



# EXOCOSMØS

MEMÒRIA DEL PROJECTE

Joan Sabata Moreno  
23/05/2025  
Projecte de Síntesi

Exocosmos és una aplicació web educativa i creativa que permet a qualsevol usuari crear, gestionar i visualitzar sistemes planetaris ficticis en 3D, combinant dades científiques reals amb una experiència visual immersiva. Està pensada tant per a l'aprenentatge com per a la creació lliure dins d'un entorn segur i personalitzable.

Es compon d'un **backend** creat amb **Express** i **MySQL**, que actua com a API REST per gestionar dades i autenticació; i un **frontend** desenvolupat amb **React**, **Vite** i **React Three Fiber**, encarregat de renderitzar escenes tridimensionals i oferir una interfície d'usuari moderna i accessible.

Els **objectius principals** del projecte eren:

- Aplicar tecnologies modernes del desenvolupament web tant al client com al servidor.
- Integrar visualització 3D mitjançant Three.js i React Three Fiber.
- Crear un sistema complet amb autenticació, perfil d'usuari i gestió de contingut.
- Oferir una experiència educativa personalitzable, basada en dades i visualment atractiva.

Els **resultats clau aconseguits** inclouen:

- Desenvolupament complet i funcional del backend i frontend.
- Visualització en temps real de planetes, estrelles i sistemes en escenes 3D interactives.
- Gestió segura d'usuaris amb edició de perfil i pujada d'imatges.
- Creació de sistemes planetaris amb composició química, colors i textures configurables.  
Desplegament final a Fly.io (backend) i Vercel (frontend).
- Superació de múltiples reptes tècnics com tests automatitzats, rendiment gràfic o validació de dades.

**Exocosmos** es presenta com un projecte complet, sòlid i amb potencial de futur, capaç de créixer en funcionalitats, impacte educatiu i interacció comunitària.

# Índex

<b>Índex</b>	<b>2</b>
<b>Introducció</b>	<b>3</b>
Objectius personals i d'aprenentatge	3
Objectius específics del projecte	4
<b>Planificació</b>	<b>5</b>
Pla de treball general	5
Fases del projecte	5
Cronograma original	6
Cronograma amb les fites finals	6
Desviacions de la planificació	6
Recursos necessaris	7
Recursos formatius	7
Eines d'intel·ligència artificial	7
Plataformes de desplegament	7
Altres recursos	7
<b>Desenvolupament del Projecte</b>	<b>8</b>
Descripció de la solució implementada	8
Tecnologies utilitzades	8
Base de dades	8
Backend	8
Frontend	8
Desplegament	9
Altres	9
Arquitectura de la solució	9
Backend	9
Frontend	9
Funcionalitats principals	10
Decisions tècniques rellevants	10
<b>Avaluació i Resultats</b>	<b>12</b>
Problemes trobats i solucions aplicades	12
Lliçons apreses	13
<b>Conclusions</b>	<b>14</b>
Resum dels resultats	14
Valoració personal	14
Propostes de millora	14
Línies futures de desenvolupament	15
<b>Referències i Bibliografia</b>	<b>16</b>

# Introducció

Fa uns mesos, vaig descobrir la llibreria Three.js, que permet crear gràfics 3D al navegador. Em va cridar molt l'atenció i vaig començar a explorar-la pel meu compte, com a aprenentatge personal. Quan vaig haver de triar el tema pel meu projecte final, vaig tenir clar que volia utilitzar aquesta eina d'alguna manera, ja que em semblava interessant, potent i diferent del que havia fet fins ara.

Després de pensar en diverses opcions, vaig acabar decidint-me per crear una aplicació educativa centrada en la creació de planetes i sistemes planetaris, ja que els objectes astronòmics tenen formes geomètriques senzilles, i això em permetia experimentar amb gràfics 3D sense haver d'importar models externs.

A partir d'aquesta idea, vaig anar construint el projecte Exocosmos, intentant combinar el coneixement tècnic adquirit durant el cicle amb l'experimentació amb Three.js, per fer una aplicació web que fos tant visual com funcional.

