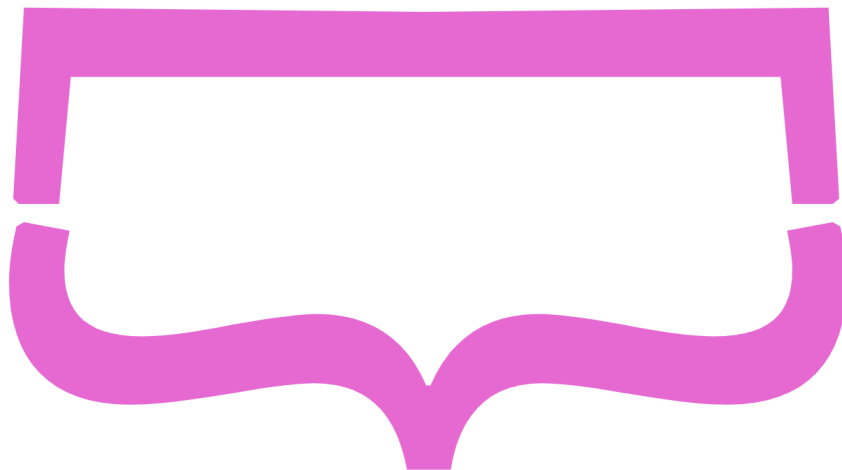


## Code Coach

La pàgina web intel·ligent per a la correcció de codi



[codecoach.world](https://codecoach.world)

## **Agraïments**

En primer lloc, m'agradaria agrair als meus tutors, el Carlos García i l'Angel Ramos, per al seu acompanyament.

També agraeixo la col·laboració desinteressada dels meus pares, Victor Sala, per guiar-me des de l'expertesa en l'àmbit de la programació i fer possible el desenvolupament de la web i Roser Guixà, per la seva ajuda i suport en donar forma i coherència a la memòria escrita.

Per acabar, m'agradaria regradar a la intel·ligència artificial, per ser el motor darrere de cada línia de codi corregida.

## **Resum**

La correcció de codi ha esdevingut imprescindible per als programadors. En les darreres dècades la utilització de la intel·ligència artificial per aquesta tasca ha anat en augment.

A partir de la meua experiència com a programadora principiant, aquest treball proposa la creació des de zero d'una pàgina web intel·ligent que corregeix codi utilitzant la tecnologia d'un model d'intel·ligència artificial: Code Coach.

En aquest projecte s'abordaran i detallaran totes les fases que comporta la creació d'una pàgina web, des de l'estructura inicial, fins al producte final, passant pels aspectes gràfics i visuals i tots els processos necessaris per connectar-la amb la intel·ligència artificial.

## **Abstract**

Code correction has become indispensable for programmers. In recent decades, the use of artificial intelligence for this task has been on the rise.

From my experience as a novice programmer, this project proposes the creation from scratch of an intelligent website that corrects code using the technology of an artificial intelligence model: Code Coach.

In this project, all the stages involved in creating a website will be addressed and detailed, from the initial structure to the final product, covering graphic and visual aspects and all the necessary processes to connect it with artificial intelligence.

# Índex

<b>1. Introducció.....</b>	<b>4</b>
<b>2. La intel·ligència artificial en l'àmbit de la programació.....</b>	<b>7</b>
2.1. Definició, història i evolució.....	7
2.2. Com aprèn la IA?.....	8
2.3. Conceptes clau: 'prompt'.....	9
<b>3. La correcció en l'àmbit de la programació.....</b>	<b>10</b>
3.1. Conceptes clau: front-end, back-end.....	10
3.2. HTML, CSS, JavaScript i PHP.....	11
<b>4. Flux de comunicació amb la IA.....</b>	<b>12</b>
4.1. Interfície d'usuari: definició i utilització.....	12
4.2. Interfície API: definició i utilització.....	12
<b>5. OpenAI.....</b>	<b>13</b>
<b>6. Desenvolupament de Code Coach.....</b>	<b>14</b>
6.1. Abans de començar, organització.....	14
6.1.1. Curs d'Udemy: formació en desenvolupament web.....	14
6.2. Disseny i desenvolupament del front-end.....	15
6.2.1. Disseny de la interfície d'usuari.....	15
6.2.2. Nom i logotip.....	17
6.2.3. Desenvolupament de la interfície d'usuari.....	18
6.3. Desenvolupament del back-end.....	23
6.3.1. Alta com a desenvolupadora de l'API d'OpenAI.....	23
6.3.2 Flux de comunicació.....	25
6.4.2. Playground.....	27
6.4.3. Programació d'interfície back-end.....	28
<b>7. Resultats i conclusions.....</b>	<b>31</b>
<b>8. Bibliografia.....</b>	<b>32</b>
<b>9. Annexos.....</b>	<b>34</b>

## 1. Introducció

La programació ha experimentat una transformació important i remarcable des dels seus inicis. El que va començar com a unes màquines analítiques primitives, ha evolucionat cap a un món de plataformes web, algorismes avançats i sistemes d'intel·ligència artificial sofisticats. A mesura que la complexitat augmenta, la correcció del codi de programació esdevé imprescindible.

La correcció de codi també ha evolucionat molt al llarg dels anys, adaptant-se als canvis tecnològics i a les necessitats dels desenvolupadors. Inicialment, la correcció de codi es feia manualment, requerint una revisió minuciosa línia per línia per detectar errors.

Amb el temps, la introducció dels sistemes de control de versions va facilitar la col·laboració entre desenvolupadors i la correcció conjunta del codi. La comunitat d'internet de programadors hi ha tingut un rol fonamental amb el sorgiment de fòrums de preguntes i respostes, la col·laboració en codi obert i proliferació de comunitats específiques.

En l'era actual, amb l'auge de la intel·ligència artificial els programadors s'han adonat que és una eina excepcional per corregir errors de programació, perquè pot analitzar codi ràpidament, té accés a una extensa base de dades de codis anteriors, opera amb consistència i es pot adaptar contínuament gràcies a l'aprenentatge automàtic.

Des que vaig començar a explorar el món de la programació, vaig adonar-me ràpidament de la importància de l'atenció al detall en programar, ja que una petita errada pot complicar tota l'execució del codi.

Els errors de sintaxi i de coherència són un dels problemes principals dels programadors principiants.

De la necessitat d'una ajuda ràpida i eficaç que m'acompanyes durant el procés d'aprenentatge, va sorgir Code Coach, una web que guia, aconsella i

corregeix el codi utilitzant la tecnologia d'una intel·ligència artificial. La participació de la IA permet no només detectar errors, sinó que també proporciona suggeriments i guies per millorar el codi, convertint-la en una eina perfecta per als principiants sigui quin sigui el llenguatge de programació que estiguin aprenent.

En aquest sentit, Code Coach pot sostenir gairebé tots els llenguatges, independentment del seu nivell de dificultat. Això inclou tant llenguatges populars com Python, HTML, o JavaScript, fins a llenguatges més especialitzats com Fortran, Scheme, o Prolog.

La meva intenció per Code Coach és dissenyar-lo de manera que sigui eficaç, permetent als usuaris copiar i enganxar el codi i prémer un botó per obtenir correccions, facilitats que el diferenciï d'altres eines de correcció de codi.

**Objectiu principal:**

- Desenvolupar "Code Coach", una interfície web que utilitza la tecnologia d'un model d'intel·ligència artificial per corregir codi de programació de manera automatitzada, sense la necessitat d'un 'prompt' específic.

**Objectius tècnics:**

- Aprendre a programar per a la creació web des de zero.
- Integrar efectivament l'API d'OpenAI.
- Assegurar l'estabilitat, rendiment i disponibilitat de la plataforma.
- Implementar característiques de personalització segons les necessitats de l'usuari.
- Fer una anàlisi de les necessitats i preferències del públic objectiu i tenir-les en compte en el disseny.

**Intenció artística:**

- Establir una identitat distintiva per a Code Coach.
- Dissenyar una interfície d'usuari atractiva.

## **2. La intel·ligència artificial en l'àmbit de la programació**

### **2.1. Definició, història i evolució**

La intel·ligència artificial (IA) es defineix com la capacitat de les màquines o programaris de demostrar intel·ligència. Aquesta branca de la informàtica dedicada al desenvolupament d'algorismes imita les funcions cognitives humanes, com l'aprenentatge, el descobriment, el raonament, la presa de decisions i la resolució de problemes. Mitjançant l'ús de determinades tècniques la IA permet a les màquines aprendre a partir de dades, analitzar-les i prendre decisions basades en la informació que rep.

