos para cada target

Como foi referido anteriormente cada target irá representar um objeto. De seguida irão ser apresentados cada um desses objetos.

Frutas

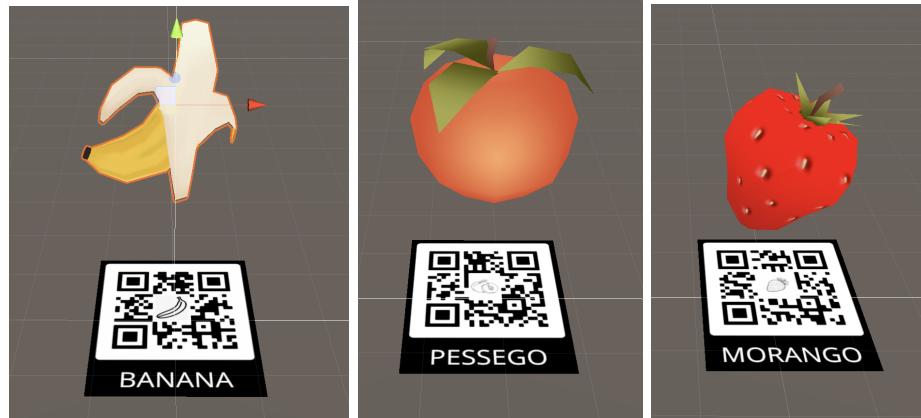


Figura 3.5: Objects 3D que representam as frutas.

Utensílios

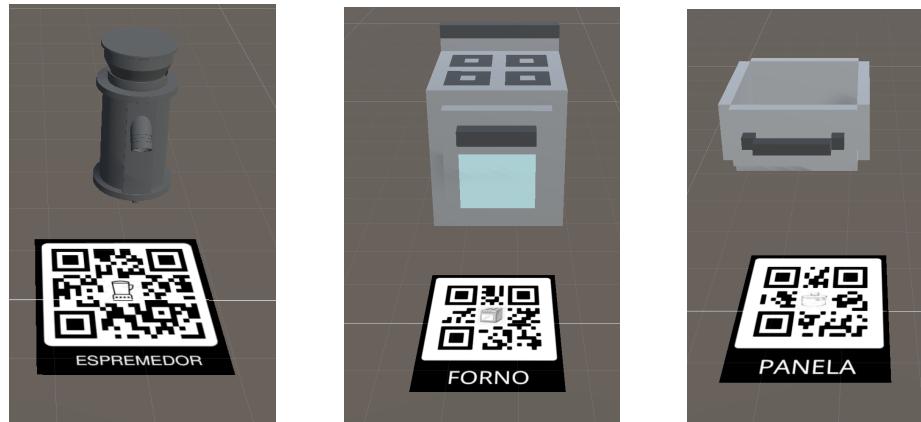


Figura 3.6: Objects 3D que representam os utensílios.

Ingredientes

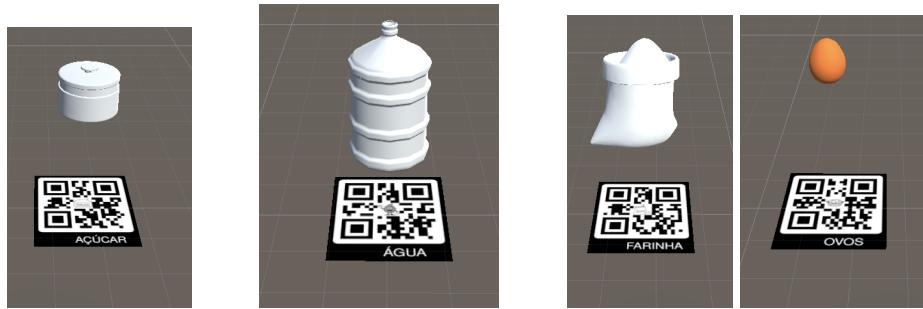
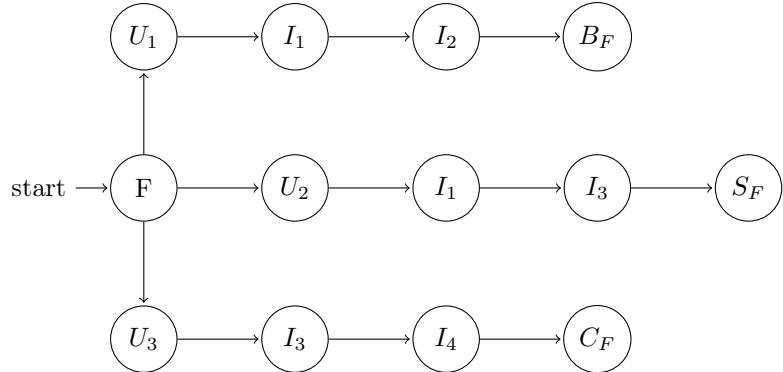


Figura 3.7: Objects 3D que representam os ingredientes.

