

## Viatges UOC

**Sergi Planes Ortiz**

Grau d'Enginyeria Informàtica  
Enginyeria del programari

**Josep Maria Camps Riba**

**Santi Caballe Llobet**

26/06/2023



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

## FITXA DEL TREBALL FINAL

<b>Títol del treball:</b>	Anàlisi i disseny d'una App de viatges - Viatges UOC-
<b>Nom de l'autor:</b>	Sergi Planes Ortiz
<b>Nom del consultor/a:</b>	Josep Maria Camps Riba
<b>Nom del PRA:</b>	Santi Caballe Llobet
<b>Data de lliurament (mm/aaaa):</b>	06/2023
<b>Titulació o programa:</b>	Grau d'Enginyeria Informàtica
<b>Àrea del Treball Final:</b>	Enginyeria del programari
<b>Idioma del treball:</b>	Català
<b>Paraules clau</b>	Requeriments, anàlisi, disseny
<b>Resum del Treball (màxim 250 paraules):</b> <i>Amb la finalitat, context d'aplicació, metodologia, resultats i conclusions del treball</i>	
<p>Aquest projecte centra la seva atenció en el disseny d'una aplicació web similar a Booking, el seu objectiu primari és solidificar els coneixements adquirits durant el grau i proporcionar una visió completa de totes les fases que componen la creació d'un projecte enginyeria del programari, a més de generar el disseny funcional d'una aplicació per facilitar la cerca i reserva de vols i allotjaments arreu del món, amb l'esperança de millorar l'experiència d'usuari.</p> <p>El context d'aplicació sorgeix a causa de l'increment constant del moviment global, alliberar la càrrega a l'hora de cercar, pot facilitar una millor experiència d'usuari i incentivar l'ús.</p> <p>El projecte combina tant les metodologies de cascada i àgil, concentrant-se principalment en l'anàlisi i el disseny de l'aplicació, sense abordar la implementació, el projecte es divideix en tres parts: planificació del projecte, anàlisi dels requisits i finalment, el disseny de l'aplicació, aquest últim inclou tant l'arquitectura del sistema com el disseny de la base de dades.</p> <p>El resultat d'aquest treball ha estat, el disseny completament funcional d'una aplicació web i l'adquisició d'una visió global de totes les fases que conformen un projecte d'enginyeria.</p> <p>De cara a les conclusions, es manifesta l'èxit en l'aplicació pràctica dels coneixements obtinguts durant el grau en un context real.</p> <p>Tot i que s'han assolit tots els objectius inicials, s'han identificat també possibles àrees de millora, com implementar alguna part de l'aplicació.</p>	

Finalment, aquest projecte ha donat lloc a la creació d'una nova aplicació, ViatgesUOC, que està preparada per a ser implementada.

**Abstract (in English, 250 words or less):**

This project focuses its attention on the design of a web application similar to Booking, its primary objective is to solidify the knowledge acquired during the degree and provide a complete view of all the phases that make up the creation of a software engineering project, in addition to generating the functional design of an application to facilitate the search and booking of flights and accommodations around the world, with the hope of improving the user experience.

The application context arises due to the constant increase of global movement, relieving the burden when searching, can facilitate a better user experience and encourage use.

The project combines both waterfall and agile methodologies, focusing mainly on the analysis and design of the application, without addressing the implementation, the project is divided into three parts: project planning, requirements analysis and finally, the application design, the latter includes both the system architecture and the database design.

The result of this work has been, the fully functional design of a web application and the acquisition of a global vision of all the phases that make up an engineering project.

Looking at the conclusions, the success in the practical application of the knowledge obtained during the degree in a real context is manifested.

Although all the initial objectives have been achieved, areas for improvement have also been identified, such as implementing some part of the application.

Finally, this project has given rise to the creation of a new application, ViatgesUOC, which is ready to be implemented.