A l'hora de plantejar aquest projecte, em vaig marcar una sèrie d'objectius tant **personals com tècnics**, amb la idea de fer un treball que m'ajudés a créixer com a desenvolupador i que pogués servir com a mostra per al meu futur portfoli.

## Objectius personals i d'aprenentatge

- **Fer servir tecnologies modernes** del desenvolupament web, tant al frontend com al backend.
- **Aprendre React**, una llibreria molt utilitzada al món professional i compatible amb Three.js.
- **Treballar amb Three.js** i aplicar el que havia après prèviament, especialment en la creació d'escenes 3D.
- **Tenir un resultat visual i interactiu.**
- Entendre millor com integrar la part gràfica 3D dins d'un entorn real, connectat a dades i a un backend.

## Objectius específics del projecte

- Desenvolupar una **API REST amb Express i MySQL**, completament separada del frontend, que gestionés la informació dels usuaris i dels cossos còsmics (planetes, estrelles, sistemes...).
- Crear un **frontend modern amb React**, amb bones pràctiques i una arquitectura i disseny clar.
- Implementar la visualització 3D mitjançant **React Three Fiber (R3F)** per mostrar planetes i sistemes de forma interactiva.
- Afegir funcionalitats com **autenticació, edició de perfil, creació i edició de sistemes planetaris**, etc.
- Oferir una **experiència educativa** i divertida dirigida a persones de totes les edats interessades en l'astronomia.

El desenvolupament del projecte no ha seguit una metodologia estricta ni formal. Al principi, em vaig centrar en explorar les possibilitats de Three.js per veure com es podria simular un planeta, com personalitzar-lo i quins paràmetres serien necessaris per definir-lo. Aquesta primera fase va ser essencial per entendre els límits de la tecnologia i decidir l'enfocament del projecte.

Un cop tenia més clar què volia fer, vaig passar a definir l'estructura de dades i les taules de la base de dades. Amb això fet, vaig desenvolupar el backend (amb Express i MySQL) i després el frontend (amb React + Vite), sempre seguint un ordre lineal. És a dir, no vaig treballar les parts en paral·lel, sinó una després de l'altra, fent les proves necessàries en cada pas abans de continuar.

Tot i haver fet una planificació inicial, la realitat del projecte va fer que la descartés bastant aviat. Van sorgir problemes i errors que no havia previst i que em van fer perdre molt de temps. Altres coses, que pensava que serien complicades, van resultar molt més fàcils gràcies a un ús de la intel·ligència artificial, que m'ha ajudat molt a avançar.

Per mantenir un bon control sobre aquest ús de la IA i garantir que entenia realment el que estava fent, he treballat sempre amb Git. Això m'ha permès veure l'evolució del projecte pas a pas, revisar els canvis amb calma i assegurar-me que no depenia cegament de res que no pogués justificar o comprendre.

En resum, ha estat una metodologia improvisada, però efectiva dins les meves possibilitats, adaptant-me sobre la marxa i resolent els entrebancs a mesura que anaven sorgint.

# Planificació

## Pla de treball general

El desenvolupament del projecte va seguir un ordre seqüencial, començant per la fase d'experimentació i disseny, seguida per la implementació del backend, i finalitzant amb el frontend i el desplegament. Tot i haver elaborat un cronograma inicial, aquest es va veure alterat durant el procés per canvis de prioritats, dificultats imprevistes i reajustaments tècnics.

## Fases del projecte

### 1. Exploració amb Three.js

Abans de definir l'abast concret del projecte, es van fer proves amb la llibreria Three.js per avaluar-ne les possibilitats i limitacions, especialment pel que fa a la representació de planetes i estrelles sense necessitat de models externs.

### 2. Disseny i implementació de la base de dades

Un cop clarificada la idea, es va procedir a dissenyar l'estructura relacional necessària per gestionar sistemes planetaris, planetes, estrelles i usuaris. El diagrama de base de dades està datat del 22 de maig de 2025.

### 3. Desenvolupament del backend (API REST amb Express)

El backend es va iniciar a l'abril (primer commit: 8 d'abril de 2025) i va resultar ser una de les parts més exigents. La seva complexitat, sumada a errors inesperats, va generar retards significatius. Al mateix temps, es va aprofitar per començar a aprendre React i R3F.