3.2.3 Receitas

No desenvolvimento desta aplicação foram pré definidas três receitas de modo a que o utilizador necessite de usar pelo menos 5 targets para chegar ao resultado final assim, estas receitas irão ser representadas por um autómato onde,

- F representa todas as frutas.
- U_1 representa o Forno.
- U_2 representa o Espremedor.
- U_3 representa o Forno.
- I_1 representa o Açúcar.
- I_2 representa o Farinha.
- I_3 representa o Água.
- I_4 representa o Ovo.
- B_F representa o Bolo da fruta.
- S_F representa o Sumo da fruta.
- C_F representa a Compota da fruta.



Capítulo 4

Desenvolvimento do sistema

4.1 Alternativas, Decisões e Problemas de Implementação

Antes mesmo da escolha do produto que viríamos a desenvolver, tivemos que pensar num sistema a usar para criar o nosso futuro jogo. Apesar de haver vários sistemas que nos permitissem criar a nossa aplicação, a escolha do *Unity* com a biblioteca *Vuforia* foi principalmente relacionada com a estabilidade que o *Unity* proporciona para a leitura dos marcadores e a facilidade que existe em interagir com os mesmos.

Usamos a versão de 2018, pois, as versões que se seguiram causavam problemas ao criar o *apk* para sistemas *IOS* e para evitar tais problemas optamos pela versão mais antiga, visto que, em termos de performance, não havia diferenças que valham a pena denotar.

Durante o desenvolvimento da aplicação deparamo-nos com algumas dificuldades que fomos ultrapassando com resiliência e também com auxílio dos professores. De seguida irão ser apresentados os algoritmos que se destacam após a finalidade deste projeto.

Aqui é possível verificar uma parte da implementação na nossa máquina de estados, quando é escolhido um marcador de uma banana e outro de um forno vamos acabar por obter um bolo de banana.

```
switch (fruit.name)
{
    case "Banana":
        switch (utens.name)
        {
            case "Forno":

                var bananaCake =
                    GameObject.Find("Resultado").transform;

                bananaCake.Find("BananaCake").gameObject.SetActive(true);

                break;

            case "Panela":

                var bananaJam =
                    GameObject.Find("Resultado").transform;
                bananaJam.Find("BananaJam").gameObject.SetActive(true);

                break;

            case "Espremedor":

                var bananaJuice =
                    GameObject.Find("Resultado").transform;
                bananaJuice.Find("BananaJuice").gameObject.SetActive(true);
                break;
        }
        break;
}
```

No entanto, este *switch case* irá ter vários outros casos para as restantes frutas.

A Porção do código seguinte representa o conjunto de verificações necessárias para que o utilizador siga as regras base para que o jogo funcione.

```
if (checkTrackableQtf(activeTrackables)){

    erTra.Unable(errottrackables);

    if (ntrackables < 5){
        foreach (TrackableBehaviour tb in
            activeTrackables){

            if (checkTrackableFruit(tb) && ntrackables
                == 0){
                listOfActiveTrackables.Add(tb);
                f.Unable(fruitFirst);
                ntrackables++;
                ok.Enable(objectOkButton);
                okbutton.onClick.AddListener(() => {
                    okclicked = true;
                    ok.Unable(objectOkButton); });
            }

            else if (!checkTrackableFruit(tb) &&
                ntrackables == 0){
                    f.Enable(fruitFirst);
                }

            else if (checkTrackableUtens(tb) &&
                ntrackables == 1 && okclicked){
                okclicked = false;
                listOfActiveTrackables.Add(tb);
                u.Unable(utenSecond);
                ntrackables++;
                ok.Enable(objectOkButton);
                okbutton.onClick.AddListener(() =>
                { okclicked = true;
                    ok.Unable(objectOkButton); });
            }

            else if (!checkTrackableUtens(tb) &&
                ntrackables == 1 && okclicked){
                    u.Enable(utenSecond);
                }
        }
    }
}
```

```

else if (checkTrackableIngredientes(tb)
&& (ntrackables == 2 || ntrackables == 3)
&& okclicked){
    string utensilio =
        listOfActiveTrackables[1].name;
    rpIngre.Unable(repeatedIngredient);
    wrongIngre.Unable(wrongIngredient);
    chooseIngre.Unable(chooseIngredient);

    if
        (receitas[utensilio].Contains(tb.name)
&&!listOfActiveTrackables.Contains(tb)){
            okclicked = false;
            ntrackables++;
            ok.Enable(objectOkButton);
            listOfActiveTrackables.Add(tb);
            okbutton.onClick.AddListener(() =>
            { okclicked = true;
                ok.Unable(objectOkButton); });

        }
    else if
        (listOfActiveTrackables.Contains(tb))
    {
        rpIngre.Enable(repeatedIngredient);
    }
    else if
        (!receitas[utensilio].Contains(tb.name))
    {
        wrongIngre.Enable(wrongIngredient);
    }
}
else if (!checkTrackableIngredientes(tb)
&& (ntrackables == 2 || ntrackables == 3)
&& okclicked)
{
    Debug.Log("Entrou");
    chooseIngre.Enable(chooseIngredient);
}
else if (tb.name == "Resultado"
&& ntrackables == 4 && okclicked)
{
    ntrackables++;
}
Debug.Log(ntrackables);
}