Les intel·ligències artificials tenen una història curta però intensa. El concepte de raonament artificial apareix per primera vegada a principis del segle XIX, quan matemàtics i filòsofs ja confeccionen lògiques formals. L'esclat de les IA, però, no van donar-se fins a principis del segle XXI. N'és un exemple la victòria del supercomputador Watson al concurs televisiu de cultura general Jeopardy l'any 2011, demostrant que els ordinadors poden respondre preguntes de la mateixa manera que ho faria un ésser humà. Un altre moment memorable que posa en evidència la capacitat d'una IA per superar la d'un humà va tenir lloc el 2015, quan el programari AlphaGo es va convertir en el primer a derrotar un jugador professional de Go, un joc de taula tradicional xinès d'estratègia.

Des d'aleshores, la IA ha experimentat un creixement exponencial i s'ha aplicat a molts camps diferents, com la medicina (per ajudar amb la diagnosi de malalties i disseny de tractaments) o a la robòtica (facilitant les tasques de configuració i augmentant la productivitat de treballadors). S'especula que la IA esdevindrà en un futur pròxim tan present en la vida quotidiana, com ho són les xarxes socials actualment. De fet, el final de l'any 2022 ja va estar marcat per l'aparició de diferents productes d'IA revolucionaris, noves aplicacions de la tecnologia que fan tasques de tota mena: des de transformar text en àudio fins a generar imatges segons les teves instruccions.



## 2.2. Com aprèn la IA?

Amb aquesta ràpida expansió de la IA i la seva presència en la vida quotidiana d'alguns, és més habitual fer-se una pregunta fonamental: com aprèn exactament una màquina? Tot i que avui en dia, la intel·ligència artificial ajuda pilots a volar avions comercials i a urbanistes a preveure el trànsit, els científics informàtics que l'han dissenyat, amb freqüència no saben exactament com ho fa. Això és degut al fet que la IA sovint s'autoensenya i amb un simple conjunt d'instruccions crea una sèrie de patrons que l'ajuden a anar més enllà.

Hi ha 3 maneres diferents de construir aquests programes d'autoaprenentatge o aprenentatge automàtic (AA): l'aprenentatge no supervisat (UL), l'aprenentatge supervisat (SL) i l'aprenentatge per reforç (RL).

Aprenentatge no supervisat és una tècnica d'AA on una màquina o programa intenta trobar patrons en un conjunt de dades (exemples) sense guia o supervisió externa. Aquest tipus d'aprenentatge requereix una gran quantitat de dades i d'entrenament. Per exemple, imaginem que volen entrenar a un model d'IA perquè reconegui imatges de cocos. El que faríem seria donar-li quantitats ingents d'imatges de fruites, explicar-li com és un coco i el model hauria de buscar les característiques del coco i trobar-ne un patró.

En l'aprenentatge supervisat, el model també aprèn a partir d'exemples, però, a diferència de l'anterior i com indica el nom, en aquest mètode un desenvolupador o supervisor li proporciona el valor de cada entrada i sortida. És a dir, en aquest cas sí que li indicàriem si a la imatge hi apareix un coco o no i aquest utilitzaria la informació per aprendre a distingir el coco de la resta de fruites.

L'aprenentatge per reforç és un mètode que usen els sistemes que interactuant amb el seu entorn aprenen a triar automàticament les accions

que funcionen millor. En aquest mètode l'agent d'aprenentatge fa servir aquesta retroalimentació per decidir quin és el camí més eficaç amb el major rendiment. Un exemple que ajuda a entendre aquesta tècnica és la idea d'un laberint. El model no coneix la forma del laberint, però a base de prova i error en troba la sortida.

Cap d'aquests tres paradigmes bàsics de l'AA és essencialment més intel·ligent que l'altre. Tot i que alguns requereixen una major intervenció humana, tots ells tenen punts forts que els fan adequats per a determinades tasques. De fet, emprant-los junts, els programadors poden construir sistemes d'IA complexos, on els programes individuals poden supervisar-se i ensenyar-se els uns als altres.

### **2.3. Conceptes clau: 'prompt'**

Un cop explicats els mètodes que s'utilitzen per fer aprendre a una màquina, és important entendre com interactua amb els éssers humans. Un 'prompt' es refereix a l'entrada (declaració, instrucció o pregunta) que la persona fa servir per a sol·licitar una resposta o fer executar una tasca específica a un model d'IA. En el context d'IA per a la programació, els 'prompts' es fan servir amb l'objectiu de rebre correccions, exemples i suggeriments, refaccions o indicacions dins del codi.

Aquests models d'IA fan ús de tècniques i algorismes de processament de llengües naturals (PLN), analitzant el lèxic i morfologia per interpretar semànticament el text en llenguatge natural.

### 3. La correcció en l'àmbit de la programació

En el món digital, les persones necessitem una manera de comunicar instruccions a les màquines. Els llenguatges de programació serveixen aquest propòsit per als ordinadors. HTML, CSS, JavaScript i PHP són llenguatges de programació que s'utilitzen per al desenvolupament web i són els quatre llenguatges principals que he utilitzat per a la programació de Code Coach.

Com qualsevol llenguatge, el de programació té una sintaxi pròpia, similar a la gramàtica dels idiomes humans. De la mateixa manera que una errada gramatical pot donar lloc a malentesos, un error de sintaxi en programació pot causar problemes en l'execució del codi. Per aquest motiu és necessària la correcció, per evitar errors d'execució. La "correcció en l'àmbit de la programació" no només es refereix a la identificació d'errors sinó també a l'optimització i el garantiment d'un codi net i ben estructurat, per facilitar la comprensió a altres desenvolupadors col·laboradors.

#### 3.1. Conceptes clau: *front-end*, *back-end*

El desenvolupament web es pot dividir en dues àrees principals: el *front-end* i el *back-end*. El *front-end* és essencialment la part de la web que l'usuari pot veure i amb la que pot interaccionar. Inclou elements de disseny com botons, text i imatges. Per exemple, quan visites una pàgina web i veus elements que pots presionar, estas interactuant amb el *front-end*. El back end, en canvi, és aquella part que s'encarrega del funcionament de la web i que l'usuari no veu. En el context de Code Coach, el *back-end* agafa la petició de l'usuari del *front-end*, es comunica amb el sistema d'OpenAI per a corregir el codi, i després envia aquest codi corregit de nou al *front-end*.

### 3.2. HTML, CSS, JavaScript i PHP

Els llenguatges que he fet servir per a desenvolupar Code Coach són :

- HTML (HyperText Markup Language) és un llenguatge de programació que permet definir el contingut i estructurar els elements de la web, com els botons, les imatges i els enllaços.
- CSS (Cascading Style Sheets) és el llenguatge que utilitzem per indicar l'aparença dels elements d'HTML, és a dir la seva estètica (el color, la mida, posició...).
- JavaScript fa un pas més enllà i aporta interactivitat i funcionalitat als elements a la pàgina, permetent als usuaris comunicar-se en temps real amb la interfície.
- PHP a diferència dels tres llenguatges ja mencionats, es fa servir per programar el *back-end* de la web i permet la comunicació via API entre JavaScript i OpenAI.

## **4. Flux de comunicació amb la IA**

### **4.1. Interfície d'usuari: definició i utilització**

Interfície d'usuari és el mitjà amb què l'usuari pot comunicar-se amb una màquina. És a dir, el que veiem a simple vista quan entrem a qualsevol plataforma. L'objectiu d'aquesta interfície és proporcionar un disseny visual que faciliti la interacció entre els usuaris i el sistema.

Hi ha tres tipus d'interfícies d'usuari, les interfícies alfanumèriques (intèrprets de comandes), les gràfiques d'usuari (comunicació amb l'ordinador) i les tàctils (comunicació amb la tauleta o telèfon mòbil).

Code Coach es tracta d'una interfície gràfica d'usuari (GUI) dissenyada per ser intuïtiva. La GUI de Code Coach permet als usuaris interactuar directament amb elements visuals com botons, menús desplegable i caixes de text.

### **4.2. Interfície API: definició i utilització**

La interfície API (Interfície de Programació d'Aplicacions) és una forma de comunicació entre dos programaris o serveis. A través de l'API, els desenvolupadors podem accedir a la tecnologia d'una plataforma i integrar-la a les seves webs.

Utilitzant l'API d'una IA es pot tenir accés a la tecnologia, i a partir d'aquesta crear una aplicació completament nova. Pel que fa a Code Coach, faig servir la tecnologia d'OpenAI i el seu servei d'API.

## 5. OpenAI

OpenAI és un laboratori de recerca en IA format per dues organitzacions: la corporació OpenAI LP i la fundació sense ànim de lucre OpenAI Inc. És una empresa líder dedicada a la creació de models d'IA i la desenvolupadora d'una de les innovacions més destacades de l'any, ChatGPT. Es tracta d'un model d'IA capaç de mantenir converses en llenguatge natural amb els seus usuaris.

La missió d'OpenAI és garantir que la intel·ligència artificial beneficiï a tota la humanitat. Treballen per construir IA segura i per promoure la seva adopció àmplia entre la comunitat investigadora.