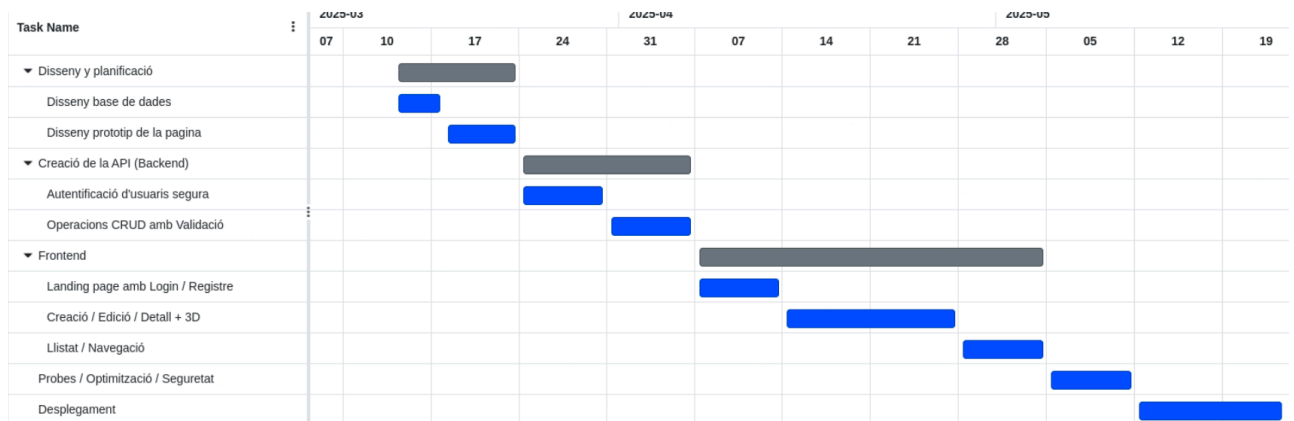
### 4. Desenvolupament del frontend amb React + Vite

El frontend es va iniciar a principis de maig (primer commit: 6 de maig de 2025). En una primera setmana es va completar l'editor d'estrelles, i en la segona es va implementar el formulari de creació de planetes, una de les parts més complexes. Finalment, es va dedicar una tercera setmana a l'estilització de tota la interfície utilitzant l'eina v0.

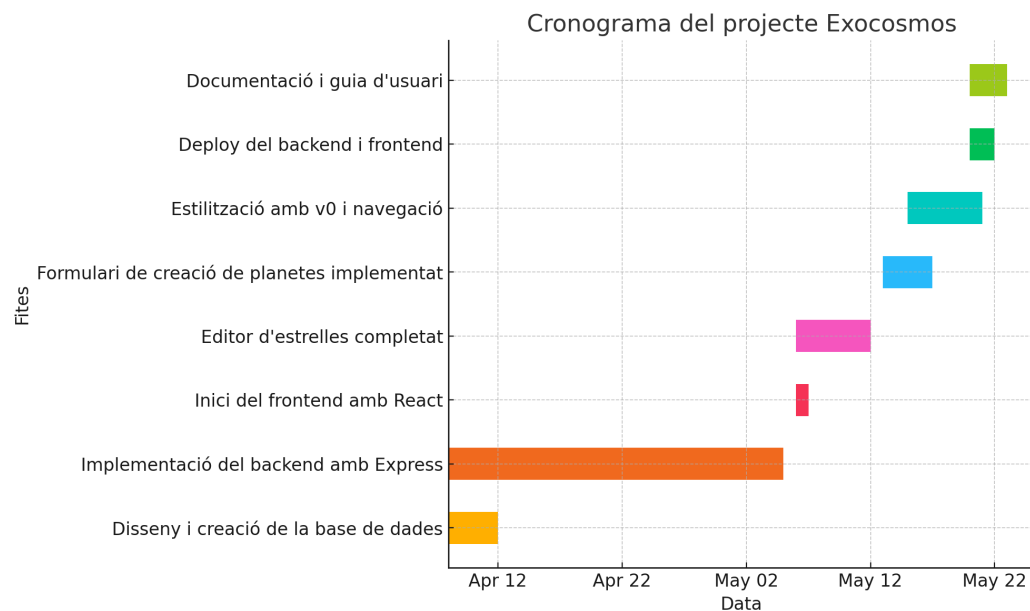
### 5. Desplegament i documentació

Un cop completat el desenvolupament, es va realitzar el desplegament del backend a Fly.io i del frontend a Vercel. Paral·lelament, es va redactar tota la documentació del projecte i la guia d'usuari.

