



Escola d'Enginyeria de Telecomunicació i
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

TÍTOL DEL TFG: Desenvolupament d'una aplicació educativa per al disseny de turbines des del punt de vista termodinàmic

TITULACIÓ: Grau en Enginyeria d'Aeroports

AUTOR: Nuria Hostench Muelas

DIRECTOR: Antonio Marzoa Domínguez i Òscar Bertran Cànovas

DATA: 07 de juliol del 2023

Títol: Desenvolupament d'una aplicació educativa per el disseny de turbines des del punt de vista Termodinàmic

Autor: Nuria Hostench Muelas

Director: Antonio Marzoa Domínguez i Òscar Bertran Cànovas

Data: 07 de juliol del 2023

Resum

Aquest treball de fi de grau és un projecte de programació l'objectiu del qual és crear una pàgina web per a desenvolupar una aplicació interactiva que permeti il·lustrar els processos i cicles termodinàmics que es donen lloc en una turbina d'avió. L'aplicació també té un parell d'exercicis de turbojets i turbofans, on l'usuari ha de fixar les condicions de treball i altres paràmetres, segons requeriments del disseny, en treballs posteriors s'ampliarà aquesta part. Dins el lloc web també hi ha una part de teoria de termodinàmica i exercicis dels gasos ideals. La finalitat de la pàgina web és ajudar als estudiants i els professors a entendre millor la termodinàmica i facilitar l'hora de fer exercicis i teoria. En la memòria inicialment s'explica quin llenguatge de programació s'ha fet servir i perquè, també hi ha una comparació entre el llenguatge Matlab i Python, ja que eren els dos llenguatges que es podien fer servir, finalment es va decidir que la millor opció era utilitzar Python. Un cop explicat el llenguatge de programació s'explica com començar a programar els exercicis de cicles i processos termodinàmics que s'inclouen a la pàgina web utilitzant PyCharm, per fer això inicialment es va haver de decidir el format dels exercicis i quina informació havia de donar l'usuari, en aquest punt també s'explica com crear les gràfiques. El següent punt parla de les eines necessàries per dur a terme el lloc web, s'explica que és el Visual Studio Code i com funciona, aquest programa és el que s'ha utilitzat per dur a terme la pàgina web, també s'explica les extensions que s'han utilitzat i l'HTML. I per acabar el capítol 1 s'explica l'estructura interna de la pàgina web, la pàgina web consta d'un arxiu anomenat `app.py` que és l'encarregat de connectar els diferents arxius i després d'una carpeta de *templates* que consta de sis carpetes més on estan els arxius HTML per a cada idioma. En el següent capítol s'explica el resultat de la pàgina web, el primer punt del capítol dos explica el seu disseny i el seu funcionament, és a dir els diferents apartats que té i com passar d'un lloc a l'altre, també s'explica com funcionen els exercicis. I el següent i últim punt del capítol 2, explica com publicar la pàgina web, inicialment només es volia compartir la pàgina web amb els tutors, doncs, per fer això es va utilitzar un programa anomenat Ngrok, i després es va fer servir un *hosting* per poder publicar el lloc web i fer-lo accessible per a qualsevol usuari. Finalment hi ha les conclusions. Per acabar aquest resum cal dir que s'ha aconseguit l'objectiu principal del projecte .

Título: Desarrollo de una aplicación educativa para el diseño de turbinas desde el punto de vista Termodinámico

Autor: Nuria Hostench Muelas

Director: Antonio Marzoa Domínguez y Óscar Bertran Cànovas

Data: 07 de julio del 2023

Resumen

Este trabajo de fin de grado es un proyecto de programación cuyo objetivo es crear una página web para desarrollar una aplicación interactiva que permita ilustrar los procesos y ciclos termodinámicos que se dan lugar en una turbina de avión. La aplicación tiene un par de ejercicios de turbojets y turbofanos, donde el usuario debe fijar las condiciones de trabajo y otros parámetros, en trabajos posteriores se ampliará esta parte. Dentro del sitio web hay una parte de teoría de termodinámica y ejercicios de los gases ideales. La finalidad de la página web es ayudar a los estudiantes y los profesores a entender mejor la termodinámica y facilitar la hora de hacer ejercicios y teoría. En la memoria inicialmente se explica qué lenguaje de programación se ha usado y porque, también hay una comparación entre el lenguaje Matlab y Python, ya que eran los dos lenguajes que se podían usar, finalmente la mejor opción era Python. Una vez explicado el lenguaje de programación se explica como se empezó a programar los ejercicios de ciclos y procesos termodinámicos que hay en la página web utilizando PyCharm, para hacer esto inicialmente se tiene que decidir cómo el formato de los ejercicios y qué información que tiene que dar el usuario, en este punto también se explica cómo se han creado las gráficas. El siguiente punto habla de las herramientas necesarias para llevar a cabo el sitio web, se explica que es el Visual Studio Code y cómo funciona, este programa es el que se ha utilizado para llevar a cabo la página web, también se explica las extensiones utilizadas y el HTML. Y para finalizar el capítulo 1 se explica la estructura interna de la página web, la página web consta de un archivo llamado app.py que es el encargado de conectar los diferentes archivos y después de una carpeta de *templates* que consta de seis carpetas más donde están los archivos HTML para cada idioma. En el siguiente capítulo se explica el resultado de la página web, el primer punto del capítulo dos explica su diseño y su funcionamiento, es decir los diferentes apartados que tiene y cómo pasar de un lugar a otro, también se explica cómo funcionan los ejercicios. Y el siguiente y último punto del capítulo 2, explica cómo publicar la página web, inicialmente solo se quería compartir la página web con los tutores, pues, para hacer esto se ha utilizado un programa llamado Ngrok, y luego un *hosting* para poder publicar el sitio web y hacerlo accesible para cualquier usuario. Finalmente están las conclusiones. Para acabar este resumen me gustaría decir que se ha logrado el objetivo principal.

Title: Development of an educational application for the design of turbines from the point of view Thermodynamics

Author: Nuria Hostench Muelas

Director: Antonio Marzoa Domínguez and Òscar Bertran Cànovas

Date: July 07, 2023

Overview

This final degree project is a programming project whose objective is to create a website to develop an interactive application that illustrates the thermodynamic processes and cycles that occur in an airplane turbine. The application also has a couple of turbojet and turbofan exercises, where the user must set working conditions and other parameters, according to design requirements, in the future this part will be expanded. Within the website there is also a part of thermodynamics theory and ideal gas exercises. The purpose of the website is to help students and teachers better understand thermodynamics and facilitate the time of doing exercises and theory. The report initially explained what programming language I used and why, there was also a comparison between the Matlab language and Python, since they were the two languages I could use, finally I decided that the best option was to use Python. Once explained the programming language explains how I started programming the exercises of cycles and thermodynamic processes that I wanted to include on the website using PyCharm, to do this initially I had to decide how I wanted to do the exercises and what information I wanted the user to give me, at this point it also explains how I created the graphics. The next point talks about the tools needed to carry out the website, it explains what the Visual Studio Code is and how it works, this program is the one I used to carry out the website, it also explains the extensions I used and the HTML. And to finish chapter 1 explains the internal structure of the website, the website consists of a file called app.py that is responsible for connecting the different files and then a folder of templates consisting of six more folders where the HTML files for each language are. The next chapter explains the result of the website, the first point of chapter two explains its design and operation, that is, the different sections it has and how to move from one place to another, it also explains how the exercises work. And the next and last point of chapter 2, explains how to publish the website, initially I just wanted to share the website with my tutors, so, to do this I used a program called Ngrok, and then I used a hosting to be able to publish the website and make it accessible to any user. Finally, there are the conclusions. To finish this summary, it must be said that I have achieved the objective of the project that was to create a website of educational thermodynamics, I think I have created a clear and easy website.

Als meus pares, que sempre m'ho han donat tot i m'han animat dia a dia a seguir endavant i no han deixat que mai em donés per vençuda, sense ells això seria un full en blanc. I a la meva germana Mar, per recolzar-me sempre i ser un exemple a seguir.

Als meus avis Olfe i Gonzalo per donar-me tot el seu amor i al meu tiet Gonzalo per creure en mi i estar sempre al meu costat.

Els que m'han acompanyat de la mà durant aquests anys d'universitat i s'han tornat part de la meva família, sense ells no hagués arribat fins aquí: Paola i Xavi.

A les meves 3 grans amigues: Andrea, Lorena i Marta. Per la paciència, els ànims i el carinyo.

I als meus tutors, per donar-me l'oportunitat de fer aquest treball, gràcies.

ÍNDEX

INTRODUCCIÓ	9
CAPÍTOL 1. DESENVOLUPAMENT DE LA PÀGINA WEB	11
1.1. Programació.....	11
1.1.1. Llenguatge de programació.....	11
1.1.2. Entorns de desenvolupament i llibreries de Python	18
1.1.3. Creació del projecte amb PyCharm	22
1.2. Eines necessàries per a la creació de la pàgina web	26
1.2.1. Extensions del Visual Code Studio	27
1.2.2. Framework Flask	30
1.2.3. HTML	33
1.2.4. PyScript	38
1.3. Estructura de la pàgina web	39
CAPÍTOL 2. FUNCIONAMENT DE LA PÀGINA WEB.....	48
2.1. Disseny de la pàgina web	48
2.1.1. Pàgina d'inici.....	49
2.1.2. Pàgina principal	49
2.1.3. Encapçalat	49
2.1.4. Peu de pàgina	52
2.2. Publicació i accés al lloc web	53
2.2.1. Ngrok	53
2.2.2. Publicació del lloc web a Internet	54
CAPÍTOL 3. CONCLUSIONS	57
3.1. Treball futur	57
BIBLIOGRAFIA	59
CAPÍTOL 4. ANNEXOS	63
4.1. Annex 1. Conceptes termodinàmics.....	63
4.1.1. Lleis de la termodinàmica.....	63
4.1.2. Variables termodinàmiques	68
4.1.2. Processos termodinàmics	72
4.1.3. Cicles termodinàmics	77

INTRODUCCIÓ

La Termodinàmica sorgeix com a branca científica a començaments del segle XIX com a resposta als problemes derivats del desenvolupament de les emergents màquines tèrmiques [1]. No obstant això, el seu estudi ha evolucionat més enllà de la conversió d'energia calorífica en mecànica, i a partir dels seus principis s'estén a l'anàlisi de qualsevol transformació i transferència d'energia, la seva viabilitat energètica, reversibilitat i espontaneïtat. Els models i les Lleis de la Termodinàmica es fan servir per estimar propietats termodinàmiques de substàncies i mescles, i estudiar l'equilibri de reaccions químiques i processos relacionats, com operacions de mescla i separació. Tractant-se així d'una ciència amb un gran àmbit d'estudi que no només serveix de base per al disseny de màquines i motors tèrmics, sinó també de processos químics i tecnològics: l'enginyeria de la reacció química.

Els fonaments de la termodinàmica de l'enginyeria [2] juguen un paper important en l'avenç cap a un món millor, a través de la millora en el rendiment de la planta, l'equip i el seu disseny general. Els factors que són crítics per avaluar l'acompliment de l'equip són elements com la sortida del producte final, el consum de matèria primera d'entrada, el cost de producció i l'avaluació de l'efecte sobre el medi ambient. Els enginyers d'avui estan utilitzant el concepte de termodinàmica per examinar i reinventar les coses que estan destinades a la seguretat i la comoditat humanes.

En el sector de l'aviació la termodinàmica també juga un paper molt important, per aquest motiu l'objectiu del present Treball de Fi de Grau (TFG) és desenvolupar una pàgina web online que tracti d'un curs de termodinàmica pels estudiants de l'EETAC, aquesta pàgina també permet realitzar exercicis i simulacions de cicles i processos termodinàmics i exercicis de *turbojets* i *turbofans*. L'objectiu d'aquesta pàgina és que pugui ser una eina útil pels estudiants i també pels professors, agafant com a marc de referència l'assignatura de Termodinàmica del grau aeroespacial de l'EETAC (veure [3]).

Retornant al contexte de l'EETAC, en acabar l'assignatura de termodinàmica, l'estudiant/a ha de ser capaç de:

- Conèixer les lleis de la termodinàmica i la seva aplicació en sistemes tancats i oberts.
- Definir les variables termodinàmiques d'interès en enginyeria: Energia Interna, Entalpia i Entropia. Explicar el seu significat i la seva relació amb el manteniment i conservació dels dispositius en enginyeria.
- Explicar el comportament de líquids i gasos en diferents condicions de pressió i temperatura i els canvis de fase entre ells.
- Definir el rendiment ideal i isentròpic dels dispositius d'enginyeria: toveres, difusors, turbines, compressors, intercanviadors de calor, cambres de mescla, vàlvules d'estrenyiment, etc.

- Explicar el funcionament d'un motor tèrmic i calcular el seu rendiment. Explicar les diferències entre els motors que utilitzen diferents cicles termodinàmics (Otto, Diesel, Brayton) i els respectius camps d'aplicació.
- Explicar el principi de propulsió d'un motor aeronàutic i calcular l'empenta i la potència de propulsió en diferents condicions d'operació.
- Explicar les tecnologies de propulsió disponibles i les seves limitacions tecnològiques. Explicar les propietats bàsiques dels combustibles i les reaccions de combustió.
- Explicar la diferència entre aire sec i humit, i les seves propietats. Explicar el funcionament dels sistemes de condicionament d'aire d'edificis.
- Identificar els mecanismes físics de transmissió de calor. Explicar la seva importància relativa en diferents materials i geometries i calcular el flux de calor en els casos d'interès en enginyeria. Explicar la diferència entre règim estacionari i règim transitori de conducció de la calor.

La pàgina web no té tots els punts que s'acaben d'explicar degut a que es necessita molt més temps per poder fer-ho, però en un futur s'anirà ampliant el seu contingut per poder tenir un curs de termodinàmica de l'EETAC complet. Actualment estan els punts que he considerat més essencials. En l'Annex 1 s'expliquen els conceptes de termodinàmica que es troben a la pàgina web.

L'ús d'aplicacions per a al docència de la Física té un ampli recorregut en la literatura. Es poden trobar exemples que van des de la Física general com poden ser les simulacions PhET (veure [4]) o els treballs desenvolupats per el professor de la UPC Dr. Joan Soler Ruiz (veure [5]), entre molts d'altres. Val la pena destacar el JOC: JavaOptics Course (veure [6]) desenvolupat per la Universitat de Barcelona. El JOC és un curs interactiu en Java per a ensenyar Òptica Geomètrica i Física a nivell universitari i pre-universitari amb publicacions científiques (veure [7]). De fet, el JOC va inspirar la realització d'aquest treball.

Particularitzant en el cas de la Termodinàmica, són molt destacables els treballs de Ghajar (veure [8]) i de Brown(veure [9]), que es recomanen pel lector interessat en aprofundir en aquests aspectes.

La memòria consta de 2 capítols principals, el primer tracta de tota la programació del lloc web i de la part interna del projecte i el segon capítol del disseny, el funcionament i la publicació de la pàgina.

CAPÍTOL 1. DESENVOLUPAMENT DE LA PÀGINA WEB

Una pàgina web consisteix en una sèrie de codis que els navegadors poden llegir. En base a la informació que inclou aquest codi, podran mostrar la pàgina de forma adequada.

El codi li va dient al navegador quines parts són text, quines parts són referències, quines parts són imatges. També on s'ha de mostrar cada cosa, els colors, etc. Amb aquesta informació, el navegador després construeix el que es veu realment quan s'entra a una pàgina web.

1.1. Programació

La programació [10] és l'acte de programar, és a dir, organitzar una seqüència de passos ordenats que s'han de seguir per fer una cosa específica.

En el camp de la informàtica, la programació es refereix a la creació de programes o aplicacions mitjançant el desenvolupament de codi, que es basa en un conjunt d'instruccions que segueix una computadora per executar un programa. La programació és el que fa que una computadora funcioni i realitzi les tasques sol·licitades per l'usuari.

L'objectiu principal de la programació és definir instruccions perquè una computadora pugui executar sistemes, programes i aplicacions que siguin eficients, accessibles i fàcils d'usar.

1.1.1. Llenguatge de programació

El primer pas que vaig haver de fer per començar el projecte va ser decidir amb quin llenguatge de programació volia treballar. En informàtica, es coneix com a llenguatge de programació [11] un programa destinat a la construcció d'altres programes informàtics. El seu nom es deu a que comprèn un llenguatge formal que està dissenyat per organitzar algoritmes i processos lògics que seran després duts a terme per un ordinador o sistema informàtic, permetent controlar així el seu comportament físic, lògic i la seva comunicació amb l'usuari humà.

Aquest llenguatge està compost per símbols i regles sintàctiques i semàntiques, expressades en forma d'instruccions i relacions lògiques, mitjançant les quals es construeix el codi font d'una aplicació o peça de programari determinat. Es pot anomenar també llenguatge de programació al resultat final d'aquests processos creatius.

No s'han de confondre amb els diferents tipus de llenguatge informàtic que seran explicats més endavant. Aquests últims representen una categoria molt

més àmplia, on hi ha continguts els llenguatges de programació i molts altres protocols informàtics, com l'HTML de les pàgines web.

Els dos llenguatges de programació que vaig estudiar per realitzar el treball van ser Matlab i Python.

1.1.1.1. *Matlab*

Matlab [12] és un llenguatge de programació similar a altres llenguatges com Python, Java, C++, etc. Ve amb el seu propi IDE (entorn de desenvolupament integrat) i un conjunt de biblioteques.

Matlab és una abreviatura del terme *Matrix Lab* perquè originalment se'l coneixia com un llenguatge de programació matricial. En poques paraules, aquesta és una versió avançada de la calculadora que es pot executar en una computadora o dispositiu mòbil. Es pot utilitzar per realitzar operacions d'interfície de línia de comandaments, així com un editor de text. També es pot utilitzar per escriure programes i funcions que realitzen tasques contínues.

Algunes de les característiques principals de Matlab són les següents:

- Es pot comportar com una calculadora o com un llenguatge de programació.
- Combina molt bé el càlcul i el traçat gràfic.
- És relativament fàcil d'aprendre.
- S'interpreta (no es compila), els errors són fàcils de corregir.
- Està optimitzat per ser relativament ràpid quan es realitzen operacions matricials.
- Té alguns elements orientats a objectes.

Matlab fa que el seu codi sigui ràpid, aquest es compila amb un compilador JIT (*just-in-time*) les crides a la biblioteca s'optimitzen i les tasques per realitzar operacions matemàtiques es distribueixen entre els nuclis de la computadora. Matlab li permet executar algorismes en paral·lel, cosa que accelera l'execució.

L'IDE de Matlab proporciona accés a aplicacions interactives que permeten realitzar operacions computacionals de manera interactiva i proporciona imatges d'aquestes operacions. També pot crear les seves pròpies aplicacions i usar aplicacions desenvolupades per altres usuaris de Matlab.

Fent servir aquest llenguatge es pot implementar i dissenyar diferents algorismes. Pot carregar dades de diferents fonts, com arxius, bases de dades

o la web per analitzar les seves dades i visualitzar-les utilitzant l'aplicació de visualització Matlab que li ofereix una àmplia gamma de gràfics per triar.

Matlab és un programari que s'adapta a les necessitats dels processos de desenvolupament, recerca i anàlisi, en les enginyeries i les ciències. A continuació s'expliquen els avantatges [13] d'usar aquest llenguatge de programació:

- És un llenguatge de programació que permet expressar matrius matemàtiques i matrius vectorials directament. El mateix s'aplica a l'anàlisi de dades, processament de senyals i imatges, desenvolupament de control i altres aplicacions. Proporciona l'IDE més ràpid per al càlcul matemàtic de matrius i àlgebra lineal.
- Els noms de les funcions són familiars i fàcils de recordar, l'entorn d'escriptori està ajustat per a fluxos de treballs científics i d'enginyeria i la documentació està escrita per a enginyers i científics, no per a informàtics.
- Les aplicacions de Matlab són aplicacions interactives que combinen l'accés directe a grans col·leccions d'algoritmes amb retroalimentació visual immediata. Es pot visualitzar instantàniament com funcionen els diferents algoritmes amb les dades entrades i iterar fins a obtenir els resultats esperats i després generar automàticament un codi de Matlab per reproduir o automatitzar el treball.
- Conté les millors biblioteques de paquets matemàtics per proporcionar suport per a tots els camps de les matemàtiques, que van des de la suma simple fins a la inversió de matrius, etc.
- Proporciona suport multifil i recol·lecció d'escombraries per facilitar l'execució paral·lela d'algoritmes.
- El seu sistema de gràfics (Simulink) inclou comandaments per a visualització de dades bidimensionals i tridimensionals, processament d'imatges, presentació de gràfics i animació que proporciona una visualització d'alta qualitat de gràfics.
- Un equip d'enginyers de MathWorks verifica contínuament la qualitat del programari executant milions de proves sobre el codi base de Matlab tots els dies.

Matlab té molts avantatges, però també té alguns inconvenients [14]:

- Problemes eventuais de velocitat.
- Procés laboriós i susceptible a cometre errors en la programació d'algoritmes; és convenient depurar correctament els codis realitzats.
- Distribució d'executables.

- Cal utilitzar llibreries numèriques i gràfiques i és usual que depenguin del llenguatge de programació i del sistema operatiu.
- L'ordinador on l'aplicació ha de ser utilitzada necessita MCR (Matlab Component Runtime) perquè els arxius funcionin correctament.

1.1.1.2. Python

Python [15] és un llenguatge de programació comunament utilitzat per crear llocs web i programari, automatitzar tasques i analitzar dades. Python és un llenguatge de propòsit general, cosa que significa que es pot fer servir per crear molts programes diferents i no està dissenyat per resoldre un problema específic. Aquesta versatilitat, juntament amb la seva facilitat per als principiants, l'ha convertit en un dels llenguatges de programació més utilitzats en l'actualitat.

Python es fa servir comunament per al desenvolupament de llocs web i programari, automatització de tasques, visualització i anàlisi de dades.

Aquestes són algunes de les tasques que es poden realitzar amb aquest llenguatge:

- Anàlisi de dades i aprenentatge automàtic: Python s'ha convertit en un element bàsic en la ciència de dades, la qual cosa permet als analistes de dades i altres professionals utilitzar el llenguatge per realitzar càlculs estadístics complexos, crear visualitzacions de dades, crear algorismes d'aprenentatge automàtic, processar i analitzar dades i realitzar altres tasques relacionades amb les dades.
- Desenvolupament web: Python es fa servir sovint per desenvolupar el *backend* d'un lloc web o una aplicació, les parts que l'usuari no veu. El paper de Python en el desenvolupament web pot incloure l'enviament de dades cap a i des dels servidors, el processament de dades i la interacció amb les bases de dades, l'enrutament d'URL i la seguretat. Python ofereix diversos marcs de desenvolupament web. Els més utilitzats són Django i Flask.
- Automatització o *scripting*: Quan una tasca s'ha de fer repetidament, per treballar de manera més eficient es pot automatitzar-la amb Python, el codi utilitzat per crear aquests processos s'anomena *scripting*. L'automatització es pot fer servir per buscar errors en arxius, convertir arxius, executar matemàtiques simples i eliminar duplicats en les dades.
- Proves i prototips de programari: Python pot ajudar amb tasques com el control de compilació, el seguiment d'errors i les proves. Aquest llenguatge de programació permet als desenvolupadors de programari

automatitzar les proves de nous productes o funcions. Algunes eines de Python utilitzades per provar el programari són Green i Requestium.

Python té diversos usos en les empreses de desenvolupament de programari, com *frameworks*, aplicacions web, creació de prototips, etc. Això li dona a Python un avantatge competitiu sobre altres llenguatges de programació utilitzats en aquest camp. Moltes persones el fan servir a causa dels seus avantatges [16], entre ells destaquen els següents:

- És un llenguatge d'alt nivell, per la qual cosa és més fàcil d'usar que els llenguatges de baix nivell. Els elements del llenguatge natural es poden fer servir per a la programació de Python perquè la seva sintaxi és similar a l'anglès, la qual cosa facilita la lectura, l'escriptura i l'aprenentatge.
- Un dels grans actius de Python és la seva versatilitat i multiparadigma. És un llenguatge de propòsit general i es pot fer servir per a una varietat de propòsits. És una excel·lent opció per al desenvolupament de programari, també es pot fer servir per escriure *scripts* web i desenvolupar una GUI per a escriptors o processar dades. Així mateix, a Python se'l considera un llenguatge de paradigmes múltiples, que admet programació estructurada, funcional i orientada a objectes.
- Té una àmplia col·lecció de biblioteques i *frameworks*.
- És compatible amb tots els sistemes operatius (macOS, Linux, UNIX i Windows).
- És un llenguatge de programació desenvolupat sota la llicència de codi obert aprovada per OSI, tothom pot usar-lo i distribuir-lo lliurement.
- La simplicitat de la sintaxi de Python permet escriure programes completament funcionals en només unes poques línies de codi, per la qual cosa la corba d'aprenentatge és molt petita.

Python té molts avantatges, però també té alguns desavantatges, ja que no és un llenguatge perfecte. Python té els següents inconvenients [10]:

- La lentitud de Python es deu principalment a la seva naturalesa dinàmica i versatilitat.
- No és la millor opció quan la tasca requereix molta memòria. El consum de memòria de Python és molt alt, cosa que es deu a la flexibilitat dels tipus de dades.
- És excel·lent per a plataformes d'escriptori i servidors, però no és un llenguatge gaire adequat per al desenvolupament mòbil.

1.1.1.3. *Matlab vs Python*

Matlab i Python poden fer coses completament diferents i increïbles, el que fa que Matlab vs Python sigui una pregunta interessant. Doncs ara en aquest punt toca comparar un llenguatge amb l'altre per decidir amb quin dels dos dur a terme el projecte [17].

Python és un llenguatge de programació de propòsit general amb una gran quantitat de biblioteques que s'utilitzen per a una gran varietat d'aplicacions, com desenvolupament web, desenvolupament d'aplicacions empresarials i ciència de dades.

Python i la majoria de les seves biblioteques es poden descarregar i usar de forma gratuïta, tot i que molts usuaris fan servir Python a través d'un servei de pagament. Els serveis de pagaments ajuden les organitzacions de tecnologia de la informàtica a administrar els riscos associats amb l'ús de programari de codi obert, com seguretat, llicències i control d'accés. A més, proporcionen accés a distribucions empaquetades o allotjades, que eliminen gran part de la dificultat de configurar una instal·lació de Python amb versions compatibles de llibreries.