Tot i que algunes de les seves investigacions són privades per raons de seguretat, OpenAI es compromet a publicar la major part del seu treball, afavorint la transparència i la col·laboració amb altres investigadors, com per exemple, la seva associació amb altres organitzacions tecnològiques i laboratoris de recerca en el context de la seguretat en la IA.

OpenAI no només vol ser al capdavant de la tecnologia IA, sinó que també busca col·laborar amb altres institucions de recerca per resoldre els reptes globals associats amb la IA.

OpenAI ofereix una API que m'ha permès implementar la seva tecnologia a Code Coach. L'empresa proporciona directament la seva API amb l'objectiu de monetitzar el servei i controlar-ne l'ús ètic i segur.

## 6. Desenvolupament de Code Coach

### 6.1. Abans de començar, organització

Un dels principis de la programació consisteix en una bona gestió que faciliti el manteniment i la detecció d'errors. És per això que abans de submergir-me en el procés de desenvolupament de la plataforma, volia assegurar-me de tenir sempre un codi organitzat i net. Per garantir que es conservava un registre clar i actualitzat del codi vaig crear un repositori<sup>1</sup> a GitHub.

GitHub és una mena de xarxa social per a desenvolupadors que permet gestionar el codi font, controlar les versions del programari i treballar en un marc de codi obert<sup>2</sup>.

Després d'establir la base de codi a GitHub, vaig comprar el domini<sup>3</sup> '[codecoach.world](https://codecoach.world)' per 1,71 € a través de Hostinger<sup>4</sup>, i el vaig vincular al repositori. D'aquesta manera, qualsevol actualització que fes al codi es veuria directament a la web.

#### 6.1.1. Curs d'Udemy: formació en desenvolupament web

Com que els meus coneixements de programació no eren suficients per dur a terme el projecte, vaig decidir inscriure'm a unes classes de formació del llenguatge de programació JavaScript. Va ser un curs de 36 hores totals on vaig aprendre i practicar els fonaments del llenguatge, necessaris per programar els elements interactius de Code Coach.

---

<sup>1</sup>Un repositori és un espai centralitzat on es guarda i gestiona versions de codi de programació.

<sup>2</sup>Codi obert fa referència al programari distribuït i desenvolupat lliurement de forma pública i oberta a modificacions.

<sup>3</sup>Un domini web és la adreça utilitzada per identificar una ubicació específica a Internet, com ara un lloc web.

<sup>4</sup>Hostinger és una empresa internacional proveïdora de dominis web.

## **6.2. Disseny i desenvolupament del front-end**

### **6.2.1. Disseny de la interfície d'usuari**

El primer pas abans de prendre qualsevol decisió sobre el disseny d'una plataforma web és fer una anàlisi del públic objectiu que potencialment en farà ús. Aquesta anàlisi implica identificar el perfil de coneixement tècnic de l'usuari, és a dir, tenir en compte la seva experiència en el camp de la programació.

La meua intenció és fer una eina útil per un públic de programadors veterans, però també tenia clar que volia arribar als programadors principiants. Per assolir aquest objectiu necessitava una plataforma no només funcional i rica en possibilitats, sinó també amb un disseny intuïtiu i fàcil d'entendre.

Un cop clar el públic objectiu, vaig començar amb els primers esbossos sobre paper de les pàgines principals de la web: la pàgina d'inici i la pàgina de correcció.

*Imatge 2 - Esbós pàgina correcció*

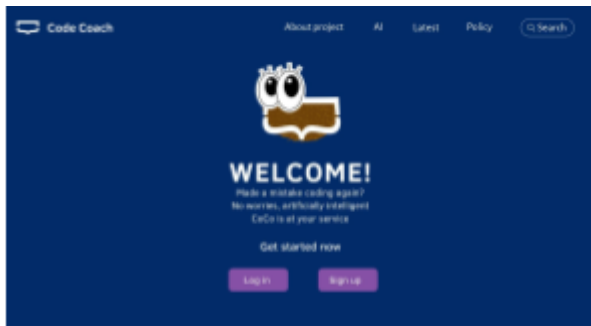




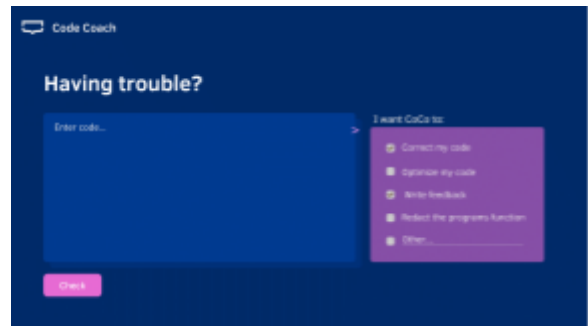
Imatge 3 - Colors

Un detall crític del disseny ha estat la selecció de la paleta de colors. He optat per una paleta de colors que combina tons de blau marí per simular un 'mode fosc'<sup>5</sup> -popular en les plataformes de desenvolupament- amb accents més vius per captar l'atenció en àrees clau que actuen com a elements de guia.

Els esquemes a paper els vaig fer servir com a guia per fer el mockup<sup>6</sup>, que vaig dissenyar amb Figma<sup>7</sup> per poder veure el conjunt amb color i tipografia.



Imatge 4 - Mockup de la pàgina d'inici



Imatge 5 - Mockup de la pàgina correcció

Aquest és un disseny provisional que s'utilitza per ajudar a estructurar les idees a l'hora de programar. He hagut de fer canvis a mesura que avançava el projecte, adaptant-me a les meves capacitats i les noves idees que sorgien. Més endavant, a la Imatge 11 es pot veure el resultat de l'última versió.

<sup>5</sup> El 'dark mode' és una opció de la interfície d'usuari que mostra el contingut amb text clar sobre un fons fosc.

<sup>6</sup>Presentació visual similar a un prototip que simula l'aspecte final d'un disseny.

<sup>7</sup>Figma és un programa d'edició gràfica vectorial.

### 6.2.2. Nom i logotip

Una part molt important del disseny és el nom i el logotip. Volia un nom amb ritme, fàcil de recordar i que capturés l'essència i l'objectiu principal de la web: la correcció de codi. Després de valorar diverses opcions, em vaig decantar per "Code Coach", una traducció a l'anglès que significa literalment 'entrenador de codi'.

A més a més, vaig decidir incorporar un 'personatge' que actués com un assistent virtual per a l'usuari. Vaig optar per anomenar-lo CoCo, que surt d'unir les dues primeres síl·labes de cada paraula del nom, amb la intenció de personificar el model d'IA que corregeix errors de programació.

Pel que fa al logotip, he utilitzat dos signes de puntuació molt emprats en diversos llenguatges de programació: la clau oberta i el claudàtor tancat, que junts queden així "{}".

Aquest símbol és fàcilment reconeixible per als programadors i, juntament l'assistent, completa la identitat visual de la plataforma.



*Dibuix 1 - Claudàtor + parèntesi*



*Imatge 6 - CoCo*

### 6.2.3. Desenvolupament de la interfície d'usuari

Un cop clara l'estètica de la web, era moment de donar-li forma. El procés del desenvolupament de la interfície d'usuari consta de tres etapes principals, que es desenvolupen de forma seqüencial seguint una metodologia iterativa, a base d'assaig i error, retrocedint i avançant a mesura que és necessari.

#### Fase 1: **Maquetació** (HTML)

Creació de l'esquelet de la pàgina web

#### Fase 2: **Estilització** (CSS)

Edició del color, posició, mida i tipografia dels elements definits

#### Fase 3: **Interactivitat** (JavaScript)

Implementació de funcionalitats dinàmiques

Per mostrar com funciona aquest procés i les seves fases veurem com ha estat en el cas de la barra de navegació (el primer element que vaig programar) i en el cas de la pàgina de correcció (l'espai principal de la interfície d'usuari, el *front end*).

La barra de navegació, coneguda en anglès com a *navigation bar* o simplement *navbar*, és una secció d'una pàgina web que normalment es troba a la part superior en forma horitzontal o a la part esquerra en vertical. La seva funció principal és facilitar als visitants enllaços que condueixen a diferents seccions de la web o a pàgines externes.

A la imatge 2 es poden veure les línies de codi necessàries per definir la barra de navegació, diversos enllaços i el logotip:

```

<!--Navbar-->
<nav class="navbar">
  <div class="navbar__container">
    <a href="/" class="navbar__logo">
      <span class="navbar__icon">{</span> Code Coach
    </a>
    <div id="mobile-menu" class="navbar__menu">
      <ul class="navbar__items">
        <li class="navbar__item about">
          <a href="AboutProject.html" class="navbar__links">About Project</a>
        </li>
        <li class="navbar__item">
          <a href="https://openai.com/blog/openai-api" class="navbar__links">AI</a>
        </li>
        <li class="navbar__item">
          <a href="Latest.html" class="navbar__links">Latest</a>
        </li>
        <li class="navbar__item">
          <a href="Policy.html" class="navbar__links">Policy</a>
        </li>
      </ul>
    </div>
  </div>
</nav>

```

Imatge 7 - Codi HTML navbar

L'estructura del codi es presenta en tres parts ("navbar", "navbar\_\_container" i "navbar\_\_items") i es basa en contenidors i etiquetes. S'utilitza la indentació per mostrar jerarquia entre els elements i quins continguts estan englobats i dins de quins.