## Cronograma original



## Cronograma amb les fites finals



## Desviacions de la planificació

La planificació inicial no es va poder seguir degut a una subestimació del temps necessari per desenvolupar la API i gestionar les seves complexitats. El backend va resultar ser molt més exigent del que es preveia inicialment, amb errors inesperats i dificultats sobretot en la fase de validació i test. Cal destacar que era la primera vegada que es feia tests automatitzats d'una API i que també era el primer projecte amb React, cosa que va afegir dificultats i temps d'aprenentatge extra. Tot i això, el projecte es va completar amb èxit, incloent-hi el desplegament i una presentació de la pàgina satisfactòria.

## Recursos necessaris

Durant el desenvolupament del projecte s'han utilitzat diversos recursos formatius, tècnics i tecnològics:

### Recursos formatius

- **Three.js Journey:** curs en línia especialitzat en Three.js, realitzat durant el període nadalenc. Va ser essencial per entendre la lògica de les escenes 3D, càmeres, il·luminació, materials i interacció.
- **Fullstack Open (Universitat d'Hèlsinki):** curs gratuït i molt complet per aprendre React i altres eines modernes del desenvolupament web. Va servir de base per estructurar el frontend del projecte.

### Eines d'intel·ligència artificial

- **ChatGPT:** utilitzat com a assistent de desenvolupament, especialment per resoldre dubtes, entendre errors i generar fragments de codi o millorar-los.
- **Gemini:** provat breument per comparar capacitats amb altres assistents.
- **v0 (de Vercel):** eina emprada per generar components visuals estilitzats amb Tailwind CSS amb resultats ràpids i personalitzables que han agilitzat l'estilització del frontend.

### Plataformes de desplegament

- **Fly.io:** utilitzat per desplegar l'API REST amb Express i la base de dades MySQL.
- **Vercel:** emprat per al desplegament del frontend amb React i Vite, aprofitant la seva integració amb Git i desplegament automàtic.

### Altres recursos

- **Textura i contingut visual:** s'han utilitzat diverses plataformes per obtenir textures de planetes, efectes visuals, shaders i inspiració per a la interfície gràfica. Tots aquests recursos es detallaran a l'apartat final de **Referències i Bibliografia**.



# Desenvolupament del Projecte

## Descripció de la solució implementada

Exocosmos és una aplicació web educativa que permet a usuaris de totes les edats crear, visualitzar i gestionar sistemes planetaris de manera interactiva. El projecte es compon de dues parts clarament diferenciades: el backend, una API REST desenvolupada amb Express i una base de dades MySQL; i el frontend, una SPA creada amb React i Vite, amb capacitats de visualització 3D gràcies a Three.js mitjançant la llibreria React Three Fiber (R3F).

L'aplicació ofereix funcionalitats com ara el registre d'usuari, creació de sistemes i cossos celestes, personalització visual, exploració de contingut creat per altres usuaris, i una guia d'ús integrada.

## Tecnologies utilitzades

### Base de dades

- MySQL com a sistema gestor de base de dades.
- MySQL Workbench per al disseny i consultes durant el desenvolupament.

### Backend

- Express amb TypeScript per a la creació de l'API REST.
- Zod per a la validació de dades.
- JWT per a l'autenticació mitjançant cookies HttpOnly.
- Multer per a la gestió de pujades d'imatges.
- Vitest i Supertest per a les proves automàtiques.
- PubChem API per a la consulta i validació de compostos químics externs.

### Frontend

- React i Vite com a entorn de desenvolupament.
- Tailwind i v0 per a l'estilització moderna.
- React Hook Form i Zod per a la gestió i validació de formularis.
- React Three Fiber i Drei per a la visualització 3D basada en Three.js.

## Desplegament

- Fly.io per al backend.
- Vercel per al frontend.