Matlab és una plataforma de càlcul que s'utilitza per a aplicacions científiques i d'enginyeria, com anàlisi de dades, processament de senyals i imatges, sistemes de control, comunicacions sense fil i robòtica. Inclou un llenguatge de programació, aplicacions interactives, biblioteques altament especialitzades per a aplicacions d'enginyeria i eines per generar automàticament codi en línia. Matlab també és la base de Simulink, un entorn de diagrama de flux per simular sistemes complexos de múltiples dominis.

Matlab no és gratuït [18], tot i que és més accessible i costa menys del que comunament es creu. Aproximadament 8 milions de persones tenen accés sense restriccions a Matlab a través de les seves escoles, instituts de recerca o empreses, incloses la majoria de les universitats d'enginyeria i ciències de tot el món. Les institucions trien invertir en Matlab perquè reconeixen el valor dels guanys de productivitat en usar eines especialment dissenyades.

Aquests dos llenguatges de programació es poden fer servir de forma interactiva (un comandament alhora) i poden desenvolupar aplicacions a gran escala. Ambdós ofereixen suport per a seqüències de comandaments, així com programació procedimental i orientada a objectes.

A continuació es mostren una sèrie de preguntes que em vaig haver de plantejar per prendre la decisió.

La primera pregunta que em vaig fer per escollir entre un dels dos és quin és el millor per a la productivitat. La resposta va ser Python. La sintaxi neta és fàcil de llegir i escriure, el que significa que es poden fer més coses.

Quin és el llenguatge més utilitzat per la gent que treballa en el món de la programació i la informàtica? Va ser la següent pregunta que em vaig fer, en aquest cas la resposta va ser que Python es considera la llengua franca per a la informàtica.

Quin és el millor pel processament d'imatges? El processament d'imatges és un mètode per extreure o millorar una imatge. Matlab és realment bo per dissenyar i afinar algoritmes. Ofereix un munt de eines per ajudar a processar imatges. Aquestes eines també es poden utilitzar per a la segmentació, extracció i anàlisi de dades d'imatges. Python ofereix alguns paquets externs pel processament d'imatges, com Numpy i SciPy. Doncs es pot veure que tant Matlab com Python són capaços de produir els mateixos resultats, però Matlab arriba més ràpidament.

Què és el millor per a l'aprenentatge automàtic? L'aprenentatge automàtic [19] (*Machine learning*) és una forma d'intel·ligència artificial que permet a un sistema aprendre de les dades en lloc d'aprendre mitjançant la programació explícita. La resposta a aquesta pregunta és Python. Python té un munt de biblioteques i paquets per l'aprenentatge automàtic. A més, és el llenguatge més utilitzat per a la investigació moderna d'aprenentatge automàtic a la indústria i l'acadèmia. Matlab també es pot utilitzar, disposa d'una gran col·lecció d'algorismes que poden oferir retroalimentació visual immediata. Però, aquests algorismes són propietaris, hi ha molt poca portabilitat de codi. Tot i que és més fàcil començar amb l'aprenentatge automàtic, realment no té futur tret que només es treballa amb altres persones que utilitzen Matlab.

Què és el millor per a l'aprenentatge profund? L'aprenentatge profund [20] (*deep learning*) és un subconjunt de l'aprenentatge automàtic, que és essencialment una xarxa neuronal amb tres o més capes. Aquestes xarxes neuronals intenten simular el comportament del cervell humà, encara que lluny de coincidir amb la seva capacitat, permetent aprendre a partir de grans quantitats de dades. Tot i que una xarxa neuronal amb una sola capa encara pot fer prediccions aproximades, les capes ocultes addicionals poden ajudar a optimitzar i perfeccionar la precisió. L'aprenentatge profund impulsa moltes aplicacions i serveis d'intel·ligència artificial que milloren l'automatització, realitzant tasques analítiques i físiques sense intervenció humana. Teòricament, els models d'aprenentatge profund haurien de ser capaços de determinar per si sols si les prediccions són precises, en canvi l'aprenentatge automàtic requereix una orientació més humana. Per aquesta raó es considera Python com la millor opció per a l'aprenentatge profund. Com que Matlab és propietari, significa que les innovacions són més lentes. Python és de codi obert, el que significa que qualsevol persona pot crear paquets que siguin àmpliament adoptats per la comunitat de Python. Paquets com PyTorch, Tensorflow, Caffe, etc. S'utilitzen per a l'aprenentatge profund.

Quin és el millor per a la velocitat? En els punts anteriors ja s'ha parlat de la velocitat dels dos llenguatges, i s'ha comentat que Matlab és molt ràpid, és indubtablement més ràpid que Python. Tot i això, cal assenyalar que Python té la possibilitat d'utilitzar més de 12 nuclis de processament quan s'executen treballs en paral·lel. Doncs s'arriba a la conclusió de que Matlab és més ràpid que Python, però Python és millor per executar diverses feines en paral·lel.

Una de les preguntes més importants és quin és el millor pel futur? Els llenguatges de programació van i venen. Aquells que semblen permanents encara poden començar a perdre influència i els nous desafien les

llengües existents tot el temps. La innovació, especialment en camps com la informàtica i la ciència de dades, és absolutament crítica per a la progressió. És per això que preguntar-se quin llenguatge és millor pel futur és realment important. Tot i que Matlab és més antic, crec que les llengües de codi obert són el futur. Els llenguatges i les interfícies de pagament no arribaran lluny a mesura que la ciència de dades i la informàtica es democratitza. Això es veu en l'acceleració d'aplicacions per a Python, MatLab no pot mantenir-se al dia perquè no és de codi obert. Python ha demostrat la seva capacitat per superar molts obstacles, inclòs el fiasco de Python 2 a Python 3, l'auge de les llengües més de moda que van desafiar el seu torn, etc. Matlab, per la seva banda, té una influència minvant fins i tot a les zones on abans dominava com els centres acadèmics. Per això la resposta a la pregunta és Python.

Després d'un amplia recerca d'aquest dos llenguatges vaig decidir utilitzar Python per dur a terme la pàgina web. Vaig escollir Python perquè és amb el llenguatge de programació que em sento més còmoda i també perquè a l'hora de comparar els dos llenguatges em va semblar que Python era millor per dissenyar i desenvolupar llocs web. Un altre motiu és que vaig veure que Python té unes llibreries de termodinàmica que inicialment em van semblar útils per realitzar els exercicis que inclou la pàgina web, però finalment no les he fet servir per crear el codi. La versió de Python que s'ha utilitzat és Python 3.9.7.

1.1.2. Entorns de desenvolupament i llibreries de Python

En els punts anteriors ja s'han explicat les principals característiques de Python, però en aquest punt s'ha aprofundit més i s'explica com s'ha treballat amb Python per obtenir l'objectiu del treball.

Per a aquest treball no s'ha utilitzat una versió "pura" de Python, sinó la distribució Conda, que permet crear entorns virtuals configurant diferents versions de les eines utilitzades i proporcionant eines de programació com Pycharm, que s'utilitzarà per a la programació en si.

Anaconda Navigator [21] és una interfície gràfica d'usuari d'escriptori (GUI) inclosa en Anaconda® que permet llançar aplicacions i gestionar paquets conda, entorns, i canals sense fer servir els comandaments d'ordres d'interfície de línia (CLI). Pot cercar paquets en Anaconda.org o en un repositori local d'Anaconda. A la figura 1.1 es veu els diferents entorns que té el navegador anaconda.

En un principi Conda va ser creada per al maneig de programes de Python, però actualment permet compilar i gestionar diferents tipus de programari. La distribució utilitzada és Anaconda®.

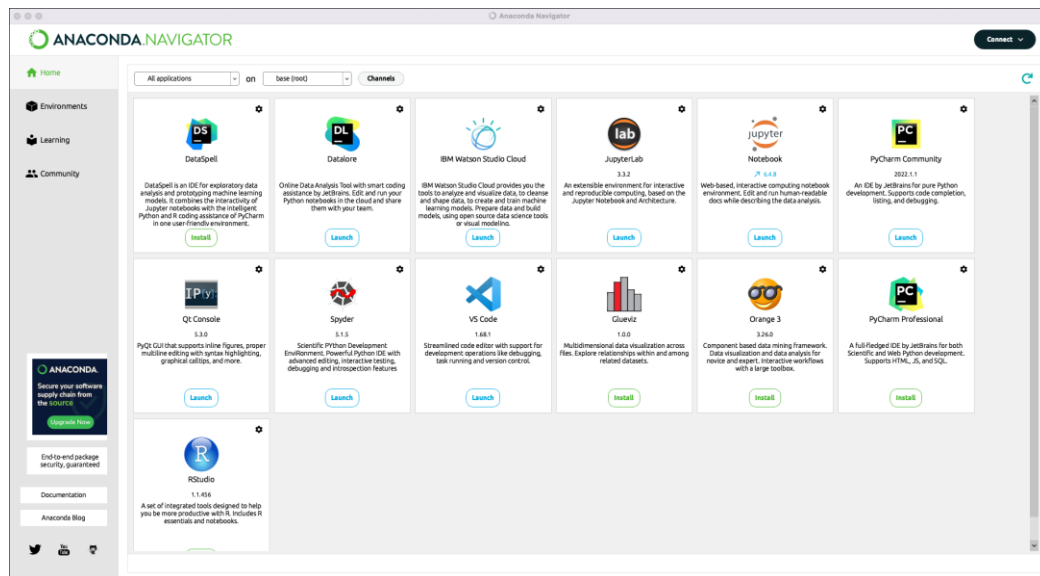


Fig. 1.1 Entorns que inclou el navegador anaconda [21].

Per a la programació es va instal·lar la versió 4.14.0 de Conda que ve junt amb la distribució Anaconda3 amb Python 3.9.7.

1.1.2.1. PyCharm

PyCharm és l'entorn de desenvolupament integra (IDE) que s'ha fet servir per escriure el codi dels exercicis de termodinàmica, és la primera eina de programació que s'utilitza al iniciar al projecte, en el punt 1.1.3 s'explica com es va escriure el codi.

Amb PyCharm [22] es poden crear aplicacions, pàgines web i treballar amb *data science* de manera eficient, ja que compta amb un suport per a cadascun d'aquests aspectes. La seva meta és millorar l'ús de Python i crear un ambient que inclogui tot el necessari per construir aplicacions simples o projectes complexos.

Fer els projectes de Python amb PyCharm permet aprofitar algunes característiques com:

- Editor intel·ligent: Ja integrat dins de l'aplicació, ajuda els programadors a llegir el codi més ràpid i eficientment, tant el seu com el d'altres.
- Depurador integrat: Revisa, prova i corregeix cada línia del codi que s'escriu.

- Vistes: Permet buscar segments específics en els programes, mostrar l'estructura del projecte i localitzar ràpidament elements clau amb el seu mode lent.
- Refactorització: Facilita la modificació de l'estructura del codi sense afectar el resultat final desitjat. Així es millora l'acompliment del sistema, aprofitant els recursos disponibles.
- Plantilles: Ideals per crear segments de codi amb poc esforç.

La funció de recerca facilita trobar codi en segons. Amb PyCharm és fàcil canviar el nom de les variables, facilita el treball dels seus usuaris amb dreceres de teclat que agilitzen les tasques repetitives durant el codi. Modificar l'aparença o agregar funcions personalitzades a PyCharm és factible gràcies a la gran varietat d'extensions disponibles. Algunes són creades per l'empresa matriu i altres per la comunitat d'usuaris freqüents de l'editor.

PyCharm té extensions útils per donar-li una nova dimensió al treball programat, a continuació es descriuen les que més s'utilitzen:

- Duckly: Elemental quan es treballa de manera remota, permet compartir la mateixa vista amb una altra persona mentre s'està treballant. Es poden fer correccions, revisions, depuracions i altres activitats.
- Mongo Plugin: Útil quan es treballa amb bases de dades MongoDB. Connexió, consultes i modificacions en integrar aquesta extensió en l'entorn de desenvolupament.
- ESLint: Per detectar els errors que hi hagi en el codi i corregir-los. Programa les teves pròpies condicions per a la seva execució i redueix la possibilitat d'errors en el projecte.
- EduTools: Brinda l'oportunitat d'aprendre amb tasques de programació senzilles, mentre ofereix la possibilitat de consultar material educatiu de qualitat.

Si parlem d'edició de codi, l'IDE integrat que ofereix Python en instal·lar-se és prou bo i simple per a edició de codi, però, no ofereix tots els avantatges que ofereix de sèrie PyCharm, havent d'acudir a altres eines per realitzar diferents tasques com pot ser crear mòduls o paquets.

1.1.2.2. Visual Studio Code

Com s'ha comentat en el punt interior, inicialment es va començar a fer el projecte amb PyCharm, però finalment l'editor de codi que s'ha fet servir per dur a terme la pàgina web ha sigut Visual Studio Code.

Visual Studio Code (VS Code) [23] és un editor creat per Microsoft per codificar. El programari és gratuït i es pot fer servir en múltiples plataformes com Windows, GNU/Linux i macOS. VS Code s'integra bé amb Git, ofereix depuració de codi i moltes extensions per escriure i executar codi en qualsevol llenguatge de programació.

VS Code té una gran varietat de característiques útils per agilitzar la feina:

- **IntelliSense:** Aquesta funció facilita l'escriptura de codi mitjançant l'edició, autocompletat i ressaltat de sintaxi. Aquest programari ofereix recomanacions intel·ligents de codis i termes basats en variables i funcions. Les extensions milloren l'IntelliSense i permeten la personalització de qualsevol llenguatge.
- **Depuració:** Ofereix depuració per detectar fallades al codi. Així, evitem revisar manualment línia per línia a la recerca d'errors. VS Code pot identificar errors petits de manera automàtica abans d'executar el codi.
- **Ús del control de versions:** És compatible amb Git, així que es poden revisar canvis amb git diff, organitzar i confirmar arxius des de l'editor, i actualitzar des de qualsevol servei de gestió de codi font (SMC). Altres opcions de SMC estan a disposició mitjançant extensions.
- **Extensions:** Visual Studio Code és un editor poderós a causa de les seves extensions. Les extensions són mòduls personalitzats que afegixen funcionalitats extra de manera aïllada. Exemples d'usos són programar en diversos llenguatges, afegir temes a l'editor i vincular amb altres serveis. Les extensions milloren l'experiència del usuari sense afectar el rendiment de l'editor, ja que s'executen en processos independents. La botiga d'extensions compta amb més de 16.500 extensions, les quals es poden explorar una a una o usar el cercador per trobar ràpidament la que es vol fer servir en funció de les necessitats del codificador.

Aquest editor inclou una terminal amb totes les funcions, la qual s'inicia fàcilment en el directori de treball. La terminal integrada pot utilitzar qualsevol *Shell* instal·lat a l'equip, com *PowerShell*, *Bash* o qualsevol altre. Tenir una terminal a l'editor és molt útil per executar comandaments necessaris durant el desenvolupament.

VS Code té sorprenents funcions útils per a qualsevol professional de la tecnologia de la informàtica (IT) i no està restringit a un sol tipus d'aplicació. Es pot accedir a màquines virtuals, contenidors i WSL (*Windows Subsystem for Linux*) de forma remota amb extensions apropiades de SSH. Això permet la

gestió des de la terminal i accés al sistema d'arxius. Es pot utilitzar per desenvolupar aplicacions en contenidors i administrar grups a *Kubernetes*. La integració de Microsoft Azure amb VS Code és excel·lent, cosa que proporciona més opcions de treball.

Com s'ha explicat anteriorment, VS Code és un editor de codi optimitzat que proporciona moltes facilitats per escriure, depurar i provar codi. És un editor de codi que està agafant molta força últimament és la versió gratuïta i *Open Source* del famós editor de Microsoft,. Aquesta versió és multiplataforma i aporta moltes més funcionalitats que l'IDE per defecte de Python. Els programadors busquen millorar la seva eficiència i rendiment, i Visual Studio Code ho permet amb les seves característiques, per això és l'entorn de desenvolupament que he escollit per dur a terme el treball.

1.1.2.3. *Llibreria Numpy*

NumPy [24] és una llibreria de Python per la computació científica. A més dels seus obvis usos científics, NumPy pot ser utilitzat com un eficient contenidor multidimensional de dades genèriques. Tipus de dades arbitràries poden ser definides. Això permet que NumPy s'integri sense problemes i amb rapidesa amb una àmplia varietat de bases de dades.

El principal avantatge de NumPy és que permet generar i processar dades de forma extremadament ràpida. NumPy té la seva pròpia estructura de dades integrada anomenada *array*, que és similar a una llista normal de Python però pot emmagatzemar i manipular dades de manera molt més eficient.

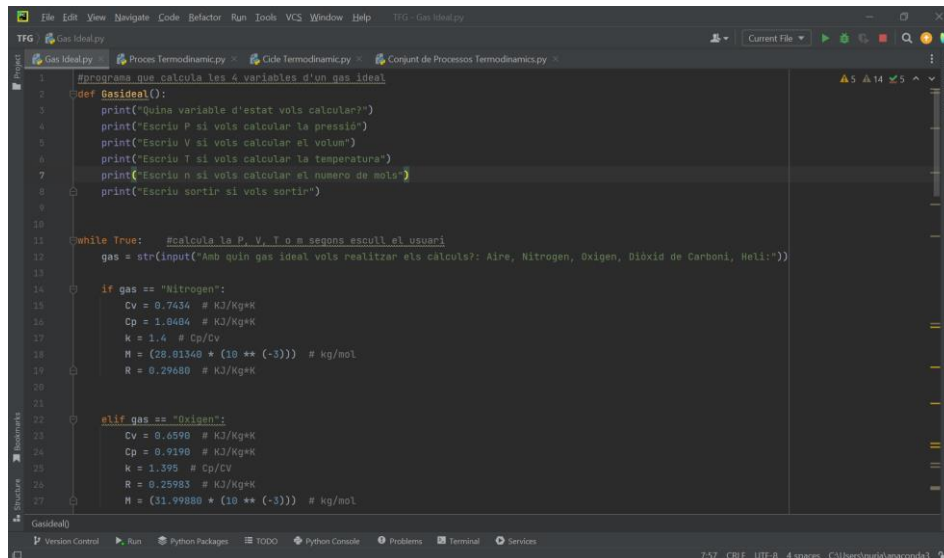
Per utilitzar la llibreria NumPy en el arxiu de Python es fa servir el següent comandament: *import numpy as np*.

1.1.2.4. *Llibreria Matplotlib*

Matplotlib és una llibreria de Python especialitzada en la creació de gràfics en dues dimensions. Per utilitzar la llibreria Marplotlib en un arxiu de Python es fa servir el següent comandament: *import matplotlib as plt*.

1.1.3. **Creació del projecte amb PyCharm**

Després de configurar i obrir PyCharm el primer que es va fer va ser crear un arxiu anomenat GasIdeal.py i crear una funció anomenada Gasideal, com es pot veure en la figura 1.2.



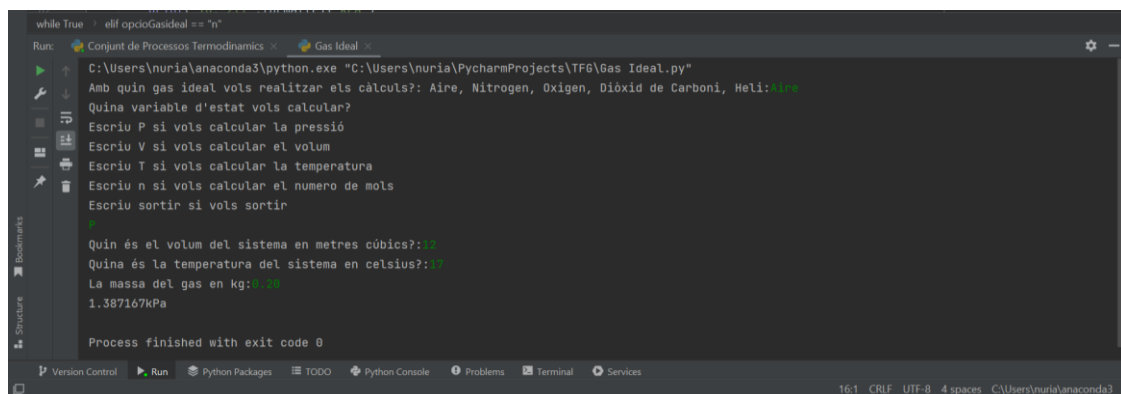
```

1 #programa que calcula les 4 variables d'un gas ideal
2 def Gasideal():
3     print("Quina variable d'estat vols calcular?")
4     print("Escriu P si vols calcular la pressió")
5     print("Escriu V si vols calcular el volum")
6     print("Escriu T si vols calcular la temperatura")
7     print("Escriu n si vols calcular el numero de mols")
8     print("Escriu sortir si vols sortir")
9
10
11 while True: #calcula la P, V, T o n segons escull el usuari
12     gas = str(input("Amb quin gas ideal vols realitzar els càlculs?: Aire, Nitrogen, Oxigen, Diòxid de Carboni, Heli:"))
13
14     if gas == "Nitrogen":
15         Cv = 0.7434 # KJ/kgK
16         Cp = 1.0484 # KJ/kgK
17         k = 1.4 # Cp/Cv
18         M = (28.01340 * (10 ** (-3))) # kg/mol
19         R = 0.29680 # KJ/kgK
20
21     elif gas == "Oxigen":
22         Cv = 0.6590 # KJ/kgK
23         Cp = 0.9190 # KJ/kgK
24         k = 1.395 # Cp/Cv
25         R = 0.25983 # KJ/kgK
26         M = (31.99880 * (10 ** (-3))) # kg/mol
27

```

Fig. 1.2 Arxiu Gas Ideal.py.

Aquesta funció el que fa és calcular una variable termodinàmica amb la fórmula dels gasos ideals. L'usuari ha de escollir amb quin gas vol treballar i quina variable vol calcular, a la figura 1.3 es mostra un exemple de com funciona.



```

while True: elif opcióGasideal == "n":
Run: C:\Users\nuria\anaconda3\python.exe "C:\Users\nuria\PycharmProjects\TF6\Gas Ideal.py"
Amb quin gas ideal vols realitzar els càlculs?: Aire, Nitrogen, Oxigen, Diòxid de Carboni, Heli:Heli
Quina variable d'estat vols calcular?
Escriu P si vols calcular la pressió
Escriu V si vols calcular el volum
Escriu T si vols calcular la temperatura
Escriu n si vols calcular el numero de mols
Escriu sortir si vols sortir
Quin és el volum del sistema en metres cúbics?:1
Quina és la temperatura del sistema en celsius?:10
La massa del gas en kg:1
1.387167kPa
Process finished with exit code 0

```

Fig. 1.3 Funció Gasideal.

Després es van crear dos arxius més, un que serveix per calcular processos termodinàmics i un altre per calcular els cicles. En aquest arxiu s'utilitzen dues llibreries: NumPy i Matplotlib.

Els dos arxius comentats anteriorment tenen un funcionament molt semblant al del GasIdeal.py, en la figura 1.4 es mostra un exemple de com funciona l'arxiu Proces Termodinamica.py i en la figura 1.5 es mostra el diagrama de pressió-volum del procés escollit. En la figura 1.6 es mostra un exemple de com funciona l'arxiu Cicle Termodinamica.py i en la figura 1.7 i en la figura 1.8 es

representen els gràfics de pressió-volum i de temperatura-entropia del cicle escollit respectivament.

```

while True:
    elif Procés == "Adiabàtic":
        if W > 0:
            Run: Conjunt de Processos Termodinàmics x
            Procés Termodinàmic x
            C:\Users\nuria\anaconda3\python.exe "C:\Users\nuria\PycharmProjects\TF6\Procés Termodinàmic.py"
            Amb quin gas ideal vols realitzar els càlculs?: Aire, Nitrogen, Oxigen, Diòxid de Carboni, Heli: Nitrogen
            Quin procés termodinàmic vols realitzar?
            Isobàric, Adiabàtic, Isocòric o Isotèrmic
            Isobàric, Adiabàtic, Isocòric o Isotèrmic
            Escriu sortir per sortir del programa
            sortir
            Quin és el volum inicial del sistema en metres cúbics?: 10
            Quina és la temperatura inicial del sistema en celsius?: 0
            Quina és la pressió inicial del sistema en KPA?: 100
            Saps el volum, la pressió o la temperatura final del procés?, Escriu pressió, volum o temperatura: volum
            Quin és el volum final del sistema en metres cúbics?: 200
            El valor de la pressió final és: 1.5085440841362914 KPA. I el valor de la temperatura final és: 85.9870127957686 K
            El treball realitzat per aquest procés és: 1745.7279579318547 kJ. Es tracta d'una expansió adiabàtica
  
```

Fig. 1.4 Funció ProcésTermodinamic.

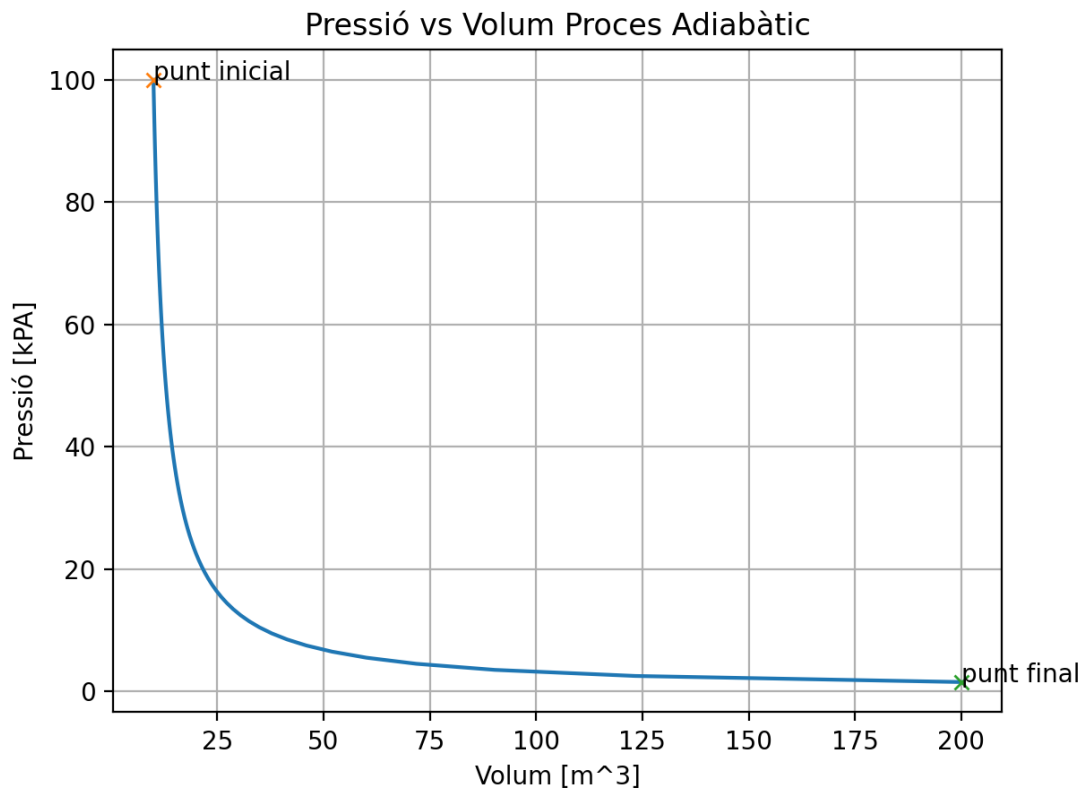


Fig. 1.5 Gràfic pressió-volum del procés seleccionat.


```

while True:
    if Cicle == "Carnot":
        Run: Conjunt de Processos Termodinamics Cicle Termodinamic
        C:\Users\nuria\anaconda3\python.exe "C:\Users\nuria\PycharmProjects\TF6\Cicle Termodinamic.py"
        Amb quin gas ideal vols realitzar el proces?: Aire, Nitrogen, Oxigen, Diòxid de Carboni, Heli: Aire
        Quin cicle termodinàmic vols realitzar? Carnot, Diesel, Otto o Brayton
        Escriu sortir per sortir del programa
        Escriu:
        Massa del gas en Kg: 10
        Calor utilitzat a l'expansió isotèrmica en KJ: 10
        Eficiència de la màquina en %: 0
        Quin és el volum inicial del cilindre-pistó en metres cúbics?: 0.1
        Quina és la pressió inicial del cilindre-pistó en KPA?: 400
        El treball net que realitza la màquina és: 18.000000000000007 kJ
        La variació d'energia interna total és: 0.0 kJ
        La entropia total del cicle de carnot és: 0.0 kJ/K
    
```

Fig. 1.6 Funció CicleTermodinamic.