## Índex

1. Introducció.....	3
1.1 Context i justificació del Treball.....	3
1.2 Objectius del Treball.....	3
1.3 Enfocament i mètode seguit.....	4
1.4 Planificació del Treball.....	4
1.5 Breu sumari de productes obtinguts.....	4
1.6 Breu descripció dels altres capítols de la memòria.....	5
2. Pla de treball.....	7
2.1 Requisits de l'aplicació.....	7
2.2 Planificació detallada.....	8
2.3 Eines a utilitzar per a l'anàlisi i el disseny.....	10
3. Requisits i anàlisi.....	10
3.1 Definició d'objectius.....	10
3.2 Anàlisi de requisits.....	11
3.2.1 Requisits funcionals.....	12
3.2.2 Requisits no funcionals.....	12
3.3 Històries d'usuari.....	13
3.4 Diagrama de casos d'ús.....	14
3.5 Casos d'ús.....	15
3.5.1 Dependències entre casos d'ús.....	26
3.6 Model d'actors.....	27
3.7 Disseny de la interfície d'usuari.....	30
3.8 Limitacions observades.....	30
3.9 Conclusió.....	31
4. Disseny.....	31
4.1 Disseny de la arquitectura del sistema.....	31
4.2 Arquitectura hexagonal: Microserveis (Perquè?).....	37
4.3 Disseny del sistema.....	38
4.3.1 Llista de comandes.....	39
4.3.2 Especificació de les comandes.....	42
4.3.3 Diagrama de classes.....	50
4.3.4 Subdominis.....	50
4.3.5 Mapatge de funcions amb el seu subdomini.....	51
4.3.6 Mapa hexagonal.....	52
4.4 Disseny relacional de la base de dades.....	60
4.5 Disseny en capes.....	62
4.5.1 Disseny de les capes del sistema.....	63
4.6 Patrons de disseny.....	64
5. Conclusions.....	67
6. Bibliografia.....	70
7. Annexos.....	71
7.1 Entitats, relacions, classes i funcions.....	71

## Llista de figures

Il·lustració 1: Diagrama de Gantt.....	9
Il·lustració 2: Diagrama de casos d'ús.....	15
Il·lustració 3: Arquitectura Monolítica .....	32
Il·lustració 4: Arquitectura Client-Servidor .....	33
Il·lustració 5: Arquitectura de Microserveis.....	34
Il·lustració 6: Arquitectura Orientada a Serveis (SOA) .....	35
Il·lustració 7: Arquitectura sense servidor (Serverless) .....	36
Il·lustració 8: Diagrama de classes.....	50
Il·lustració 9: Mapatge complet dels subdominis .....	52
Il·lustració 10: Diagrama general.....	53
Il·lustració 11: Catàleg de serveis .....	54
Il·lustració 12: Perfil d'usuari .....	56
Il·lustració 13: Sistema de reserva i passarel·la de pagament .....	57
Il·lustració 14: Rendiment.....	58
Il·lustració 15: Sistema de comunicació i feedback .....	59
Il·lustració 16: Aplicació del patró factory .....	65
Il·lustració 17: Aplicació del patró Observer .....	66
Il·lustració 18: Aplicació del patró DAO .....	67

# 1. Introducció

## 1.1 Context i justificació del Treball

El punt de partida del treball és generar una aplicació web similar a Booking, per tal de permetre als usuaris poder cercar i reservar vols o allotjaments a diferents parts del món, la principal idea és que els usuaris puguin trobar una destinació de manera fàcil i senzilla i a l'hora, permetre als propietaris promocionar els seus serveis.

Aquesta és una necessitat important, ja que avui en dia moltíssima gent viatja a reu del món i es necessari poder trobar vols o allotjaments de manera senzilla, ràpida i fiable, cert és que Booking ja és una aplicació prou popular, però una aplicació similar que pugui oferir una millor experiència d'usuari pot ser una opció molt interessant.