## Altres

- Git i GitHub com a sistema de control de versions.

## Arquitectura de la solució

### Backend

El backend implementa una arquitectura basada en el patró Model-Vista-Controlador (MVC):

- **Models:** Gestionen la comunicació directa amb la base de dades. Per exemple, UserModel, PlanetModel, etc.
- **Controladors:** Gestionen la lògica de les peticions, cridant als models, validant dades amb Zod i retornant la resposta adequada.
- **Rutes:** Defineixen els endpoints i deleguen la gestió als controladors. Estan agrupades per recursos: /auth, /stars, /planets, etc.
- **Middlewares:** S'utilitzen per a la validació, gestió d'errors, autenticació, i pujades d'imatges.

### Frontend

L'aplicació de client és una SPA amb React. Utilitza React Router per a la navegació entre pàgines i un context global (AuthContext) per a gestionar l'estat de l'usuari. La visualització 3D s'integra mitjançant React Three Fiber, amb escenes que permeten rotació, zoom i interacció amb cossos celestes.

El frontend es manté modular i organitzat, amb components reutilitzables per a targetes, formularis, seccions i vistes detallades. També incorpora animacions i una interfície cuidada, especialment en la guia i en els formularis avançats de creació.

## Funcionalitats principals

El sistema d'**autenticació** permet el **registre** i **inici de sessió** d'usuaris, amb la gestió segura de la sessió mitjançant **cookies HttpOnly**, garantint una major protecció de la informació personal. Un cop autenticat, l'usuari pot accedir al seu **perfil** i modificar-ne les dades bàsiques com el **nom**, el **correu electrònic** i la **contrasenya**, així com pujar una **imatge de perfil** personalitzada.

A nivell de creació, l'usuari pot generar **sistemes planetaris** definint un nom i una descripció, i dins d'aquests, afegir-hi **estrelles** amb propietats com el **tipus espectral**, el **radi**, la **temperatura** i el **color**. També és possible crear **planetes** amb paràmetres físics com la **massa**, el **radi** i l'**òrbita**, a més de configuracions visuals com el **color**, la **textura** i el **tipus d'atmosfera**, donant lloc a escenaris diversos i personalitzats.

La **visualització** d'aquests sistemes es realitza en **3D** mitjançant una **càmera orbital** que permet explorar les escenes de manera interactiva, amb **textures realistes**, **llums dinàmiques** i **controls suaus**, tot integrat amb **React Three Fiber**. A més, els usuaris poden accedir a la secció d'**exploració**, on es mostren sistemes i cossos celestes creats per altres persones, afavorint així el **descobriment** i l'**aprenentatge compartit**.

També es poden assignar **compostos químics** a l'atmosfera o superfície dels planetes. Aquests compostos es poden seleccionar lliurement i es validen automàticament amb la base de dades de **PubChem**, assegurant-ne l'existència real i proporcionant un context científic. Els compostos es desen en una memòria cau pròpia per agilitzar consultes i reutilitzar informació de manera eficient.

Finalment, l'aplicació incorpora una **guia d'usuari** estructurada i accessible des de qualsevol punt del frontend, amb contingut explicatiu per a cada secció i funcionalitat, pensada per orientar l'usuari al llarg del seu recorregut dins d'**Exocosmos**.

## Decisions tècniques rellevants

Entre les **decisions tècniques** més destacades del projecte, cal mencionar l'ús de **MySQL** com a sistema gestor de base de dades. Aquesta elecció es va fonamentar en la seva naturalesa **relacional**, que permet estructurar la informació de manera clara i coherent mitjançant **taules** i **claus foranes**. Això facilita tant la **consulta** com la **integració directa amb Express** mitjançant llibreries compatibles, assegurant una connexió eficient i estable amb la base de dades.

Pel que fa a la **seguretat**, es va optar per implementar un sistema d'autenticació basat en **cookies HttpOnly**, una pràctica que evita que el token d'autenticació sigui accessible des del JavaScript del navegador. D'aquesta manera, es redueix significativament el risc d'atacs de tipus **XSS** (Cross-Site Scripting), i es garanteix una millor protecció de les sessions dels usuaris. Aquesta elecció es va considerar adequada per al tipus de projecte, ja que proporciona un bon equilibri entre **seguretat** i **simplicitat**.