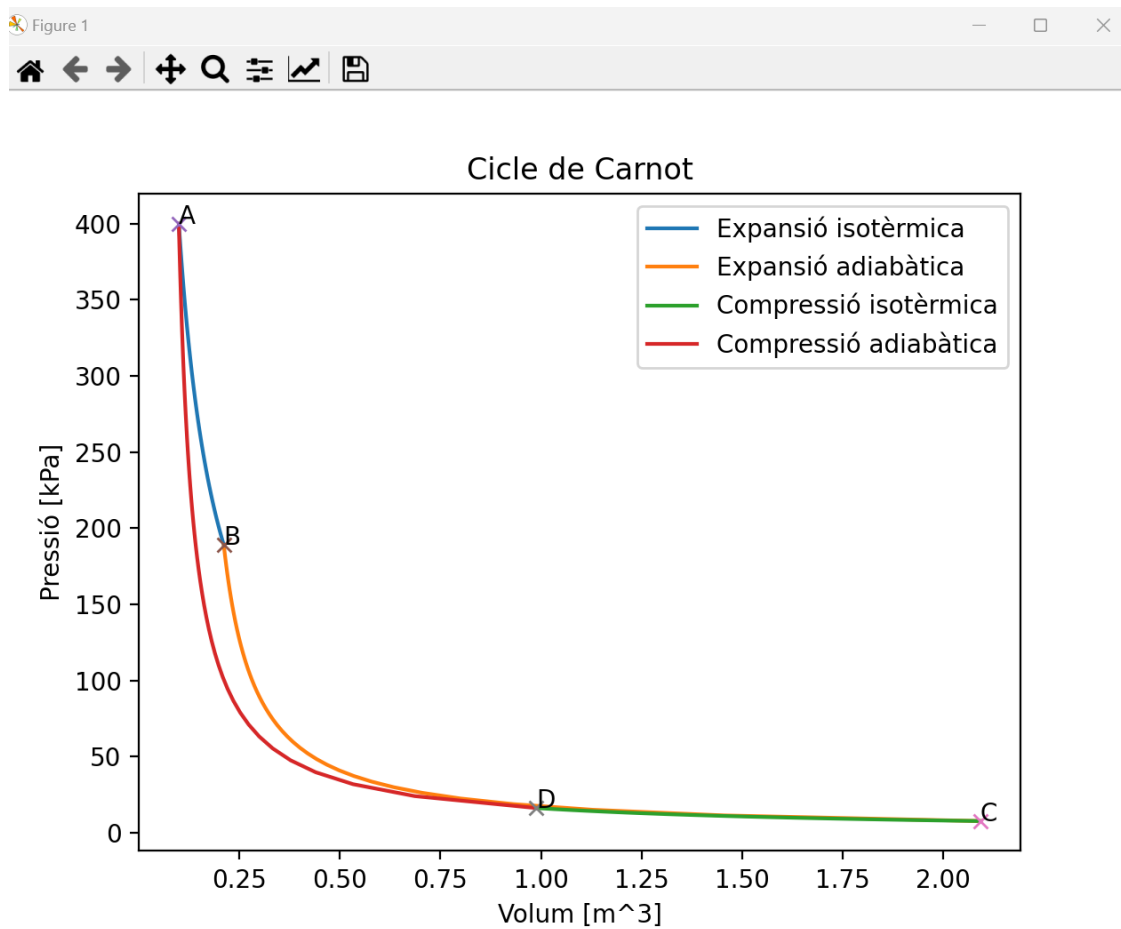


Fig. 1.7 Gràfic pressió-volum del cicle seleccionat.

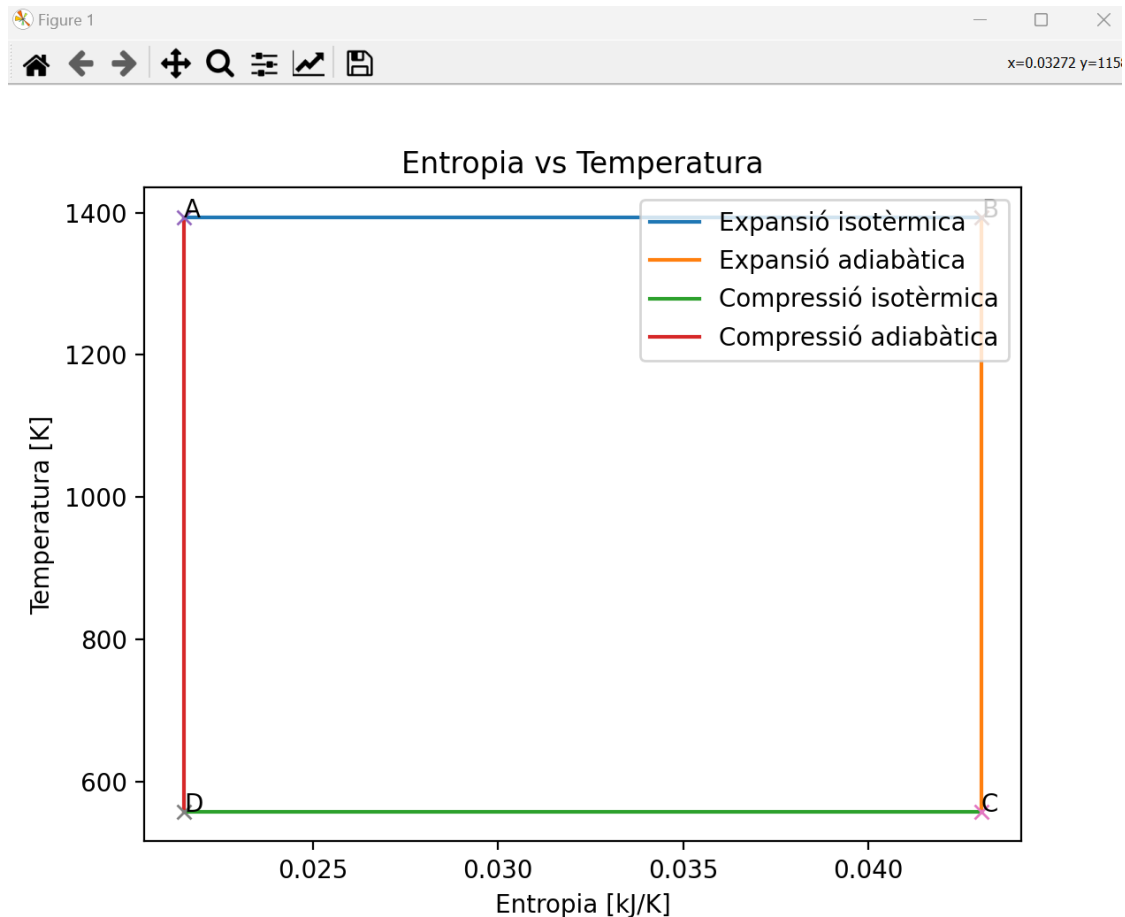


Fig. 1.8 Gràfic temperatura-entropia del cicle seleccionat.

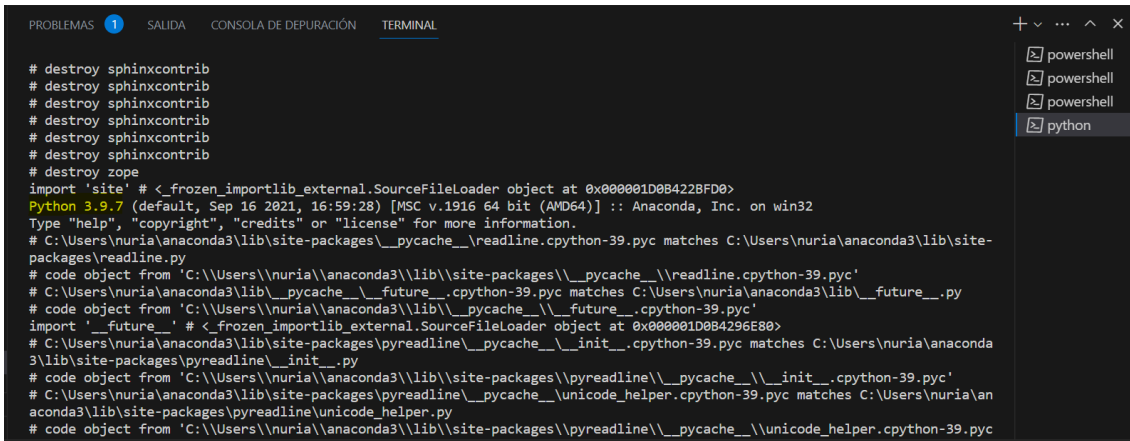
Aquesta va ser la primera idea de com serien els exercicis que es volien a la pàgina web i a partir de aquí es va començar a treballar amb el Visual Studio Code.

1.2. Eines necessàries per a la creació de la pàgina web

Desenvolupar una pàgina web amb Python implica codificar pel servidor en comptes del navegador per aconseguir un disseny adaptable. L'ús de Python en el desenvolupament web ha augmentat molt, per la qual cosa hi ha molts mòduls dissenyats per ajudar en la creació de pàgines web funcionals.

En el desenvolupament web amb Python, hi ha molts *frameworks* que dominen l'escena: Django, WebApp2, Pyramid, Flask, TurboGears, Bottle, i CherryPy. En el treball el *frameworks* que s'ha utilitzat és Flask, en el següent punt s'explica com funciona.

El primer pas per començar a crear la pàgina web és crear una carpeta en el l'escriptori anomenada *sitioweb*, doncs aquesta carpeta es va obrir amb el VS Code. A continuació s'ha de mirar que la ubicació del intèrpret de Python estigui inclosa a la variable d'entorn PATH, això es fa obrint un terminal i executant el següent comandament: `python -v`, en la figura 1.9 es mostra la versió de Python que té instal·lada.



```
PROBLEMAS 1 SALIDA CONSOLA DE DEPURACIÓN TERMINAL
# destroy sphinxcontrib
# destroy sphinxcontrib
# destroy sphinxcontrib
# destroy sphinxcontrib
# destroy sphinxcontrib
# destroy sphinxcontrib
# destroy zope
import 'site' # < frozen_importlib_external.SourceFileLoader object at 0x000001D084228FD0>
Python 3.9.7 (default, Sep 16 2021, 16:59:28) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
# C:\Users\nuria\anaconda3\lib\site-packages\__pycache__\readline.cpython-39.pyc matches C:\Users\nuria\anaconda3\lib\site-packages\readline.py
# code object from 'C:\Users\nuria\anaconda3\lib\site-packages\__pycache__\readline.cpython-39.pyc'
# C:\Users\nuria\anaconda3\lib\__pycache__\__future__.cpython-39.pyc matches C:\Users\nuria\anaconda3\lib\__future__.py
# code object from 'C:\Users\nuria\anaconda3\lib\__pycache__\__future__.cpython-39.pyc'
import '__future__' # < frozen_importlib_external.SourceFileLoader object at 0x000001D084296E80>
# C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\__pycache__\__init__.cpython-39.pyc matches C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\__init__.py
# code object from 'C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\__pycache__\__init__.cpython-39.pyc'
# C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\__pycache__\unicode_helper.cpython-39.pyc matches C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\unicode_helper.py
# code object from 'C:\Users\nuria\anaconda3\lib\site-packages\pyreadline\__pycache__\unicode_helper.cpython-39.pyc'
```

Fig. 1.9 Comanda `python -v` i versió de Python.

El següent pas és executar el comandament `pip list` això serveix per veure quins paquets té instal·lats el VS Code, s'ha de tornar a mirar de que Python estigui instal·lat correctament.

A continuació es revisen les extensions del Visual Code Studio i s'instal·len les que són necessàries i les que poden ajudar a crear el codi de forma més ràpida.

1.2.1. Extensions del Visual Code Studio

En aquesta punt es mostren i s'expliquen les extensions que es van haver d'instal·lar a Visual Studio Code per treure-li el màxim profit, augmentar la productivitat del desenvolupament d'aplicacions i evitar errors innecessaris durant la codificació utilitzant la terminació de codi.

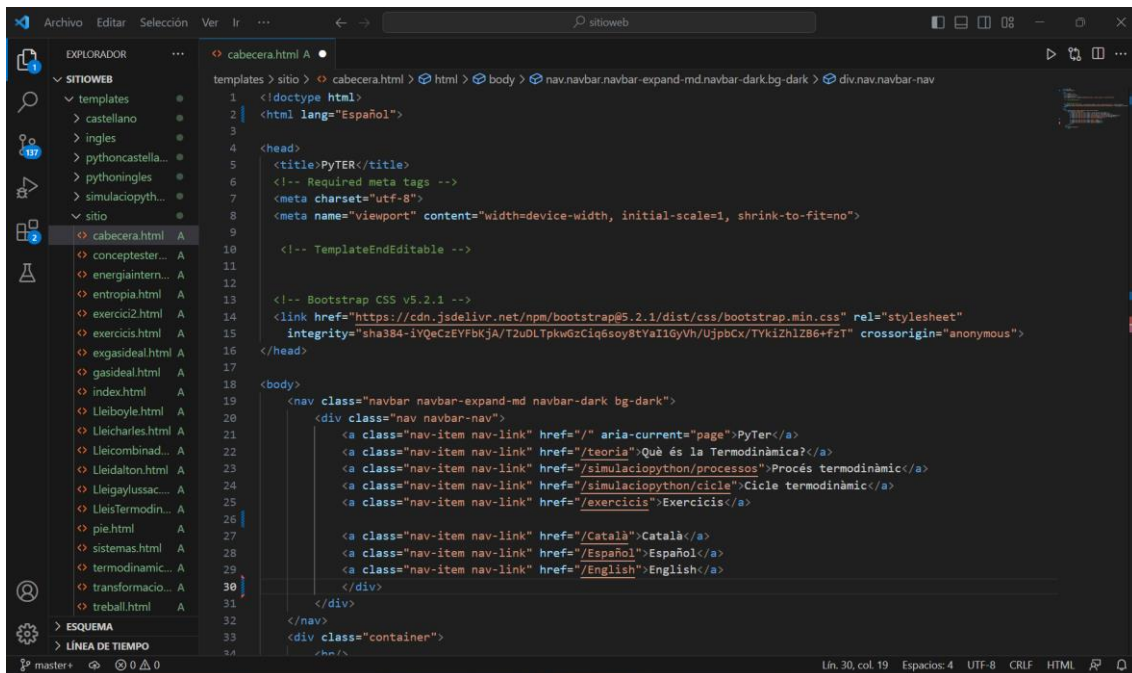
1.2.1.1. Bootstrap 5 Quick Snippets

Bootstrap [25] és una biblioteca d'eines de codi obert optimitzada pel disseny del *front-end*, és a dir, a la pantalla d'interfície amb l'usuari de llocs i aplicacions. Aquesta plataforma es basa en llenguatge HTML i CSS, i inclou

una àmplia gamma d'elements de disseny, com formularis, botons i menús que s'adapten a diferents formats de navegació.

Bootstrap va ser desenvolupat originalment per GitHub per resoldre problemes de navegació mòbil. Això significa que el codi està optimitzat per a dispositius amb pantalles petites i superfícies tàctils. A partir d'aquí, Bootstrap s'ha expandit a altres formats com pantalles d'escriptori. Això li brinda a l'usuari la tranquil·litat de saber que el seu lloc es carregarà correctament i crearà interfícies veritablement interactives quan es construeixi amb Bootstrap.

L'extensió de Bootstrap que s'ha fet servir per crear l'estructura de la capçalera de la pàgina web, en la figura 1.10 es pot observar el codi i en la figura 1.11 es mostra com es veu la capçalera a la pàgina web. Els comandaments que s'han utilitzat a la capçalera són *bs5- $\$$* , que serveix per crear una estructura i *bs5-nav-minimal-a* que serveix per crear la capçalera i fer que cada apartat de la capçalera sigui un botó i ens redirigeixi a un nou arxiu HTML.



```

1 <!doctype html>
2 <html lang="Español">
3
4 <head>
5 <title>PyTER</title>
6 <!-- Required meta tags -->
7 <meta charset="utf-8">
8 <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
9 <!-- TemplateEndEditable -->
10
11 <!-- Bootstrap CSS v5.2.1 -->
12 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap.min.css" rel="stylesheet"
13 integrity="sha384-1yQeCzEVFBKJA/T2uDLTpkwGzCiq6soy8tYaIIGyVh/UjpbCx/TYkiZHLZ86+FzT" crossorigin="anonymous">
14 </head>
15
16 <body>
17 <nav class="navbar navbar-expand-md navbar-dark bg-dark">
18 <div class="nav navbar-nav">
19 <a class="nav-item nav-link" href="/" aria-current="page">PyTer</a>
20 <a class="nav-item nav-link" href="/teoria">Què és la Termodinàmica?</a>
21 <a class="nav-item nav-link" href="/simulaciopython/processos">Procés termodinàmic</a>
22 <a class="nav-item nav-link" href="/simulaciopython/cicle">Cicle termodinàmic</a>
23 <a class="nav-item nav-link" href="/exercicis">Exercicis</a>
24 <a class="nav-item nav-link" href="/Català">Català</a>
25 <a class="nav-item nav-link" href="/Español">Español</a>
26 <a class="nav-item nav-link" href="/English">English</a>
27 </div>
28 </nav>
29 <div class="container">
30
31 </div>
32 </body>
33 </html>

```

Fig. 1.10 Codi de l'extensió de Bootstrap per la capçalera.

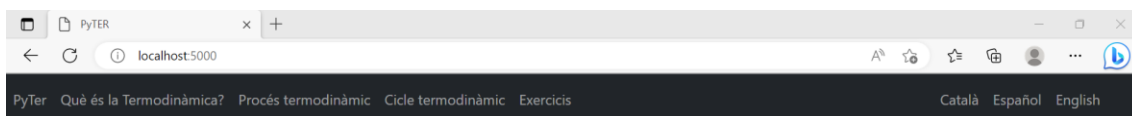


Fig. 1.11 Visualització de la capçalera a la pàgina web.

Aquesta extensió també s'ha fet servir per crear els botons de la pàgina web i les taules. Pels botons s'han utilitzat dos tipus d'extensions, la *bs5-button-a* aquesta és vàlida quan el propòsit del botó és enviar a l'usuari a un altre arxiu HTML, la seva estructura es pot veure a la línia superior de la figura 1.12 i l'altre és *bs5-button-default* que serveix per enviar dades al servidor, la seva estructura es pot veure a la línia inferior de la figura 1.12.

Per les taules s'ha utilitzat l'extensió *bs5-table-default*, la seva estructura es mostra a la figura 1.13.

```
<a name="" id="" class="btn btn-primary" href="#" role="button">Button</a>  
<button type="button" class="btn btn-primary">Button</button>
```

Fig.1.12 Botons de Bootstrap.

```
263  
264 <div class="table-responsive">  
265   <table class="table table-primary">  
266     <thead>  
267       <tr>  
268         <th scope="col">Column 1</th>  
269         <th scope="col">Column 2</th>  
270         <th scope="col">Column 3</th>  
271       </tr>  
272     </thead>  
273     <tbody>  
274       <tr class="">  
275         <td scope="row">R1C1</td>  
276         <td>R1C2</td>  
277         <td>R1C3</td>  
278       </tr>  
279       <tr class="">  
280         <td scope="row">Item</td>  
281         <td>Item</td>  
282         <td>Item</td>  
283       </tr>  
284     </tbody>  
285   </table>  
286 </div>  
287
```

Fig. 1.13 Taules de Bootstrap.

En la pàgina oficial de Bootstrap (veure [25]), hi ha uns quants exemples de com utilitzar aquesta extensió i les plantilles que té.

1.2.1.2. Extensió Python

Treballar amb Python al Visual Studio Code, mitjançant l'extensió Microsoft Python [26], és senzill i productiu. L'extensió fa de VS Code un excel·lent editor de Python, i funciona en qualsevol sistema operatiu amb una varietat

d'intèrprets de Python. L'extensió de Python instal·la més extensions de forma automatitzada.

L'extensió Python admet la compleció de codi i *IntelliSense* mitjançant l'intèrpret seleccionat actualment. *IntelliSense* és un terme general per a diverses funcions, inclosa la compleció intel·ligent de codi (mètode en context i suggeriments de variables) en tots els fitxers i per a mòduls integrats i de tercers. *IntelliSense* mostra ràpidament mètodes, membres de la classe i documentació mentre s'escriu.

Linting analitza el codi de Python en busca de possibles errors, cosa que facilita la navegació i la correcció de diferents problemes.

VS Code inclou un gran suport de depuració per a Python, que permet establir punts d'interrupció, inspeccionar variables i utilitzar la consola de depuració per veure en profunditat com s'executa el programa pas a pas. Depura diversos tipus d'aplicacions Python, incloses aplicacions multifil, web i remotes.

L'extensió Python detecta automàticament intèrprets de Python instal·lats en ubicacions estàndard. També detecta entorns conda així com entorns virtuals a la carpeta de l'espai de treball.

L'extensió de Python l'he utilitzat en l'arxiu *app.py* aquest arxiu consta de diferents funcions en la que cada funció té una ruta assignada i un arxiu HTML, doncs al executar la funció sobre l'arxiu amb la ruta assignada.

1.2.2. Framework Flask

Un *framework* es tracta d'una estructura o plantilla que els programadors utilitzen per desenvolupar un programari, crear una aplicació, etc. Es pot definir com un esquema o marc de treball que ofereix una estructura base per elaborar un projecte amb objectius específics. Generalment, els *framework* són usats per programadors perquè permeten accelerar el treball i afavorir que aquest sigui col·laboratiu, reduir errors i obtenir un resultat de més qualitat.

El *framework* que he utilitzat s'anomena Flask, aquest ens ajudar a agilitzar el procés de construcció de codi a través de Python. Per instal·lar Flask en VS Code s'ha d'obrir la terminal i escriure *pip install flask*, un cop hem fet aquest comandament ja estan les extensions disponibles. El següent pas va ser a instal·lar *MySQL* per utilitzar la base de dades *MySQL* mitjançant la terminal amb el següent comandament *pip install flask-mysql*.

Flask [27] és un "micro" *framework* de Python que ajuda al desenvolupament d'aplicacions web amb el patró MVC. La paraula "micro" no vol dir que sigui un projecte petit o que ens permet fer pàgines web petites sinó que en instal·lar Flask tenim les eines necessàries per crear una aplicació web funcional, però si es necessita en algun moment una nova funcionalitat hi ha un conjunt molt gran que es poden instal·lar amb Flask.

MVC és un mètode per separar el model de dades, la vista i el controlador en aplicacions web. El model és la informació emmagatzemada en una base de dades, la vista és la pàgina HTML i el controlador gestiona les sol·licituds de l'aplicació.

Aquest *framework* inclou un servidor web de desenvolupament, això vol dir que no cal una infraestructura amb servidor web per provar aplicacions. És possible correr un servidor web de manera fàcil per veure els resultats obtinguts.

Compta amb depurador i suport per a proves unitàries. Si hi ha errors en el codi en construcció, es poden depurar i visualitzar els valors de les variables. També s'hi poden afegir proves unitàries.

1.2.2.1. MySQL

MySQL [28] és un sistema de gestió de bases de dades de codi obert amb un model client/servidor, les seves sigles en anglès són RDBMS. Un RDBMS és un programari o servei per crear i administrar bases de dades basades en un model relacional.

En un entorn RDBMS els clients i els servidors es comuniquen utilitzant un llenguatge específic de domini, el llenguatge de consulta estructurat (SQL). SQL li diu al servidor que fer amb les dades. És similar a la funció d'una contrasenya o codi de WordPress. S'ingressa en el sistema per obtenir accés a l'àrea del panell de control. En aquest cas, les declaracions de SQL poden indicar-li al servidor que realitzi certes operacions:

- Consulta de dades: Demanar informació específica de la base de dades existent.
- Manipulació de dades: Agregar, eliminar, canviar, ordenar i altres operacions per modificar les dades, els valors o els elements visuals.
- Identitat de dades: Definir tipus de dades, per exemple, canviar dades numèriques a nombres enters. Això també inclou la definició d'un esquema o la relació de cada taula a la base de dades.
- Control d'accés a les dades: Proporcionar tècniques de seguretat per protegir les dades, la qual cosa inclou decidir qui pot veure o usar qualsevol informació emmagatzemada a la base de dades.

El sistema de gestió de base de dades MySQL funciona de la següent manera:

1. MySQL crea una base de dades per emmagatzemar i manipular dades, definint la relació de cada taula.
2. Els clients poden realitzar sol·licituds escrivint instruccions SQL específiques a MySQL.

