

META 02

RELATÓRIO DO PROJETO LICA

Livros Interativos para Crianças Aprenderem

Trabalho realizado por Gabriela Nogueira nº2021176188 e Jorge Antunes
nº2022136207 em contexto académico para a cadeira de Tecnologias de Interface, leccionada pelo professor Sérgio Rebelo do Mestrado de Design e Multimédia da Universidade de Coimbra

01 INTRODUÇÃO

O presente relatório tem como intenção descrever o conceito do projeto, detalhar todas as etapas da metodologia de processo usando imagens e esboços para clarificar o referido no texto, como também, explicar a estratégia de implementação e todo o desenvolvimento que foi realizado para chegar a um produto físico e funcional. Serve como forma de planeamento de todo o trabalho que foi realizado para dar vida ao nosso conceito de livro interativo para crianças aprenderem.

02 ÍNDICE

02 Introdução

03 Descrição do conceito

**04 Etapas de Metodologias de Design
de Interação**

4.1 User needs and wants

4.2 Analysis

4.3 Design da interface e Design do Produto

4.3 Prototipagem

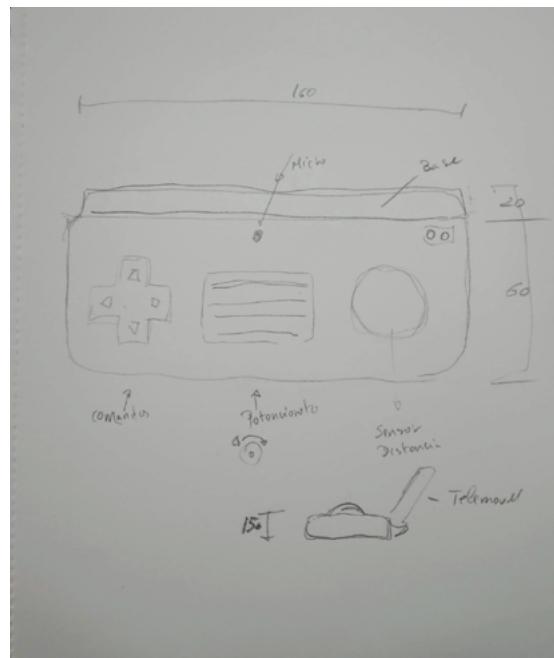
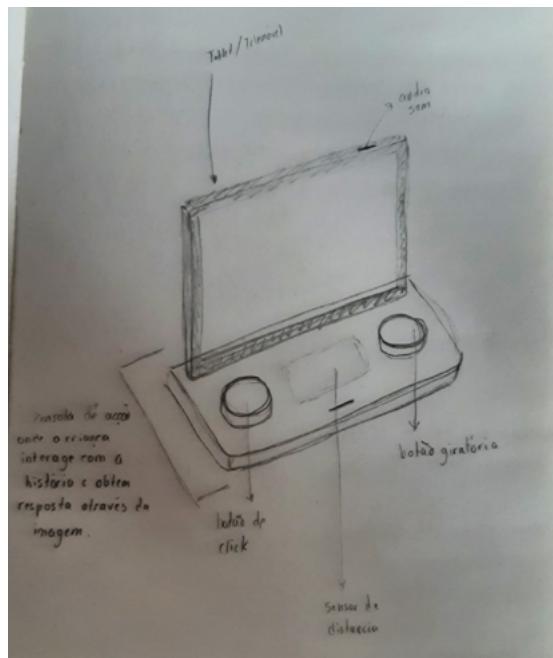
4.3 Implementação e Desenvolvimento

4.3 Teste e Feedback

05 Conclusão

03 DESCRIÇÃO DO CONCEITO

O conceito do nosso projeto surge do desejo em criar um livro interativo que promova a aprendizagem das crianças sobre vários tópicos importantes da nossa sociedade, como é o caso do tema da reciclagem que, apesar de ser já frequentemente destacado, ainda não se encontra enraizado nos nossos hábitos diários e é muitas vezes ignorado. Assim, considerámos a história infantil dos três porquinhos - conhecida por todas as crianças e adultos - e reinventámos o conceito do livro para esta temática, de forma a melhor adequar-se aos tempos digitais.



Assim, o projeto consiste numa aplicação para tablets que disponibiliza um conjunto de livros interativos, onde a criança pode ler e interagir com as histórias e aprender de forma mais lúdica. A aplicação será possível de descarregar através de um *QRcode*, associado à consola de interação, possível de ser adquirida em lojas aderentes ao projeto. Nesta consola existe uma entrada de ligação para o tablet e componentes de inputs com os quais a criança pode interagir ao longo da história.

A história “Os 3 Porquinhos & Reciclagem” é uma das inseridas nesta aplicação - LICA - e é através dela que damos vida ao conceito de livro interativo, do seu design, das ideias de interação, da implementação e do seu desenvolvimento.



Livros Interativos para Crianças Aprenderem



imagens 03 e 04
Imagen de marca da aplicação LICA e da história dos 3 porquinhos e a Reciclagem



imagen 05
Aplicação de Livros Interativos para Crianças Aprenderem num contexto real idealizado



imagen 06
QR code associado à consola para descarregar mais facilmente a aplicação.

04 ETAPAS DE METODOLOGIAS DE DESIGN DE INTERAÇÃO

Um dos fatores determinantes para a qualidade de um produto ou serviço é a metodologia de processo utilizada, que envolve diversas etapas até à obtenção da solução final, que lhe confere estrutura e organização, de forma a facilitar o desenvolvimento do produto.

O processo por nós considerado partiu da necessidade de criar uma interface tecnológica para as crianças e para a sua aprendizagem - para isso utilizámos como referência o processo de design de interação da *Interaction Design Foundation*, constituído por 5 etapas. Para além destas, acrescentámos mais um passo, considerado por nós fundamental, que corresponde à testagem e à recolha de feedback. Desta forma, torna-se possível analisar, definir melhorias e aplicar mudanças no protótipo até que este responda, da melhor forma possível, às necessidades das crianças.

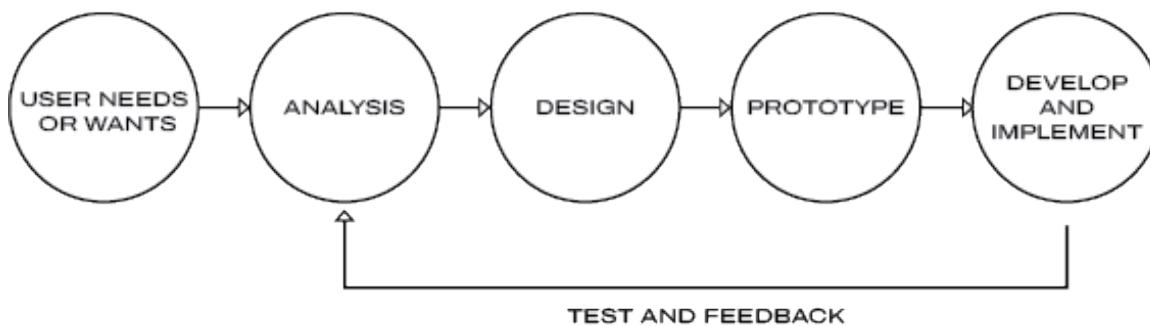


imagem 07
imagem criada e inspirada no processo de design de interação da Interaction Design Foundation

<https://www.interaction-design.org/literature/topics/interaction-design-process>

4.1 User Needs or Wants

Nesta primeira fase de investigação, tentámos identificar e compreender as necessidades das crianças, tendo descoberto que:

- Existe uma área de HCI dedicada somente às crianças, denominada por Child Computer Interaction;
- Existem quatro diferentes fases de aprendizagem cognitiva da criança que, segundo Jean Piaget, são: a fase Sensorimotor, a fase Pre-Operational, a fase Concrete Operational e fase Formal Operational;
- O desenvolvimento das capacidades motoras difere de criança para criança.
- As crianças aprendem melhor através de imagens e materiais didáticos.