Un altre aspecte tècnic clau és la **validació de dades** amb **Zod**, tant al backend com al frontend. Aquesta biblioteca permet definir **esquemes de validació reutilitzables**, la qual cosa assegura que les dades compleixin els requisits esperats a ambdós costats de l'aplicació. Gràcies a aquesta **validació compartida**, s'aconsegueix una major coherència i es redueixen els errors, ja que les mateixes regles s'apliquen a l'hora d'enviar o rebre informació.

Pel que fa al **desplegament**, es va escollir **Fly.io** per al backend i **Vercel** per al frontend. Aquestes plataformes ofereixen una gran simplicitat a l'hora de desplegar projectes permetent **automatitzar el desplegament** cada vegada que es vulgui fer una actualització. A més, estan especialment pensades per a projectes petits o de mida mitjana, com és el cas d'aquest treball, i ofereixen un rendiment estable i una experiència de desplegament àgil.

Per optimitzar l'ús d'imatges dins l'aplicació, especialment les que es **guarden al backend** com a **textures personalitzades dels planetes**, es va optar per utilitzar el format **WebP**. Aquest format permet **mantenir la qualitat visual** amb una **compressió superior** respecte a JPEG o PNG, reduint significativament el pes dels arxius. A més, **admet transparència**, una característica necessària per representar correctament la **textura de l'atmosfera** dels planetes en les visualitzacions 3D. Gràcies a aquesta elecció, s'ha aconseguit millorar tant el **rendiment de càrrega** com la **fluïdesa en la visualització** interactiva de planetes dins l'escena tridimensional.

Finalment, una decisió important va ser la creació de **rutes especials** anomenades **/full** per a determinats recursos. Aquestes rutes permeten obtenir **tota la informació d'un recurs**, incloent-hi les seves relacions (per exemple, un sistema amb totes les seves estrelles i planetes), en una única petició. Tot i que aquesta solució s'allunya del patró **REST pur**, va ser una decisió intencionada per millorar l'eficiència del frontend, reduint la **complexitat** i el **nombre de peticions** necessàries per mostrar informació completa a l'usuari. Aquesta flexibilitat en el disseny de l'API ha permès **simplificar el flux de dades** i millorar el **rendiment** de manera notable.

# Avaluació i Resultats

Malgrat diversos imprevistos i dificultats tècniques al llarg del desenvolupament, es pot concloure que el projecte ha assolit els objectius marcats amb un grau alt de satisfacció. L'aplicació final, Exocosmos, compleix amb les funcionalitats bàsiques i avançades previstes, i ofereix una experiència interactiva, educativa i atractiva tant a nivell visual com funcional.

Els objectius generals establerts al principi del projecte eren, entre d'altres:

- Utilitzar tecnologies modernes de desenvolupament web.
- Aplicar els coneixements adquirits de Three.js en un projecte real.
- Aprendre i posar en pràctica React i la seva integració amb entorns 3D com React Three Fiber.

Tots aquests objectius s'han complert satisfactòriament. El projecte ha estat una oportunitat real d'aprenentatge intensiu, especialment pel que fa a React, i el resultat aconseguit reflecteix aquest esforç. L'aspecte final de l'aplicació és modern i funcional, i el codi mostra una separació clara de responsabilitats i bones pràctiques.

## Problemes trobats i solucions aplicades

- **Tests paral·lels no aïllats (Vitest):**  
Els tests automatitzats escrits amb Vitest generaven errors intermitents i difícils de reproduir, especialment quan s'executaven en paral·lel.  
**Solució:** Es va garantir que cada test fos totalment independent, inicialitzant una base de dades buida i aïllada per a cada execució. Això va millorar la fiabilitat i la claredat dels resultats.
- **Error difícil de depurar en tests:**  
Quan un test fallava, la informació proporcionada per Vitest no era suficient per identificar l'error amb precisió.  
**Solució:** Es va implementar una funció *logIfFailed* que extreu informació addicional de la resposta HTTP i l'imprimeix només si el test falla, facilitant enormement la depuració.
- **Creació de planetes amb compostos inexistents:**  
El formulari permetia seleccionar compostos que no existien a la base de dades, causant errors posteriors.  
**Solució:** Es va afegir una verificació per comprovar si cada compost estava registrat. En cas contrari, es realitza una consulta automàtica a PubChem per obtenir les dades; si no es troben, es mostra un error informatiu.