3. L'aplicació del servidor respondrà amb la informació demanada i aquesta apareixerà davant els clients.

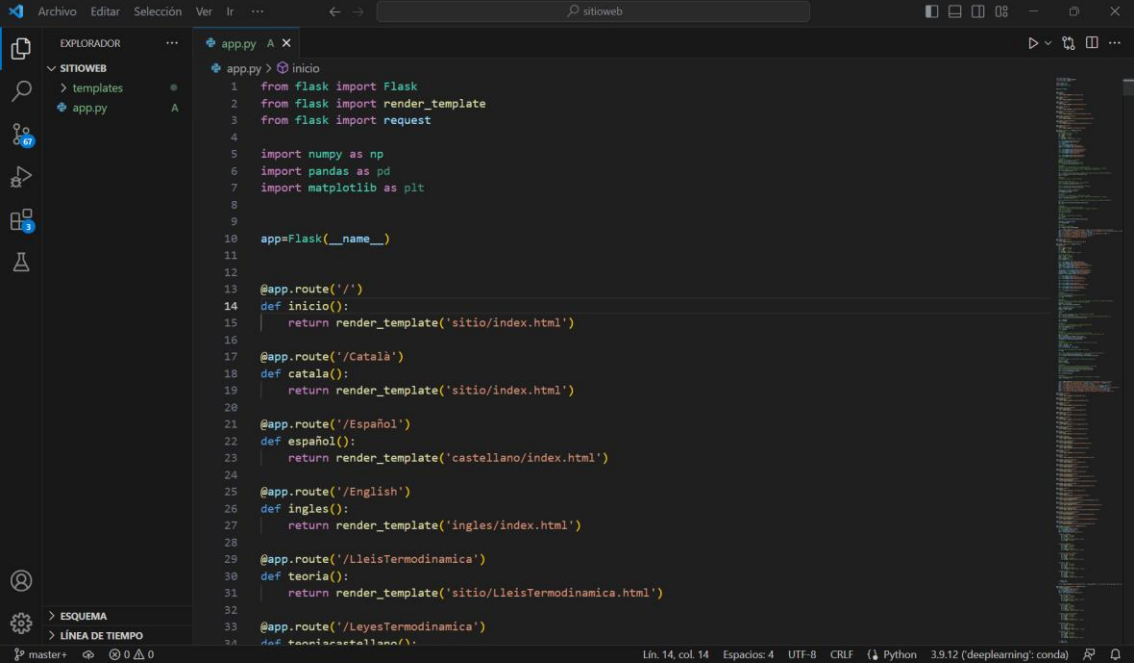
1.2.2.2. *Render template*

El render *template* de Flask [29] es pot definir com una funció que s'encarrega de presentar una plantilla en el text realitzat amb HTML, el qual permet generar una resposta. Això ocasiona una sortida de la plantilla a partir de la dependència on es troba un paquet de plantilles, és a dir, de la dependència de Jinja. Jinja és un paquet de plantilles per a Python que es troba configurat automàticament en el *framework* Flask.

El primer pas és crear una plantilla HTML. Aquesta plantilla ha de contenir etiquetes especials per a les variables de plantilla. Aquestes etiquetes es reemplacen amb dades dinàmiques de l'aplicació. Un cop la plantilla HTML està llesta, el següent pas és escriure el codi de l'aplicació per renderitzar la plantilla HTML.

El procés per renderitzar la plantilla de Flask en l'aplicació web és bastant senzill. Aquest procés s'ha de realitzar en el fitxer que controla les vistes del programa de codi, és a dir, en el fitxer `app.py`. Una vegada creat aquest fitxer, s'ha d'importar des de la dependència de la Flask el mètode `render_template`. Això es fa usant la funció `render_template()`, aquesta funció pren la plantilla HTML i les dades de l'aplicació com a arguments i retorna l'HTML renderitzat. L'HTML renderitzat s'envia al navegador web de l'usuari i el navegador web l'interpreta i mostra la pàgina web a l'usuari.

A la figura 1.14 es mostra el fitxer `app.py`, que com s'ha explicat és el fitxer que controla les visites. En la primer línia de codi es mostra com s'ha importat la funció a través del comandament `from flask import render_template`. En les altres línies del codi es mostra com es fa servir aquesta funció.



```
1 from flask import Flask
2 from flask import render_template
3 from flask import request
4
5 import numpy as np
6 import pandas as pd
7 import matplotlib as plt
8
9
10 app=Flask(__name__)
11
12
13 @app.route('/')
14 def inicio():
15     return render_template('sitio/index.html')
16
17 @app.route('/Català')
18 def catala():
19     return render_template('sitio/index.html')
20
21 @app.route('/Español')
22 def español():
23     return render_template('castellano/index.html')
24
25 @app.route('/English')
26 def ingles():
27     return render_template('ingles/index.html')
28
29 @app.route('/LleisTermodinamica')
30 def teoria():
31     return render_template('sitio/LleisTermodinamica.html')
32
33
34 @app.route('/LeyesTermodinamica')
35 def teoriaCastellano():
```

Fig. 1.14 Funció `render_template`.

1.2.2.3. Request

Un objecte *request* [30] conté tota la informació que un client (com un navegador web) envia a un servidor (en aquest cas, una aplicació Flask). Aquesta informació inclou encapçalats HTTP, paràmetres d'URL, codificació preferida i, per suposat, les dades enviades a través del formulari. Per poder utilitzar l'objecte *request* s'ha d'importar-lo al principi del programa Flask, a través del comandament *from flask import request*.

Una de les de les funcions que pot fer l'objecte *request* és saber quin tipus de sol·licitud està realitzant el client (GET, POST, DELETE, etc.), per això aquest objecte proporciona un atribut *.method*.

L'objecte *request* serveix per accedir a la informació que envia el client. La principal informació que envia són els paràmetres, ja siguin paràmetres tipus GET o paràmetres tipus POST.

1.2.3. HTML

HTML (*HyperText Markup Language*) [31] no és en sentit estricte un llenguatge de programació, sinó un llenguatge de marcatge que desplega i estructura una pàgina o l'aplicació web, però sense donar-li format, ni estil, ni funcionalitat més enllà de hiperenllaços. HyperText fa referència als enllaços que connecten entre si a les pàgines web, ja sigui en un mateix lloc web o entre llocs web.

HTML utilitza etiquetes per mostrar text, imatges i altres contingut en un navegador web. Aquestes són algunes de les etiquetes que té HTML: <head>, <title>, <body>, <header>, <footer>, <article>, <section>, <p>, <div>, , , <aside>, <audio>, <canvas>, <datalist>, <details>, <embed>, <nav>, <output>, <progress>, <video>, , , , etc.

1.2.3.1. Elements HTML

Un element és part d'una pàgina web. En XML i HTML, un element pot contenir un element de dades o un fragment de text o una imatge, o res.

Les parts principals d'un element són:

1. Etiqueta d'obertura. S'escriu entre "<" i ">" el tipus d'element. Estableix on comença l'element. En el cas de la figura 1.15, on comença el paràgraf.
2. Etiqueta de tancament. Igual que la d'obertura, però s'utilitza "</" i ">". Estableix on acaba l'element.
3. Contingut. En el cas de la figura 1.15, el contingut és només text.
4. Element. El conjunt format per l'etiqueta d'obertura, l'etiqueta de tancament i el contingut, formen l'element. En la figura 1.15 l'element és un paràgraf, ja que l'element que s'utilitza <p>.

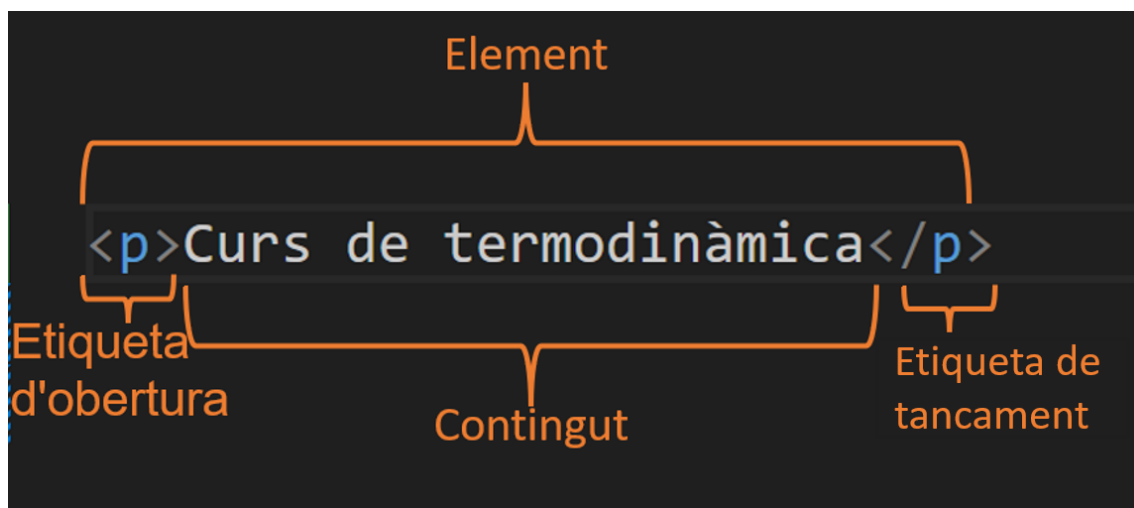


Fig. 1.15 Anatomia d'un element HTML.

Els elements també poden contenir atributs descrits a l'etiqueta d'obertura. Proporcionen informació addicional a l'element que l'usuari no desitja que es

mostri en el seu contingut. A la figura 1.16 es mostra un exemple on *class* és el nom de l'atribut i *text-primary* és el valor de l'atribut. Aquest tipus d'atribut li dóna a l'element un nom identificatiu, per exemple en agregar un estil. L'atribut *text-primary* serveix per escriure el text de color blau.

```
<p class="text-primary">Curs de termodinàmica</p>
```

Fig. 1.16 Especificació d'atributs d'un element HTML

Es poden col·locar elements dins d'altres. En la figura 1.17 es crea un element paràgraf (<p>), amb la paraula "termodinàmica" ressaltada, això s'aconsegueix fent servir .

```
<p>Crus de <strong>termodinàmica</strong></p>
```

Fig. 1.17 Element dins d'un element.

Els elements s'han d'obrir i tancar correctament perquè estiguin clarament dins o fora un de l'altre. Quan se solapen, els navegadors web intenten endevinar el que l'usuari està tractant de transmetre, cosa que pot generar resultats inesperats.

També existeixen els elements buits, són aquells elements que no posseeixen contingut. La figura 1.18 és un exemple d'un element buit utilitzant .

```

```

Fig. 1.18 Element buit.

Com es pot veure a la figura 1.18 l'element té dos atributs, però no té etiqueta de tancament ni contingut adjunt. Això es deu a que l'element d'imatge no conté cap contingut afectat. El seu propòsit és mostrar la imatge a la pàgina HTML.

Els elements d'encapçalat permeten especificar que una part particular del contingut és un encapçalat o subencapçalat del seu contingut. Així com un llibre té un títol principal, que pot contenir capítols individuals i títols de subtítols, també el pot tenir un document HTML. HTML posseeix sis nivells d'encapçalats, que s'escriuen de la següent manera: `<h1></h1>`.

1.2.3.2. Document HTML

En el punt anterior s'han explicat els coneixements bàsics d'elements HTML individuals, però aquests no són gaire útils per si sols. Ara s'explica com els elements individuals són combinats per formar una pàgina HTML [31] sencera.

```
18
19
20
21 <!DOCTYPE html>
22 <html>
23   <head>
24     <meta charset="utf-8">
25     <title>Pàgina de prova</title>
26   </head>
27   <body>
28     
29   </body>
30 </html>
31
```

Fig. 1.19 Anatomia d'un document HTML.

A la figura 1.19 es mostren les diferents parts d'un document HTML:

- `<!DOCTYPE html>`: És el tipus de document i un preàmbul requerit. Al voltant de 1991 i 1992, el tipus de document servia com a enllaç a un conjunt de regles que havien de seguir perquè el codi HTML de la pàgina es considerés correcte, això implicava la comprovació automàtica d'errors i altres funcions de seguretat. Però avui això és només una

relíquia antiga que a ningú li importa, però s'ha d'incloure perquè tot funcioni correctament.

- `<html></html>`: Aquest element engloba tot el contingut de la pàgina i també es coneix com element arrel (*root element*).
- `<head></head>`: Aquest element actua com un contenidor de tot allò que el programador vol incloure a la pàgina HTML que no és contingut visible pels visitants de la pàgina. Inclou coses com paraules clau, una descripció de la pàgina que vols que aparegui en resultats de recerques, codi CSS per donar estil al contingut, declaracions del joc de caràcters, etc.
- `<meta charset="utf-8">`: Aquest element especifica el conjunt de caràcters que utilitza el document en utf-8. Això inclou gairebé tots els caràcters en tots els idiomes. Bàsicament, pot manejar qualsevol contingut de text que puguis incloure.
- `<title></title>`: Estableix el títol de la pàgina, que és el títol que apareix a la pestanya o a la barra de títol del navegador quan la pàgina és carregada, i es fa servir per descriure la pàgina quan és afegida als marcadors o com a favorita.
- `<body></body>`: Això inclou text, imatges, vídeos, jocs, pistes d'àudio reproduïbles i qualsevol altre contingut que mostri als usuaris web que visiten la pàgina web.

1.2.3.3. Imatges

A la figura 1.18 es mostra com és un element imatge, com s'ha comentat abans, incrusta una imatge a la pàgina, en la posició en què apareix. Això es fa a través de l'atribut *src* (*source*) que conté la ruta (o ubicació) de l'arxiu d'imatge. També conté un atribut *alt* (*alternative*), que conté text per descriure la imatge, aquest atribut és útil per als usuaris que no poden veure la imatge.

1.2.3.4. Vincles

Els vincles o enllaços són molt importants. Per implementar un vincle, es necessari fer servir un vincle simple, `<a>`, l'abreviatura de la paraula anglesa *anchor* (ancora). Per convertir algun text dins d'un paràgraf en un vincle, s'han de seguir aquests passos:

1. Tria algun text. En aquest cas he escollit «Curs de termodinàmica».
2. Tanca el text en un element `<a>`, com es mostra a la figura 1.20.

```
<a>Curs de termodinàmica</a>
```

Fig. 1.20 Pas 2 de la creació d'un vincle.

3. Proporcionar a l'element `<a>` un atribut `href` i completa el valor d'aquest atribut amb un adreça web amb la qual es vol connectar al vincle: com es mostra a la figura 1.21. Un cop s'ha creat el vincle al fer-li clic dirigeix a l'usuari a la direcció desitjada.

```
<a href="https://eetac.upc.edu/es">Curs de termodinàmica</a>
```

Fig. 1.21 Vincle.

1.2.4. PyScript

PyScript [32] és un *framework* que permet als usuaris crear aplicacions Python al navegador utilitzant la interfície d'HTML i la potència de Pyodide, WASM i les tecnologies web modernes. PyScript és un sistema per interconnectar el codi Python en un script HTML. Això significa escriure i executar codi de Python a HTML, trucar a biblioteques de JavaScript a PyScript i fer desenvolupament web exclusivament amb programació de Python.

Per instal·lar PyScript dins d'un arxiu HTML simplement s'ha d'afegir dins de l'element `<head>` les instruccions que es mostren a la figura 1.22.

```
<link rel="stylesheet" href="https://pyscript.net/latest/pyscript.css" />  
<script defer src="https://pyscript.net/latest/pyscript.js"></script>
```

Fig. 1.22 Instal·lació de PyScript.

PyScript és *open source* i els principals comandaments per usar-lo en un document HTML són:

- `<py-env>`: Defineix els paquets necessaris per executar el codi de Python.
- `<py-script>`: És on s'escriu el codi Python que després s'executarà dins de la pàgina web.
- `<py-repl>`: Crea un component REPL (*read-eval-print-loop*) que avalua el codi inserit i mostra els resultats.

1.3. Estructura de la pàgina web

En aquest punt s'explica com construït l'estructura del lloc web. Primer de tot s'ha de crear una carpeta anomenada *templates*, dins d'aquesta carpeta hi hauran arxius HTML i el que fa Flask es entrar dins de la carpeta a buscar les estructures HTML per poder mostra-les.

Dins de la carpeta *templates* es troben les altres que ajuden a administrar la pàgina web. La figura 1.23 mostra la carpeta *templates* i les carpetes que es troben dins d'aquesta carpeta. Dins de totes les carpetes hi ha arxius HTML. Les carpetes *sitio*, *ingles* i *castellano* tenen el mateix contingut però en diferents idiomes. Les carpetes *pythoncastellano*, *pythoningles* i *simulaciopython* també tenen el mateix contingut però en diferents idiomes, en les últimes tres carpetes es troben els arxius en els que he utilitzat el *framework* PyScript.

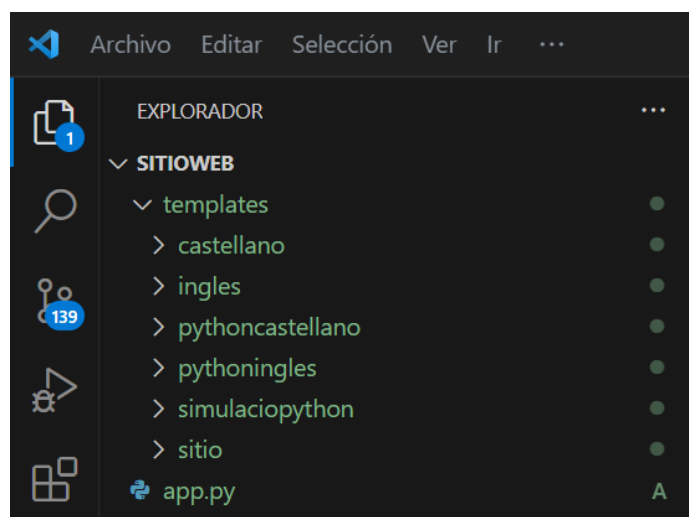


Fig. 1.23 Carpetes utilitzades en VS Code per administrar la pàgina web.

Dins de la carpeta *templates* es crea l'arxiu *app.py* que és on està escrit tot el cos de l'aplicació per poder accedir directament i mostrar els arxius HTML. El primer pas en aquest arxiu és importar Flask, *render_template* i *request* des de el *framework* Flask, tot això es pot veure a la figura 1.24. En les següents línies de codi s'importen les llibreries necessàries per Python. I després amb el comandament de la línia 9 es crea l'aplicació. Per últim el pas més important és la línia 11, ja que *@app.route* és el comandament necessari per accedir a la URL, el que es fica dins el parèntesis és el que es mostra a la barra del buscador, es defineix com la via que es busca per accedir a l'arxiu que es defineix a les següents línies, a la figura 1.25 es mostra un exemple. A continuació s'ha de crear una funció, que retorna un *render_template()*, dins dels parèntesis es defineix la ruta per arribar a l'arxiu que es vol mostrar amb la via definida anteriorment. En la figura 1.24 l'arxiu HTML està dins de la carpeta *sitio*.



```
app.py A ●
app.py > inicio
1  from flask import Flask
2  from flask import render_template
3  from flask import request
4
5  import numpy as np
6  import matplotlib as plt
7
8
9  app=Flask(__name__)
10
11 @app.route('/Català')
12 def catala():
13     return render_template('sitio/index.html')
14
```

Fig. 1.24 Inici del arxiu *app.py*.

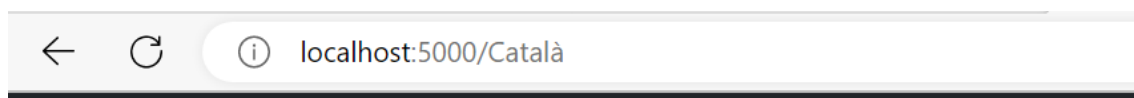


Fig. 1.25 Via */Català* en el buscador.

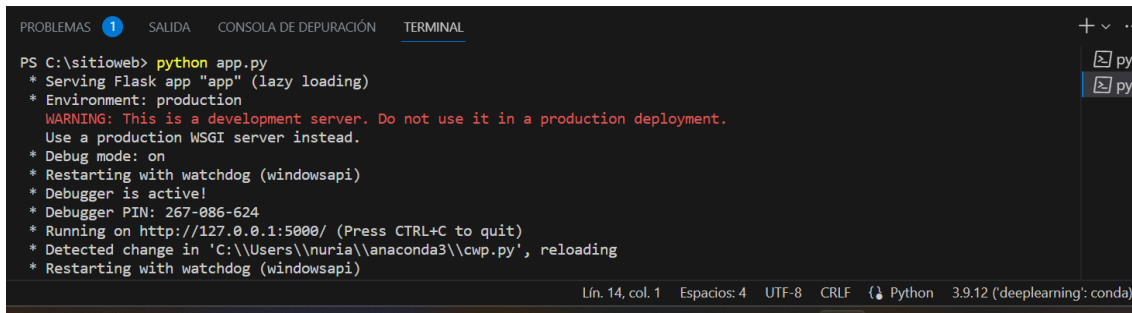
El codi de les línies 11, 12 i 13 de la figura 1.24 és per quan l'usuari accedeix directament al servidor amb la ruta */Català*, es buscarà l'arxiu *index.html* i es mostrarà. Però perquè això funcioni s'ha de ficar alguna instrucció més que es mostra a la figura 1.25.

El comandament de la figura 1.26 serveix per que quan l'aplicació està llesta s'executi en mode *debug* per a que cada cop que jo actualitzi o faci alguna modificació en algun arxiu HTML, s'actualitzi de forma automàtica.


```
8910     if __name__ == '__main__':
8911         app.run(debug=True)
8912
```

Fig. 1.26 Mode debug.

Per veure com funciona la pàgina web s'ha d'obrir la consola i escriure *python app.py*, a la figura 1.27 es mostra el resultat d'executar el comandament, doncs es veu que l'aplicació corre només en una direcció que és <http://127.0.0.1:5000>. Doncs al obrir el navegador i al escriure la ruta es mostra a la pàgina web com es mostra a la figura 1.28, on es pot veure que la barra del buscador apareix la direcció que ens dona el terminal i el primer arxiu HTML que es mostra és l'inici que es troba dins la carpeta *sitio*.



```
PROBLEMAS 1 SALIDA CONSOLA DE DEPURACIÓN TERMINAL
PS C:\sitioweb> python app.py
* Serving Flask app "app" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Restarting with watchdog (windowsapi)
* Debugger is active!
* Debugger PIN: 267-086-624
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Detected change in 'C:\Users\nuria\anaconda3\cwp.py', reloading
* Restarting with watchdog (windowsapi)
Lín. 14, col. 1 Espacios: 4 UTF-8 CRLF Python 3.9.12 (deeplearning: conda)
```

Fig. 1.27 Comandament i ruta de la pàgina web.

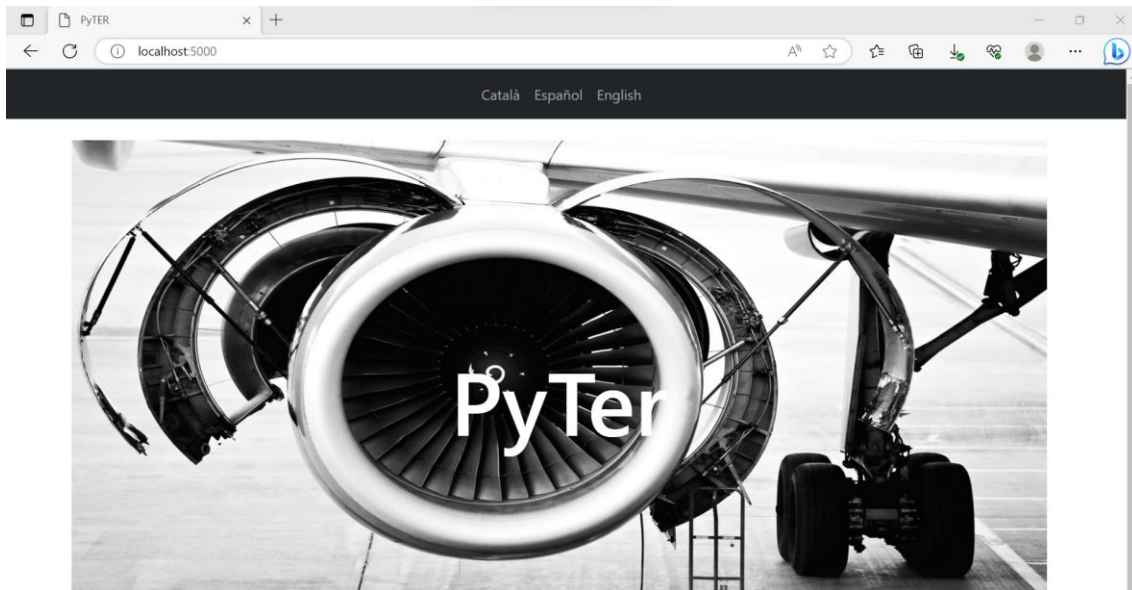


Fig. 1.28 Arxiu inici.html de la carpeta *sitio*.

Els arxius HTML en els que només hi ha textos o imatges, són arxius senzills i fàcils d'escriure. Les complicacions van aparèixer a l'hora de fer els exercicis. A la figura 1.29 es mostra un arxiu HTML d'un exercici de la part de cycle termodinàmic.

Exercici 3. Una massa de kg de segueix el cycle de Brayton. El gas entra al compressor amb una temperatura de °C i amb una pressió de kPa. La relació de pressions és igual a i el gas entra a la turbina amb una temperatura de kPa. Determina la pressió, el volum i la temperatura en tots els estats. El diagrama p-V i T-s. La calor, la variació d'energia interna i el treball en tots els processos.

Fig. 1.29 Exercici del cycle de Brayton.

Els exercicis de la pàgina web tenen tots una estructura molt semblant. Tots tenen un enunciat en el que l'usuari ha de ficar les dades que ell vol i doncs clicar el botó calcular per obtenir el resultat, doncs en aquest arxiu és on s'ha utilitzat l'objecte *request*. A la figura 1.29 es mostra l'arxiu HTML dels exercicis de Brayton i a la figura 1.30 és el codi per calcular a l'exercici que està escrit dins l'arxiu *app.py*.

```

186 <form action="{{url_for('calcularbrayton')}}" method="post">
187
188 <p><strong>Exercici 3.</strong>Una massa de <input type="text" name="massa" placeholder="Massa" required="required">kg
189 de <select name="gas" size="1" required="required">
190 <option>Aire</option>
191 <option>Nitrogen</option>
192 <option>Oxigen</option>
193 <option>Diòxid de Carboni</option>
194 <option>Heli</option>
195 </select> segueix el cicle de Brayton. El gas entra al compressor amb una temperatura de <input type="text" name="temperaturaComperssor"
196 amb una pressió de <input type="text" name="pressio" placeholder="Pressió" required="required">kPa. La relació de pressions és igual
197 a <input type="text" name="relacio" placeholder="Relació pressions" required="required"> i el gas entra a la turbina amb una
198 temperatura de <input type="text" name="temperaturaTurbina" placeholder="Temperatura trubina" required="required">kPa.
199 Determina la pressió, el volum i la temperatura en tots els estats. El diagrama p-V i T-s. La calor, la variació d'energia interna i
200 el treball en tots els processos.
201 </p>
202 <button type="submit" class="btn btn-dark btn-block btn-large">Calcular</button>
203
204
205 </form>

```

Fig. 1.30 Arxiu HTML exercicis del cicle de Brayton.