Estas conclusões foram obtidas através da nossa análise de vários projetos, em diversas áreas, cuja finalidade é fomentar a aprendizagem das crianças.

4.2 Analysis

Depois da pesquisa e observação de casos de estudo já existentes, definimos como objetivo criar um livro interativo, com o qual as crianças possam brincar de forma livre e autónoma, mas também aprender. Assim, decidimos adotar como referência os livros Montessori, concordantes com o nosso conceito e objetivo, pois são livros coloridos, texturados e com pequenas ações que a criança pode realizar a fim de obter uma reação, visual ou sonora.

Ao longo de todo este processo, houve a necessidade de listar e organizar as tarefas a realizar, de forma a concretizar o projeto adequadamente, tendo em conta cada etapa de entrega. Por esta razão, foi criado um Diagrama de Gantt que nos permitiu delinear tempos de execução para cada uma das tarefas e a atribuição de cada uma, de forma a melhor organizar o projeto e cumprir os objetivos.



imagem 08
Diagrama Gantt para o planeamento das tarefas

Imagem anexada no final do documento em ponto maior

A estruturação das tarefas neste diagrama revelou-se extremamente importante, visto que nos possibilitou uma maior consciência dos tempos que cada tarefa precisava para serem realizadas, prazos de entrega e de compromisso entre os elementos do grupo.

4.3 Design da Interface

Como referido anteriormente, o nosso projeto passa pela ideia de reinventar o conceito do livro e da história dos três porquinhos e, por isso, decidimos que o primeiro passo seria a elaboração de um *storyboard* da nossa narrativa.

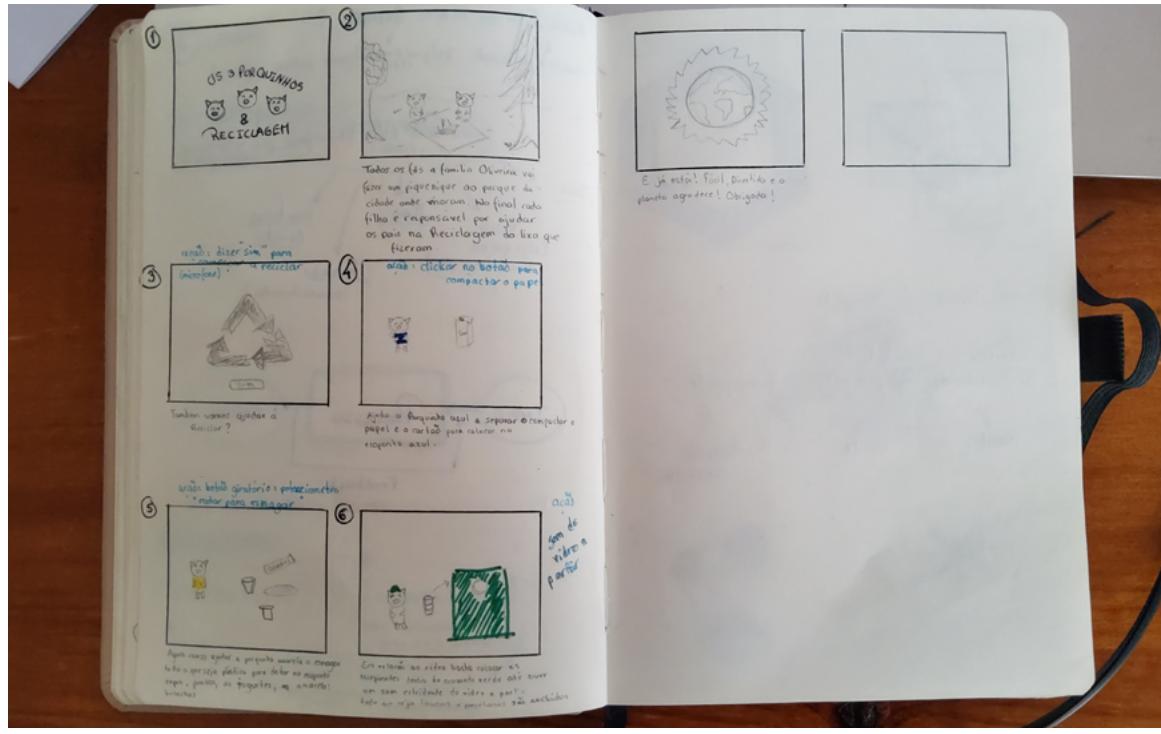


imagem 09
Storyboard da narrativa criada dos três porquinhos e a reciclagem.

Posteriormente, desenhámos as ilustrações necessárias e definimos a sua ordem, de forma a criar a história.

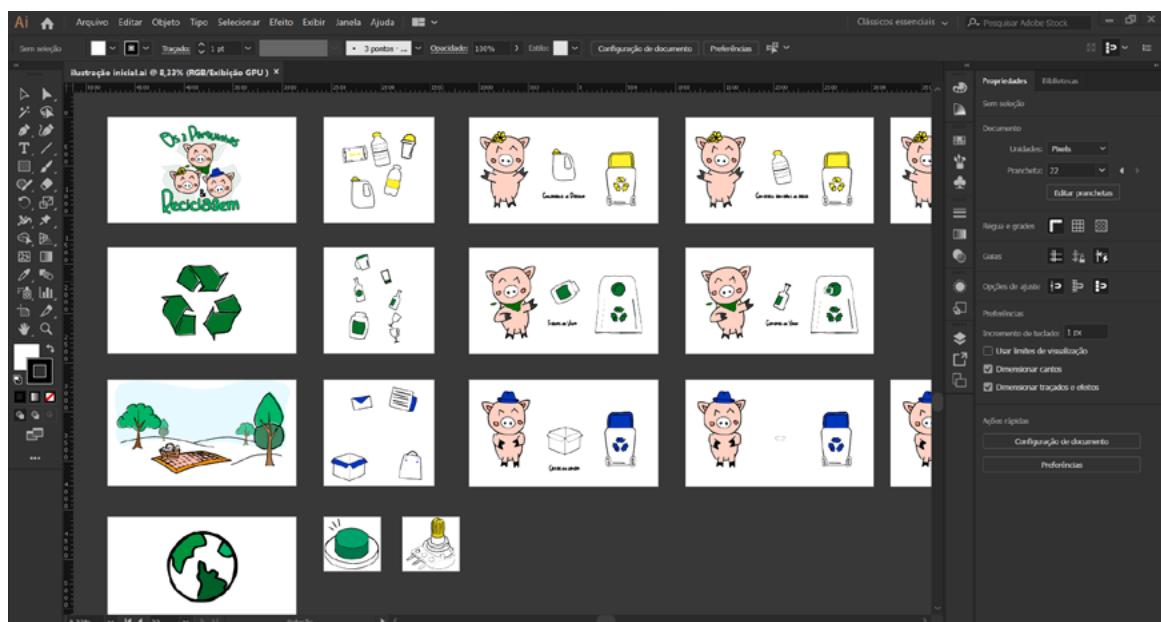


imagem 10
Ilustrações da história feitas no Adobe Illustrator

Tendo em conta os princípios de *CCI* e de Design de Interação descritos por *Don Norman*, começámos a definir meios para a criança interagir com a história criada, mas de forma a respeitar alguns destes princípios como:

- Visibilidade : Como é que a criança descobre as funções da interface?