El primer contenidor "navbar", encapsula tota la barra. Dins d'aquest contenidor, trobem un segon contenidor, "navbar\_\_container" que recull el logotip i el menú. Dins del menú, identificat com a "navbar\_\_menu", tenim una llista d'elements ("navbar\_\_items") que són els enllaços.

L'enllaç 'About Project' dirigeix cap a aquest treball de recerca on l'usuari podrà veure el propòsit, la història i el codi font del projecte. El segon, 'AI' té una vinculació directa amb la web d'OpenAI per referenciar i donar crèdit a la tecnologia que faig servir. Seguidament, trobem el botó 'Corrector', que dona accés directe a la pàgina de correcció sense haver de registrar-se a la web. Finalment, a l'enllaç 'Policy' es detallen les polítiques d'ús i altres informacions rellevants.



Imatge 8 - Execució del codi HTML la barra de navegació

Per estilitzar el disseny visual, he utilitzat el llenguatge de programació CSS. La imatge 3 es el codi necessari per personalitzar "navbar" i "navbar\_\_container":

```
*{
  box-sizing: border-box;
  margin: 0;
  padding: 0;
  font-family: 'Clear Sans', sans-serif; /*aplica l'estilització a tots els elements de la pàgina*/
}

body {
  background-color: #022C69;
  margin: 0;
  padding: 0;
}

.navbar{
  background: #01204D;
  height: 70px;
  display: flex;
  justify-content: center; /*aplica l'estilització a la barra de navegació*/
  align-items: center;
  font-size: 1.2rem;
  position: sticky;
  top: 0;
  z-index: 1000;
}

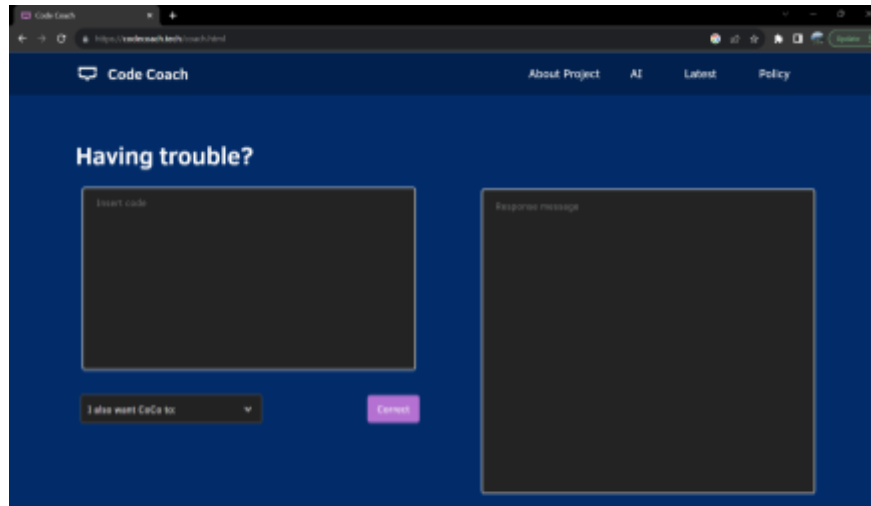
.navbar__container {
  display: flex;
  justify-content: space-between;
  height: 70px;
  z-index: 1; /*aplica l'estilització al elements de la barra de navegació*/
  width: 100%;
  max-width: 1300px;
  margin: 0 auto;
  padding: 0 50px;
}
```

*Imatge 9 - Codi CSS navbar*

El codi CSS està dividit en 3 blocs. En cadascun d'aquests blocs es veuen les etiquetes específiques que criden als elements programats anteriorment al codi HTML i els assigna estils visuals concrets. (*Per veure el codi complet: ANNEX 5*)



*Imatge 10 - Producte final de la barra de navegació*



Imatge 11 - Resultat final de la pàgina de correcció

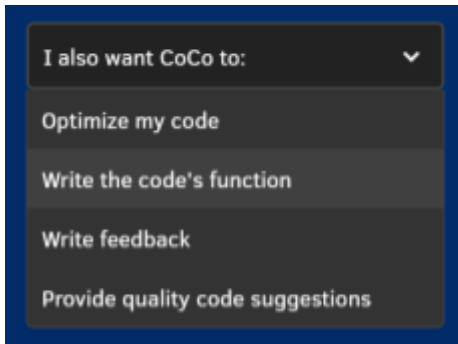
Per programar la pàgina de correcció a més a més d'HTML i CSS, he fet servir JavaScript. Amb JavaScript he programat el menú desplegable que permet a l'usuari escollir que vol que faci CoCo.

Un menú desplegable, també conegut com a "dropdown menu", és un element d'interfície d'usuari que desplega una llista de possibles opcions quan l'usuari fa clic.

```
,  
document.addEventListener("DOMContentLoaded", function () {  
  const optionMenu = document.querySelector(".select-menu"),  
        selectBtn = optionMenu.querySelector(".select-btn"),  
        options = optionMenu.querySelectorAll(".option"),  
        sBtn_text = optionMenu.querySelector(".sBtn-text");  
  
  selectBtn.addEventListener("click", () => optionMenu.classList.toggle("active"));  
  
  options.forEach(option => {  
    option.addEventListener("click", () => {  
      let selectedOption = option.querySelector(".option-text").innerText;  
      sBtn_text.innerText = selectedOption;  
      optionMenu.classList.remove("active");  
      console.log(selectedOption);  
    });  
  });  
});
```

Imatge 12 - Codi JavaScript pàgina correcció

El codi que es mostra a la imatge 12 és el que permet que en el botó principal del menú es mostri una llista d'opcions i un cop seleccionada amb un clic l'opció desitjada, el menú es tanqui i la tria quedi fixada.



*Imatge 13 - Menú desplegable*

Un cop acabades les 3 fases per cada element que volia programar, ja tenim la interfície d'usuari completa. Ara, per donar funcionalitat a tots aquests elements i fer possible la correcció, és necessari el desenvolupament del *back-end*.

## 6.3. Desenvolupament del *back-end*

### 6.3.1. Alta com a desenvolupadora de l'API d'OpenAI

Per fer ús del model d'intel·ligència artificial de l'empresa OpenAI, cal interactuar amb la seva API. Per aconseguir això, vaig haver de donar-me d'alta com a desenvolupadora a la seva plataforma. Un cop fet el registre vaig generar una "API key" i vaig dedicar temps a llegir la documentació proporcionada per OpenAI per assegurar-me de com fer-ne un ús correcte i estar al corrent de les millors pràctiques recomanades.

Una "API key" és la clau que permet a les aplicacions comunicar-se entre elles de forma confidencial i segura. En el context de Code Coach, era essencial per establir una connexió amb el model d'intel·ligència artificial. A més a més, la creada clau serveix com a identificador per a monitorar i autenticar-ne l'ús. Això és crucial, ja que l'empresa cobra pels seus recursos en funció de la quantitat de processament que sol·licites.

#### API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code, in order to protect the security of your account, OpenAI may also automatically disable any API key that we've found has leaked publicly.

NAME	KEY	CREATED	LAST USED ⓘ	
Code Coach	sk-...uo76	Jul 26, 2023	Jul 26, 2023	✎ ⓘ

+ Create new secret key

*Imatge 14 - API key*

Després d'investigar les tarifes d'OpenAI, vaig decidir utilitzar el model gpt-4 durant la fase inicial de proves per la seva alta precisió i comprensió del llenguatge natural, malgrat el preu per interacció sigui més elevat. No obstant això, en el moment de llançar el producte, quan l'ús de l'API serà més intensiu, canviaré al model gpt-3.5, que té un preu més accessible i no suposa sacrificar massa la qualitat de les respostes.



Model	Input	Output
8K context	\$0.03 / 1K tokens	\$0.06 / 1K tokens
32K context	\$0.06 / 1K tokens	\$0.12 / 1K tokens

Model	Input	Output
4K context	\$0.0015 / 1K tokens	\$0.002 / 1K tokens
16K context	\$0.003 / 1K tokens	\$0.004 / 1K tokens

**Model:** versió o configuració del model d'intel·ligència artificial d'OpenAI.

**Tokens:** unitats d'informació o trossos de paraules, similar a un caràcter. 1K tokens són aproximadament 750 paraules.