A la figura 1.30, en la primer línia de codi l'element `<form>` defineix com s'envien les dades, tots els seus atributs estan dissenyats perquè pugui configurar la sol·licitud que s'enviarà quan un usuari clica un botó d'enviament, en aquest cas el botó calcular. Els dos atributs més importants són *action* i *method*.

L'atribut *action* defineix el lloc on les dades s'envien. El seu valor ha de ser una adreça URL vàlida, en aquest cas és una adreça que ve definida per una funció, que com es pot veure a les figures 1.30 i 1.31, l'adreça definida té el mateix nom que la funció, si tinguessin noms diferents donaria error.

L'atribut *method* defineix com s'envien les dades. El protocol HTTP proporciona diverses formes de realitzar una sol·licitud, les dades del formulari HTML es poden transmetre a través de diferents mètodes, els més comuns i els són el GET i el POST. El mètode GET és utilitzat pel navegador per demanar al servidor que s'envii de tornada un recurs donat. En aquest cas, el navegador envia un cos buit. Atès que el cos està buit, si un formulari s'envia utilitzant aquest mètode, les dades enviades al servidor s'annexen a la URL. El mètode POST és una mica diferent. És el mètode que el navegador utilitza per comunicar-se amb el servidor quan es demana una resposta que tingui en compte les dades proporcionades al cos de la petició HTTP. Si un formulari s'envia utilitzant aquest mètode, les dades s'annexen al cos de la petició HTTP.

```

3751 @app.route('/simulaciopython/cicle/Brayton/calcularBrayton', methods=['POST'])
3752 def calcularbrayton():
3753     PA = float(request.form['pressio'])
3754     TemA = float(request.form['temperaturaComperssor'])
3755     TemC = float(request.form['temperaturaTurbina'])
3756     r = float(request.form['relacio'])
3757     Gas = (request.form['gas'])
3758     m = float(request.form['massa'])
3759

```

Fig. 1.31 Funció càlcul exercicis de Brayton en l'arxiu app.py.

Doncs l'atribut que s'ha fet servir en els exercicis és el POST, aquest atribut el que fa és enviar les dades que introdueix l'usuari en els espais en blanc de la figura 1.29, aquest espais com es mostra a la figura 1.30 tenen un nom assignat que és el *name*, doncs a la figura 1.31, es mostra com s'agafen les dades introduïdes per l'usuari i se li assigna una variable, en els casos necessaris el text l'he convertit en *float*, és a dir, en un número decimal.

Un cop es tenen les dades necessàries s'executa el codi per calcular el que demana l'enunciat i en aquest cas la resposta es torna en un altre arxiu HTML, com es mostra a la figura 1.32, en aquest cas també envia cap a l'arxiu HTML les dades necessàries.

```
return render_template('simulaciopython/graficabrayton.html', Pressio = PA, gastreball = G, TemperaturaA = TA, TemperaturaC = TC,
rel = r, massa = m, PressioB = PB, PressioC = PC, PressioD = PD, TemperaturaB = TB, TemperaturaD = TD, VolumA = VA,
VolumB = VB, VolumC = VC, VolumD = VD, CalorAB = QAB, CalorBC = Qin, CalorCD = QCD, CalorDA = Qout, UA = UAB,
UB = UBC, UC = UCD, UD = UDA, WA = WAB, WB = WBC, WC = WCD, WD = WDA, EntropiaA = SA, EntropiaB = SB,
EntropiaC = SC, EntropiaD = SD, VarSA = SAB, VarSB = SBC, VarSC = SCD, VarSD = SDA)
```

Fig. 1.32 Resposta del càlcul de l'exercici de Brayton en l'arxiu app.py.

Aquest exercici de Brayton ens demana que es dibuixi la gràfica del cicle, doncs en l'arxiu *graficabryton.html* s'ha utilitzat el *framework* PyScript. En les línies de la 101 a la 106 de la figura 1.33, es mostra com s'agafen les dades que són enviades des de l'arxiu app.py. A la figura 1.34 es mostra el codi de la gràfica entropia vs temperatura. Finalment a la figura 1.35 i figura 1.36 es mostra el resultat que és l'arxiu *graficabryton.html* que és el que veu l'usuari en els buscador.

```
89 <body>
90
91
92 <div id="variables"></div>
93 <div id="lineplot"></div>
94
95 <py-script output="lineplot", output="variables">
96
97 import matplotlib.pyplot as plt
98 import numpy as np
99
100
101 PA = {{Pressio}}
102 m = {{massa}}
103 r = {{rel}}
104 TA = {{TemperaturaA}}
105 TC = {{TemperaturaC}}
106 gas = {{gastreball}}
107
```

Fig. 1.33 Arxiu *graficabrayton.html*.

```
243 fig1, ax1 = plt.subplots()
244
245 # PLOT de l'entropia vs T
246
247 plt.plot([SA, SB], [TA, TB], label='Compressió isentròpica')
248 plt.plot(SBC, TBC, label='Addició de calor isobàrica')
249 plt.plot([SC, SD], [TC, TD], label='Expansió isentròpica')
250 plt.plot(SDA, TDA, label='Rebuig de calor isobàric')
251 plt.legend(loc="upper left")
252
253 plt.plot(SA, TA, "x")
254 plt.plot(SB, TB, "x")
255 plt.plot(SC, TC, "x")
256 plt.plot(SD, TD, "x")
257
258 plt.text(SA, TA, "A")
259 plt.text(SB, TB, "B")
260 plt.text(SC, TC, "C")
261 plt.text(SD, TD, "D")
262 plt.title("Entropia vs Temperatura en un cicle Brayton")
263 plt.xlabel("Entropia [kJ/K]")
264 plt.ylabel("Temperatura [K]")
265
266
267 display(fig, fig1)
268
269 </py-script>
```

Fig. 1.34 Plot entropia vs temperatura del cicle de Bryton.

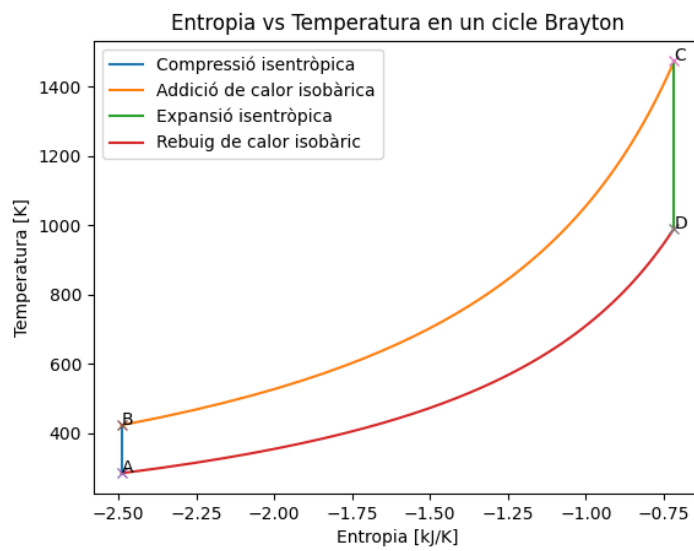
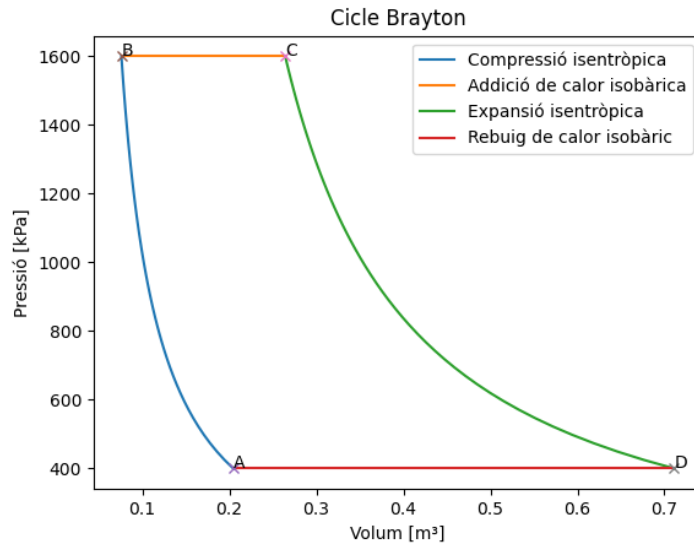


Fig. 1.35 Arxiu graficabrayton.html que veu l'usuari en el buscador.

	Volum[m ³]	Pressió[kPa]	Temperatura[k]	Entropia[kJ]
A	0.20448749999999996	400.0	285.0	-2.4909755102858733
B	0.07596681429997292	1600.0	423.50837240403024	-2.4909755102858733
C	0.264219375	1600.0	1473.0	-0.7161907563200739
D	0.7112258154194014	400.0	991.2554918737303	-0.7161907563200739

	Calor[kJ]	Variació energia interna [kJ]	Treball[kJ]	Variació entropia[kJ]
A-B	0	99.4490113860937	99.4490113860937	0
B-C	1054.9489840594688	753.7448869394254	-301.2040971200433	1.252964845308556
C-D	0	-345.8925568346616	-345.8925568346616	0
D-A	-507.0914431653383	-304.39611699757774	202.69532616776058	-1.252964845308556

Fig. 1.36 Arxiu graficabrayton.html que veu l'usuari en el buscador.

Com s'ha explicat anteriorment les dues taules que es mostren a la figura 1.36 s'han fet utilitzar l'extensió de Bootstrap.

Tots els exercicis segueixen l'estructura que s'ha explicat anteriorment, l'únic que varia és que en els exercicis que no s'han de mostrar gràfiques no s'ha utilitzat el PyScript, ja que tots els càlculs s'han fet des de l'arxiu app.py i s'han enviat les cap a l'arxiu HTML seleccionat i s'han mostrar a la pàgina web mitjançant text.

L'estructura de la pàgina web és molt senzilla, hi ha 6 carpetes i cada carpeta té els seus arxius HTML. Inicialment es va començar només em dues carpetes, la de *sitio* i *simulaciopython*, ja que al principi la pàgina web es va crear només en català. Doncs el que es fer va ser pensar quina estructura havia de tenir l'aplicació i de quins temes es volia parlar, poc a poc es van anar creant arxius HTML i es van connectar entre ells en els casos que era necessari. Un cop es va tenir tota la pàgina creada en català, es va decidir traduir-la al castellà i l'anglès.

CAPÍTOL 2. FUNCIONAMENT DE LA PÀGINA WEB

El disseny de la pàgina web és molt senzill i molt fàcil d'utilitzar, ja que s'ha creat perquè sigui més intuïtiu possible. Tots els arxius HTML tenen la mateixa estructura, és a dir, constant d'un encapçalat, el cos i el peu. A la [figura 2.1](#) es mostra la primera pàgina que es troba l'usuari al obrir el lloc web.

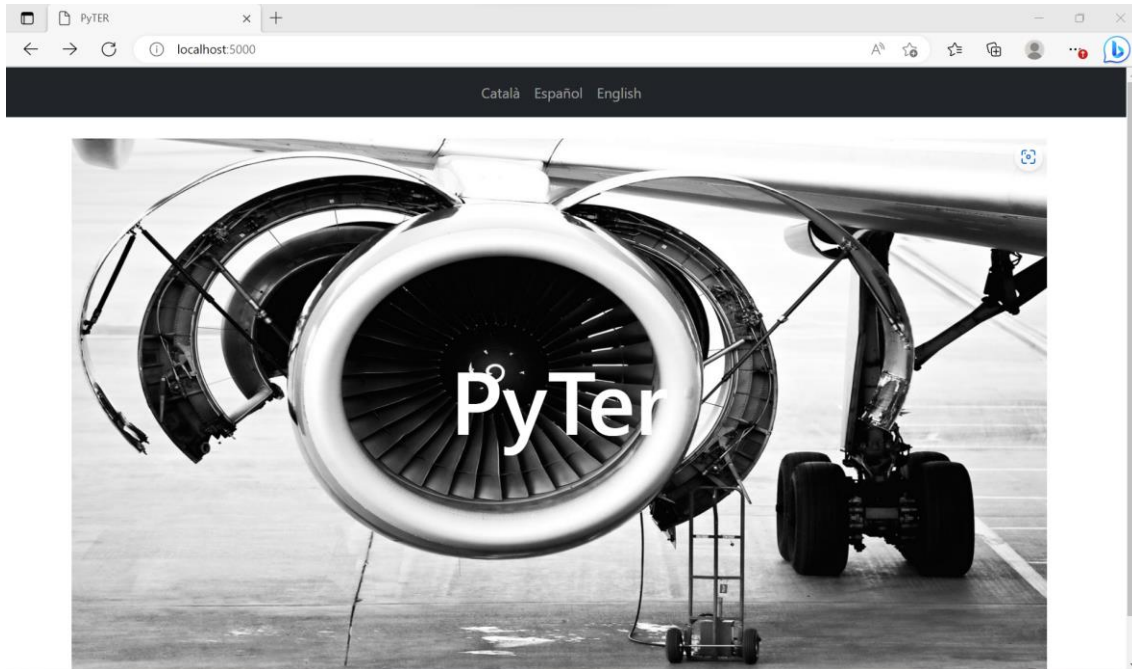


Fig. 2.1 Pàgina d'inici del meu lloc web.

2.1. Disseny de la pàgina web

Una part fonamental que ha de contemplar tota pàgina web és el contingut que s'hi al·legui. Per això, és important realitzar en primer lloc, una detecció i selecció dels continguts que s'hi abordaran. Partint de l'objectiu principal de la pàgina web, en aquest cas, un curs de termodinàmica de l'EETAC, s'ha de tenir en compte la procedència i autenticació de la informació que s'ha d'anar recaptant.

La pàgina web, està organitzada per temes d'interès que es desenvolupen en base als blocs. Cada tema, està localitzat en una pàgina diferent, per facilitar així el dinamisme al lector i per ajudar-lo a no perdre's per la pàgina web durant la seva navegació. És per això, que cada pàgina rep un subtítol de pàgina i un encapçalat, on es desenvolupa una breu descripció de l'apartat a tractar, per posteriorment, començar amb la publicació de la informació captada i elaborada. El títol de cada pàgina és descriptiu i breu, pel que fa a l'encapçalat, aquest permet estructurar el contingut.

2.1.1. Pàgina d'inici

La pàgina d'inici és la primera impressió que tindran els visitants del lloc web, l'objectiu d'aquesta pàgina és comunicar el missatge principal per mantenir els visitants interessats, en aquest cas, com és pot veure a la figura 2.1 s'ha ficat una pàgina inicial molt senzilla i visual, en la que només mostra una capçalera per escollir l'idioma preferit de l'usuari, el nom de la pàgina i el peu. El que ha de fer l'usuari en aquest arxiu HTML és escollir un dels tres idiomes, per passar a la següent subpàgina.

2.1.2. Pàgina principal

Un cop l'usuari ha escollit l'idioma s'obre la pàgina principal que es mostra a la figura 2.2, en aquest arxiu s'explica com funciona la pàgina web i de que tracta.



Fig. 2.2 Pàgina principal del lloc web.

2.1.3. Encapçalat

L'encapçalat és la part superior de color negre de la figura 2.2. Normalment, és el primer que veuen els visitants quan ingressen a un lloc per aquest motiu s'ha dissenyat de la manera més clara possible. S'ha fet una capçalera simple i el contingut està escrit en una font clara i llegible. L'encapçalat apareixer en totes les subpàgines i té el mateix disseny en cadascuna d'elles.

Com es pot veure a la figura 2.2 a la part esquerra de l'encapçalat hi ha els diferents apartats de la pàgina web i a la part dreta hi ha per seleccionar l'idioma un altre cop.

2.1.3.1. PyTer

Al seleccionar PyTer l'usuari torna a la pàgina principal del lloc web amb l'idioma que ha seleccionat inicialment.

2.1.3.2. Què és la termodinàmica?

En aquest apartat s'explica que és la termodinàmica, els gasos ideals, conceptes bàsics, etc. A la figura 2.3 es mostra aquest punt.

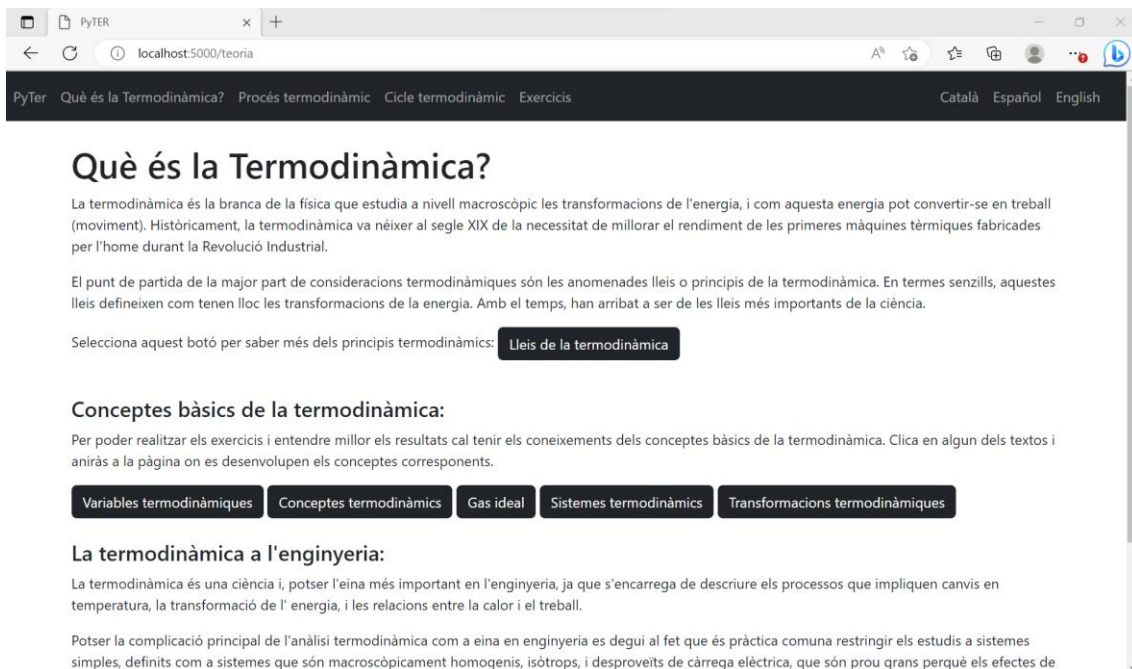


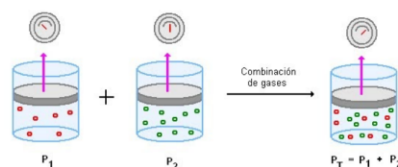
Fig. 2.3 Pàgina què és la termodinàmica?

Com es pot veure a la figura 2.3 dins d'aquest punt es troben diversos botons en els quals es pot fer clic i anar a una pàgina nova. Però principalment explicarà el punt de gas ideal, ja que dins d'aquest arxiu hi ha uns quants exercicis, els exercicis es troben al fer clic a sobre les fórmules, a la figura 2.4 mostra una explicació de la Llei de Dalton i es veu que la fórmula està escrita de color blau, al fer clic a sobre s'obre un altre arxiu que conté exercicis d'aquesta llei.

Llei de Dalton

La llei de Dalton diu que la pressió total d'una barreja de gasos és igual a la suma de les pressions que exerceixen cadascun dels gasos que la componen. La pressió que exerceix cada gas de la mescla es denomina pressió Parcial.

La Llei de Dalton s'expressa amb la següent fórmula (clica per fer exercicis utilitzant la Llei de Dalton): $P_{\text{total}} = \sum P$



A la imatge es mostra que la pressió total és igual a la pressió del gas 1 més la pressió del gas 2.

Fig. 2.4 Pàgina dels gasos ideals.

El lloc web permet fer exercicis utilitzant l'equació d'estat dels gasos ideals, la Llei de Boyle, Llei de Charles, la de Gay-Lussac, la Llei combinada dels gasos i per últim la Llei de Dalton.

2.1.3.3. Procés termodinàmic

En aquesta pàgina s'explica la teoria dels processos termodinàmics. Dintre de la teoria hi ha uns botons d'exercicis, que igual que en el apartat de gasos ideals, quan l'usuari fa clic s'obre un nou arxiu HTML amb exercicis.

Anteriorment ja s'ha explicat com funcionen els exercicis, l'usuari ha d'emplenar totes els espais en blanc amb les dades que ell vol, és molt important que tots els espais estiguin plens perquè el programa poguï resoldre l'exercici, sinó apareix un missatge d'error com el que es mostra a la figura 2.5, un cop estan totes les dades plenes l'usuari ha de donar clic al botó de calcular i es torna a enviar el mateix arxiu HTML amb el resultat, el resultat apareix de color blau i amb negreta. En alguns exercicis inicialment hi ha una explicació de com resoldre'l i quines fórmules s'han de fer servir, això està explicat de color blau com es pot veure a la figura 2.5.

Cal comentar que en alguns exercicis l'usuari ha d'escollir amb quin gas ideal vol treballar, les opcions són: Aire, Nitrogen, Diòxid de Carboni, Heli i Oxigen. També el lloc web permet fer exercicis de 4 processos termodinàmics, aquest són: isobàric, for, adiabàtic i isotèrmic.

Exercici 2. En un contenidor kg de Aire es van comprimir d'un volum de m³ a m³. Sabent que la temperatura inicial era de °C. Calcula el treball fet sobre el gas.

Es tracta d'una compressió adiabàtica irreversible. Hi ha dues equacions

Rellene este campo.

$$W = -mC_v\Delta T$$

$$W = \frac{C_v}{R} (P_1V_1 - P_2V_2)$$

Les pressions les podem calcular, però per estalviar temps és millor procedir amb la primera de les equacions. Falta calcular T₂:

$$T_1V_1^{\gamma-1} = T_2V_2^{\gamma-1}$$

S'aïlla T₂ de l'equació i finalment es substitueix en l'equació del treball.

Fig. 2.5 Exemple d'un exercici d'un procés termodinàmic.

2.1.3.4. *Cicle termodinàmic*

El següent punt que es troba és el de cicle termodinàmic, aquesta subpàgina és molt semblant a la de processos termodinàmics, primer de tot s'explica la teoria dels cicles termodinàmics i al final de tot has de seleccionar un dels quatre cicles que surten, doncs, s'explica la teoria del cicle seleccionat i finalment hi ha un botó per fer exercicis. Els cicles que es tracten a la pàgina web són: Carnot, Diesel, Brayton i Otto.

2.1.3.5. *Exercicis*

Per acabar, l'últim punt de la capçalera és el d'exercicis, en aquesta pàgina inicialment s'obre un exercici d'un turbojet, doncs el procediment és el mateix que en els exercicis anteriors. Si l'usuari baixa fins abaix de tot de la pàgina, troba un botó que fica següent, al fer clic, s'obre un exercici d'un turbofan.

2.1.4. **Peu de pàgina**

El peu de pàgina es troba a la part inferior del lloc web. És el lloc ideal per enllaçar a contingut important que no pertany a l'encapçalat o al contingut principal, així com informació bàsica de contacte i xarxes socials. Com es pot veure a la figura 2.6, el peu conté el logo de la UPC i el de l'EETAC, al clicar a sobre del logos envia a l'usuari a la pàgina web de la UPC o de l'EETAC. En el peu també apareix el nom de l'autor i el correu electrònic.

Fig. 2.6 Peu del lloc web.

2.2. Publicació i accés al lloc web

Per dur a terme la creació de la pàgina web, es necessita en primer lloc d'una connexió a Internet, ja que és vital per poder pujar la pàgina a la xarxa.

Per poder accedir a la pàgina web desde l'ordinador que conté els arxius, només s'ha d'escriure a la barra del buscador <http://127.0.0.1:5000/>, però aquesta adreça no és vàlida perquè altres persones puguin accedir. El primer problema es va tenir quan s'havia de compartir el lloc web amb el tutor, doncs, es va haver d'investigar i es va trobar un servei anomenat Ngrok que és útil per fer això. La següent investigació va ser per publica la pàgina web a internet, això s'ha fet a través del servei Banahost.

2.2.1. Ngrok

Ngrok [33] és un servei que ens permet crear un servidor local en un subdomini per poder visualitzar-lo fora de la LAN (xarxa d'àrea local), a través d'internet. Amb aquest servei es pot emmagatzemar els arxius i establir connexions.

Per utilitzar aquest servei el primer pas es crear un usuari des de la pàgina web i després instal·lar l'aplicació, després s'han de seguir els passos que estan explicats a la seva pàgina web (veure [33]). En el meu cas l'entorn de desenvolupament està en el port 5000, tenint En aquest cas l'entorn de desenvolupament està en el port 5000, tenint en compte això en la consola de Ngrok s'ha d'escriure els següent comandament: `Ngrok http 5000`. Això arrencarà el client de Ngrok i mostrarà la informació en temps real de la connexió, així com les URLS d'accés, com es mostra a la figura 2.7.

```

C:\Users\nuria\Desktop\ngrok x + v
ngrok
ngrok http https://localhost # expose a local http

ngro
ngrok start foo bar baz
ngrok (Ctrl+C to quit)

Announcing ngrok-go: The ngrok agent as a Go library: https://ngrok.com/go

Session Status      online
Account             Nuria Hostenc Muelas (Plan: Free)
Update              update available (version 3.3.1, Ctrl-U to update)
Version              3.2.2
Region              Europe (eu)
Latency              48ms
Web Interface        http://127.0.0.1:4040
Forwarding           https://5a10-92-58-171-219.ngrok-free.app -> http://localhost:5000

Connections          ttl   opn   rt1   rt5   p50   p90
146                 0     0.00  0.00  0.33  0.34

HTTP Requests
-----

```

Fig. 2.7 Informació de la connexió i ruta URL a través de ngrok.

En la figura 2.7 la ruta URL del *Forwarding* corresponen amb l'accés https al servidor local, a més es pot veure la redirecció que fa. Accedint a la ruta des d'un altre dispositiu, s'arriba finalment al lloc web. La URL *Web Interface* dona accés a una consola web d'estat de Ngrok.

La part negativa de Ngrok és que l'ordinador sempre ha d'estar encès i corrent el VS Code perquè els altres poguessin visualitzar el lloc web.

2.2.2. Publicació del lloc web a Internet

En aquest punt cal recordar que ja hi ha creada una carpeta en l'escriptori en la que hi ha carpetes amb arxius HTML i tot això és la pàgina web.