Através de inputs, como botões e sensores visíveis e através do texto que acompanha a narrativa construída, ela saberá como e quando interagir com o artefacto.

- Feedback : Qual a resposta que queremos mostrar às crianças depois de interagir com os inputs? Através das ilustrações, a criança irá conseguir visualizar o efeito das suas ações com os componentes de interação.

- Mapping : Quais são as ações, dentro do sistema, que advém das interações?

Qualquer componente de interação, ao ser utilizado, deve acionar uma função dentro do sistema e, para isto acontecer, decidimos usar componentes de inputs como affordance.

Casos do nosso projeto:

- Botão de parar para accionar o som do vidro a estilhaçar;



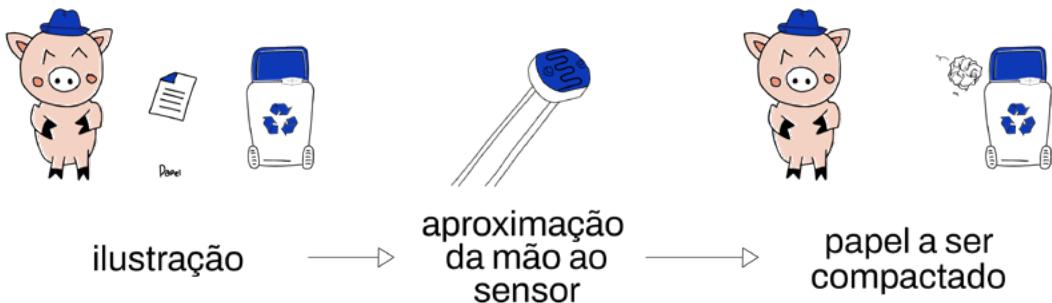
ilustração

click

som do vidro
a estilhaçar

imagem 11
Ilustrações da funcionalidade do botão

- Sensor LDR para accionar, visualmente, a compactação do papel/cartão;



ilustração

aproximação
da mão ao
sensor

papel a ser
compactado

imagem 12
Ilustrações da funcionalidade do sensor de distância

- Botão giratório/potenciômetro para accionar o girar da garrafa até ao contentor

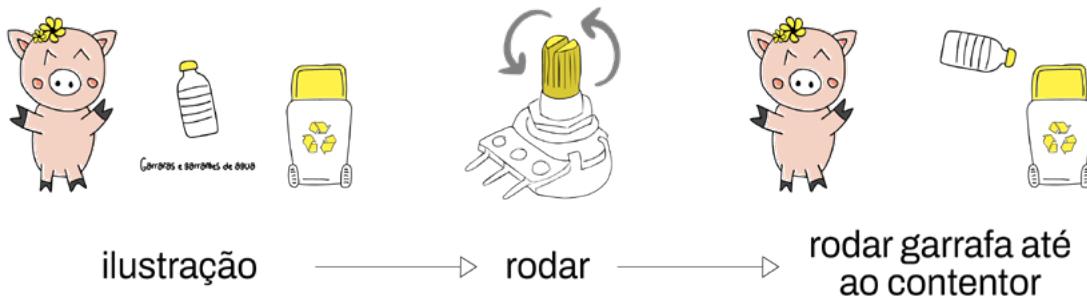


imagem 13
Ilustraçōe da funcionalidade do botāo potenciometro

- Restrições: Quais as consequências de ações incorrectas com o artefacto?

Os componentes só são funcionais durante a leitura da história, ou seja, quando a narrativa está na fase da reciclagem do vidro, apenas o botão está disponível e os outros formatos de interação do livro não. Para completar, adicionou-se um *buzzer* para dar sinal quando houver a interação de um componente não correspondente ao momento certo.

Para além das ilustrações, a história será acompanhada por texto legendado para que a criança possa ler a narrativa e assim perceber quando deve interagir com ela.

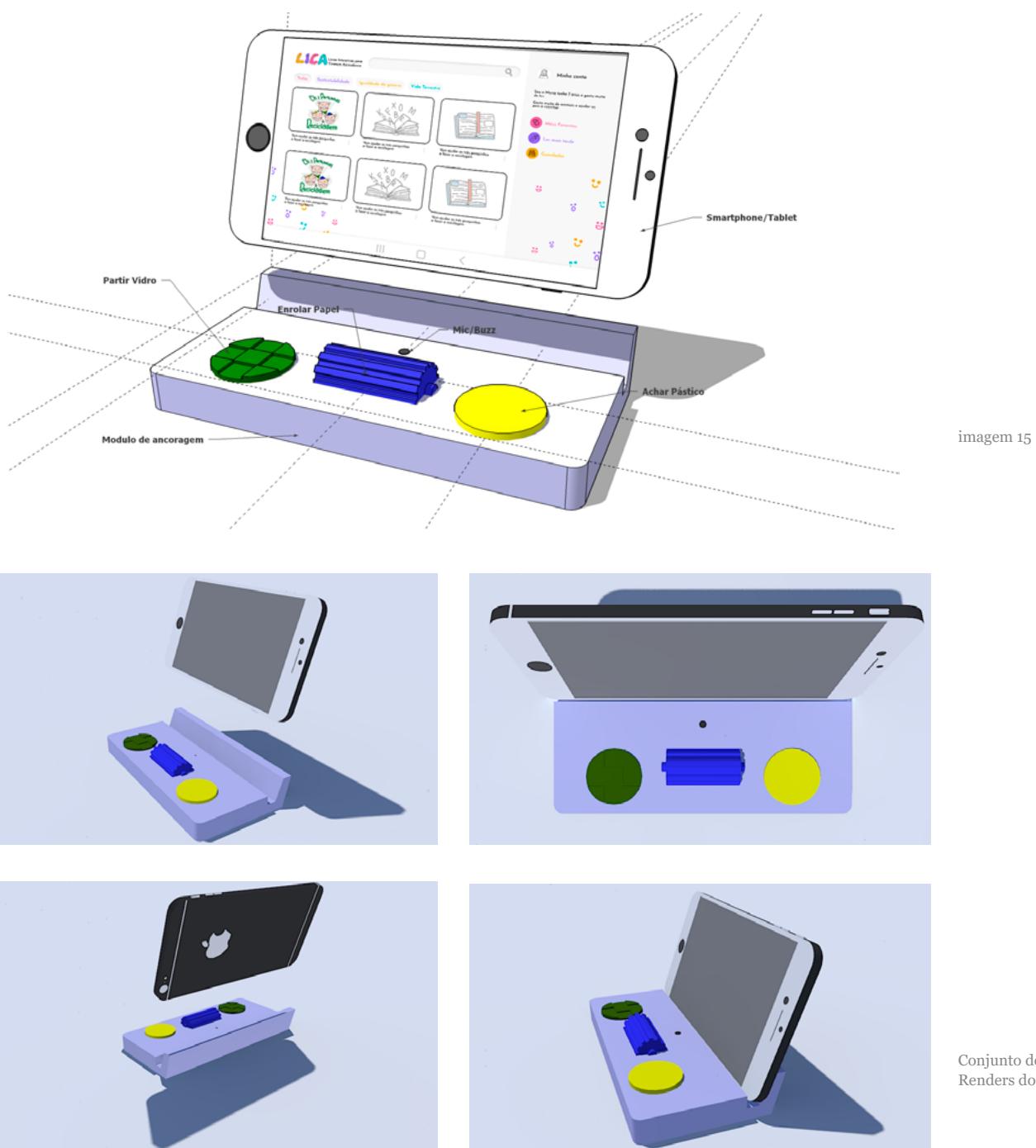


imagem 14
Exemplo da história com a ilustraçōe e o texto legendado com a descriçōe da ação.