**Context:** la quantitat màxima d'informació (en tokens) que permet el model d'una sola vegada

**Input:** l'entrada d'informació, el que s'envia al model. En el context de Code Coach, el codi a corregir.

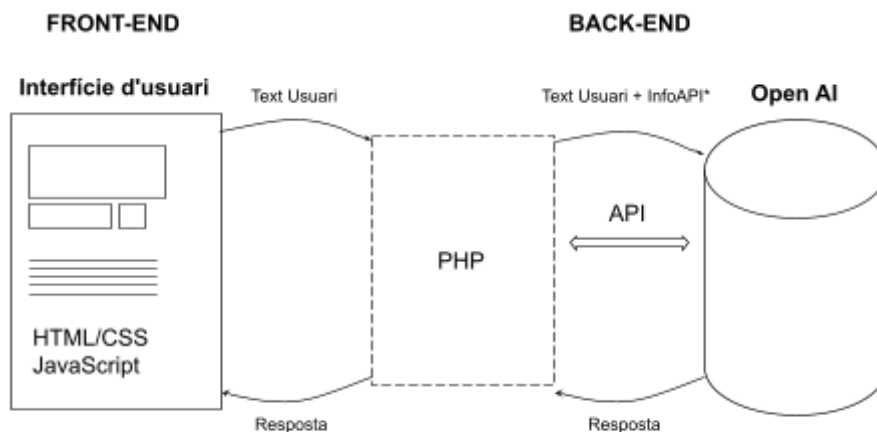
**Output:** la sortida d'informació, la resposta que el model dona com a resultat de l'input.

Com a exemple, considerem una comparació de preus basada en una interacció de 1000 tokens -el que aproximadament equival a 200-250 línies de codi CSS, amb una mitjana de 4 paraules per línia-:

Model gpt-4	Model gpt-3.5
Context: 8K	Context: 4K
Entrada (Input): \$0.03 / 1K tokens	Entrada (Input): \$0.0015 / 1K tokens
Sortida (Output): \$0.06 / 1K tokens	Sortida (Output): \$0.002 / 1K tokens
Total per interacció: <b>\$0.09</b>	Total per interacció: <b>\$0.0035</b>

Taula 1 - Comparació de preus entre els dos models emparats

### 6.3.2 Flux de comunicació



Dibuix 2 - Esquema del flux de comunicació de Code Coach

Aquest diagrama representa l'arquitectura del sistema que connecta la interfície d'usuari amb OpenAI a través *back-end* impulsat per PHP. El flux de comunicació funciona de la següent manera: l'usuari fa una sol·licitud (enganxa el codi problemàtic a la interfície), la sol·licitud s'envia al *back-end*, que es comunica amb OpenAI per rebre el codi corregit i enviar la resposta de nou al *front-end*.

Però perquè és necessari l'intermediari PHP?

La meva primera intenció era establir una comunicació directa entre interfície i OpenAI. Aquí em vaig trobar amb un inconvenient estructural i una alerta inesperada, ja que OpenAI té unes restriccions de seguretat que no permeten publicar el codi que implica fer pública l'API key. Aquesta mesura és essencial per evitar l'ús indegut de claus.

*Imatge 17 - Avis de Github*

*Imatge 18 - Avis d'OpenAI*

Així doncs, davant d'aquest error de procediment, vaig adonar-me que necessitava un intermediari fora de Github, la plataforma on es gestiona la resta de codi de manera pública.

Vaig optar per fer servir HostMonster, un proveïdor d'allotjament que m'ajuda a gestionar el servidor back-end i ofereix un entorn segur i privat per executar i gestionar el fitxer PHP.

Amb aquesta configuració en marxa, el fitxer PHP ja actua com a intermediari entre la interfície d'usuari i OpenAI. Cada vegada que un usuari fa una petició, el fitxer PHP, allotjat a HostMonster, processa la sol·licitud, es comunica amb OpenAI i envia la resposta de tornada a la interfície d'usuari. D'aquesta manera, s'evita exposar l'API key.



## [lanasals/CodeCoach] pages build and deployment workflow run

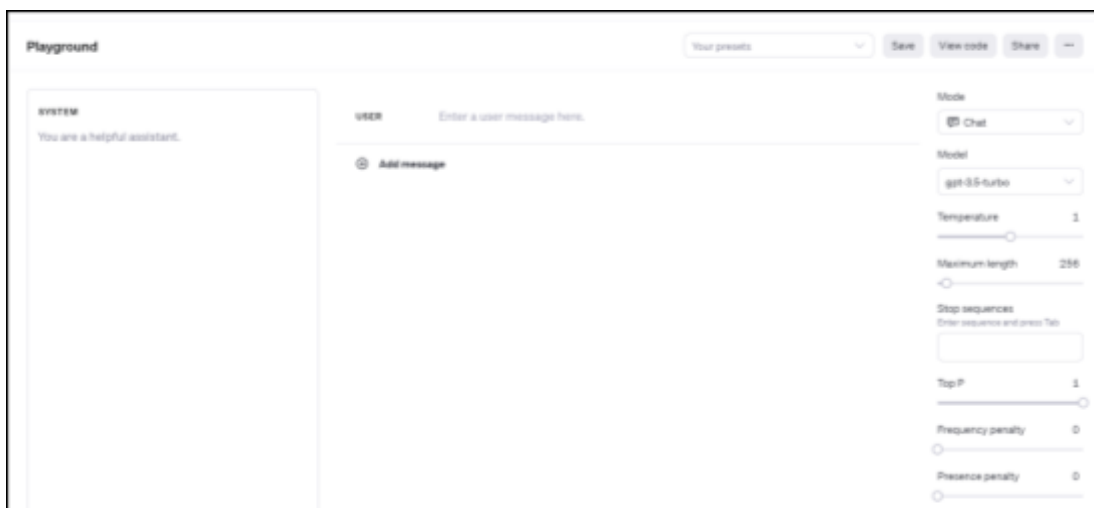
pages build and deployment: All jobs were cancelled

[View workflow run](#)

	pages build and deployment / build Cancelled	
	pages build and deployment / deploy Cancelled	
	pages build and deployment / report-build-status Cancelled	

### 6.4.2. Playground

Per programar el fitxer PHP, Open AI proporciona una eina que permet als desenvolupadors provar les capacitats dels models sense necessitat de programar: Playground. És a dir, permet experimentar amb diferents característiques del model abans de donar-lo per definitiu.



Imatge 19 - Playground d'OpenAI

Dins del playground pots descriure el rol que vols que adopti el model i assignar valors a paràmetres de comportament i estructura de la resposta. Si és necessari, la plataforma escriu el codi en funció d'aquests ajustos, un procediment que em va facilitar la programació del fitxer PHP.

Code Coach no és únicament un corrector de codi, sinó que ofereix a l'usuari l'opció d'anar més enllà. Justament aquesta possibilitat és la que el diferencia de plataformes similars. Sense necessitat d'escriure un 'prompt' i amb pocs clics l'usuari pot rebre consells i recomanacions que l'ajuden a aprendre i millorar les seves habilitats.

Permet a l'usuari escollir quina mena de retorn vol, amb 4 possibles opcions: optimitzar el codi, comprendre la funció del codi, rebre retroalimentació o obtenir suggeriments de qualitat. Per aconseguir això vaig assignar un rol específic a cadascun dels quatre retorns possibles.

#### 6.4.3. Programació d'interfície *back-end*

Com ja he mencionat anteriorment, el *back-end* fa d'intermediari entre la interfície d'usuari de Code Coach i l'API d'OpenAI; processa les peticions de l'usuari, comunicant-se amb l'API per obtenir una resposta, i després retornant aquesta resposta a l'usuari.

A la imatge següent es poden veure les línies de codi del programa 'response.php':

```
<?php

$api_key = "sk-2jHe[REDACTED]";

if(!isset($_GET['content'])) {
    die(json_encode(['error' => 'Content parameter is missing.']));
}
if(!isset($_GET['CoCoJob'])) {
    die(json_encode(['error' => 'CoCoJob parameter is missing.']));
}

$CoCoJob = $_GET['CoCoJob'];
$content = $_GET['content'];

// Optimitzar --> CoCoJob = 1
// Funcionalitat --> CoCoJob = 2
// Retroalimentació --> CoCoJob = 3
// Qualitat --> CoCoJob = 4

switch ($CoCoJob) {
    case 1:
```



El codi segueix la següent estructura:

**Inicialització:** Primer definim la clau API que hem generat a la plataforma d'OpenAI per a la comunicació amb la seva tecnologia

**Verificació:** Comprovem que els paràmetres 'content' (el missatge d'entrada de l'usuari, és a dir, el codi problemàtic) i 'CoCoJob' (tipus de servei que vol l'usuari) s'hagin establert correctament. Si no es proporcionen aquests paràmetres, el servidor retorna un missatge d'error.

**Assignació:** El valor de 'CoCoJob' assigna una tasca específica al model. Aquestes són les tasques i la seva respectiva línia de codi:

"Optimize my code": Rebre suggeriments sobre com fer que el sigui més eficient. Escurçar el codi eliminant parts redundants o implementar mètodes més adequats.