```

```
        }
    }
else
{
    ok.Unable(object0kButton);
    erTra.Enable(errotrackables);
}

if(ntrackables == 5)
{
    st.fsm(listOfActiveTrackables);
}
```

Capítulo 5

Apresentação do sistema

5.1 Testes realizados e Resultados

Neste capítulo irão ser demonstrados alguns testes com o intuito de mostrar o funcionamento da aplicação. Inicialmente irá ser mostrado todo o processo até formar um bolo de banana.

Receita bolo de banana

Começa-se por escolher a peça de fruta.



Figura 5.1: Leitura da fruta

Depois escolhe-se o utensílio.



Figura 5.2: Leitura do utensílio

Posteriormente escolhem-se os dois ingredientes.



Figura 5.3: Leitura do primeiro ingrediente



Figura 5.4: Leitura do segundo ingrediente

Por fim, obtém-se o resultado.



Figura 5.5: Leitura do resultado obtido

Erros na escolha dos marcadores

Este projeto foi implementado de forma a que o primeiro marcador a ser lido tenha que ser obrigatoriamente um de fruta, caso contrário obtém-se um erro.



Figura 5.6: Erro na escolha de fruta

O mesmo se aplica quando o utilizador tem de escolher um utensílio e utiliza um marcador que não é um utensílio.



Figura 5.7: Erro na escolha de utensílio

Outra ocasião onde o utilizador é informado que tem de escolher outro marcador é na escolha dos ingredientes, visto que cada receita tem os ingredientes específicos então, quando é escolhido um ingrediente que não pertence a essa receita, um erro aparece na tela para informar o utilizador que tem de mudar de marcador.

No seguinte caso o objetivo era fazer um bolo onde os ingredientes necessários são **Açucar** e **Farinha**.



Figura 5.8: Erro na escolha do ingrediente

Capítulo 6

Conclusão

Com a conclusão deste projeto é possível afirmar que o grupo concretizou o desafio esperado: implementar uma aplicação cujo o objetivo é introduzir o Pensamento Computacional aos mais jovens de forma a introduzi-los na programação.

A realização deste projeto contribuiu para que os elementos do grupo entendessem como funciona o *Unity* e o *Vuforia*. Os desafios que fomos confrontando ao longo do desenvolvimento do projeto contribuíram para a consolidação e a progressão do conhecimento.

Consideramos que a aplicação desenvolvida permite ao utilizador chegar a vários *outputs* a partir do mesmo *input* o que, seria o principal objetivo deste projeto pois permite ao utilizador perceber que o resultado final irá sempre depender do conjunto de instruções escolhido.

Em suma, todos os elementos estão satisfeitos com o trabalho que foi feito ao longo do semestre visto que o resultado é uma aplicação *android* prática e funcional.

No futuro, poderão ser inseridas mais opções tanto de frutas como de utensílios e ingredientes o que, consequentemente, aumentaria o número de marcadores da aplicação e permitiria ao utilizar ter um maior número de combinações. Seria interessante investir algum tempo na interface gráfica de modo a tornar a aplicação mais apelativa e melhor a comunicação entre a aplicação e o utilizador.

Bibliografia

- [1] Ronald T. Azuma. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments*, 6(4):355–385, 08 1997.
- [2] M.-O. Berger. Resolving occlusion in augmented reality: a contour based approach without 3d reconstruction. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 91–96, 1997.
- [3] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.
- [4] Mark R. Mine, Jeroen van Baar, Anselm Grundhofer, David Rose, and Bei Yang. Projection-based augmented reality in disney theme parks. *Computer*, 45(7):32–40, 2012.
- [5] Mohd Khalid Mokhtar, Farhan Mohamed, Mohd Shahrizal Sunar, Ahmad Ashraf Abd Aziz, Mohd Azhar M. Arshad, and Mohd Kufaisal Mohd Sidik. Image features detection and tracking for image based target augmented reality application. In *2019 IEEE Conference on Graphics and Media (GAME)*, pages 38–43, 2019.
- [6] Rémi Paucher and Matthew Turk. Location-based augmented reality on mobile phones. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 9–16, 2010.
- [7] Susanne Stadler, Kevin Kain, Manuel Giuliani, Nicole Mirnig, Gerald Stollberger, and Manfred Tschelegi. Augmented reality for industrial robot programmers: Workload analysis for task-based, augmented reality-supported robot control. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 179–184, 2016.
- [8] Ivan Sutherland. The ultimate display. 1965.