L'objectiu principal del treball és l'anàlisi i el disseny d'una aplicació web (aquest treball no contindrà la part d'implementació de l'aplicació, només se centrarà en les fases d'anàlisi i disseny) similar a Booking però intentant posar èmfasi en l'experiència d'usuari en àmbits com ara la comunicació.

La principal motivació i l'objectiu personal d'aquest treball és solidificar els coneixements adquirits durant el grau, no s'ha decidit realitzar una aplicació com Booking esperant treure una millor versió, Booking ja té prou força, el que es busca és dissenyar una aplicació que de ser implementada «poguéu intentar competir» amb els estàndards que hi ha ara mateix.

En la mateixa línia, el TFG aporta sobretot experiència i coneixement personal, i genera una visió sòlida de les fases d'un projecte d'enginyeria del programari, el domini de l'aplicació podria haver estat indiferentment qualsevol altre.

L'aportació realitzada en aquest treball serà un document formal amb el disseny funcional d'una aplicació web, juntament amb una visió de totes les fases que componen l'anàlisi i disseny d'un projecte d'enginyeria del programari.

## 1.2 Objectius del Treball

- Analitzar i dissenyar una aplicació «clon» de Booking, que permeti als usuaris poder reservar vols o allotjaments en qualsevol part del món.
- Permetre als usuaris cercar vols o allotjament introduint filtres com ara el preu, la ubicació, el nombre de persones, etc.
- Permetre als propietaris disposar d'una plataforma on poder promocionar els seus serveis.
- Treballar de cara a una millora de l'experiència d'usuari.
- Utilitzar les eines vistes durant el grau per poder dissenyar una aplicació web.



### 1.3 Enfocament i mètode seguit

Per al desenvolupament d'aquest TFG, s'han volgut utilitzar tots els coneixements adquirits al llarg del grau, i plasmar aquests coneixements a l'hora de dissenyar una aplicació web.

La metodologia escollida serà un mix entre cascada i metodologia àgil, primerament en cascada perquè el projecte no tindrà cap MVP fins al final, però tampoc no s'entregarà un sol bloc al final del termini, s'aniran entregant lliurables per tal de tenir una visió general del percentatge de realització del TFG.

### 1.4 Planificació del Treball

S'ha planificat el treball en 3 grans blocs:

El **primer bloc o pla de treball** és bàsicament un bloc introductori on s'ha treballat un **diagrama de Gantt** i s'ha fet una estimació del projecte que ha servit de guia i que s'ha anat modificant al llarg de tot el projecte.

En el **segon bloc o requisits i anàlisi**, s'ha analitzat l'aplicació a baix nivell i s'han treballat les seves característiques com ara: **els objectius, els requisits (tant funcionals com no funcionals), les històries d'usuari, els casos d'ús** tanmateix també s'han analitzat altres aspectes com **el models d'actors o les limitacions**, amb el objectiu de poder donar un context global perfectament detallat del que es dissenyarà.

En el **tercer bloc esdevé el disseny** pròpiament dit, al principi del capítol es pot trobar una **descripció dels diferents tipus d'arquitectures** amb **els seus avantatges i inconvenients**, seguidament **una elecció de l'arquitectura**, després es pot observar **llista de comandes a més de la seva especificació completa, diagrames tant del model de domini com de l'arquitectura**.

### 1.5 Breu sumari de productes obtinguts

Aquest treball només se centrarà en **les fases d'anàlisi i disseny** del cicle d'un projecte d'enginyeria de programari.

El que sí que s'obtidran és una sèrie de documents, gràfics o diagrames que s'inclouran en aquesta memòria.

Al llarg del semestre s'entregaran 4 documents, tot i que a grans trets només seran versions actualitzades del mateix document a excepció de l'entrega final que contindrà 3 arxius:

- **Pla de treball**, o PAC1 El primer lliurable del projecte, contindrà una visió global del projecte amb una especificació de l'organització juntament amb un diagrama de Gantt
- **Entregable parcial: Document d'especificació de requisits i diagrama de casos d'ús**, o PAC2, contindrà l'especificació dels

requisits explorats a més a més dels diagrames de casos d'ús, històries d'usuari i el model d'actors.