4.3 Design do Produto

Da mesma forma que desenhámos as ilustrações da narrativa, criámos também toda a identidade da aplicação onde a história está inserida, bem como os esboços em 3D, para melhor demonstrar a nossa ideia para o artefacto.

O design inicial do artefacto constituía um apoio ao telemóvel ou tablet, capaz de comunicar com estes por bluetooth - desta forma, iria ter três botões de interação com o utilizador (onde cada um tem a cor correspondente à da interação apresentada na interface da narrativa), um microfone e um *buzzer*.



Conjunto de imagens 16
Renders do produto em 3D

4.4 Prototype

Depois da definição das ideias e do design, foram desenvolvidos os primeiros protótipos - com a ajuda do *software figma*, criámos um protótipo de alta fidelidade do fluxo da narrativa, com as ilustrações desenhadas, e da interação da aplicação.



link do protótipo inicial:

<https://www.figma.com/proto/rqDiF5Cx6mDZfXthe2Q86/os-3-porquinhos-e-a-Reciclagem?type=design&node-id=136-338&scaling=scale-down&page-id=0%3A1&starting-point-node-id=136%3A338>

De forma a melhor compreender a montagem do circuito em Arduino, foi criado um protótipo inicial no *tinkercad* do circuito e programado o código responsável pelas interações, para poder testar as funcionalidades.

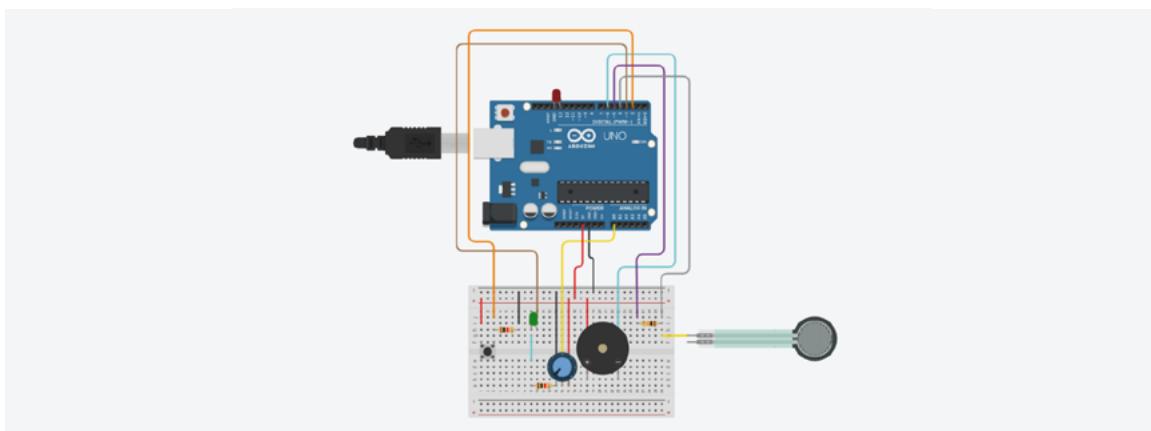


imagem 15
Esquema do circuito dos componentes

```

LivreInterativo_P1_Botao.ino
1 const int buttonPin = 2;
2 int buttonState = 0;
3
4 void setup() {
5   pinMode(buttonPin, INPUT);
6   Serial.begin(9600);
7 }
8
9 void loop() {
10   buttonState = digitalRead(buttonPin);
11   if (buttonState == HIGH) {
12     Serial.println("play_sound");
13     delay(100);
14 }
15 }

LivreInterativo_P1_Botao.pde
1 import processing.serial.*;
2 import processing.sound.*;
3
4 Serial serial;
5 SoundFile sound;
6
7 void setup() {
8   size(400, 400);
9   printArray(Serial.list()); //run this code to determine which port to use
10  serial = new Serial(this, Serial.list()[2], 9600);
11  sound = new SoundFile(this, "e_cair.wav");
12 }
13
14 void draw() {
15   background(255);
16   if (serial.available() > 0) {
17     String message = serial.readStringUntil('\n');
18     if (message != null & message.trim().equals("play_sound")) {
19       sound.play();
20       fill(255, 0, 0); // Define a cor vermelha para o círculo
21       ellipse(width/2, height/2, 50, 50); // Desenha um círculo no centro da tela
22     }
23   }
24 }
25
26
27
28
29
30
31
32
33
34

```

link do video da funcionalidade Botão click

<https://kdrive.infomaniak.com/app/share/723395/5a46e2ac-53d7-4108-boe4-0803beaff4f/preview/video/5458>