El primer per publicar una pàgina web es fer-nos amb un domini propi (.com, .es, .net, .org, etc.), això es pot fer a través de *hostings* que ajuden a crear una pàgina web amb un domini propi, com per exemple banahosting.com.

2.2.2.1. Hosting

Un *hosting* [34] és un servei d'allotjament per a llocs web. El *hosting* web allotja els continguts d'un lloc web i el correu electrònic del client perquè puguin ser visitats en tot moment des de qualsevol dispositiu connectat a Internet. El contingut d'una web, ha d'estar emmagatzemat en algun lloc i perquè pugui ser visitat a través d'Internet haurà d'estar emmagatzemat en un servidor web.

En lloc d'identificar-se entre si pel seu nom, diverses computadores connectades a Internet es comuniquen mitjançant un codi numèric anomenat IP. Però recorda la direcció IP de cada lloc web és impossible, és per això que es va crear el Sistema de Noms de Domini (DNS) per facilitar aquesta tasca i permetre utilitzar noms, fàcils de recordar, en lloc d'IPs. El que el sistema DNS

és traduir la IP del servidor que allotja el contingut i els comptes de correu electrònic.

Un cop es compta amb un *hosting* web per allotjar els arxius de la pàgina, les visites poden arribar fins a ells a través del domini. El domini s'utilitza per dir-li al navegador quin contingut mostrar. El navegador anirà a buscar els continguts del lloc web al *hosting* on està allotjat per mostrar-los, per saber a quin allotjament web ha d'anar a buscar-los es fan servir les DNS. El servei *hosting* l'he fet amb Banahosting.

2.2.2.2. *Banahosting*

Després d'una llarga investigació de quina empresa de web *hosting* fer servir, es va decidir que la millor opció és Banahosting.

Banahosting [35] és una de les empreses líders en tot allò relacionat amb l'allotjament web. Els serveis que l'empresa ofereix als seus clients inclouen allotjament dedicat, allotjament compartit, VPS i registre de domini. Els serveis de l'empresa són de molt alta qualitat i el seu centre de suport brinda als clients solucions ràpides i eficients. Utilitza IP dels Estats Units i Europa per a l'allotjament de les pàgines web dels seus clients.

En la pàgina web de la empresa (veure [35]) es poden veure tots els serveis que ofereix depenen del pla escollit, jo he escollit el Bana Professional.

Per utilitzar aquest servei, primer s'ha de crear un compte i escollir el pla. El següent pas és decidir el nom del domini, i el nom que s'ha escollit és webpyter.com. Un cop escollit el *hosting* i el domini, s'ha d'utilitzar cPanel per pujar el lloc web. El primer pas és crear una aplicació de Python a cPanel i crear una carpeta anomenada PythonAppTFG, doncs dins d'aquesta carpeta s'ha d'afegir l'arxiu app.py i la carpeta de *templates* amb els arxius html. Per últim s'ha d'obrir el terminal de cPanel i instal·lar Flask, Numpy, Matplotlib i MySQL, un cop instal·lat tot lo necessari s'ha de reiniciar l'aplicació de Python que s'ha creat inicialment.

I per últim s'ha d'instal·lar la base de dades, per a això, s'ha utilitzat l'assistent de MySQL al qual es pot accedir des de CPanel des de l'apartat de Databases. Un cop seleccionat l'assistent, s'escriu el nom de la base de dades. En un *hosting* compartit, el nom de la base de dades està normalment precedit pel nom d'usuari i l'id, en aquest cas rybkkaфу_nuriahostenchemuelas. Un cop creat la base de dades, s'ha de crear un compte d'usuari per poder accedir-hi especificant nom d'usuari i contrasenya. L'últim pas, és proporcionar a l'usuari prèviament creat, els permisos necessaris per poder treballar sobre la base de dades.

Ara el lloc web ja està pujat a Internet i a l'abast de tot el mon, com s'ha comentat anteriorment el *link* per entrar a la pàgina web és el següent:

<http://webpyter.com/>, la informació de pàgina web s'ha tret de les referències [36], [37] i [38].

CAPÍTOL 3. CONCLUSIONS

Un cop finalitzat el present Treball de Fi de Grau, es pot afirmar que s'han complert tots els objectius i requisits proposats en un principi.

L'objectiu del treball era crear una plataforma que ajudés als usuaris a entendre millor la termodinàmica i a resoldre exercicis. La realització d'aquest projecte és molt beneficiosa per estudiants i professors en el marc de l'assignatura de Termodinàmica amb el codi 300209 (veure [3]) del Grau en Enginyeria de Sistemes Aeroespacials de l'EETAC i, potencialment, per a altres comunitats educatives.

Cal destacar l'enorme esforç en poder realitzar un disseny adaptatiu del contingut i crear una plataforma clara i en que la informació no és faci fatigosa per l'usuari.

Crear un lloc web ha sigut tot un repte per mi, vaig haver de fer una gran investigació dels serveis necessaris per realitzar un la pàgina web, però estic molt orgullosa perquè gràcies a aquest projecte he ampliat els meus coneixements de programació, cosa que crec que em serà molt útil pel meu futur laboral.

A nivell personal estic molt contenta amb el resultat obtingut i amb els coneixement apresos, espero que aquest projecte sigui útil per a moltes persones i també m'agradaria ser una motivació per totes les nois que volen entrar en el món de la tecnologia i l'enginyeria.

3.1. Treball futur

El present Treball de Fi de Grau pot continuar sent ampliat aportant noves millores partint dels fonaments que s'han construït en el present projecte. Gràcies al sistema de gestió de continguts i al treball realitzat, tot el que es comenta ara pot ser implementat perfectament sense gaires dificultats.

M'agradaria posar-me amb contacte amb alumnes i professors i que em donessis la seva opinió i algunes idees per millorar i fer la plataforma d'un lloc més professional.

També es preveu ampliar tant la part de teoria com la part dels exercicis. En la part teòrica es preveu profunditzar més en el tema de motors d'avions i les reaccions que pateixen. Per altre banda, també s'implementaran alguns exercicis amb preguntes teòriques i també ficar més gasos per escollir en els exercicis. En resum l'objectiu és que els alumnes puguin assolir els coneixements esmentats a la introducció del treball utilitzant el lloc web.

Utilitzant AWStats (que el proporciona CPanel) i Google Analytics, seria interessant realitzar un estudi sobre el nombre de visitants en un rang de temps (des que es va iniciar el lloc web actual), la procedència de les visites, la

durada, veure com han arribat els visitants al lloc, què fer perquè continuïn visitant-lo, etc.

També es preveu poder emprar l'eina en el futur pròxim a les classes de Termodinàmica a l'EETAC, tant com a recurs per la resolució de problemes a l'aula o de suport a l'estudi dels estudiants. Amb els professors de l'assignatura s'ha començat a dissenyar un seguit de possibles activitats per a dur amb els alumnes i de la traçabilitat del seu impacte en els eu aprenentatge en la matèria. Els resultats posteriors a aquesta implementació i estudi pendents s'espera que es puguin presentar, en el futur, en congressos especialitzats.

Sens dubte, aquest projecte es pot ampliar en grans dimensions en un futur, i m'alegro d'haver aportat els fonaments i l'estructura del mateix.

BIBLIOGRAFIA

[1] Llorente, J. (2017, 31 agosto). *Breve historia de la Termodinámica - Blog de Ingeniería Industrial*. Blog de Ingeniería Industrial.

<https://blogs.udima.es/ingenieria-industrial/breve-historia-de-la-termodinamica/>

[2] What is Thermodynamic Engineering? (2023). *All the Science*.

<https://www.allthescience.org/what-is-thermodynamic-engineering.htm>

[3] *Guia docent 300209 - TER - Termodinàmica*. (2023).

https://www.upc.edu/estudispdf/guia_docent.php?codi=300209

[4] *PhET Interactive Simulations*. (s. f.). PhET. <https://phet.colorado.edu/>

[5] Javier, V. M. F. (2016, 28 enero). *Detección y corrección de ideas previas erróneas en la praxis docente de la física con apoyo de las TIC*.

<https://upcommons.upc.edu/handle/2117/96198?show=full>

[6] *JOptics*. (s. f.). <http://www.ub.edu/javaoptics/index-en.html>

[7] *Understanding Optical Trapping Phenomena: A Simulation for Undergraduates*. (2011b, febrero 1). IEEE Journals & Magazine | IEEE Xplore.

<https://ieeexplore.ieee.org/document/5451098>

[8] Ghajar, A., Delahoussaye, R. D., & Nayak, V. V. (2023). Development and Implementation of Interactive/Visual Software for Simple Aircraft Gas Turbine Design. *ResearchGate*.

https://www.researchgate.net/publication/228856367_Development_and_Implementation_of_InteractiveVisual_Software_for_Simple_Aircraft_Gas_Turbine_Design

[9] Brown, B. P., & Singh, C. (2022). Student understanding of thermodynamic processes, variables and systems. *European Journal of Physics*, 43(5), 055705.

<https://doi.org/10.1088/1361-6404/ac7af2>

[10] *Programación (Informática) - Qué es, información, lenguajes*. (s. f.).

Concepto. <https://concepto.de/programacion/>

[11] *Lenguaje de Programación - Concepto, tipos y ejemplos*. (s. f.). Concepto.

<https://concepto.de/lenguaje-de-programacion/>

[12] Pedamkar, P. (2023). What is Matlab? *EDUCBA*.

<https://www.educba.com/what-is-matlab/>

[13] David. (2018). 7 ventajas de usar MATLAB. *Componentes Electrónicas LTDA*.

<https://www.compelect.com.co/2018/01/26/7-ventajas-de-usar-matlab/>

[14] Andree-Gutierrez-Suclla. (2022). MATLAB Ventajas y Desventajas.

vdocuments.site. <https://vdocuments.site/matlab-ventajas-y-desventajas.html?page=1>

- [15] Coursera. (2023). ¿Qué es Python y para qué se usa? Guía para principiantes. Coursera. <https://www.coursera.org/mx/articles/what-is-python-used-for-a-beginners-guide-to-using-python>
- [16] KeepCoding, R. (2022, 13 diciembre). Ventajas y Desventajas de Python | KeepCoding Bootcamps. *KeepCoding Bootcamps*. <https://keepcoding.io/blog/ventajas-y-desventajas-de-python/>
- [17] Rane, Z. (2022, 5 enero). Matlab vs Python: 9 Comparisons For Which Language is Best for You | Towards Data Science. *Medium*. <https://towardsdatascience.com/theres-a-clear-winner-between-matlab-vs-python-f6bb56b2b930>
- [18] *MATLAB vs. Python ¿Cuál se adapta mejor a sus necesidades?* (s. f.). MATLAB & Simulink. <https://es.mathworks.com/products/matlab/matlab-vs-python.html>
- [19] *Machine Learning*. (s. f.). <https://www.ibm.com/mx-es/analytics/machine-learning#:~:text=Aprendizaje%20iterativo.%20Machine%20learning%20permite%20modelos%20a%20entrenar,de%20asociaciones%20hechas%20entre%20os%20elementos%20de%20datos.>
- [20] *What is Deep Learning?* | IBM. (s. f.). <https://www.ibm.com/topics/deep-learning>
- [21] *Anaconda Navigator — Anaconda documentation*. (s. f.). https://docs.anaconda.com/free/navigator/?utm_source=anaconda_navigator&utm_medium=nav-docs#anaconda-navigator
- [22] *IDE PyCharm: definición, características y extensiones*. (s. f.). <https://openbootcamp.com/aprender-programar/tipos-de-ide-pycharm>
- [23] Flores, F. (2023, 13 abril). Qué es Visual Studio Code y qué ventajas ofrece. *OpenWebinars.net*. <https://openwebinars.net/blog/que-es-visual-studio-code-y-que-ventajas-ofrece/>
- [24] Cardellino, F. (2021). La guía definitiva del paquete NumPy para computación científica en Python. *freeCodeCamp.org*. <https://www.freecodecamp.org/espanol/news/la-guia-definitiva-del-paquete-numpy-para-computacion-cientifica-en-python/>
- [25] Contributors, M. O. J. T. A. B. (s. f.). *Bootstrap*. <https://getbootstrap.com/>
- [26] *Python in Visual Studio Code*. (2021, 3 noviembre). <https://code.visualstudio.com/docs/languages/python>
- [27] Muñoz, J. D. (2023, 14 abril). Qué es Flask. *OpenWebinars.net*. <https://openwebinars.net/blog/que-es-flask/>

[28] B, G., & B, G. (2023). ¿Qué es MySQL? Explicación detallada para principiantes. *Tutoriales Hostinger*. <https://www.hostinger.es/tutoriales/que-es-mysql>

[29] KeepCoding, R. (2022a, agosto 10). ¿Qué es render template de Flask? | KeepCoding Bootcamps. *KeepCoding Bootcamps*.

<https://keepcoding.io/blog/que-es-render-template-de-flask/>

[30] *Request Flask*. (2023, 6 junio). Manual Web.

<https://www.manualweb.net/flask/request-flask/>

[31] *HTML: Lenguaje de etiquetas de hipertexto | MDN*. (2023, 13 marzo).

<https://developer.mozilla.org/es/docs/Web/HTML>

[32] Anaconda Inc. (s. f.). *Pyscript.net*. <https://pyscript.net/>

[33] *Ngrok*. (s. f.). <https://ngrok.com/>

[34] Núñez, M. (2023). Qué es un hosting y cómo funciona. *Webempresa*.

<https://www.webempresa.com/hosting/hosting-que-es-comofunciona.html#:~:text=Un%20hosting%20es%20un%20servicio%20de%20alojamiento%20para,todo%20momento%20desde%20cualquier%20dispositivo%20conectado%20a%20Internet.>

[35] Provider, B. W. H. (s. f.). *Mejor Hosting 2022 - Dominios - WordPress - VPS Server - Servidores Dedicados - BanaHosting.com*. Copyright by

banaHosting Web Hosting. All Rights Reserved. <https://banahosting.com/es/>

[36] Cengel, Y. A., & Boles, M. A. (2015). *Thermodynamics: An Engineering Approach*.

[37] Zemansky, M. W., & Dittman, R. (1997). *Heat and Thermodynamics: An Intermediate Textbook*. McGraw-Hill Science, Engineering & Mathematics.

[38] Fermi, E. (1956). *Thermodynamics*. Dover Publications.

[39] *Thermodynamics*. (s. f.). <https://www.grc.nasa.gov/WWW/k-12/airplane/thermo.html#:~:text=Thermodynamics%20is%20a%20branch%20of%20physics%20which%20deals,which%20we%20can%20observe%20and%20measure%20in%20experiments.>

[40] *Third Law of Thermodynamics: Definition, Equation & Examples*. (2021, 29 abril). Sciencing. <https://sciencing.com/third-law-of-thermodynamics-definition-equation-examples-13722779.html>

[42] *Variables Termodinámicas*. (2016, 24 septiembre). Termodinámica.

<https://termodinamicaweb.wordpress.com/variables-termodinamicas/>

- [42] Redacciones. (2023). ENERGÍA INTERNA » Qué es, Características, Fórmula - Cumbre Pueblos. *Cumbre Pueblos*.
https://cumbrepuebloscop20.org/energias/interna/?utm_content=cmp-true
- [43] Planas, O. (s. f.). ¿Qué es la electricidad? [Vídeo]. <https://solar-energia.net/termodinamica/propiedades-termodinamicas/entalpia>
- [44] Entropía_ (termodinámica). (s. f.).
https://www.quimica.es/enciclopedia/Entrop%C3%ADa_%28termodin%C3%A1mica%29.html
- [45] Entropy of a Gas. (s. f.). <https://www.grc.nasa.gov/WWW/K-12/VirtualAero/BottleRocket/airplane/entropy.html>
- [46] Física universitaria volumen 2 | OpenStax. (s. f.).
<https://openstax.org/books/f%C3%ADsica-universitaria-volumen-2/pages/1-introduccion>
- [47] Ciclo Diesel. (2015, 25 mayo). TERMODINÁMICA.
<https://pilarospina96.wordpress.com/tercer-corte/ciclo-diesel/>
- [48] Admin, & Admin. (2019). Ciclo de Brayton. *Explicacion .Net*.
<https://www.explicacion.net/ciclo-de-brayton/#:~:text=El%20ciclo%20b%C3%A1sico%20Brayton%20ideal%20est%C3%A1%20compuesto%20por,remove%20el%20calor%20del%20fluido%20a%20presi%C3%B3n%20constante>
- [49] Zapata, F. (2019). Ciclo Otto: fases, rendimiento, aplicaciones, ejercicios resueltos. *Lifeder*. <https://www.lifeder.com/ciclo-otto/>

CAPÍTOL 4. ANNEXOS

4.1. Annex 1. Conceptes termodinàmics

La termodinàmica [39] és una branca de la física que s'ocupa de l'energia i el treball d'un sistema. Va néixer al segle XIX quan els científics van descobrir per primera vegada com construir i operar màquines de vapor. La termodinàmica s'ocupa només de la resposta a gran escala d'un sistema que podem observar i mesurar en experiments. Les interaccions de gasos a petita escala són descrites per la teoria cinètica dels gasos. Els mètodes es complementen; alguns principis s'entenen més fàcilment en termes de termodinàmica i alguns principis s'expliquen més fàcilment per la teoria cinètica.

4.1.1. Lleis de la termodinàmica

Les lleis de la termodinàmica expliquen el comportament de tres quantitats físiques bàsiques dels sistemes termodinàmics: temperatura, energia, i entropia. Matemàticament, aquests principis es descriuen mitjançant un conjunt d'equacions que expliquen el comportament dels sistemes termodinàmics, definits com qualsevol objecte d'estudi.

Hi ha quatre lleis fonamentals de la termodinàmica que són essencials per a la comprensió dels principis físics universals i la impossibilitat d'alguns fenòmens.

4.1.1.1. Principi zero de la termodinàmica

La llei zero de la termodinàmica [39] comença amb una definició senzilla d'equilibri termodinàmic. S'observa que alguna propietat d'un objecte, com la pressió en un volum de gas, la longitud d'una vareta metàl·lica o la conductivitat elèctrica d'un cable, pot canviar quan l'objecte s'escalfa o es refreda. Si dos d'aquests objectes es posen en contacte físic, inicialment hi ha un canvi en la propietat d'ambdós objectes. Però, finalment, el canvi de propietat s'atura i es diu que els objectes es troben en equilibri tèrmic o termodinàmic.

L'equilibri termodinàmic condueix a la definició a gran escala de la temperatura. Quan dos objectes estan en equilibri tèrmic es diu que tenen la mateixa temperatura. Durant el procés d'arribar a l'equilibri tèrmic, la calor, que és una forma d'energia, es transfereix entre els objectes. Els detalls del procés d'arribar a l'equilibri tèrmic es descriuen en la primera i segona lleis de la termodinàmica.

La llei zero de la termodinàmica és una observació. Quan dos objectes en equilibri termodinàmic estan separats per un tercer objecte, doncs, estan en equilibri entre ells. Els objectes en equilibri termodinàmic tenen la mateixa temperatura.

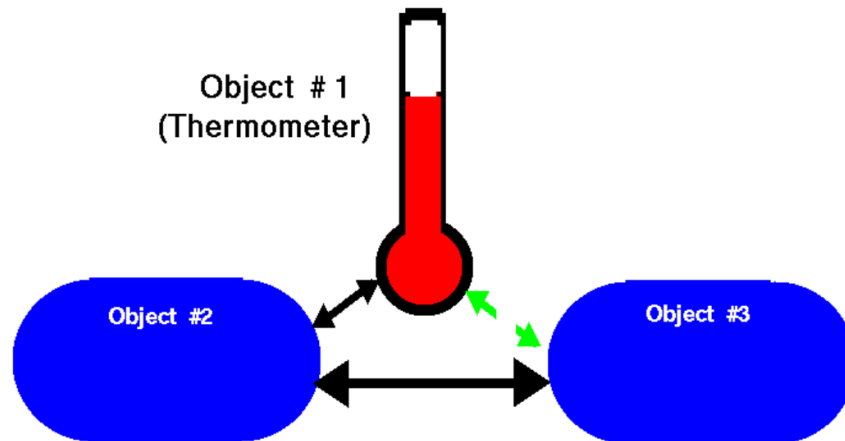


Fig. 4.1 Principi zero de la termodinàmica [39].

En la figura 4.1 es representen tres objectes, l'objecte #1 i l'objecte #2. L'objecte #1 i l'objecte #2 es troben en contacte físic i en equilibri tèrmic. L'objecte #2 també es troba en equilibri tèrmic amb l'objecte #3. Inicialment no hi ha contacte físic entre l'objecte #1 i l'objecte #3, però, si l'objecte #1 i l'objecte #3 es posen en contacte, s'observa que es troben en equilibri tèrmic. Aquesta senzilla observació permet crear un termòmetre.

Es pot calibrar el canvi d'una propietat tèrmica, com la longitud d'una columna de mercuri, posant el termòmetre en equilibri tèrmic amb un sistema físic conegut en diversos punts de referència. Els termòmetres Celsius tenen els punts de referència fixats en el punt de congelació i ebullició de l'aigua pura.

4.1.1.2. Primera llei de la termodinàmica

La primera llei de la termodinàmica [39] diu que entre dos estats d'equilibri, el canvi d'energia interna (E) és igual a la transferència de calor (Q) al sistema i al treball realitzat pel sistema (W).

$$\Delta U = Q - W \quad (4.1)$$

El treball realitzat en, o per un gas i la quantitat de calor transferida a o des d'un gas no depèn només dels estats inicials i finals del gas, sinó també del procés, o camí que produeix fins arribar a l'estat final. Moltes observacions de gasos reals han demostrat que la diferència del flux de calor cap al gas i el treball realitzat pel gas depèn només dels estats inicial i final del gas i no depèn del procés o trajectòria que produeix l'estat final. Això suggereix l'existència d'una variable addicional, anomenada energia interna del gas, que depèn només de l'estat del gas i no de cap procés. L'energia interna és una variable d'estat, igual que la temperatura o la pressió.

A l'equació 4.1 la calor eliminada d'un sistema i el treball realitzat al sistema se li assignaria un signe negatiu.

L'energia interna és només una forma d'energia com l'energia potencial d'un objecte a certa altura sobre la terra, o l'energia cinètica d'un objecte en moviment. De la mateixa manera que l'energia potencial es pot convertir en energia cinètica conservant l'energia total del sistema, l'energia interna d'un sistema termodinàmic es pot convertir en energia cinètica o potencial. Igual que l'energia potencial, l'energia interna es pot emmagatzemar al sistema. Però, la calor i el treball no es poden emmagatzemar ni conservar de manera independent, ja que depenen del procés.

Doncs la primera llei de la termodinàmica també se li pot dir llei de la conservació de l'energia perquè dicta que en qualsevol sistema físic aïllat del seu entorn, la quantitat total d'energia serà sempre la mateixa, tot i que pugui transformar-se d'una forma d'energia a altres de diferents. O dit en altres paraules: l'energia no es pot crear ni destruir, només transformar-se, doncs podem afirmar que l'energia es conserva en tots els processos termodinàmics.

4.1.1.3. Segona llei de la termodinàmica

La primera llei de la termodinàmica permet que existeixin molts estats possibles d'un sistema, no obstant això, es poden imaginar processos termodinàmics que conserven l'energia però que mai es produeixen a la natura. Per exemple, si es posa en contacte un objecte calent amb un objecte fred, l'objecte calent es refreda i l'objecte fred s'escalfa fins a arribar a un equilibri. La transferència de calor va de l'objecte calent a l'objecte fred. Es pot imaginar un sistema en que la calor es transfereix de l'objecte fred a l'objecte calent, i aquest sistema no viola la primera llei de la termodinàmica. L'objecte fred es torna més fred i l'objecte calent s'escalfa, però es conserva l'energia. Òbviament aquest sistema no es troba a la natura, la segona llei de la termodinàmica ajuda a explicar aquest fenomen.

La segona llei estableix [39] que existeix una variable d'estat útil anomenada entropia. El canvi d'entropia (ΔS) és igual a la transferència de calor (ΔQ) dividida per la temperatura (T).

$$\Delta S = \Delta Q/T \quad (4.2)$$

La segona llei de la termodinàmica estableix que l'entropia total de l'univers o d'un sistema aïllat mai disminueix. En termodinàmica, un sistema aïllat és aquell en el qual ni la calor ni la matèria poden entrar o sortir dels límits del sistema.

S'ha de tenir en compte que per a un procés físic determinat, l'entropia del sistema i de l'entorn seguirà sent una constant si es pot invertir el procés, doncs en aquest cas es pot afirmar que l'entropia inicial del procés és igual a la final.

En un procés reversible l'estat del gas tornaria a les seves condicions originals i el canvi d'entropia del sistema seria zero. La segona llei estableix que si el procés físic és irreversible, l'entropia del sistema i del medi ambient ha d'augmentar; l'entropia final ha de ser més gran que l'entropia inicial. En un procés irreversible l'entropia final és major a l'entropia inicial.

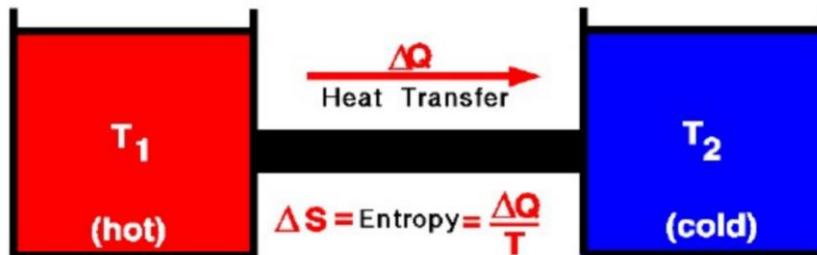


Fig. 4.2 Segona llei de la termodinàmica [39].