- **Entregable parcial: Document de disseny de la base de dades i diagrama de model de dades UML**, o PAC3, contindrà a grans trets l'especificació del sistema, el diagrama de classes, l'especificació i disseny de la base de dades i els diagrames d'arquitectura.
- **Entrega final**, finalment l'últim lliurable serà l'entrega final que contindrà la memòria completa i la presentació (presentació més vídeo).

## 1.6 Breu descripció dels altres capítols de la memòria

S'ha generat un capítol per PAC, a més a més cada PAC conte diversos subapartats que també seran creats com a capítols.

Concretament, l'estructura serà la següent:

- 2. Pla de Treball
  - 2.1 Requisits funcionals
  - 2.2 Planificació detallada.
  - 2.3 Eines a utilitzar per a l'anàlisi i el disseny.
- 3. Requisits i anàlisi
  - 3.1 Definició d'objectius
  - 3.2 Anàlisi de requisits
    - 3.2.1 Requisits funcionals
    - 3.2.2 Requisits no funcionals
  - 3.3 Històries d'usuari
  - 3.4 Diagrama de casos dus
  - 3.5 Casos dus
    - 3.5.1 Dependències entre casos d'ús.
  - 3.6 Disseny de la interfície d'usuari
  - 3.7 Limitacions observades
  - 3.8 Conclusió
- 4. Disseny
  - 4.1 Disseny de la arquitectura del sistema
  - 4.2 Arquitectura hexagonal: Microserveis amb capes (Perquè?)
  - 4.3 Disseny del sistema
    - 4.3.1 Llista de comandes
    - 4.3.2 Especificació de les comandes
    - 4.3.3 Diagrama de classes
    - 4.3.4 Subdominis
    - 4.3.5 Mapeig de funcions amb el seu subdomini
    - 4.3.6 Mapa hexagonal
  - 4.4 Disseny relacional de la base de dades
  - 4.5 Disseny en capes
    - 4.5.1 Disseny de les capes del sistema
  - 4.6 Patrons de disseny

**Pla de Treball:** En aquest apartat, es pot trobar una descripció més detallada del pla de treball en el qual es basarà aquest projecte.

Primerament, es mostren els requisits que es creuen «necessaris» perquè l'aplicació pugui funcionar correctament, tot seguit es pot veure una visió global del TFG amb el Diagrama de Gantt i una explicació detallada. Per acabar es descriuran les eines a utilitzar en la realització d'aquest projecte d'enginyeria de programari.

**Requisits i anàlisi:** En aquest apartat és descriuen els requisits del sistema, els casos d'ús, les històries d'usuari i model d'actors del sistema.

Per finalitzar s'ha generat un prototip de la interfície d'usuari, a més a més és s'han afegit les limitacions i una petita conclusió a tall de síntesi del capítol.

**Disseny:** En aquest apartat, conté tota la informació relacionada amb el disseny funcional del sistema.

Per començar, es pot observar una comparació entre les diferents arquitectures, seguidament d'una llista de comandes extretes dels casos d'ús del tercer capítol amb la seva especificació completa, a més dels diagrames pertinents (tant del model de domini com dels microserveis)

Per finalitzar es pot veure una explicació dels patrons de disseny aplicats més una breu explicació del perquè de l'aplicació del patró i les conclusions.