imagem 16
Código no Arduino e no process-ing para o botão

```

LivreInterativo_P1-Fotosensor.ino
1 //Arduino code to test photo sensor
2 //Eduardo Mark 3/23/17
3
4 int photocellPinB = 1; // the cell and 10K pulldown are connected to a0
5 int photocellReadings;
6
7 void setup() {
8   // initialize serial communication:
9   Serial.begin(9600);
10 }
11
12 void loop(){
13   photocellReadings = analogRead(photocellPinB); //read photosensor
14   Serial.print(" - Analog reading = ");
15   Serial.println(photocellReadings);
16 }

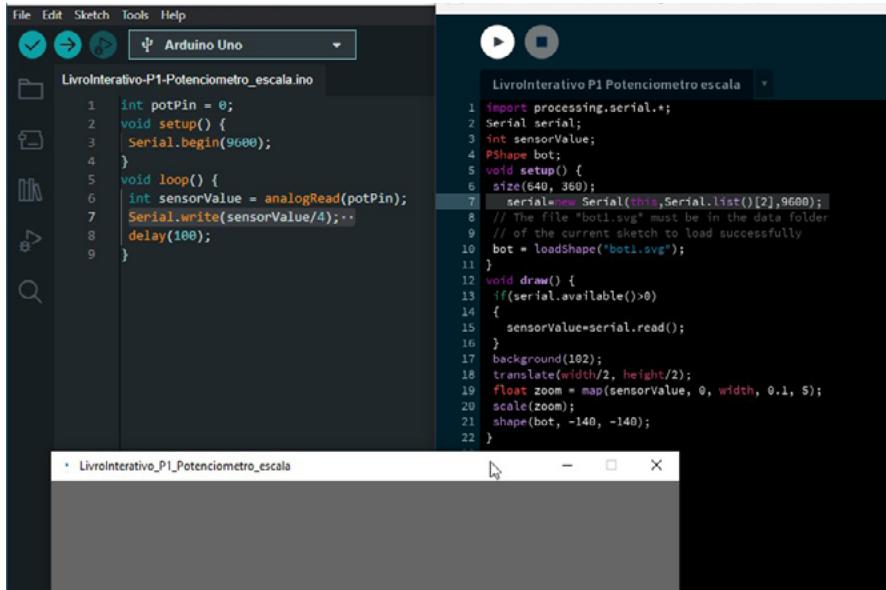
LivreInterativo_P1_Fotosensor.pde
1 import processing.serial.*;
2 import processing.core.*;
3
4 Serial serial;
5
6 void setup() {
7   size(200, 520);
8   String[] serialPortList; //list all available serial ports
9   int myPort; //int value from arduino
10  float np; //float value from arduino
11
12  serial = new Serial(this, Serial.list()[2], 9600);
13  np = serial.bufferUntil('t');
14
15  np = float(np);
16
17  if (np < 100) {
18    myPort = 1;
19  } else if (np > 100) {
20    myPort = 2;
21  }
22
23  background(150,200,255);
24
25  //read photosensor value from serial port
26  photosensorRead = split(npString, "+"); //subdivide text string at +
27  photosensorRead[0] = photosensorRead[0].substring(0, photosensorRead[0].length - 1);
28
29  //parse text value of photosensor
30  if (null != photosensorRead[0] && photosensorRead[0].length > 1) {
31    String num = photosensorRead[0];
32    np = float(num.substring(0, num.length - 1));
33  }
34
35  np = float(np);
36
37  randomSeed(0);
38  float scale = map(np, 0, 100, 0.5, 5.75);
39  np = width * np;
40
41  void owl(Flot x, Flot y, int g, Flot s) { // função que desenha pinguim
42    noStroke();
43    translate(x, y);
44    scale(s);
45    fill(255, 128, 0); // Set the color value
46    strokeWeight(10);
47    line(0, -25, 0, -45); // Body
48    noFill();
49    fill(255);
50    ellipse(-27.5, -45, 35, 35); // Left eye done
51    ellipse(27.5, -45, 35, 35); // Right eye done
52    arc(0, -45, 70, 70, 0, PI); // Chin
53    arc(0, -45, 40, 40, 0, PI);
54  }
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
99

```

link do video da funcionalidade LDR

<https://kdrive.infomaniak.com/app/share/723395/f76442a8-7boa-44c2-850e-43eob-7ffb73a/preview/video/5456>

imagem 17
Código no Arduino e no process-ing para o LDR



```

File Edit Sketch Tools Help
Arduino Uno
LivroInterativo_P1_Potenciometro_escala.ino
1 int potPin = 0;
2 void setup() {
3   Serial.begin(9600);
4 }
5 void loop() {
6   int sensorValue = analogRead(potPin);
7   Serial.write(sensorValue/4);...
8   delay(100);
9 }

LivroInterativo_P1_Potenciometro_escala
1 import processing.serial.*;
2 Serial serial;
3 int sensorValue;
4 PShape bot;
5 void setup() {
6   size(640, 360);
7   serial=new Serial(this,Serial.list()[2],9600);
8   // The file "bot1.svg" must be in the data folder
9   // of the current sketch to load successfully
10  bot = loadShape("bot1.svg");
11 }
12 void draw() {
13   if(serial.available()>0)
14   {
15     sensorValue=serial.read();
16   }
17   background(102);
18   translate(width/2, height/2);
19   float zoom = map(sensorValue, 0, width, 0.1, 5);
20   scale(zoom);
21   shape(bot, -140, -140);
22 }

```

imagem 17
Código no Arduino e no processign para o Potenciômetro

link do video da funcionalidade do Potenciômetro

<https://kdrive.infomaniak.com/app/share/723395/51b5703b-ob93-4138-b6a3-ed-7656c780d4/preview/video/5457>

O protótipo inicial foi montado numa caixa de cartão com, aproximadamente, as dimensões do design inicial, para suporte do telemóvel. Com este protótipo foram realizados todos os testes relativos ao *Arduino* e *Processing*, bem como da manipulação dos inputs e outputs, resultantes da interação com o utilizador.

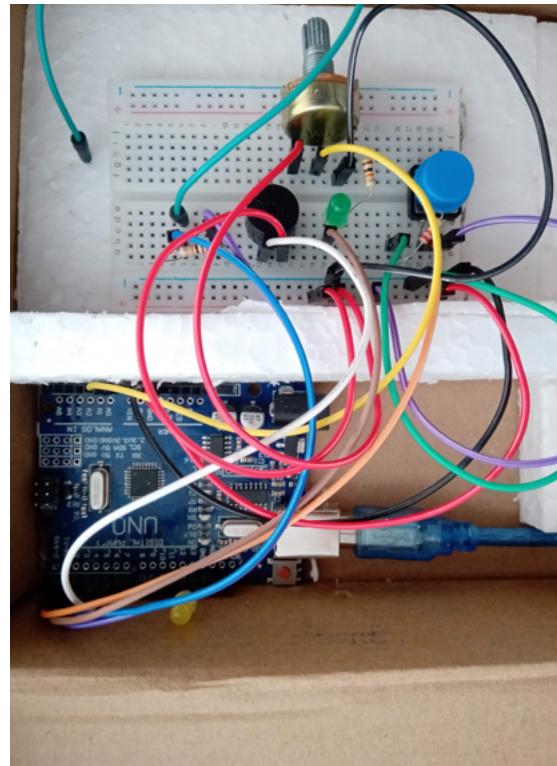
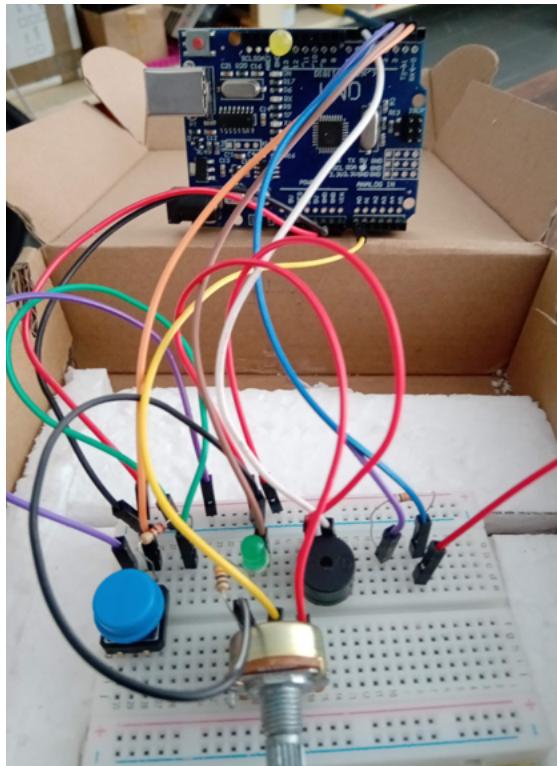


imagem 18 e 19
Circuito eletrico do arduino
montado com os componentes de
input e output usados no projeto

Tendo em conta o design do artefacto inicia em cartão e o protótipo em Arduino deu-se início à materialização do projeto, com recurso ao *ThinkerCAD*. As dimensões foram ajustadas durante a modelação do artefacto, constituído por oito peças.