L'aplicació de la segona llei descriu per què es transfereix calor de l'objecte calent a l'objecte fred. A la figura 4.2 es suposa que la calor es transfereix de l'objecte calent (objecte 1) a temperatura T_1 a l'objecte fred (objecte 2) a temperatura T_2 . La quantitat de calor transferida és Q , la temperatura d'equilibri final per a tots dos objectes és T_f . La temperatura de l'objecte calent canvia a mesura que la calor es transfereix doncs la temperatura mitjana d'aquest objecte durant el procés s'anomena T_h i és igual a la mitjana de T_1 i T_f , l'equació 4.3 defineix el valor de T_h .

$$T_h = (T_1 + T_f)/2 \quad (4.3)$$

Per a l'objecte fred, la temperatura final és T_f i la temperatura mitjana durant el procés és T_c , que és la mitjana de T_f i T_2 , l'equació 4.4 defineix el valor de T_c .

$$T_c = (T_2 + T_f)/2 \quad (4.4)$$

T_h és més gran que T_c , perquè T_1 és més gran que T_2 . El canvi d'entropia de l'objecte calent es defineix a l'equació 4.5, el signe menys significa que la calor es transfereix de l'objecte cap a l'entorn.

$$\Delta S_h = -Q/T_h \quad (4.5)$$

Per a l'objecte fred, el canvi d'entropia es defineix a l'equació 4.6, en aquest cas el signe es postiu perquè la calor es transfereix de l'entorn a l'objecte.

$$\Delta S_c = Q/T_c \quad (4.6)$$

Així doncs, el canvi total d'entropia per a tot el sistema vindria donat per l'equació 4.7.

$$S_f = S_i + Q/T_c - Q/T_h \quad (4.7)$$

En aquest cas el terme Q/T_c sempre serà més gran que $-Q/T_h$ perquè T_h és més gran que T_c . Per tant, S_f serà més gran que S_i , tal com preveu la segona llei. Si, en canvi, s'hagués assumit que la calor s'estava transferint de l'objecte fred a l'objecte calent, la equació final seria la de l'equació 4.8.

$$S_f = S_i - Q/T_c + Q/T_h \quad (4.8)$$

En aquest cas S_f és menor que S_i i l'entropia del sistema disminueix i això viola la segona llei de la termodinàmica.

4.1.1.4. Tercera llei de la termodinàmica

La tercera llei de la termodinàmica [40] estableix que a mesura que la temperatura s'acosta al zero absolut en un sistema, l'entropia absoluta del sistema s'acosta a un valor constant.

El zero Kelvin absolut és la temperatura a la qual cessa tot moviment molecular. Com que la calor és defineix com un moviment molecular en el sentit més senzill, si no hi ha moviment significa que no hi ha calor i això vol dir una temperatura de zero Kelvin.

El més important és que la tercera llei descriu una veritat important de la natura: Qualsevol substància a una temperatura superior al zero absolut (per tant, qualsevol substància coneguda) ha de tenir una quantitat d'entropia positiva. A més, com que defineix el zero absolut com a punt de referència, es pot quantificar la quantitat relativa d'energia de qualsevol substància a qualsevol temperatura. Aquesta és una diferència clau respecte a altres mesures termodinàmiques, com l'energia o l'entalpia, per a les quals no hi ha un punt de referència absolut. Aquests valors només tenen sentit en relació amb altres valors.

La tercera llei també dona suport a les implicacions de la primera llei de la termodinàmica.

4.1.2 Variables termodinàmiques

Les variables [41] que es relacionen amb l'estat intern d'un sistema es denominen variables termodinàmiques o coordenades termodinàmiques. Les més importants per a l'estudi de la termodinàmica són: la massa, el volum, la densitat, la temperatura i la pressió.

En termodinàmica és molt important estudiar les seves propietats, les quals es poden dividir en dos:

- Propietats intensives: Propietat que no depèn de la quantitat de matèria ni de la mida del sistema, i no és additiva perquè el seu valor no canvia quan el sistema inicial es divideix en diferents subsistemes.
- Propietats extensives: Aquests depenen de la quantitat de matèria en el sistema i són mútuament equivalents als concentrats. Per tant, les propietats extensives depenen de la mida del sistema. Les característiques d'expansió són additives en el sentit que si un sistema es divideix en dues o més parts, el valor de la quantitat d'expansió per a tot el sistema és la suma dels valors d'aquesta quantitat per a cada part.

Exemples de propietats extensives són la massa, el volum, el pes, la quantitat de matèria, l'energia, l'entropia i l'entalpia. En general, el quocient de dues magnituds extensives dóna una magnitud intensiva. Per exemple, en dividir la massa pel volum s'obté la densitat. A continuació s'explicaran en més profunditat tres variables extensives.

4.1.1.5. *Energia interna*

L'energia interna [42] d'un sistema s'identifica com l'energia relativa al moviment aleatori i desordenat de les molècules. Es representa amb la lletra U.

Aquesta energia en un sistema inclou energia potencial i cinètica. Això contrasta amb l'energia externa, que és una funció de la mostra respecte a l'entorn exterior.

Aquest tipus d'energia inclou energia a escala microscòpica. És la suma de totes les energies microscòpiques com ara: energia cinètica de translació, energia cinètica vibracional i rotacional, energia potencial de forces intermoleculares.

A continuació es descriuen les principals característiques que ens permeten comprendre l'estudi d'aquesta energia:

- L'energia interna és una propietat d'estat i el seu canvi no depèn del camí pel qual s'assoleix l'estat final.
- És una propietat extensiva, això vol dir que depèn de la mida del sistema o de la quantitat de substància que conté.

- No hi haurà cap canvi en l'energia interna en el procés cíclic, l'energia interna del gas ideal és una funció de la temperatura només.
- Depèn de la quantitat de la substància, la seva temperatura, naturalesa química, pressió i volum.

L'equació per calcular la variació total d'energia interna d'un sistema és:

$$\Delta U = Q + S \quad (4.9)$$

En l'equació 4.9, la variable Q representa calor i la variable W treball.

Si l'energia que ingressa al sistema és positiva, significa que la calor s'absorbeix ($Q > 0$). El treball es realitza cap el sistema ($W > 0$). Si l'energia que surt del sistema és negativa, significa que el sistema emet calor ($Q < 0$) i el sistema realitza el treball ($W < 0$).

Per consideracions pràctiques en termodinàmica o enginyeria, rara vegada és necessari, convenient, o tan sols possible, considerar totes les energies que pertanyen a l'energia intrínseca total d'un sistema de mostra, com l'energia donada per l'equivalència de massa. Normalment, les descripcions només inclouen components rellevants per al sistema en estudi. De fet, en la majoria dels sistemes sota consideració, especialment a través de la termodinàmica, és impossible calcular l'energia interna total. Per tant, es pot triar un punt de referència nul convenient per a l'energia interna. A qualsevol temperatura més gran que el zero absolut, l'energia potencial microscòpica i l'energia cinètica es converteixen constantment entre si, però la suma roman constant en un sistema aïllat. En la imatge clàssica de la termodinàmica, l'energia cinètica s'esvaeix a temperatura zero i l'energia interna és purament energia potencial.

4.1.1.6. Entalpia

L'entalpia [43] és la quantitat d'energia que un sistema termodinàmic pot intercanviar amb el seu entorn. L'entalpia també es coneix com a entalpia absoluta o calor de reacció i es representa amb la lletra H. Tal com es defineix, l'entalpia abasta en si no només l'energia interna del sistema, sinó també l'energia emmagatzemada en el medi ambient del sistema (rodalies).

En definitiva, l'entalpia és la suma de l'energia interna de la matèria i el producte del seu volum per la pressió d'acord l'equació 4.10. Per tant, l'entalpia és igual a l'energia d' un sistema termodinàmic més el treball mecànic que se li aplica.

$$H = U + P \cdot V \quad (4.10)$$

Aquesta propietat és una funció d'estat quantificable. És a dir, tot i que l'entalpia total d'un sistema no es pot mesurar directament, es pot mesurar la capacitat de rebre o aportar energia d'un sistema.

L'entalpia serveix per mesurar amb exactitud les variacions d'energia que es produeixen en un sistema, bé sigui en el moment de prendre o alliberar energia a l'ambient.

4.1.1.7. Entropia

Es denomina entropia [44] a la magnitud que indica l'energia que no pot realitzar un treball útil en un procés termodinàmic. En general l'univers tendeix al desordre. En conseqüència, l'entropia, és a dir a l'augment de l'entropia, per tant, la variació positiva d'aquesta magnitud indica el sentit natural en què ocorre qualsevol esdeveniment en un sistema aïllat. Es representa amb la lletra S.

L'entropia és una magnitud termodinàmica definida originalment com a criteri per predir l'evolució dels sistemes termodinàmics. En tot procés irreversible, el desordre del sistema augmenta i per tant, l'entropia augmenta. Si el procés és reversible, la variació d'entropia és nul·la.

L'entropia d'un sistema és una funció d'estat de caràcter extensiu. El valor d'aquesta magnitud física, en un sistema aïllat, creix en el transcurs d'un procés que es dona de forma natural. El concepte d'entropia descriu com és d'irreversible un sistema termodinàmic.

Durant un procés termodinàmic [45], la temperatura d'un objecte canvia a mesura que s'aplica o s'extreu calor. Una definició més correcta de l'entropia és la forma diferencial que explica aquesta variació i es representa a l'equació 4.11.

$$dS = dQ / T \quad (4.11)$$

El canvi d'entropia és llavors l'invers de la temperatura integrada sobre el canvi en la transferència de calor. Per als gasos, hi ha dues maneres possibles d'avaluar el canvi en l'entropia. Les equacions es poden formular en termes de l'energia interna i la definició del treball per a un gas.

$$dQ = de + p \cdot dV \quad (4.12)$$

Utilitzant l'equació d'estat, es pot modificar el segon terme d'aquestes equacions en termes de la constant dels gasos, la pressió, la temperatura i el volum.

$$dQ = de + R \cdot TdV/V \quad (4.13)$$

La transferència de calor d'un gas és igual a la constant de calor específica vegades el canvi de temperatura, s'expressa en forma diferencial a l'equació 4.14.

$$dQ = \text{constant} \cdot dT \quad (4.14)$$

Si hi ha un procés de volum constant, la formulació de la primera llei s'expressa com l'equació 4.15.

$$de = dQ = C (\text{volum constant}) \cdot dT \quad (4.15)$$

Combinant amb la primera llei:

$$dQ = C (\text{volum constant}) \cdot dT + R \cdot TdV/V \quad (4.16)$$

Llavors el canvi d'entropia ve donat per:

$$dS = C (\text{volum constant}) \cdot dT / T + R \cdot dV/V \quad (4.17)$$

Les equacions també es poden formular en termes de l'entalpia del gas amb el resultat com s'expressa a l'equació 4.18.

$$dS = C (\text{pressió constant}) \cdot dT / T - R \cdot dp/p \quad (4.18)$$

Aquestes dues relacions són equacions diferencials que es poden integrar fàcilment i s'obté l'equació 4.19 i l'equació 4.20.

$$s_2 - s_1 = C (\text{volum constant}) \cdot \ln (T_2/T_1) + R \cdot \ln(v_2/v_1) \quad (4.19)$$

$$s_2 - s_1 = C (\text{pressió constant}) \cdot \ln (T_2/T_1) - R \cdot \ln(p_2/p_1) \quad (4.20)$$

4.1.2. Processos termodinàmics

El comportament tèrmic d'un sistema es descriu utilitzant variables termodinàmiques. Pels gasos ideals, aquestes variables són la pressió, el volum, la temperatura i el nombre de molècules o mols de gas. Els diferents tipus de sistemes sovint es caracteritzen per diferents conjunts de variables.

L'estat d'un sistema pot canviar a través de la interacció amb l'entorn. Els canvis del sistema poden ser ràpids o lents, grans o petits. La forma en què l'estat d'un sistema canvia del seu estat inicial al seu estat final es denomina procés termodinàmic [46]. Per a fins analítics en termodinàmica, és útil dividir els processos en quasi-estàtics i no quasi-estàtics.

4.1.2.1. Processos quasi-estàtics i processos no quasi-estàtics

Un procés quasi-estàtic es refereix a un procés idealitzat o imaginari en què el canvi d'estat és infinitament lent i es pot suposar que el sistema està en equilibri termodinàmic amb si mateix i amb l'entorn en tot moment. El procés quasi-estàtic és prou lent com perquè el sistema sempre romangui en equilibri termodinàmic fins i tot quan el sistema canvia amb el temps. L'equilibri termodinàmic d'un sistema és necessari perquè el sistema tingui valors ben definits de propietats macroscòpiques com la temperatura i la pressió del sistema en qualsevol punt del procés. Per tant, els processos quasi-estàtics es poden mostrar com a trajectòries ben definides en l'espai d'estats de sistema.

Atès que els processos quasi-estàtics no es poden fer completament per a qualsevol canvi finit del sistema, tots els processos de la natura són no quasi-estàtics.

A la figura 4.3 es mostra uns exemples de processos quasi-estàtics i no quasi-estàtics entre els estats A i B d' un gas. En un procés quasi-estàtic, la trajectòria del procés entre A i B es pot dibuixar en un diagrama d'estats, ja que es coneixen tots els estats pels quals passa el sistema. En un procés no quasi-estàtic, no es coneixen els estats entre A i B, per la qual cosa no es pot traçar cap trajectòria. Pot seguir la línia discontinua com es mostra en la figura o prendre una trajectòria molt diferent.

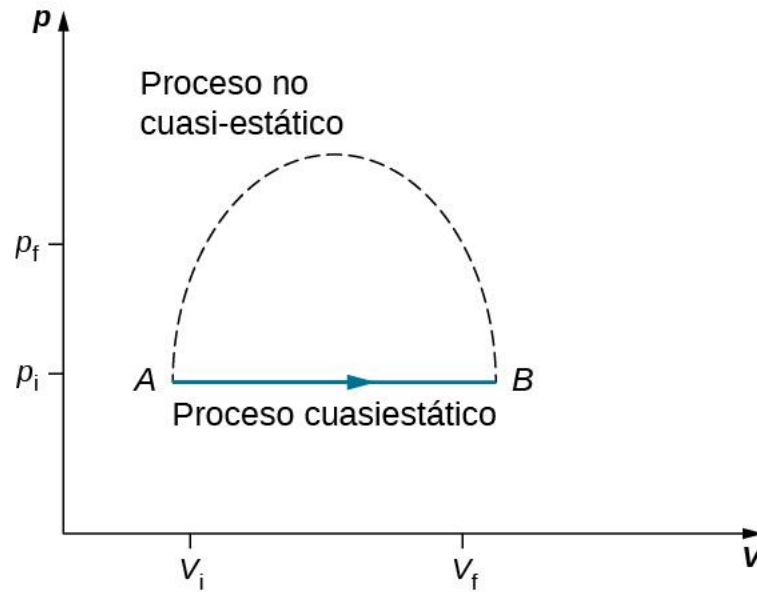


Fig. 4.3 Processos quasi-estàtics i no quasi-estàtics entre els estats A i B d'un gas [46].

4.1.2.2. Procés isotèrmic

Un procés isotèrmic [46] és un canvi d'estat d'un sistema a una temperatura constant. Aquest procés s'assoleix mantenint el sistema en equilibri tèrmic utilitzant un gran bany de calor durant el procés.

Per il·lustrar un procés isotèrmic, es considera un cilindre de gas amb un pistó en moviment submergit en un gran tanc d'aigua mantingut a temperatura constant. Atès que el pistó es pot moure lliurement, la pressió interna s'equilibra amb la pressió externa mitjançant un pes sobre el pistó, com es mostra en la figura 4.4. L'eliminació de les peses al pistó provoca un desequilibri de forces, cosa que fa que es mogui cap amunt. Quan el pistó es mou cap amunt, la temperatura es redueix momentàniament, cosa que fa que la calor flueixi des del bany de calor cap al sistema. L'energia per moure el pistó prové finalment del bany de calor.

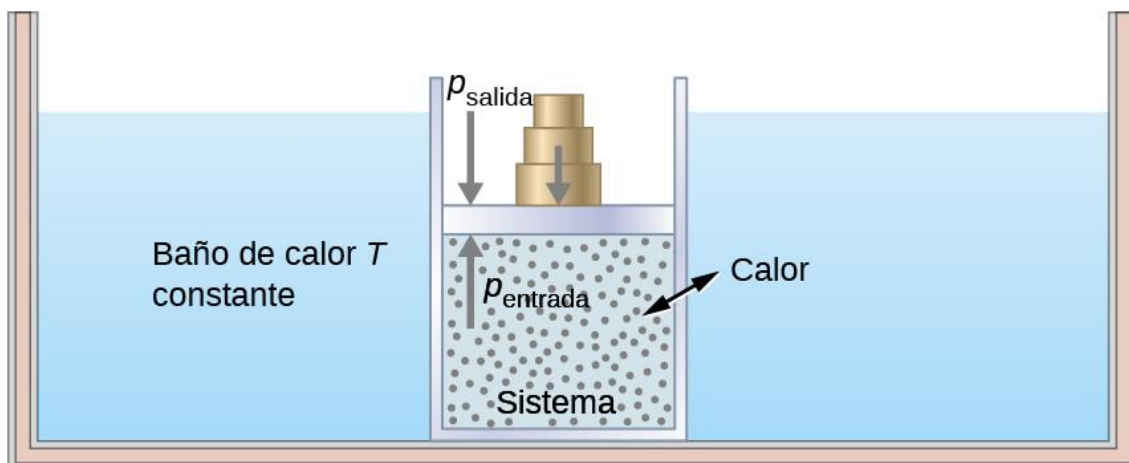


Fig. 4.4 Expansió d'un sistema a temperatura constant [46].

L'expansió del gas el refreda a una temperatura més baixa i la calor flueix cap al sistema des del bany de calor fins que la temperatura del gas torna a la temperatura del bany de calor. Si es treu el pes en petits passos fins que desaparegui, la pressió en el sistema disminuirà fins que desaparegui. D'aquesta manera, els processos isotèrmics poden executar-se quasi-estàticament. Una isoterma en un diagrama (p, V) es representa mitjançant una corba des del punt inicial A fins al punt final B, com es mostra en la figura 4.5. En un gas ideal, el procés isotèrmic és hiperbòlic perquè està a temperatura constant.

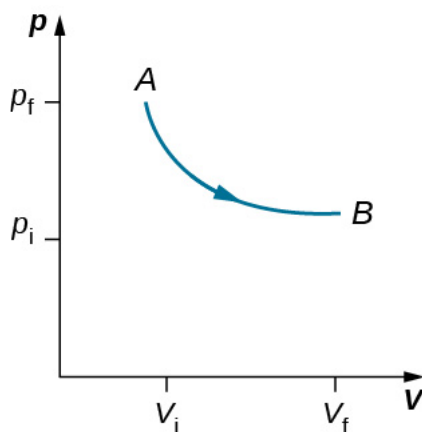


Fig. 4.5 Una expansió isotèrmica d'un estat A a un estat B [46].

Un procés isotèrmic en aquest treball es realitza de forma quasi-estàtica, ja que perquè sigui isotèrmic al llarg del canvi de volum, s'ha de poder indicar la temperatura del sistema en cada pas, la qual cosa és possible només si el sistema està en equilibri tèrmic de forma contínua. El sistema ha de sortir de l'equilibri perquè l'estat canviï, però per als processos quasi-estàtics, imaginem

que el procés es fa en passos infinitesimals, de manera que aquestes sortides de l'equilibri poden ser tan breus i petites com es vulgui.

4.1.2.3. Procés isobàric

El procés isobàric [46] és aquell en què la pressió es manté constant ($\Delta P = 0$). Si bé la calor que es transfereix al sistema funciona, l'energia interna d'aquest sistema canvia. En aquest procés la pressió i la força exercida són constants i el treball realitzat es dona com a $P\Delta V$.

Com a exemple tenim un pistó mòbil en un cilindre. La pressió dins del cilindre està sempre a la pressió atmosfèrica, encara que estigui aïllada de l'atmosfera.

En altres paraules, el sistema està connectat dinàmicament, per un límit mòbil, a un dipòsit de pressió constant. Si un gas s'expandeix a una pressió constant, la calor s'ha de transferir al sistema a una certa velocitat. Aquest procés s'anomena una expansió isobàrica.

En un diagrama de fase, un procés isobàric es dibuixa com una línia horitzontal, ja que es duu a terme sota una pressió constant, com es mostra a la figura 4.6.

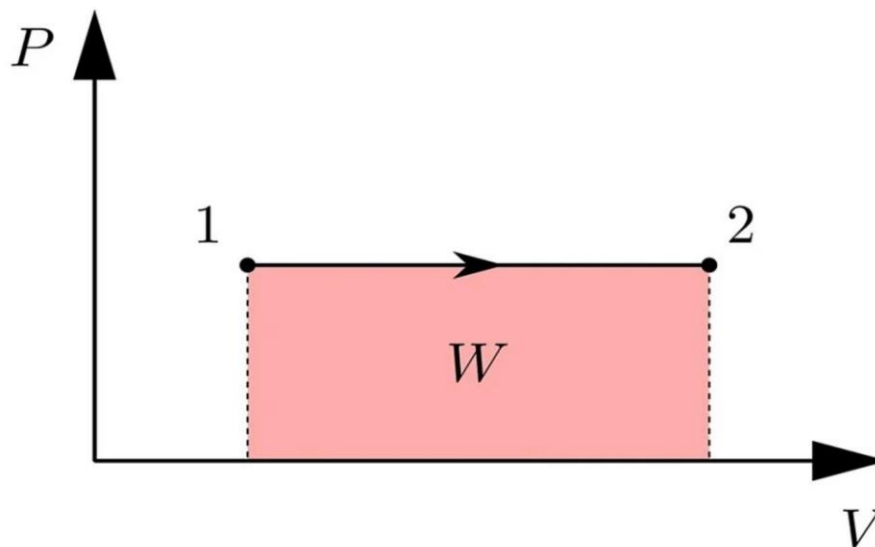


Fig. 4.6 Una expansió isobàrica d'un estat 1 a un estat 2 [46].

4.1.2.4. Procés isocor

Un procés isocor [46], també conegut com a procés a volum constant, és un tipus de procés termodinàmic en el qual el volum del sistema es manté

constant mentre es produeix un canvi en la temperatura i/o la pressió. En altres paraules, és un procés en què no hi ha canvi net en el volum del sistema. La principal característica del procés isocor que el volum del sistema es manté constant durant tot el procés. Això significa que no hi ha treball extern realitzat pel sistema, ja que el treball es defineix com la força aplicada a un objecte multiplicada per la distància que es mou. Un altre tret distintiu d'aquest procés és que, en general, es produeix un canvi en la temperatura i/o la pressió del sistema. El procés isocor també és conegut per ser un procés reversible.

En un diagrama de fase, un procés isobàric es dibuixa com una línia horitzontal, ja que es duu a terme sota un volum constant, com es mostra a la figura 4.7.

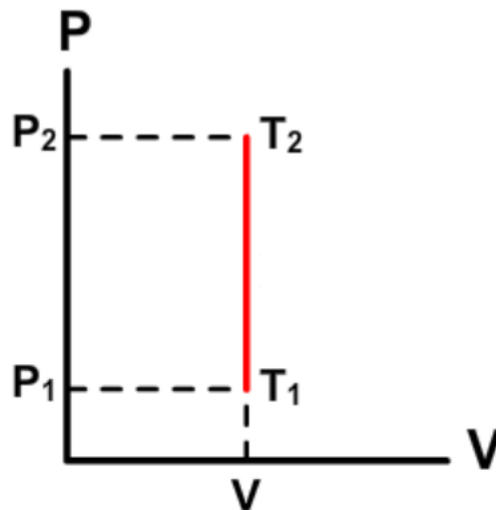


Fig. 4.7 Procés isocor d'un estat 1 a un estat 2 [46].

4.1.2.5. Procés adiabàtic

En un procés adiabàtic [46], el sistema està aïllat de l'entorn, per la qual cosa la calor no pot entrar ni sortir del sistema a mesura que canvia d'estat, com es mostra en la figura 4.8. Els processos adiabàtics poden ser processos quasi-estàtics o no quasi-estàtics. A mesura que el sistema s'expandeix adiabàticament, realitza un treball contra el món exterior, la qual cosa redueix l'energia del sistema i es reflecteix en el refredament del sistema. Una expansió adiabàtica condueix a una disminució de temperatura, i una compressió adiabàtica condueix a un augment de temperatura.

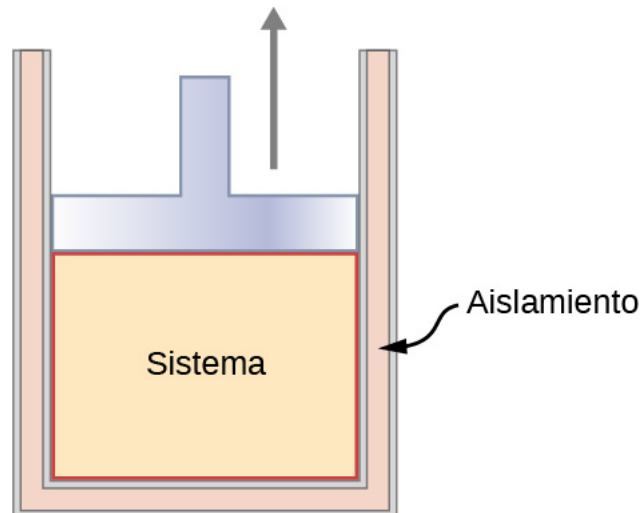


Fig. 4.8 Alliberació d'un pistó aïllat amb un gas calent i comprimit [46].

A la figura 4.8 el pistó es mou cap amunt, el volum s'expandeix i la pressió i la temperatura disminueixen. L'energia interna va al treball. Si l'expansió es produeix en un temps en què la calor pot entrar en el sistema de forma insignificant, el procés es denomina adiabàtic. Idealment, durant un procés adiabàtic no entra ni surt calor del sistema.

4.1.3. Cicles termodinàmics

Es denomina cicle termodinàmic a qualsevol sèrie de processos termodinàmics tals que, al transcurs de tots ells, el sistema torna al seu estat inicial; és a dir, que la variació de les magnituds termodinàmiques pròpies del sistema sigui nul·la. No obstant, les variables com la calor o el treball no és aplicable a l'anteriorment dit ja que aquestes no són funcions d'estat del sistema, sinó transferències d'energia entre aquest i el seu entorn. Un fet característic dels cicles termodinàmics és que la primera llei de la termodinàmica dicta que: la suma de calor i treball rebuts pel sistema ha de ser igual a la suma de calor i treball realitzats pel sistema.

4.1.3.1. *Cicle de Carnot*

El cicle de Carnot [46] és d'especial importància per diverses raons. A nivell pràctic, aquest cicle representa un model reversible per a la central elèctrica de vapor i el refrigerador, o la bomba de calor. Tanmateix, també és molt important des del punt de vista teòric, ja que exerceix un paper fonamental en el desenvolupament d'un altre important enunciat de la segona llei de la termodinàmica. Finalment, com que només intervenen dos reservoris en el seu funcionament, pot utilitzar-se juntament amb la segona llei de la termodinàmica

per definir una escala de temperatura absoluta que sigui realment independent de qualsevol substància utilitzada per mesurar la temperatura. Amb un gas ideal com a substància de treball, les etapes del cicle de Carnot, representades per la figura 4.9, són les següents.