## 2. Pla de treball

### 2.1 Requisits de l'aplicació

Els requisits al més baix nivell de l'aplicació són els següents:

- **Registre i autenticació d'usuaris:** els usuaris han de poder registrar-se a l'aplicació per tal de poder reservar viatges/vols i si és possible efectuar altres operacions.
  - **Subrequisits:**
    - Formulari per poder registrar-se
    - Sistema per poder iniciar sessió a l'aplicació.
- **Cerca i reserva d'allotjament:** Els usuaris han de poder cercar i reservar allotjament en diferents destinacions, especificant data, nombre de persones, nombre d'habitacions, tipus d'habitacions, etc.
  - **Subrequisits:**
    - Sistema de cerca que doni l'opció a poder incloure diversos filtres.
    - Calendari per poder seleccionar les dates d'entrada i sortida.
    - Sistema per poder reservar l'allotjament i poder efectuar el pagament de forma correcta.
- **Cerca i reserva de vol:** Els usuaris han de poder cercar i reservar vols des de diferents llocs d'origen i destí, introduint, data de sortida i tornada, nombre de persones, etc.
  - **Subrequisits:**
    - Sistema que doni l'opció de poder incloure diversos filtres
    - Calendari per poder seleccionar dates de sortida i tornada.
    - Sistema per poder reservar el vol i poder efectuar el pagament de forma correcta.
- **Gestió de reserves:** Els usuaris han de poder gestionar les seves reserves (veure-les, modificar-les o cancel·lar-les).
  - **Subrequisits:**
    - El sistema ha de permetre a l'usuari poder gestionar les seves reserves.
    - Funcionalitat per poder modificar o cancel·lar reserves.
- **Sistema de fidelitat:** Els usuaris han de poder gaudir d'un sistema de fidelitat, que generi ofertes i descomptes segons el nombre de reserves que l'usuari ha realitzat.
  - **Subrequisits:**
    - El sistema ha d'incloure una forma de recompensa segons el nombre de reserves efectuades.
    - El sistema ha de poder analitzar les dades per trobar patrons de comportament entre els usuaris i poder generar millors ofertes.

- El sistema ha d'assegurar la privacitat i la seguretat de les dades dels usuaris i informar l'usuari que s'utilitzen les seves dades pel sistema de fidelitat.
- **Sistema de comentaris:** Els usuaris han de poder deixar ressenyes i comentaris per poder ajudar a altres usuaris a l'hora de decidir si efectuar una reserva.
  - **Subrequisits**
    - El sistema ha de permetre poder gestionar (crear, editar, esborrar) ressenyes i comentaris.
    - Formulari per poder crear o editar el comentari o ressenya.
    - Sistema que permeti als usuaris poder valorar diferents aspectes de la reserva (Qualitat de l'allotjament, el servei de l'aerolínia) i afegir una puntuació segons la seva experiència.
- **Integració amb servei de tercers:** l'aplicació ha de poder integrar-se amb serveis de tercers per poder tenir la informació de preus i taxes actualitzada en tot moment i permetre als usuaris poder reservar altres serveis (activitats turístiques, vehicles per poder-se moure) a més a més de poder permetre efectuar pagaments des d'altres plataformes.
  - **Subrequisits**
    - Integració amb serveis bancaris de tercers (redis, Google pay, PayPal, etc.)
    - Integració amb serveis de comparació de preus per tal d'oferir una experiència més personalitzada i sempre actualitzada.
    - Integració amb ajuntaments o altres organitzacions amb ànim de lucre i sense per tal de poder reservar activitats i permetre als usuaris planificar els seus itineraris prèviament.

## 2.2 Planificació detallada

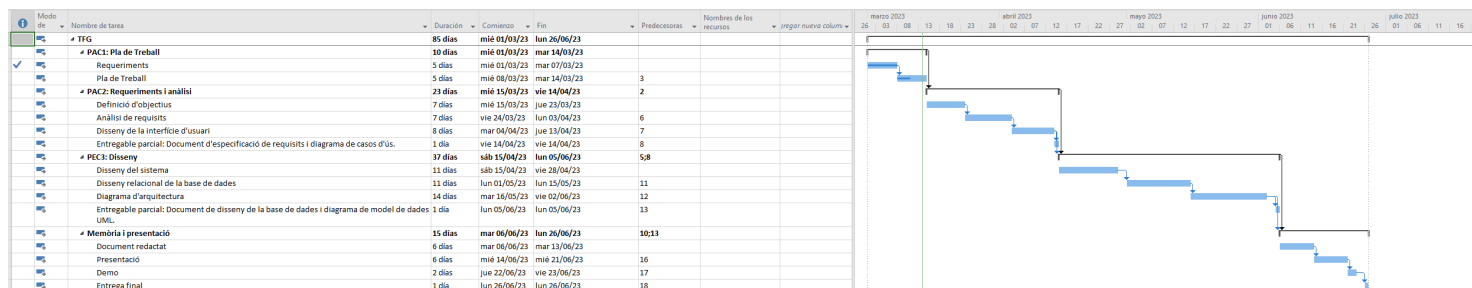
El primer pas **ha esdevingut generar un pla de treball que ha servit de guia** per anar duent a terme el projecte.