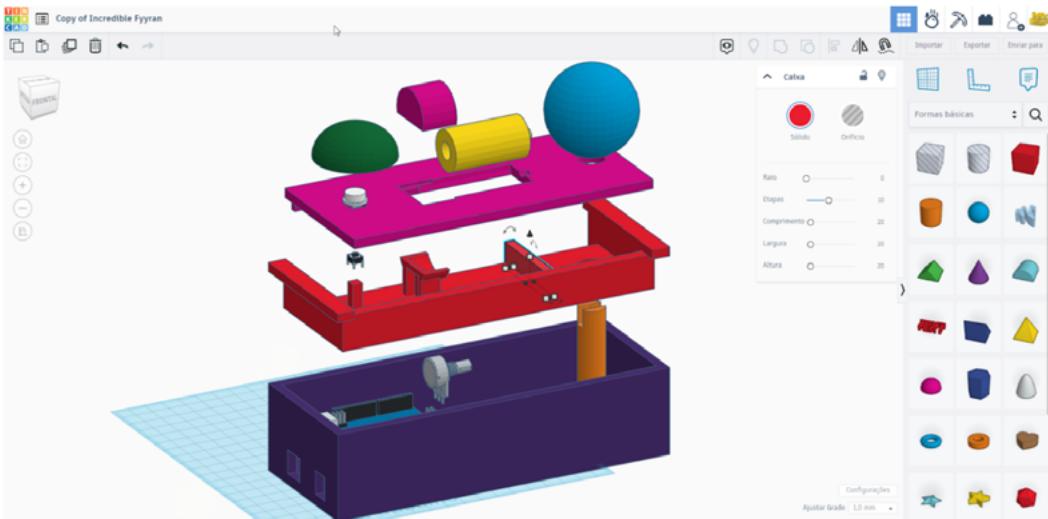


imagem 20
Protótipo da consola criado no ThinkerCAD

A consola foi modelada em *TinkerCAD* e todas as peças foram modeladas individualmente, tendo sido guardadas em ficheiros *STL*, de forma a poderem ser enviadas para um programa de “slice”, capaz de converter o modelo em *G-code*, para poder ser impresso numa impressora 3D. Com o modelo impresso, procedeu-se à montagem dos componentes e assim foi obtido o produto final, preparado para testagem.

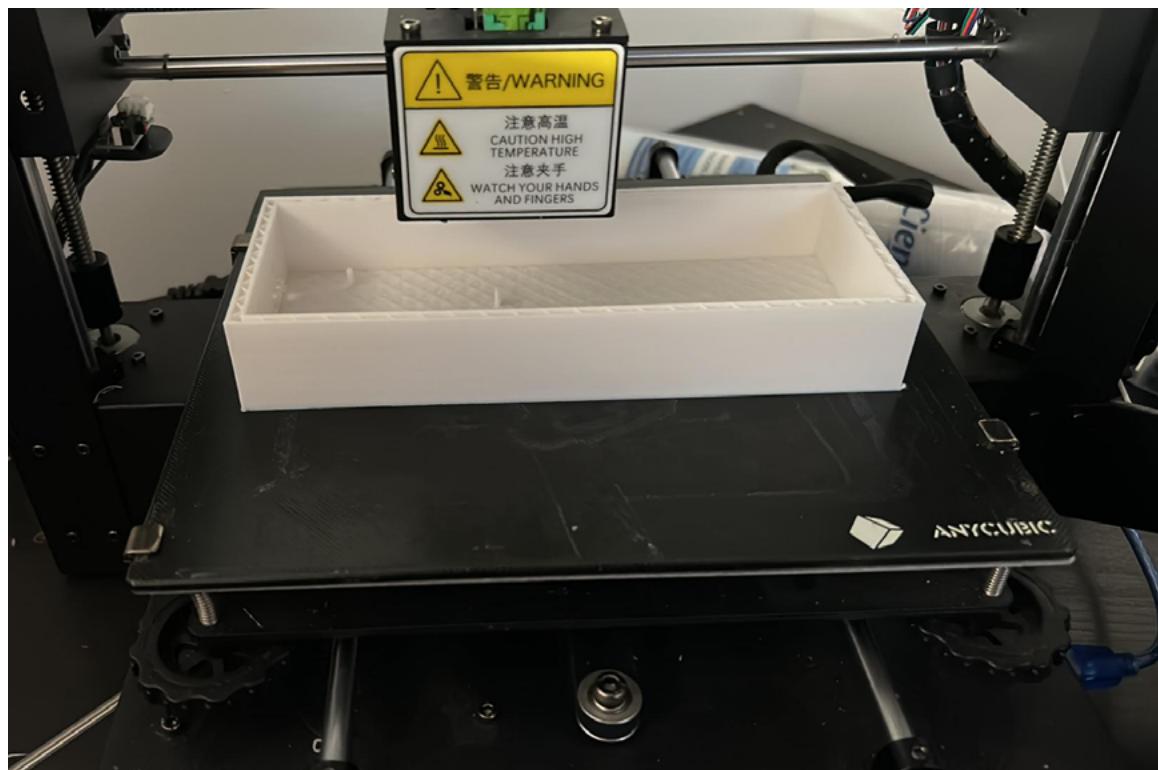


imagem 21
Impressão da maquete numa impressora 3D

4.5 Implementação e Desenvolvimento

A nossa estratégia de implementação foi faseada, de modo a reconhecer atempadamente possíveis necessidades de melhoria e a adequação ao objetivo que é pretendido alcançar. O projeto foi desenvolvido com recurso ao *software Arduino*, juntamente com *Processing* - todos os componentes foram programados e definidos no *Arduino*, enquanto que toda a parte visual foi programada em *Processing* ou seja, o primeiro executa funções de *back-end* e o segundo de *front-end*.

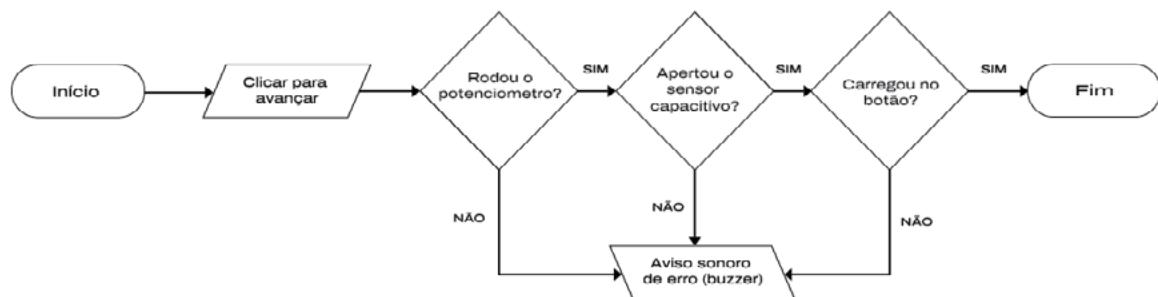


imagem 22
Fluxograma das interações do programa criado

Ao longo do desenvolvimento do código de cada uma das componentes das interações estabelecidas na etapa de design, percebemos que um dos inputs definidos, o LDR, não estava a funcionar adequadamente. Para resolver este problema, alterámos este componente e passámos a utilizar um sensor capacitivo.

Assim, as interações implementadas são a rotação do plástico através do potenciômetro, a audição do vidro a estilhaçar com o botão e a compactação do papel através de um sensor capacitivo.