**["role" => "system", "content" => "Optimize my code: You are a professional and helpful Software Developer who provides the corrected code only if necessary, and if optimizing is necessary provides the full optimized code to copy and paste."]**

Traducció al català: “Ets un desenvolupador de programes informàtics professional i servicial que proporciona el codi corregit només si cal, i si l'optimització és necassària proporciona el codi optimitzat complet per copiar i enganxar”

"Write the code's function": Obtenir una descripció de la funció del programa

**["role" => "system", "content" => "Write the code's function: You are a professional and helpful Software Developer who provides the corrected code, and a paragraph explaining its functionality"]**

Traducció al català: “Ets un desenvolupador de programes informàtics professional i servicial que proporciona el codi corregit, i un paràgraf que explica la seva funcionalitat”

"Write feedback": Proporciona una retroalimentació general sobre el codi de l'usuari per ajudar-lo a millorar.

**["role" => "system", "content" => "Write FeedBack: You are a professional and helpful Software Developer who provides the corrected code only if necessary, explains the corrections made, offers tips the current code, and gives 1 recommendation for future improvement."]**

Traducció al català: "Ets un desenvolupador de programes informàtics professional i servicial que proporciona el codi corregit només si cal, explica les correccions fetes, ofereix consells del codi actual i dona 1 recomanació per millorar en un futur"

"Provide quality code suggestions": Proporciona suggeriments sobre com millorar la qualitat del codi.

**["role" => "system", "content" => "Provide Quality Code Suggestions: You are a professional and helpful Software Developer who provides the corrected code only if necessary, and a paragraph giving code quality suggestions for the future. "]**

Traducció al català: "Ets un desenvolupador de programes informàtics professional i servicial que proporciona el codi corregit només si cal, i un paràgraf que dona suggeriments de qualitat de codi per al futur.

**Comunicació:** El codi està utilitzant una eina (cURL) per "cridar" a OpenAI i demanar-li que corregeixi el codi que li has donat.

**Resposta:** Un cop processada la petició, el codi PHP l'envia com a resposta a l'usuari.



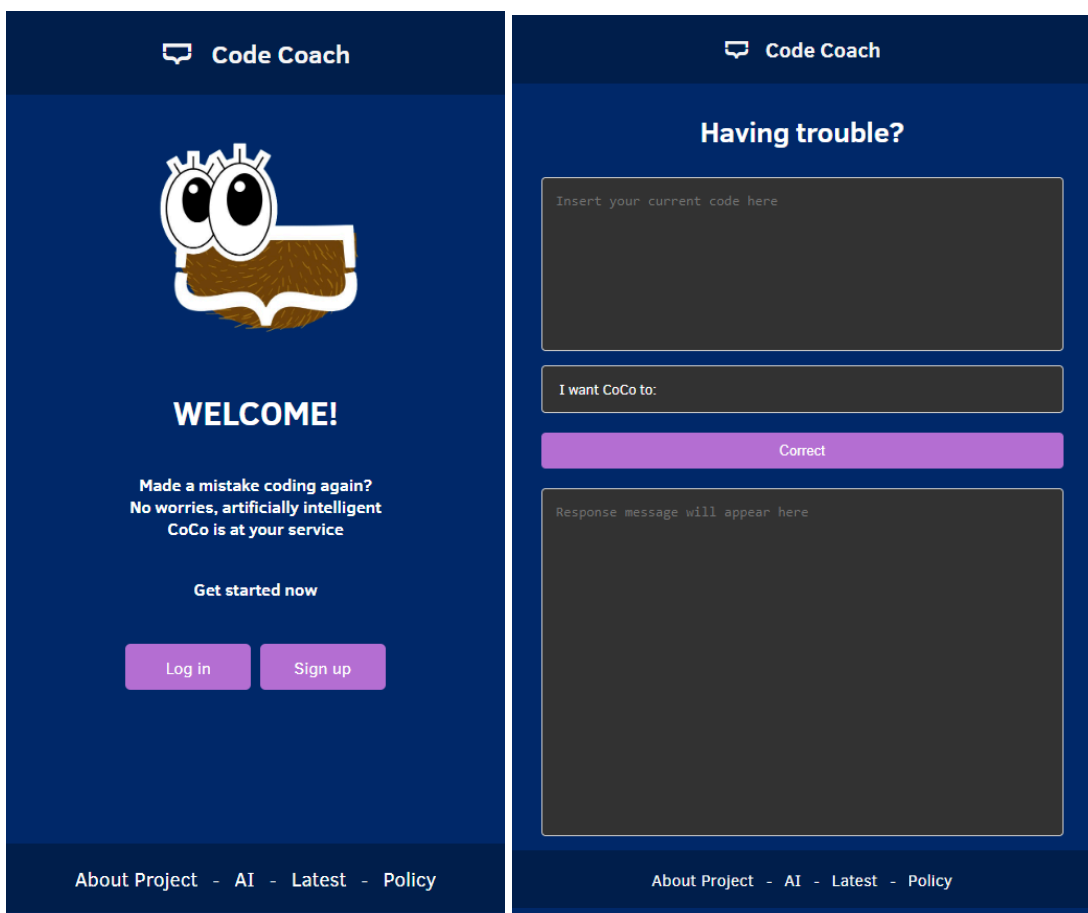
## 7. Resultats i conclusions

L'objectiu principal d'aquest treball de recerca era programar i dissenyar una interfície web intel·ligent per a la correcció de codi. Utilitzant la tecnologia d'OpenAI, pretenia desenvolupar una plataforma web en línia que permetés corregir codi de qualsevol nivell i llenguatge de programació de manera automatitzada.

Podeu experimentar la interfici a: <https://codecoach.world>

A més a més, Code Coach també tenia una intenció artística. Volia experimentar amb les possibilitats que et dona la programació per dissenyar entorns gràfics i jugar amb l'expressió del color, amb l'ambició de fusionar la tecnologia i l'art. Volia endinsar-me en un món desconegut per mi, i aprendre coses fora dels àmbits que em són més afins, sense deixar-los de banda.

Captures del resultat final en format “mobile” de les dues pàgines principal.



Estic satisfeta del resultat final.

He creat des de zero una pàgina web funcional sense tenir cap coneixement previ de programació. La pàgina té totes les característiques que em vaig proposar i és completament funcional i útil pel que estava pensada, i un disseny atractiu.

Inicialment, tenia una visió tancada de com havia de ser el producte final, durant el procés m'he anat adonant del gran marge de possibilitats que hi ha per fer-lo créixer. És per això que ara penso que Code Coach és un projecte en procés que vull mantenir viu, actualitzant-lo amb noves característiques, per continuar ampliant els meus coneixements tècnics i pràctics en l'àmbit de la programació. En aquest sentit, m'agradaria implementar un sistema de registre enllaçat amb Google, que guardi l'historial de consultes dels usuaris. També, em proposo fer-ne una versió multilingüe traduint-la com a mínim al català i al castellà.

## 8. Bibliografía

- [1] Online Courses - Learn Anything, On Your Schedule | Udemy, "Universidad JavaScript - De Cero a Experto JavaScript! Ubaldo Acosta," *Udemy*, Dec. 2022. <https://www.udemy.com> (accessed Mar. 08, 2023).
- [2] OpenAI, "Documentation OpenAI API," *platform.openai.com*. <https://platform.openai.com/docs/introduction>
- [3] Adobe, "Adobe Color," *Adobe.com*, 2019. <https://color.adobe.com/create/color-wheel>
- [4] Hostinger, "Hostinger: Mucho Más Que Alojamiento Web.," *Go Online with Hostinger*. <https://www.hostinger.es/> (accessed Oct. 21, 2023).
- [5] Figma, "Figma: the Collaborative Interface Design tool.," *Figma*, 2023. <https://www.figma.com/>
- [6] G. Roca, "Sobre la inteligencia artificial: abans d'opinar, prova-ho," *Via empresa*, Nov. 01, 2023. [https://www.viaempresa.cat/opinio/roca-artificial-novetat\\_2177346\\_102.html](https://www.viaempresa.cat/opinio/roca-artificial-novetat_2177346_102.html) (accessed Jun. 07, 2023).
- [7] V. Rusu, "Login With Google Account Using PHP & MySQL," *www.youtube.com*, Apr. 20, 2020. <https://youtu.be/M4jde7THXAI> (accessed Jun. 13, 2023).
- [8] B. Delipetrev, C. Tsinaraki, and U. Kostić, "AI Watch Historical Evolution of Artificial Intelligence," *Joint Research Centre (JRC)*, no. JRC120469, 2020, doi: <https://doi.org/10.2760/801580>.
- [9] M. Haenlein and A. Kaplan, "A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence," *California Management Review*, vol. 61, p. 000812561986492, Jul. 2019, doi: <https://doi.org/10.1177/0008125619864925>.
- [10] G. Brockman, M. Murati, and P. Welinder, "OpenAI API," *openai.com*, Jun. 11, 2020. <https://openai.com/blog/openai-api> (accessed May 20, 2023).
- [11] S. Graciela, P. Ibarra, R. Quispe, F. Mullicundo, and D. Lamas, "Herramientas Y Tecnologías Para El Desarrollo Web Desde El FrontEnd Al BackEnd," Apr. 2021. Accessed: Jul. 18, 2023. [Online]. Available: <http://sedici.unlp.edu.ar/bitstream/handle/10915/120476/Ponencia.pdf-PDFA.pdf?sequence=1&isAllowed=y>

- [12] W. Willard, *HTML : a beginner's Guide*. New York, Ny U.A.: Mcgraw Hill, 2013.
- [13] G. Greg Kohs, "Documental AlphaGo ,", *www.youtube.com*, 2017.  
<https://youtu.be/GIJ7zr4sYx4?feature=shared> (accessed Oct. 29, 2023).