A partir d'aquí, **s'ha elaborat una anàlisi dels requisits on a més a més s'han definit les característiques i les funcionalitats de l'aplicació**, tot seguit s'ha generat un prototip el qual seria una versió 1.0 de l'aplicació

Finalitzada aquesta primera part d'anàlisi, s'han enfocat tots els recursos a l'àrea de disseny, on primer de tot **s'ha realitzat una comparació de les diferents arquitectures** més una explicació de per què s'ha escollit una i no pas un altre, seguidament es pot observar **un seguit de llistes amb les comandes del sistema i la seva explicació completa**, més endavant es pot veure **una especificació dels diferents subdominis**, la part final del capítol esdevé **una sèrie de diagrames tant del model de domini com dels diferents subdominis** del sistema juntament amb **l'aplicació d'algun patró** de disseny.

Per acabar, **s'ha generat una memòria** on s'ha anat documentant tot el treball fet, a part d'aquesta memòria es disposarà d'una presentació amb un suport audiovisual per tal de fer una síntesis del treball fet.

A continuació es pot veure el diagrama de Gantt generat a l'inici del projecte.



**Il·lustració 1: Diagrama de Gantt**

A continuació també es pot veure una explicació de cada un dels apartats a l'inici del projecte.

Nombre de tarea
<b>TFG</b>
<b>PAC1: Pla de Treball – Primera PAC: Pla de Treball</b>
Requisits -> En aquesta fase, es generarà una llista de requisits del que farà l'aplicació.
Pla de Treball -> Una vegada obtinguts els requisits, es generarà un diagrama de Gantt amb totes les dates i tasques a realitzar.
<b>PAC2: Requiriments i anàlisi – Segona PAC: Requiriments i anàlisi</b>
Definició d'objectius -> En aquesta primera fase, es definiran els objectius de l'aplicació.
Anàlisi de requisits -> Juntament amb els objectius de l'aplicació, es generaran les històries d'usuari, els casos d'ús i el model d'actors que interactuarà en el sistema.
Disseny de la interfície d'usuari -> En aquest punt es generarà un prototip amb alguna aplicació per tal de visualitzar les diferents finestres de l'aplicació final.
Entregable parcial: Document d'especificació de requisits i diagrama de casos d'ús.
<b>PEC3: Disseny – Tercera PAC: Disseny</b>
Disseny del sistema -> El primer que es realitzarà serà un diagrama de classes d'alt nivell a més a més del disseny de tota l'aplicació.
Disseny relacional de la base de dades -> Una vegada es tingui l'aplicació encarada, es dissenyarà la seva base de dades.
Diagrama d'arquitectura -> Finalment una vegada es tingui tot el sistema definit a més a més de la seva base de dades, serà el moment de definir l'arquitectura que s'utilitzarà per muntar aquest sistema.
Entregable parcial: Document de disseny de la base de dades i diagrama de model de dades UML.
<b>Memòria i presentació – Entrega final</b>
Document redactat -> Aquesta fase serà el punt on s'acabarà de redactar la memòria final.
Presentació -> Una vegada finalitzat el text, es generarà una presentació per tal de tenir un suport gràfic.
Demo -> A més de la memòria i la presentació, es generarà un vídeo a tall de -demo- on es podrà veure tot el treball fet.