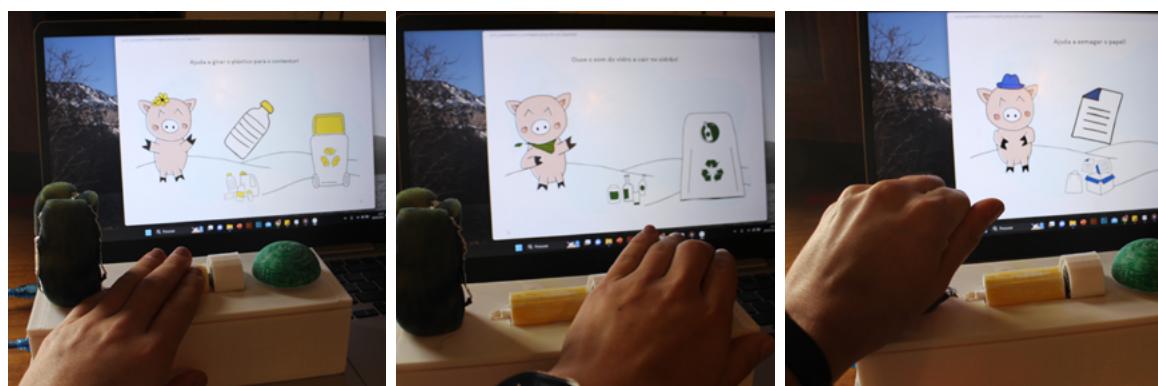


imagem 23, 24 e 25
Interação com os componentes da consola nos momentos correspondentes da história

Depois de programar e testar cada componente individualmente, os códigos foram juntos num só programa de *Arduino*. Para a comunicação em série com *Processing*, recorreu-se a uma biblioteca do *Arduino* - *Firmata* - a fim de ajudar nesta integração. Antes de proceder à integração dos códigos, criámos um código em *Processing* com as ilustrações da narrativa, através do método *switch*, por ser mais conveniente para a concretização do projeto. Desta forma, a implementação foi mais rápida, pois foi somente necessário associar as partes correspondentes a cada componente no menu criado e estruturado no *Processing*.

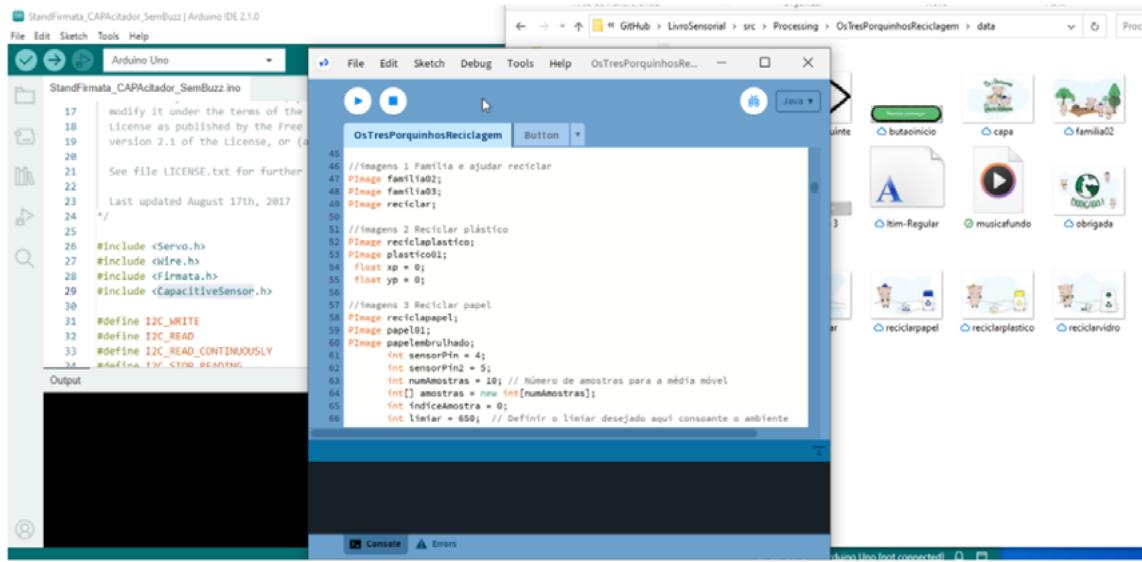


imagem 26
Código arduino em comunicação com o código do processing e as ilustrações criadas para anarrativa

Entretanto, o protótipo físico foi impresso na sua totalidade, montado com os respectivos componentes de interação e com o circuito de *Arduino*, para poder iniciar os primeiros testes de funcionalidade. Progressivamente, fomos implementando o código de cada componente e testando com o artefacto, a fim de deduzir se correspondia à ação idealizada. Algumas das características e dos valores foram ajustados progressivamente, à medida que cada um dos componentes era implementado.

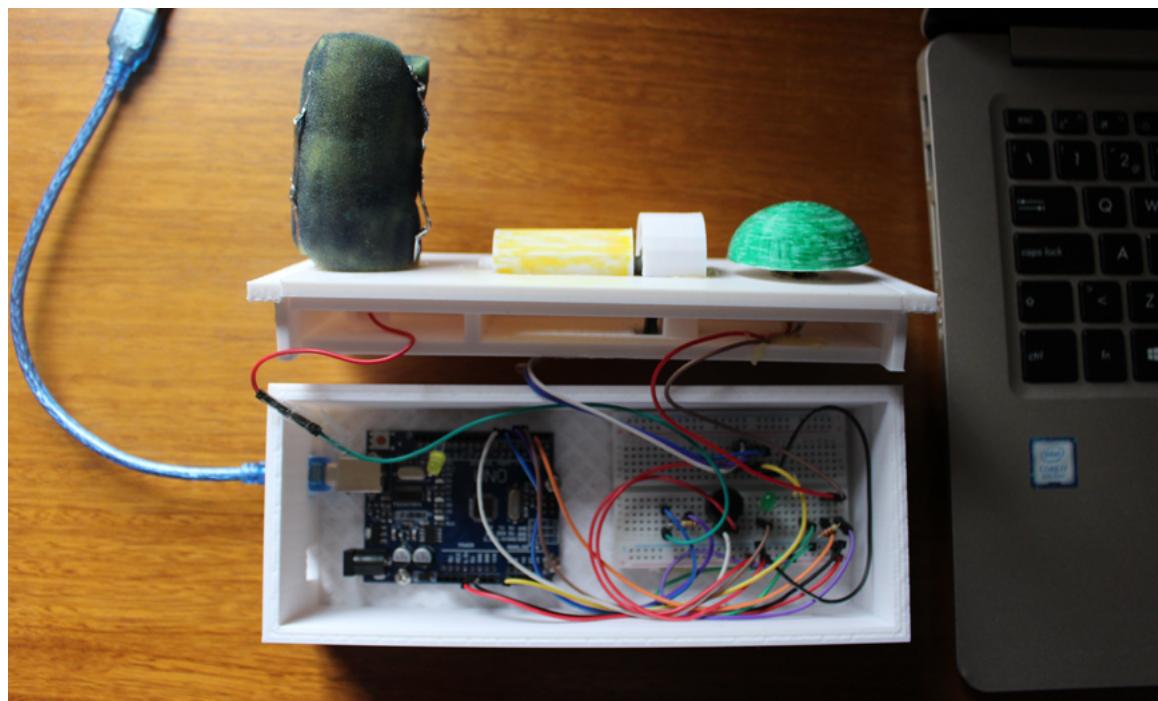


imagem 27
Maqueute da consola com os componentes montados e com o circuito de arduino à mostra

O sensor capacitivo gerou um problema devido ao material escolhido - inicialmente, a ideia era utilizar uma bola de espuma, que a criança podia apertar e/ou esmagar e assim simular o amassar do papel. Contudo, como a bola não estava a resultar, a ideia foi alterada e decidimos revestir uma esponja com pedaços de alumínio, de forma a que este sirva como condutor e reconheça a pressão gerada pela criança, quando apertado. Após esta alteração, o programa já conseguia reconhecer a força da interação, porém, surgiu outro problema com este sensor - a sua elevada sensibilidade levava a que, sem um correto mapeamento do ambiente onde se encontra, este identificasse qualquer alteração, realizando a ação pretendida na narrativa de forma independente.