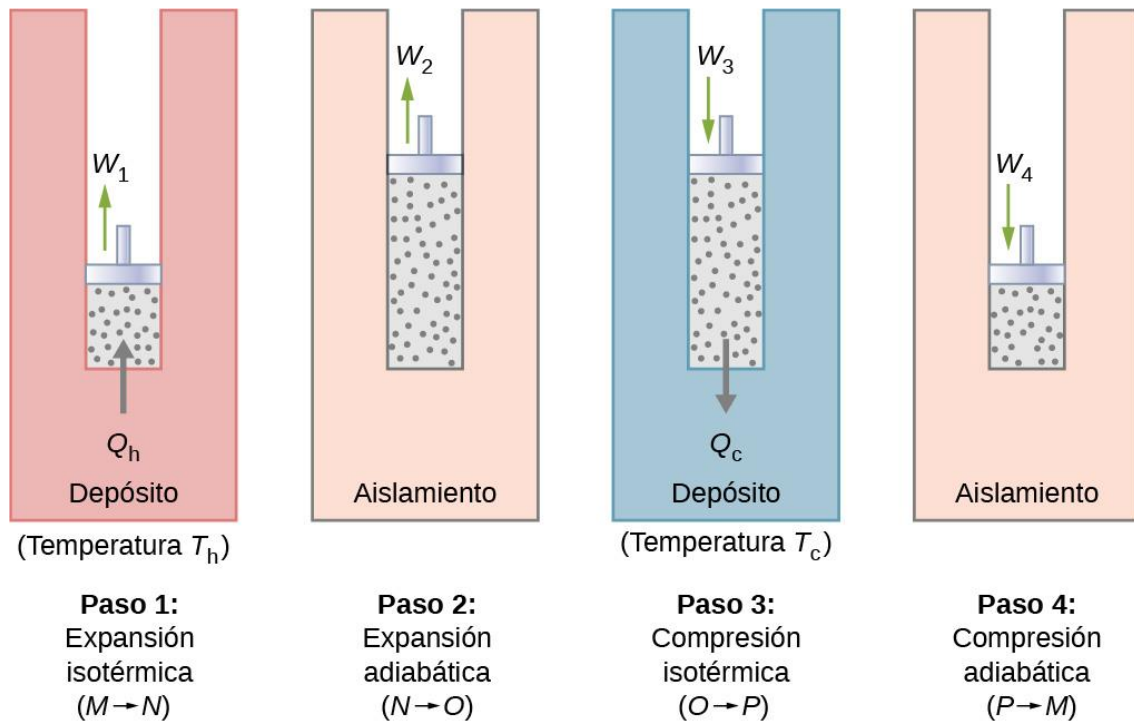


Fig. 4.9 Els quatre processos del cicle de Carnot [46].

A la figura 4.10 es mostra la trajectòria termodinàmica del cicle de Carnot. El treball total que realitza el gas en el cicle de Carnot es mostra i ve donat per l'àrea tancada pel bucle MNOPM.

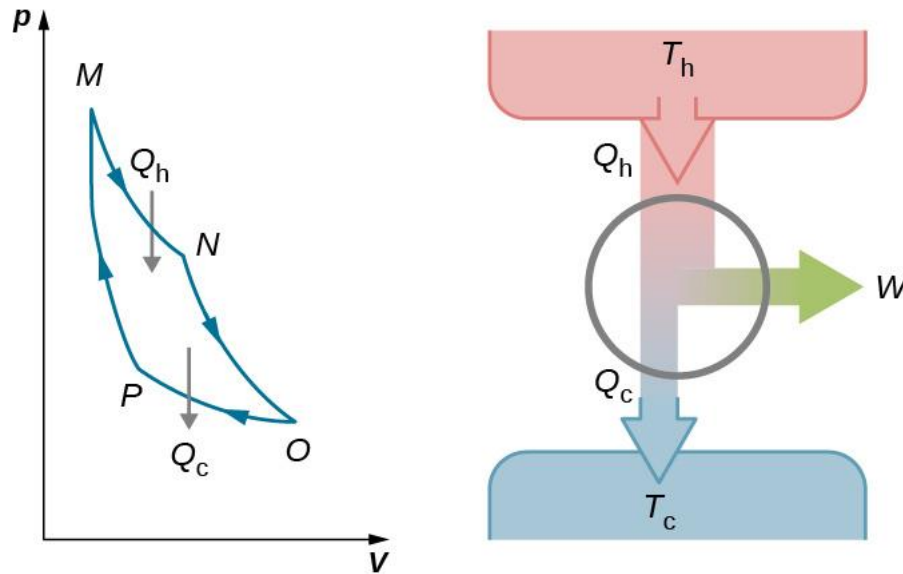


Fig. 4.10 Trajectòria termodinàmica del cicle de Carnot de la figura 4.9 [46].

- Pas 1 (expansió isotèrmica): El gas es posa en contacte tèrmic amb un reservori de calor a una temperatura T_h . El gas absorbeix la calor Q_h del reservori de calor i es deixa que s'expandeixi isotèrmicament, realitzant un treball W_1 . Ja que l'energia interna d'un gas ideal és una funció de la temperatura només, el canvi de l'energia interna és zero, durant aquesta expansió isotèrmica. Amb la primera llei de la termodinàmica, es troba la calor absorbida pel gas és:

$$Q_h = W_1 = nRT_h \ln(V_N/V_M) \quad (4.21)$$

- Pas 2 (expansió adiabàtica): El gas s'aïlla tèrmicament i es deixa que s'expandeixi més, realitzant un treball W_2 . Com que aquesta expansió és adiabàtica, la temperatura del gas cau, en aquest cas, de T_h a T_c . Doncs s'obté l'equació 4.22.

$$T_h V_N^{\gamma-1} = T_c V_O^{\gamma-1} \quad (4.22)$$

- Pas 3 (compressió isotèrmica): El gas es posa en contacte tèrmic amb un reservori fred a temperatura T_c i comprimit isotèrmicament. Durant aquest procés, el treball W_3 es fa en el gas i dona calor Q_c al reservori fred. El raonament utilitzat en el pas 1 dona com a resultat l'equació 4.23 on Q_c és la calor abocada al reservori fred pel gas.

$$Q_c = nRT_c \ln(V_O/V_P) \quad (4.23)$$

- Pas 4 (compressió adiabàtica): El gas s'aïlla tèrmicament i torna al seu estat inicial per compressió. En aquest procés, el treball W_4 es fa en el gas. Com que la compressió és adiabàtica, la temperatura del gas augmenta de T_c a T_h . . Doncs s'obté l'equació 4.24.

$$T_c V_{P^{Y-1}} = T_h V_{M^{Y-1}} \quad (4.24)$$

El treball total realitzat pel gas en el cicle de Carnot és el de l'equació 4.25.

$$W = W_1 + W_2 - W_3 - W_4 \quad (4.25)$$

Aquest treball és igual a l'àrea que tanca el bucle que es mostra en el diagrama pV de figura 4.10. Com l'estat inicial i final del sistema són els mateixos, la variació de l'energia interna del gas en el cicle ha de ser zero.

El rendiment d'aquesta màquina de Carnot és el que es representa en la l'equació 4.26.

$$\eta = 1 - T_c/T_h \quad (4.26)$$

Les màquines no necessàriament han de seguir el Cicle de Carnot. No obstant, tots els motors tenen el mateix efecte d'absorbir calor d'un dipòsit calent, produir treball i transferir calor a un dipòsit fred.

El cicle d'un refrigerador de Carnot amb un gas ideal es representa mitjançant el diagrama pV de la figura 4.11. Es tracta d'una màquina de Carnot que funciona a la inversa. El refrigerador extreu la calor Q_c d'un reservori de temperatura freda a T_c quan el gas ideal s'expandeix isotèrmicament. A continuació, el gas es comprimeix adiabàticament fins a una temperatura T_h , a continuació un compressió isotèrmica del gas resulta una calor Q_h que es diposita a un reservori d'alta temperatura a T_h . Finalment, el cicle es completa amb una expansió adiabàtica del gas, fent que la seva temperatura descendeixi fins a T_c .

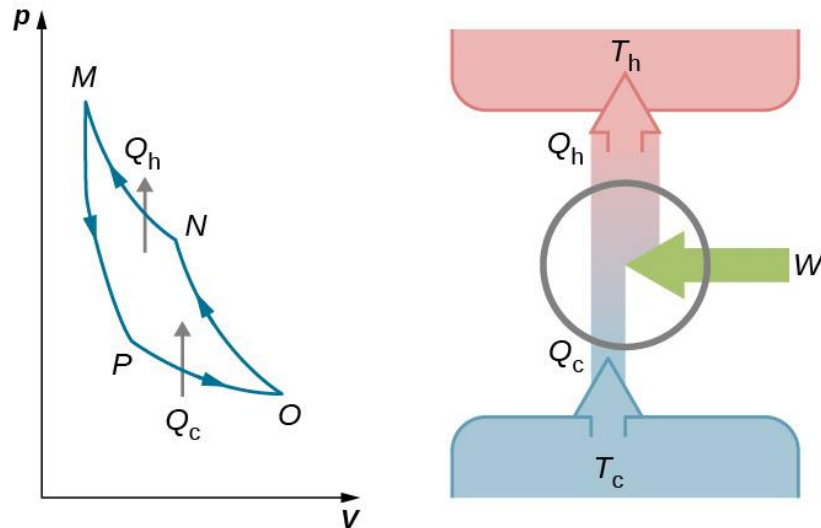


Fig. 4.11 Cicle d'un refrigerador de Carnot [46].

El treball realitzat sobre el gas en el cicle refrigerador de Carnot es mostra i ve donat per l'àrea tancada pel bucle MPONM de la figura 4.11. El treball realitzat en el gas ideal és igual a l'àrea que tanca la trajectòria del diagrama pV. A partir de la primera llei, aquest treball ve donat per l'equació 4.27.

$$W = Q_h - Q_c \quad (4.27)$$

El coeficient de rendiment d'una bomba de calor de Carnot es representa en l'equació 4.28.

$$k_P = T_h / (T_h - T_c) \quad (4.28)$$

Carnot va resumir el seu estudi de la màquina de Carnot i del cicle de Carnot en el que avui es coneix com a principi de Carnot: Cap màquina que treballi entre dos reservoris a temperatura constant pot tenir un rendiment més gran que una màquina reversible.

4.1.3.2. *Cicle Diesel*

El principi de funcionament dels motors Dièsel [47] consisteix en admetre la major quantitat d'aire que sigui possible, comprimir-lo suficientment com per generar una pressió i temperatura tals que en introduir combustible finament polvoritzat, es combuioni ràpidament i producte d'això es generi treball mecànic sostingut en el temps. Per a això és necessari que ocorrin una sèrie de transformacions termodinàmiques del fluid.

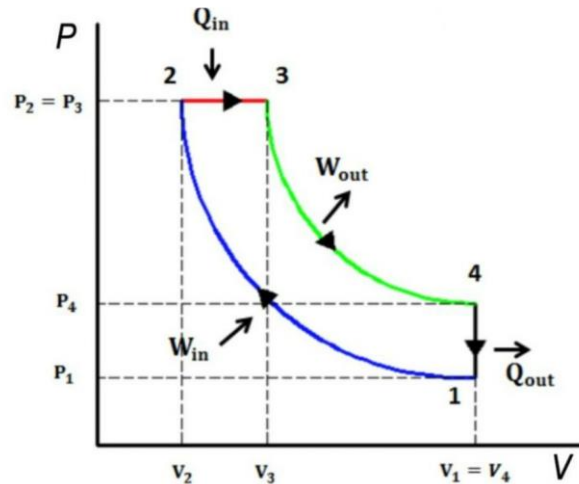


Fig. 4.12 Trajectòria termodinàmica d'un cicle Dièsel [47].

A la figura 4.12 es representa el funcionament ideal d'un motor Dièsel de quatre temps considerant que el fluid és aire pur, que es comporta com un gas perfecte. Un motor de quatre temps necessita donar dues voltes completes del cigonyal per realitzar un cicle complet (quatre curses). Les transformacions termodinàmiques que es donen en aquest cicle són:

- Procés 1-2 (compressió): És un procés de compressió adiabàtica reversible (isentròpica), és a dir sense intercanvi de calor amb l'exterior. Ve a simbolitzar el procés de compressió de la massa fresca en el motor real, en el qual en el pistó, estant en el punt mort inferior (PMI), comença la seva carrera d'ascens, comprimint l'aire contingut en el cilindre. Això eleva l'estat termodinàmic del fluid, augmentant la seva pressió, la seva temperatura i disminuint el seu volum específic, en virtut de l'efecte adiabàtic. En la idealització, el procés ve governat per l'equació isentròpica que es representa a l'equació 4.29. Els canvis en els volums i la seva relació (V_1/V_2) es coneixen com la relació de compressió. La relació de compressió d'un motor dièsel pot oscil·lar entre 12 i 24, mentre que el de gasolina pot rondar un valor de 8.

$$P \cdot v^\gamma = \text{cte} \quad (4.29)$$

- Procés 2-3 (combustió): En aquesta idealització, l'aportació de calor Q_{in} se simplifica per un procés isòbaric. Tanmateix, la combustió Diesel és molt més complexa: a l'entorn del punt mort superior (PMS) s'inicia la injecció del combustible. L'injector polvoritza i atomitza el combustible, que, en contacte amb l'atmosfera interior del cilindre, comença a evaporar-se. Com que el combustible d'un motor Dièsel ha de ser molt autoinflamable, passa que, molt abans que hagi acabat la injecció de tot el combustible, les primeres gotes de combustible injectat s'autoinflamen

i donen començament a una primera combustió caracteritzada per ser molt turbulenta i imperfecta, al no haver tingut la barreja d'aire i combustible temps suficient com per homogeneïtzar-se. Aquesta etapa és molt ràpida, i en el present cicle s'òbvia, però no així en l'anomenat cicle Diésel ràpid, en el qual se simbolitza com una compressió isòcora al final de la compressió. Posteriorment, es dona, sobre la massa que no ha estat cremada, una segona combustió, anomenada combustió per difusió, molt més pausada i perfecta, que és la que aquí se simplifica per un procés isòbar. Tanmateix, la immensa majoria del treball de pressió i de les pèrdues i irreversibilitats del cicle es donen en la combustió inicial, per la qual cosa ometre-la sense més sol conduirà a un model imperfecte del cicle Diésel. Durant aquest procés, l'energia ingressa al sistema a mesura que s'afegeix calor Q_{in} , i una part del treball es realitza movent el pistó. Els canvis en els volums i la seva relació (V_3/V_2) es coneixen com la relació de combustió.

$$Q_{in} = nC_p(T_3 - T_2) \quad (4.30)$$

- Procés 3-4 (explosió/expansió): Se simplifica per una expansió isentròpica (adiabàtica) del fluid termodinàmic, fins al volum específic que es tenia a l'inici de la compressió. En la realitat, l'expansió es produeix a conseqüència de l'elevat estat termodinàmic dels gasos després de la combustió, que empenyen el pistó des del PMS cap al PMI, produint un treball. La relació de volum (V_4/V_3) es coneix com la relació d'expansió isentròpica.
- Procés 4-1 (última etapa): Aquesta etapa és un procés isocor. En aquest procés, l'aire rebutja la calor Q_{out} cap al cos fred a volum constant fins al punt 1 on torna al seu estat original.

$$Q_{out} = nC_v(T_4 - T_1) \quad (4.31)$$

La segona llei de la termodinàmica estableix que cap màquina tèrmica pot convertir tota la calor que rep en treball mecànic. A causa d'això, només una part de l'energia química subministrada pel combustible esdevé treball i aquesta part representa l'eficiència tèrmica del motor. Per tant, es pot definir com la relació entre el treball produït en un cicle ideal i el treball equivalent de la calor consumida per produir-lo, això es mostra a l'equació 4.32.

$$\eta_t = (Q_{in} - Q_{out}) / Q_{in} \quad (4.32)$$

4.1.3.3. *Cicle de Brayton*

És un conjunt de processos termodinàmics que a mesura que ocorren, aconseguen retornar el sistema a la seva etapa inicial per permetre reiniciar els passos novament. En aquest cas el cicle Brayton [48] ideal explica el funcionament del motor d'una turbina de gas. És considerat un dels cicles termodinàmics més importants a causa de la seva àmplia aplicació en diversos camps industrials. Per exemple, la fabricació de turbines per a avions. El seu objectiu és convertir l'energia en forma de calor en treball, per la qual cosa el seu rendiment es mesura per l'eficiència tèrmica.

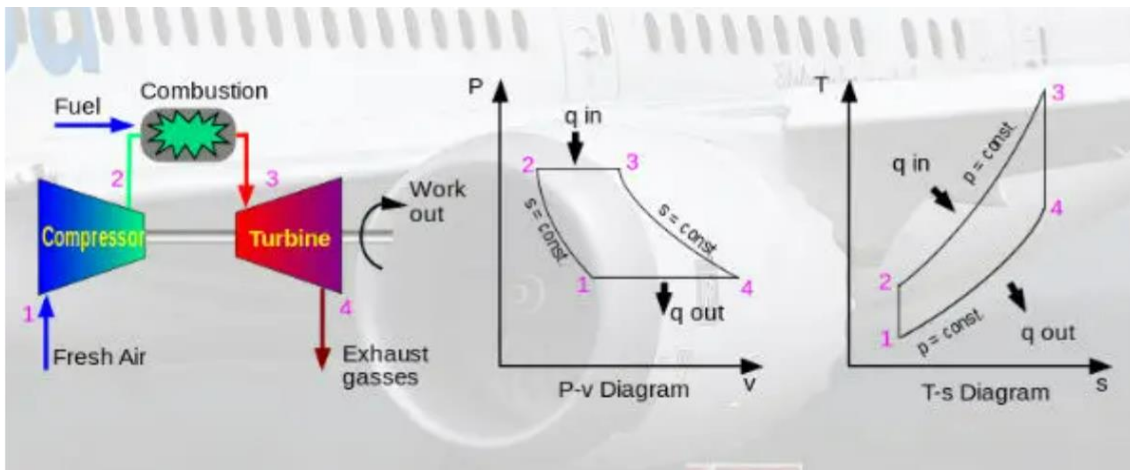


Fig. 4.13 Trajectòria termodinàmica d'un cicle de Brayton [48].

El cicle bàsic Brayton ideal està compost per quatre etapes i quatre elements com es mostra a la figura 4.13:

- Compressor, on succeeix la compressió isentròpica.
- Cambra de combustió, on s'addicionarà calor al fluid a pressió constant.
- Turbina, per realitzar l'expansió isentròpica.
- Intercanviador de calor, per remoure la calor del fluid a pressió constant.

Per entendre el funcionament del cicle es pot explicar els processos de la manera següent:

- Procés 1-2 (compressió isentròpica): El gas de treball es comprimeix isentròpicament des de l'estat 1 a l'estat 2 mitjançant el compressor. Els voltants treballen amb el gas, augmentant la seva energia interna (temperatura) i comprimint-lo (augmentant la seva pressió). D'altra banda, l'entropia roman sense canvis. Al ser un procés adiabàtic, tot

aquest treball s'inverteix a incrementar l'energia interna, elevant la seva temperatura. El treball del procés 1-2 és el de l'equació 4.33.

$$W = nC_v(T_2 - T_1) \quad (4.33)$$

- Procés 2-3 (addició de calor isobàrica): En aquesta fase hi ha una transferència de calor Q_{in} a pressió constant.

$$Q_{in} = nC_p(T_3 - T_2) \quad (4.34)$$

- Procés 3-4 (expansió isentròpica): El gas comprimit i escalfat s'expandeix adiabàticament des de l'estat 3 a l'estat 4 en una turbina. El gas perd una quantitat d'energia interna igual al treball que abandona el sistema, el treball que perd del procés 3-4 és el de l'equació 4.35.

$$W = nC_v(T_4 - T_3) \quad (4.35)$$

- Procés 4-1 (rebuig de calor isobàric): En aquesta fase, el cicle es completa mitjançant un procés de pressió constant en el qual es rebutja la calor del gas Q_{out} . La temperatura del gas de treball cau del punt 4 al punt 1.

$$Q_{out} = nC_p(T_4 - T_1) \quad (4.36)$$

El treball net realitzat sobre el gas és la suma del treball de cada procés, doncs la definició del treball net és l'equació 4.37.

$$W_{net} = nC_p(T_2 - T_1 + T_4 - T_3) \quad (4.37)$$

L'eficiència tèrmica del cicle Brayton es defineix com:

$$\eta_{th} = 1 - Q_{out} / Q_{in} = 1 - T_1 / T_2 \quad (4.38)$$

L'eficiència només depèn de la temperatura a l'inici i al final del procés de compressió, i no de la temperatura després de la combustió, o de la quantitat de calor que introdueix aquesta. Sent T_3 la temperatura màxima que assoleix l'aire, veiem ja que aquest cicle ha de tenir un rendiment menor que un cicle de Carnot que operi entre aquestes temperatures T_1 i T_3 .

Un cicle de Brayton que es condueix en direcció inversa es coneix com el cicle de Brayton invers. El seu propòsit és moure la calor del cos més fred al més calent, en lloc de produir treball. Segons la segona llei de la termodinàmica, la calor no pot fluir espontàniament del sistema fred al sistema calent sense realitzar un treball extern en el sistema. La calor pot fluir del cos més fred al més calent, però només quan és forçat per un treball extern. Això és exactament el que aconseguen els refrigeradors i les bombes de calor. Aquests són impulsats per motors elèctrics que requereixen treball del seu entorn per funcionar. Un dels cicles possibles és un cicle de Brayton invers, que és similar al cicle de Brayton ordinari però es condueix en reversa, a través de l'entrada de treball net. Aquest cicle també es coneix com a cicle de refrigeració de gas o cicle de Bell Coleman. Aquest tipus de cicle es fa servir àmpliament en avions a reacció per a sistemes d'aire condicionat que fan servir aire dels compressors del motor.

4.1.3.4. Cicle Otto

El cicle Otto [49] és un cicle termodinàmic que està conformat per dos processos isocors i dos processos adiabàtics. Aquest cicle es produeix sobre un fluid termodinàmic compressible. El cicle Otto s'aplica a motors de combustió interna que treballen amb una barreja d'aire i un combustible volàtil com gasolina, gas, o alcohol, i la combustió del qual s'inicia amb una espurna elèctrica.

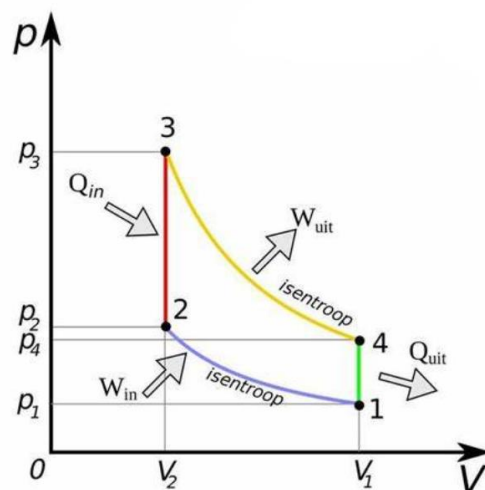


Fig. 4.14 Trajectòria termodinàmica d'un cicle de Otto [49].

Per entendre el funcionament del cicle es pot explicar els processos de la figura 4.14 de la següent manera:

- Procés 1-2 (compressió isoentròpica): El gas (barreja de combustible i aire) es comprimeix adiabàticament de l'estat 1 a l'estat 2, a mesura que el pistó es mou des del punt mort inferior al punt mort superior. L'entorn treballa amb el gas, augmentant la seva energia interna (temperatura) i comprimint-lo. D'altra banda, l'entropia roman sense canvis. Els canvis en els volums i la seva relació (V_1/V_2) es coneixen com la relació de compressió. La relació de compressió determina l'eficiència tèrmica del cicle. En general, es vol tenir una alta relació de compressió, ja que permet que un motor assoleixi una major eficiència tèrmica. En aquest procés es realitza un treball positiu sobre el gas. En ser un procés adiabàtic, tot aquest treball s'inverteix a incrementar l'energia interna, elevant la seva temperatura. El treball del procés 1-2 és el de l'equació 4.39.

$$W = nC_v(T_2 - T_1) \quad (4.39)$$

- Procés 2-3 (addició de calor isomètrica): En aquesta fase hi ha una transferència de calor Q_{in} a volum constant (el pistó està en repòs). Aquest procés està destinat a representar la ignició de la barreja de combustible i aire injectada a la cambra i la posterior combustió ràpida. La pressió augmenta i la relació (P_3/P_2) es coneix com la "relació d'explosió".

$$Q_{in} = nC_v(T_3 - T_2) \quad (4.40)$$

- Procés 3-4 (expansió isoentròpica): El gas s'expandeix adiabàticament des de l'estat 3 a l'estat 4, a mesura que el pistó es mou des del punt mort superior al punt mort inferior. El gas funciona en l'entorn (pistó) i perd una quantitat d'energia interna que és igual al treball que lliure el sistema. Novament, l'entropia roman sense canvis. La relació de volum (V_4/V_3) es coneix com la relació d'expansió isentròpica, però per al cicle d'Otto, és igual a la relació de compressió. En l'expansió és l'aire el que realitza treball sobre el pistó. De nou aquest treball útil equival a la variació de l'energia interna. Aquest treball és negatiu, per ser el sistema el que el realitza. El treball del procés 3-4 és el de l'equació 4.41.

$$W = nC_v(T_3 - T_4) \quad (4.41)$$

- Procés 4-1 (rebuig de calor isomètrica): En aquesta fase, el cicle es completa amb un procés de volum constant en què la calor es rebutja, Q_{out} , de l'aire mentre el pistó està en el punt mort inferior. La pressió del gas de treball cau instantàniament des del punt 4 al punt 1. La vàlvula

d'escapament s'obre en el punt 4. La carrera d'escapament es produeix directament després d'aquesta descompressió.

$$Q_{\text{out}} = nC_v(T_4 - T_1) \quad (4.42)$$

La diferència entre el treball realitzat pel gas i el treball realitzat sobre el gas és el treball net produït pel cicle i correspon a l'àrea tancada per la corba del cicle. El treball útil realitzat pel motor serà el treball net entregat, igual al que produeix (en valor absolut) menys el que empra a funcionar. Doncs la definició del treball net és l'equació 4.43.

$$W_{\text{net}} = nC_v(T_3 - T_4 - T_2 + T_1) \quad (4.43)$$

En tractar-se d'un procés cíclic, la variació de l'energia interna és nul·la en finalitzar el cicle. Això implica que la calor neta introduïda en el sistema ha de ser igual al treball net realitzat per aquest, en valor absolut. El treball produït pel cicle multiplicat per la velocitat del cicle (cicles per segon) és igual a la potència produïda pel motor Otto.

L'eficiència tèrmica del cicle de Otto es defineix com:

$$\eta_{\text{th}} = 1 - Q_{\text{out}} / Q_{\text{in}} = 1 - T_1 / T_2 \quad (4.44)$$