- **Bloqueig del banner de cookies en navegadors com Brave:**  
El component *CookiesBanner* era detectat com un avís de cookies clàssic i bloquejat per alguns navegadors.  
**Solució:** Es va canviar el nom del component a *ConsentBanner* i es van modificar algunes propietats visuals per evitar ser identificat com a element de seguiment.
- **Accés indegut a recursos d'altres usuaris:**  
Qualsevol usuari podia intentar modificar recursos que no li pertanyien.  
**Solució:** Es va implementar un middleware que verifica que l'usuari autenticat és el propietari del recurs abans de permetre qualsevol acció de modificació o eliminació.
- **Errors en el component 3D de planeta en diferents contextos:**  
El component de visualització 3D causava errors quan es col·locava en diferents pàgines o formularis.  
**Solució:** Es van crear versions simplifiades del component per a les miniatures i per a la pàgina principal, millorant així també el rendiment general.
- **Necessitat de comunicació entre formularis i components 3D:**  
El formulari de creació de planetes/estrelles necessitava interaccionar amb els components 3D, que es trobaven encapsulats.  
**Solució:** Es va utilitzar *FormProvider* de react-hook-form per encapsular la pàgina i permetre que diferents components accedissin i reaccionessin als canvis del formulari.
- **Rendiment pobre en dispositius amb GPU limitada:**  
En equips amb gràfics menys potents, les escenes 3D es carregaven lentes i amb poca fluïdesa.  
**Solució:** Es va afegir un selector de configuració de rendiment que permet ajustar paràmetres com la resolució i el postprocessat.

## Lliçons apreses

Un dels aprenentatges més importants d'aquest projecte ha estat desenvolupar la capacitat de **resoldre problemes de manera metòdica i eficaç**. Davant de qualsevol error, he après a no improvisar ni aplicar solucions ràpides sense entendre la causa real. El primer pas ha estat **obtenir la màxima informació possible sobre l'origen del problema**, ja sigui llegint missatges d'error, afegint logs, comprovant el comportament pas a pas o buscant documentació oficial i casos similars.

Un cop entesa millor la causa, he anat tirant del fil de manera sistemàtica, provant solucions parcials, verificant hipòtesis i adaptant-les fins a arribar a una solució definitiva. Aquesta actitud analítica i pacient m'ha permès resoldre errors complexos, especialment relacionats amb els tests automatitzats, la gestió de formularis o la integració entre components del frontend i la visualització 3D. En resum, he après que resoldre problemes no és només una qüestió tècnica, sinó també de mètode, paciència i capacitat d'anàlisi.

# Conclusions

## Resum dels resultats

El projecte **Exocosmos** ha assolit els objectius plantejats, tant a nivell funcional com tecnològic. S'ha desenvolupat una aplicació web educativa que permet a usuaris de totes les edats crear i visualitzar sistemes planetaris en 3D, combinant una arquitectura robusta al backend amb una interfície moderna i interactiva al frontend. El sistema d'autenticació, la gestió de recursos, la integració de visualitzacions 3D i la possibilitat de compartir continguts són funcionalitats implementades amb èxit.

A més, el projecte ha servit com a laboratori pràctic per posar en marxa tecnologies modernes com React, Vite, Express, Zod, React Three Fiber i Tailwind. L'ús d'intel·ligència artificial com a assistent de desenvolupament, així com plataformes de desplegament com Fly.io i Vercel, ha contribuït a donar forma a un projecte complet i real.

## Valoració personal

Aquest projecte ha estat una **experiència d'aprenentatge intensa i enriquidora**. He après a resoldre problemes tècnics complexos, a gestionar el codi de forma neta i estructurada, i a prendre decisions fonamentades, tant a nivell de disseny com de seguretat o rendiment.