## 9. Annexos

### ANNEX 1: El títol certificant de compleció del curs JavaScript



### ANNEX 2: Enllaços d'interès

- La pàgina Code Coach: <https://codecoach.world>
- Per veure el codi complet de Code Coach (Repositori de GitHub):  
<https://github.com/lanasals/CodeCoach>

### ANNEX 3: Notificació de seguretat d'OpenAI de quan vaig fer pública l'API key per error. (Traduïda al català)

Hola,

S'ha determinat que la vostra clau de l'API d'OpenAI s'ha filtrat, la qual cosa ha provocat una desactivació de la clau i aquest correu electrònic de notificació amigable.

Això pot ser degut al fet que heu enviat la vostra clau API a un servei en línia com GitHub, o la vostra clau pot haver estat compromesa d'una altra manera.

No et preocupis, encara tens accés a l'API! Aneu a la pàgina Claus de l'API per a crear una nova clau de l'API.

Si la clau de l'API s'ha emmagatzemat en qualsevol ubicació -per exemple, en el codi que esteu executant-, s'haurà d'actualitzar abans de poder tornar a executar aquest codi.

Finalment, li demanem que revisi les nostres millors pràctiques per a la seguretat de les claus de l'API.

Salutacions,

L'equip OpenAI

## ANNEX 4: Codi complet de Code Coach

### index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Code Coach</title>
  <link rel="stylesheet" href="styles.css">
  <link rel="icon" type="image/x-icon" href="faviconrosa.ico">
  <link href="https://fonts.cdnfonts.com/css/clear-sans"
rel="stylesheet">
</head>

<body>
<!--Navbar-->
  <nav class="navbar">
    <div class="navbar__container">
      <a href="index.html" class="navbar__logo">
        <span class="navbar__icon">[]</span> Code Coach
      </a>
    </div>
  </nav>
</br>
</br>
  <div class="image__container">
    
  </div>
</br>
  <h4>WELCOME!</h4>
</br>
  <h1>Made a mistake coding again?<br>No worries, artificially
intelligent<br>CoCo is at your service</h1>
</br>
  <h1>Get started now</h1></br></br>
  <div class="image__container">
    <a class="button__login" href="coach.html">Log
in</a>&nbsp;&nbsp;<a class="button__signup" href="#"
onclick="showFloatingWindow()">Sign up</a>
  </div>
</br>
</br>
</br>
```

```

</br>
    </br>
</br>
</br>
<div id="footer" class="footer">
    <a href="AboutProject.html">About Project</a> - <a
href="https://openai.com/blog/openai-api" target="_blank">AI</a> -
<a href="Latest.html">Latest</a> - <a href="Policy.html">Policy</a>
</div>

    <!-- Finestra flotant avis -->
<div id="floatingWindow" class="floating-window">
    <div class="floating-window-content">
        <p>This option will come soon.</br>Meanwhile, click on Log
in.</p>
    </div>
</div>

<!-- Javascript clic tanca -->
<script>
    function showFloatingWindow() {
        var floatingWindow =
document.getElementById("floatingWindow");
        floatingWindow.style.display = "block";
    }

    window.onclick = function(event) {
        var floatingWindow =
document.getElementById("floatingWindow");
        if (event.target == floatingWindow) {
            floatingWindow.style.display = "none";
        }
    }
</script>

</body>
</html>

```

## **styles.css:**

```
/* Reset */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

/* Global */
body {
  font-family: 'Clear Sans', sans-serif;
  background-color: #022c69;
  color: #fff;
}

/* Navbar */
.navbar {
  background: #01204d;
  height: 80px;
  display: flex;
  justify-content: center;
  align-items: center;
  font-size: 1.2rem;
  position: sticky;
  top: 0;
  z-index: 999;
}

.navbar__container {
  display: flex;
  justify-content: space-between;
  align-items: center;
  height: 80px;
  z-index: 1;
  width: 100%;
  max-width: 1300px;
  margin: 0 auto;
  padding: 0 50px;
}

.navbar__logo {
  color: #f5f5f5;
  display: flex;
  align-items: center;
  justify-content: center;
  text-decoration: none;
  font-size: 1.5rem;
  font-weight: bold;
  width: 100%;
}
```



```
.navbar__icon {  
  margin-right: 0.5rem;  
  font-size: 1.8rem;  
  transform: rotate(270deg);  
  margin-right: 25px;  
}
```

```
/* Headings */  
h4 {  
  text-align: center;  
  margin-top: 2rem;  
  font-size: 2rem;  
}
```

```
h1 {  
  text-align: center;  
  margin-top: 1rem;  
  font-size: 1rem;  
}
```

```
/* Code form styles */  
.boxes {  
  display: flex;  
  flex-direction: column;  
  align-items: center;  
  margin-top: 2rem;  
}
```

```
.box__insertcode {  
  width: 80%;  
  max-width: 600px;  
  height: 200px;  
  padding: 1rem;  
  margin-bottom: 1rem;  
  border: 2px solid #cccccc;  
  border-radius: 5px;  
  background-color: #333;  
  color: #fff;  
  font-size: 1rem;  
  resize: none;  
}
```

```
.box__response {  
  width: 80%;  
  max-width: 600px;  
  height: 400px;  
  padding: 1rem;  
  margin-bottom: 1rem;  
  border: 2px solid #cccccc;  
  border-radius: 5px;  
  background-color: #333;  
  color: #fff;
```

```
    font-size: 1rem;
    resize: none;
}

.button__correct {
    padding: 11.5px 16px;
    border: none;
    border-radius: 5px;
    background-color: rgb(255, 143, 255, 0.7);
    color: #fff;
    font-size: 1rem;
    cursor: pointer;
    width: 80%;
    max-width: 600px;
}

.button__correct:hover {
    background-color: rgb(255, 143, 255, 0.7);
}

/* Dropdown */
.select-menu {
    width: 80%;
    max-width: 600px;
}

.select-btn {
    display: flex;
    justify-content: space-between;
    align-items: center;
    height: 55px;
    background-color: #333;
    color: #fff;
    padding: 20px;
    font-size: 1rem;
    font-weight: 400;
    border-radius: 5px;
    cursor: pointer;
    border: 2px solid #cccccc;
}

.options {
    position: absolute;
    width: 80%;
    max-width: 600px;
    overflow-y: auto;
    max-height: 290px;
    padding: 10px;
    margin-top: 10px;
    border-radius: 5px;
    background-color: #757575;
    display: none;
}
```

```
        border: 2px solid #cccccc;
    }

    .select-menu.active .options {
        display: block;
    }

    .options .option {
        display: flex;
        height: 55px;
        cursor: pointer;
        padding: 0 16px;
        border-radius: 5px;
        align-items: center;
        background-color: #757575;
    }

    .options .option:hover {
        background-color: #444;
    }

    .option .option-text {
        font-size: 18px;
        color: #fff;
    }

    /* Login i signup */
    .button__login,
    .button__signup {
        background-color: rgb(255, 143, 255, 0.7);
        color: #fff;
        padding: 11.5px 16px;
        border: none;
        border-radius: 5px;
        font-size: 1rem;
        cursor: pointer;
        width: 120px;
        text-align: center;
        text-decoration: none;
    }

    /* Image container */
    .image__container {
        display: flex;
        justify-content: center;
    }

    .image__coco {
        width: 220px;
    }

    /* Floating window */
```

```
.floating-window {
  display: none;
  position: fixed;
  z-index: 999;
  left: 0;
  top: 0;
  width: 100%;
  height: 100%;
  overflow: auto;
  background-color: rgba(0, 0, 0, 0.4);
}

.floating-window-content {
  background-color: rgb(255, 143, 255, 0.9);
  color: white;
  margin: 15% auto;
  padding: 40px;
  border-radius: 5px;
  width: 400px;
  text-align: center;
}