Apesar destes problemas, conseguimos estabelecer um limiar em que bastava alterar o valor consoante o meio ambiente e realizar um teste com uma criança real.



imagem 28
Consola montada e programa pronto pronto para ser testado

Video demonstrativo da demo

<https://drive.google.com/file/d/1DA4VQPfCkSBa4Lvckxzoy05aiGKVV-wN/view?usp=sharing>

4.6 Teste e Feedback

Testar e obter feedback, é fundamental estar incluída no nosso processo para conseguirmos compreender erros que as nossas ideias e primeiros protótipos possam ter a fim de melhorar o projeto para ir de encontro às verdadeiras necessidades e desejos das crianças. Permite observar as primeiras interações com o artefacto e se corresponde ao que idealizamos no conceito e assim melhorar questões que precisam de ser afinadas ou alteradas para uma melhor experiência da criança.

Percebemos por meio da etapa anterior da implementação, que o sensor capacitivo foi o componente que dificultou a idealização do nosso conceito de projeto e de facto, quando se realizou o teste com a criança, este componente foi o que causou mais estranheza pelo seu formato e estrutura de criação. Os restantes inputs de ação foi familiar à criança e de fácil interação pela própria descrição feita na história que estava a ler.

Contudo, a criança conseguiu ler a narrativa, completar todas as tarefas do livro interativo que necessitavam de serem realizadas para chegar ao fim da história e aprender um tema de forma lúdica.

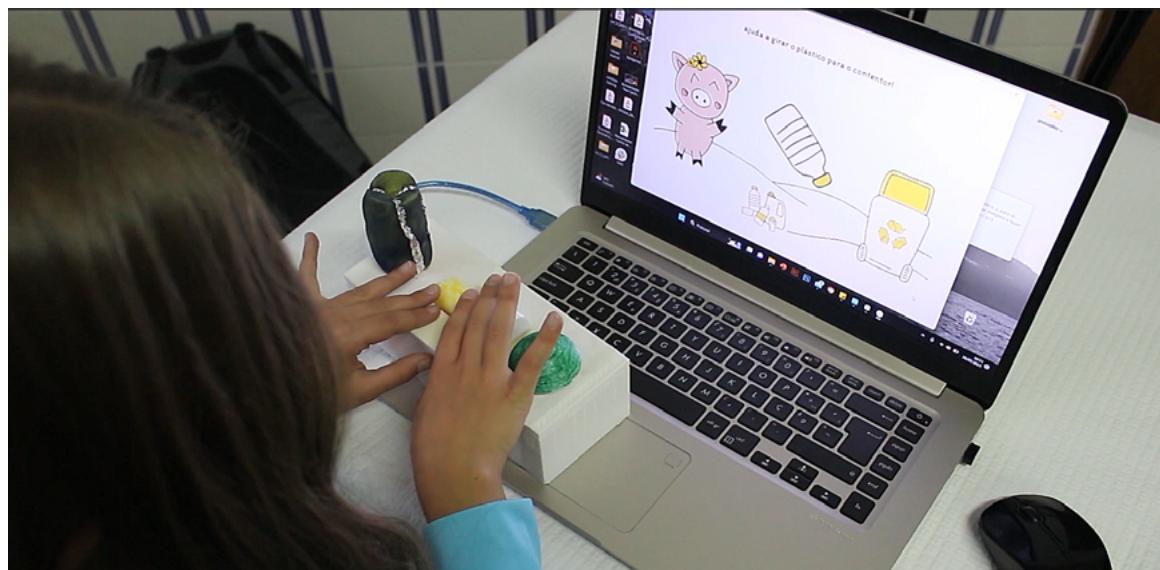


imagem 29
Criança a interagir com o protótipo

Vídeo da criança a interagir com o produto

<https://kdrive.infomaniak.com/app/share/723395/890b86db-3313-41ae-8321-d7a32a2ec91b/preview/video/8014>

05 CONCLUSÃO

Ao longo de todo este processo, conseguimos conceber o design da nossa ideia, desde a criação do *storyboard* da narrativa, as ilustrações, os desenhos em 3D da nossa maquete e a definição do design das interações da criança com o livro interativo. Também criámos os protótipos iniciais do fluxo da narrativa, do *hardware* e do *software* do programa. Foi programado todo o código associado às funcionalidades dos componentes que definimos no design: componentes de input são o botão, o potenciômetro, o sensor capacitivo e de output o *buzzer*.

No decorrer dos testes iniciais, observámos que existiam falhas no funcionamento de alguns dos componentes, principalmente no caso do sensor capacitivo - este, para além do problema com o material condutor, implica constantes ajustes, devido à sua elevada sensibilidade. Na junção das componentes do código desenvolvidas, também tivemos alguns problemas, principalmente na comunicação entre o *Processing* e a Biblioteca do sensor capacitivo do *Arduino*, o que nos levou à utilização da biblioteca *FIRMATA* - esta, contudo, não tinha a biblioteca “CapacitiveSensor.h”, de forma a conseguir funcionar com o sensor capacitivo, mas a solução encontrada foi a incorporação da biblioteca e do programa no ficheiro *StandardFirmata*.

Contudo, num próximo passo com mais testes realizados a crianças e tempo para melhorar o programa desenvolvido, poderão ser encontradas soluções para eliminar os problemas descritos anteriormente. Foram as dificuldades do projeto que nos fez evoluir e permitir adquirir vários conhecimentos ao longo das etapas do processo consoantes as tarefas que tinham de ser executadas. Fez-nos compreender que as interações criança-computador (CCI) são completamente distintas e requerem um estudo e abordagem diferente das interações humano-computador. As crianças são uma faixa etária bastante desafiante e pelo facto de terem nascido numa era mais digital cabe a nós designers criar produtos apelativos e ao mesmo tempo mais lúdicos e desafiadores para criar interesse para aprenderem melhor.

Com isto, concluo que em termos conceptuais foi atingido o que pretendemos com a ideia do nosso projeto. Porém tem algumas falhas a colmatar em termos de programa criado, como também da estrutura da consola. Num futuro, gostaríamos de criar uma consola menos robusta, mais ergonómica e apelativa para as crianças bem como mais histórias interativas com temas de responsabilidade social para que a sua aprendizagem seja adquirida de uma forma mais leve e didática e serem acrescentadas na aplicação idealizada (LICA) a fim de criar uma coleção de Livros Interativos para Crianças Aprenderem.

Todos os elementos integrantes do projeto disponível em:

<https://github.com/jmartsdesign/LivroSensorial>