També ha estat un repte en l'àmbit de l'autonomia i la constància: he treballat de forma independent, organitzant les meves pròpies tasques i horaris, i adaptant-me constantment als imprevistos. Em sento especialment satisfet del resultat final, tant pel nivell tècnic assolit com per la presentació visual i la coherència del conjunt. És un projecte que reflecteix tant allò que he après com la meua manera de treballar, i que puc incorporar amb orgull al meu portfoli.

## Propostes de millora

- **Sistema d'administració amb rols d'usuari:** Una millora important seria incorporar rols diferenciats (usuari estàndard i administrador). Aquests rols podrien controlar l'accés a determinades funcionalitats mitjançant un panell d'administració. Aquesta funcionalitat es podria implementar reutilitzant els mateixos endpoints, simplement adaptant el middleware que comprova la propietat dels recursos.
- **Millora de l'accessibilitat:** Caldria revisar l'aplicació amb criteris d'accessibilitat web (WCAG), assegurant un contrast adequat, etiquetes accessibles per a lectors de pantalla, i una navegació funcional amb teclat.
- **Opció de canvi d'idioma:** Afegir la possibilitat de traduir l'aplicació a diferents idiomes (inicialment català, castellà i anglès) milloraria l'abast educatiu i facilitaria l'accés a una audiència més àmplia.

- **Tests al frontend:** Tot i que el backend ja incorpora tests automatitzats, el frontend encara no ha estat testejat. Incloure almenys proves bàsiques per a funcionalitats crítiques com el login i el registre aportaria més confiança i mantenibilitat al projecte.

## Línies futures de desenvolupament

- **Lunes orbitant planetes:** Es podrien afegir llunes com a entitats 3D personalitzables, visibles orbitant al voltant dels planetes, amb configuració de paràmetres com el radi, la velocitat i la textura.
- **Vista completa del sistema planetari en 3D:** Crear una escena que representi tot el sistema, amb la seva estrella central i els planetes orbitant, permetria visualitzar el conjunt d'una manera molt més immersiva.
- **Exploració 3D d'estrelles:** Afegir un mapa tridimensional amb totes les estrelles creades pels usuaris permetria explorar l'univers d'Exocosmos de forma visual i interactiva, clicant una estrella per accedir al seu sistema.
- **Interacció entre usuaris:** Exocosmos podria evolucionar cap a una petita comunitat educativa, afegint funcionalitats com likes i comentaris. Això fomentaria la participació i l'intercanvi de coneixement.
- **Realitat augmentada (AR):** Gràcies al suport de Three.js per a WebAR, seria relativament senzill implementar funcionalitats de realitat augmentada per dispositius mòbils, oferint una experiència encara més immersiva.
- **Aplicació nativa per a mòbils:** Una altra línia interessant seria desenvolupar una app nativa per Android i iOS, utilitzant frameworks com React Native o altres tecnologies híbrides. Donat que el backend ja és independent, aquesta transició seria viable sense grans modificacions.



# Referències i Bibliografia

Fonts científiques i tècniques:

- **PubChem** – Base de dades de compostos químics utilitzada per validar la composició dels planetes:  
<https://pubchem.ncbi.nlm.nih.gov/>
- **NASA Eyes on Exoplanets** – Aplicació educativa de la NASA que utilitza **Three.js** per visualitzar exoplanetes i sistemes reals en 3D. Va servir com a font d'inspiració per a l'enfocament visual i l'ús de tecnologies web avançades:  
<https://eyes.nasa.gov/apps/exo/>

Textures i gràfics:

- **Solar System Scope** – Textures de planetes (terra, gasosos, lunars...):  
<https://www.solarsystemscope.com/textures/>
- **Fons espacial animat** – Animació en 2D utilitzada com a fons de la pàgina per simular un cel estrellat amb moviment:  
<https://codepen.io/NazarAzhar/pen/zqXMqP>
- **Shader per a l'estrella** – Fragment shader animat per representar estrelles dinàmiques:  
<https://codepen.io/dirkk0/pen/MWxZLeB>

Altres recursos:

- **Three.js Journey** – Curs especialitzat Three.js i R3F per a navegadors.
- **Fullstack Open (Universitat d'Hèlsinki)** – Curs complet per a desenvolupament web modern amb React.