## 2.3 Eines a utilitzar per a l'anàlisi i el disseny.

Per tal d'analitzar i dissenyar l'aplicació, es poden utilitzar infinitat d'eines, en aquest apartat es realitzarà una descripció d'aquelles que es creu que pogueren ser necessàries o que és molt possible que s'acabin utilitzant al llarg del TFG.

- Anàlisi:
  - **Diagrames de casos d'ús:** Els diagrames de casos d'ús permetran poder mostrar com interactuen els usuaris amb l'aplicació, s'utilitzarà Dia, el motiu és perquè s'ha utilitzat durant el grau.
  - **Diagrames de flux de dades:** Els diagrames de flux de dades serviran per poder analitzar el comportament de les dades del sistema i com s'interconnecten entre elles (crides externes, peticions, etc.), s'utilitzarà Dia, el motiu és perquè s'ha utilitzat durant el grau.
  - **Prototipatge UI/UX:** No és una eina com a tal, però generar un prototip de l'aplicació permetrà poder mostrar com seria una versió 1.0 de l'aplicació i com s'interconnecten les diferents finestres, s'utilitzarà Figma, el motiu és perquè la seva versió gratuïta té millors prestacions que les alternatives com poden ser Adobe XD o InVision.
- Disseny:
  - **Modelatge UML:** El modelatge UML servirà per poder explicar el comportament del sistema, s'utilitzarà Dia, el motiu és perquè s'ha utilitzat durant el grau.
  - **Patrons de disseny:** Els patrons de disseny seran una bona eina per poder estandarditzar la implementació, s'utilitzarà Dia, el motiu és perquè s'ha utilitzat durant el grau.
  - **Diagrames de classes:** Els diagrames de classes serviran per poder modelar l'estructura interna del sistema (classes, atributs, mètodes, etc.), s'utilitzarà Dia, el motiu és perquè s'ha utilitzat durant el grau.

## 3. Requisits i anàlisi

### 3.1 Definició d'objectius

L'aplicació busca ser un «clon» o una aplicació molt similar a "Booking" o "Atrapalo" entre altres.

"Booking" és una plataforma **metabuscadora de viatges per a reserves d'allotjament**.

En aquesta línia l'aplicació ViatgesUOC, pretén simular la lògica de Booking i permetre als usuaris poder gaudir de les mateixes funcionalitats intentant simplificar la interfície per tal de poder aportar una millor experiència d'usuari.

Els objectius principals de l'aplicació seran els següents:

1. Facilitar a l'usuari una plataforma segura, àgil i senzilla per poder reservar allotjament, vols, serveis entre altres, a més a més ha de poder ser accessible des de qualsevol part del món i ha de poder permetre diverses opcions per cada cerca és a dir, si se cerqués un allotjament, l'aplicació hauria de mostrar hotels, hostals, apartaments, cases compartides, viles, etc.
2. Sobre els serveis oferts en l'aplicació, s'ha de mostrar la informació amb tot grau de detall (fotografies, ubicacions, tarifes, taxes, descripcions detallades del servei, comoditats i serveis addicionals), això permet als usuaris poder prendre una decisió ben informada.
3. La plataforma ha de facilitar el procés de reserva, l'aplicació ha d'oferir un sistema fàcil, àgil i amigable per tal de poder realitzar la reserva.
4. L'aplicació ha de ser accessible des de qualsevol dispositiu.
5. Els usuaris han de tenir un sistema de gestió de reserves, l'usuari ha de poder gestionar les seves reserves (veure-les, editar-les o modificar-les).
6. Permetre als usuaris disposar d'un sistema de comentaris i valoracions, la plataforma ha de permetre als usuaris poder valorar i comentar les reserves efectuades.
7. Generar un sistema de fidelitat, la plataforma ha de conèixer els seus usuaris, per tant, ha de tenir un sistema de fidelitat que generi ofertes i descomptes i incentivi la realització de reserves.
8. L'aplicació ha de poder integrar-se amb altres sistemes de tercers.
9. Finalment, la plataforma ha de garantir la seguretat de les dades incloent les dades financeres i dades personals amb la implementació de mecanismes de seguretat adequats.