/* Footer styles */
.footer {
  flex-shrink: 0;
  background-color: #01204d;
  padding: 20px;
  text-align: center;
}

.footer a {
  color: #fff;
  font-size: 1.2rem;
  margin: 0 10px;
  text-decoration: none;
}

/* Responsive styles */
@media screen and (max-width: 960px) {
  .navbar__container {
    padding: 0 30px;
  }

  .button__correct {
    width: 80%;
    max-width: 600px;
  }

  .footer {
    padding: 10px;
  }
}
```

```
.footer a {  
  display: block;  
  margin: 10px 0;  
}  
}
```

### **coach.html:**

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <!-- Basic meta tags -->  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <meta http-equiv="X-UA-Compatible" content="IE=edge">  
  
  <!-- Page title -->  
  <title>Code Coach</title>  
  
  <!-- Stylesheets and Fonts -->  
  <link rel="stylesheet" href="styles.css">  
  <link href="https://fonts.cdnfonts.com/css/clear-sans" rel="stylesheet">  
  
  <!-- Favicon -->  
  <link rel="icon" type="image/x-icon" href="faviconrosa.ico">  
</head>  
  
<body>  
  <!-- Navbar Section -->  
  <nav class="navbar">  
    <div class="navbar__container">  
      <a href="index.html" class="navbar__logo">  
        <span class="navbar__icon">[]</span> Code  
Coach  
      </a>  
    </div>  
  </nav>  
  
  <!-- Main Content -->  
  <h4>Having trouble?</h4>  
  
  <!-- Code Correction Form -->  
  <form id="codeForm" class="boxes">  
    <textarea id="codeInput" class="box__insertcode" placeholder="Insert  
your current code here"></textarea>  
    <!-- Dropdown Section -->  
    <div class="select-menu">  
      <div class="select-btn">  
        <span class="sBtn-text">I want CoCo to:</span>  
        <i class="bx bx-chevron-down"></i>  
      </div>  
    </div>  
  </form>  
</body>  
</html>
```

```

</div>
<ul class="options">
  <li class="option">
    <span class="option-text">Optimize my code</span>
  </li>
  <li class="option">
    <span class="option-text">Write the code's function</span>
  </li>
  <li class="option">
    <span class="option-text">Write feedback</span>
  </li>
  <li class="option">
    <span class="option-text">Provide quality code
suggestions</span>
  </li>
</ul>
</div>
  <br>
  <button type="submit" class="button__correct">Correct</button>
  <br>
  <textarea id="newResultField" class="box__response"
placeholder="Response message will appear here" readonly></textarea>

</form>

<div id="footer" class="footer">
  <a href="AboutProject.html">About Project</a> - <a
href="https://openai.com/blog/openai-api" target="_blank">AI</a> - <a
href="Latest.html">Latest</a> - <a href="Policy.html">Policy</a>
</div>

<!-- JavaScript -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script
>
  <script src="app.js"></script>
  <script src="dropdown.js"></script>

</body>
</html>

```

**app.js:**

```
// Quan el document està llest
$(document).ready(function() {

    // Quan s'envia el formulari inclòs a "codeForm" que inclou on lusuari
    posa el seu codi i on mostrem la resposta obtinguda de OpenAI
    console.log(` in app.js`);
    $('#codeForm').submit(function(e) {

        // Prevenir l'acció per defecte del formulari, No volem carregar una
        pàgina nova
        e.preventDefault();

        // Esborrar el contingut del camp "newResultField" quan es fa clic al
        botó "Correct", per feedback pel usuari. que sapiga que passa algo
        $('#newResultField').val('');

        // Recollir el codi introduït per lusuari al camp "codeInput"
        var code = $('#codeInput').val();

        // Encode the code using encodeURIComponent()
        var encodedCode = encodeURIComponent(code);

        // Change the button text to "WAIT"
        $('.button__correct').text('Please, wait...');

        // Realitzar la petició POST a l'API que cridarà a OpenAI, està en
        servidor PHP per amagar l'API_KEY de lo contrari OpenAI deshabilita la
        key. No les vols públiques. Pasem 2 paràmetres, el codi a examinar (code)
        i la opció del desplegable (CoCoJob)
        $.post('https://victorsala.me/CodeCoachAPI/response.php', { content:
        code, CoCoJob: CoCoJob }, function(data) {

            // Actualitzar el contingut del camp "newResultField" amb la
            resposta de l'API
            $('#newResultField').val(data.message);

            // Change the button text back to "Correct"
            $('.button__correct').text('Correct');

        }).fail(function() {

            // Mostrar un missatge d'error si la petició falla
            $('#newResultField').val('Error occurred during the request.');
```

```

            // Change the button text back to "Correct" in case of failure
            $('.button__correct').text('Correct');
        });
    });
});

```

**response.php**

```

<?php

$api_key = "sk-2.....";

// Check if the content parameter is set in the POST data
if(!isset($_POST['content'])) {
    die(json_encode(['error' => 'Content parameter is missing.']));
}

if(!isset($_POST['CoCoJob'])) {
    die(json_encode(['error' => 'CoCoJob parameter is missing.']));
}

$CoCoJob = $_POST['CoCoJob'];
$content = $_POST['content'];

// Optimize my code --> CoCoJob = 1
// Write the code's function --> CoCoJob = 2
// Write feedback --> CoCoJob = 3
// Provide quality code suggestions --> CoCoJob = 4

switch ($CoCoJob) {
    case 1:
        $data = [
            "model" => "gpt-4",
            "messages" => [
                ["role" => "system", "content" => "Optimize my code:
You are a professional and helpful Software Developer who provides the
corrected code only if necessary, and if optimizing is necessary provides
the full optimized code to copy and paste."],
                ["role" => "user", "content" => $content]
            ]
        ];
        break;
    case 2:
        $data = [
            "model" => "gpt-4",
            "messages" => [
                ["role" => "system", "content" => "Write the code's
function: You are a professional and helpful Software Developer who
provides the corrected code, and a paragraph explaining its functionality"],
                ["role" => "user", "content" => $content]
            ]
        ];
        break;
    case 3:
        $data = [
            "model" => "gpt-4",
            "messages" => [

```



```

        ["role" => "system", "content" => "Write FeedBack: You
are a professional and helpful Software Developer who provides the
corrected code only if necessary, explains the corrections made, offers tips
the current code, and gives 1 recommendation for future improvement."],
        ["role" => "user", "content" => $content]
    ];
    break;
    case 4:
    $data = [
        "model" => "gpt-4",
        "messages" => [
            ["role" => "system", "content" => "Provide Quality Code
Suggestions: You are a professional and helpful Software Developer who
provides the corrected code only if necessary, and a paragraph giving code
quality suggestions for the future. "],
            ["role" => "user", "content" => $content]
        ]
    ];
    break;
    default:
    $data = [
        "model" => "gpt-4",
        "messages" => [
            ["role" => "system", "content" => "You are a professional
and helpful Software Developer who corrects code mistakes"],
            ["role" => "user", "content" => $content]
        ]
    ];
}

```

```

$ch = curl_init();

curl_setopt($ch,
CURLOPT_URL,"https://api.openai.com/v1/chat/completions");
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($data)); // json data
curl_setopt($ch, CURLOPT_HTTPHEADER,
array('Content-Type:application/json', 'Authorization: Bearer ' . $api_key));
curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

$server_output = curl_exec($ch);

curl_close ($ch);

$response = json_decode($server_output);

header('Content-Type: application/json');
header('Access-Control-Allow-Origin: *');

```

```
echo json_encode(['message' =>
$response->choices[0]->message->content]);
```

```
?>
```

### **dropdown.js:**

```
document.addEventListener("DOMContentLoaded", function () {
  const optionMenu = document.querySelector(".select-menu"),
    selectBtn = optionMenu.querySelector(".select-btn"),
    options = optionMenu.querySelectorAll(".option"),
    sBtn_text = optionMenu.querySelector(".sBtn-text");

  selectBtn.addEventListener("click", () =>
optionMenu.classList.toggle("active"));

  options.forEach(option => {
    option.addEventListener("click", () => {
      let selectedOption = option.querySelector(".option-text").innerText;
      sBtn_text.innerText = selectedOption;
      optionMenu.classList.remove("active"); // Close the dropdown

      switch (selectedOption) {
        case 'Optimize my code':
          CoCoJob = 1;
          break;
        case "Write the code's function":
          CoCoJob = 2;
          break;
        case 'Write feedback':
          CoCoJob = 3;
          break;
        case 'Provide quality code suggestions':
          CoCoJob = 4;
          break;
        default:
          CoCoJob = 0;
      }

      console.log(` Selected option: ${selectedOption}, CoCoJob:
${CoCoJob}` );
    });
  });
});
```

**script.js:**

```
function myFunction() {  
  var text = document.getElementById("myText").value;  
  alert(text);  
}
```