En resum, el principal objectiu **és dissenyar una aplicació que permeti cercar allotjament, vols i altres serveis**. Aportant la màxima informació possible per poder generar una experiència completa de cara a l'usuari, **facilitant els tràmits de reserva i gestió dels pagaments i disposant d'un sistema de fidelitat i la possibilitat de deixar valoracions i comentaris**.

A grans trets, el sistema, ha de permetre als usuaris poder cercar allotjaments, vols i altres serveis oferts pels diferents propietaris, a més a més el sistema ha de permetre realitzar una reserva dels serveis cercats i procedir amb el pagament. A més a més, **el sistema ha de ser fàcil d'utilitzar per als usuaris i ha de proporcionar ofertes**, d'altra banda, ha de permetre als usuaris poder afegir valoracions i comentaris i als propietaris poder-les visualitzar i respondre (si escau).

### 3.2 Anàlisi de requisits



Analitzats els objectius de l'aplicació, és el moment d'especificar els requisits.

De requisits hi ha de 2 tipus, **funcionals** (que estaran lligats a històries d'usuari o casos d'ús) i **no funcionals** (que estaran lligats a qualitats o característiques que defineixen el sistema, però que no comportant un cas d'ús o una història d'usuari).[2]

### 3.2.1 Requisits funcionals

Els **requisits funcionals de l'aplicació se centren majoritàriament en el que ha de fer el sistema**, estan estrictament lligats a històries d'usuari o casos d'ús, ja que bàsicament cada requisit funcional podria traduir-se com una acció (Per exemple «Realitzar una reserva»).

1. Cercar allotjament en una ubicació determinada per a les dates seleccionades.
2. Cercar viatges en una ubicació determinada per a les dates seleccionades.
3. Visualitzar informació detallada dels allotjaments, viatges i altres serveis disponibles (imatges, descripcions, tarifes, comoditats, serveis addicionals i valoracions d'altres usuaris).
4. Realitzar la reserva d'una forma senzilla i ràpida.
5. Realitzar la reserva de forma segura i confidencial.
6. Efectuar el pagament de forma senzilla i ràpida.
7. Gestionar les reserves per visualitzar, editar o cancel·lar.
8. Deixar comentaris i valoracions dels serveis reservats.
9. Accedir a ofertes i descomptes per estalviar diners en la reserva de diversos serveis.
10. Permetre als usuaris gaudir d'ofertes i descomptes per incentivar la utilització de la plataforma.
11. Publicar informació detallada dels allotjaments, vols i altres serveis per atraure més clients.
12. Respondre als comentaris i valoracions dels usuaris per mantenir una bona reputació i millorar l'experiència dels clients.
13. Monitorar el rendiment i l'ús de l'aplicació per garantir que sigui atractiva per als usuaris i propietaris.
14. Permetre als usuaris crear i mantenir el seu perfil amb informació personal, d'accés i preferències.
15. Gestionar la comunicació entre usuaris i proveïdors de serveis a través de missatges i notificacions en línia.
16. Gestionar la gestió de queixes i reclamacions dels usuaris i proporcionar una resolució eficaç dels problemes.
17. Garantir la seguretat i privacitat de la informació dels usuaris, incloent-hi la protecció adequada de les dades personals i financeres.

### 3.2.2 Requisits no funcionals

**Els requisits no funcionals** no estan lligats a cap història d'usuari ni cas d'ús, són requisits que no estan relacionats amb la funcionalitat del sistema, **en vers de centrar-se en el que ha de fer el sistema se centren en com ho ha de fer.**

1. **Seguretat:** el sistema ha de garantir la confidencialitat, integritat i disponibilitat de les dades dels usuaris i els serveis reservats. També ha de disposar d'un mecanisme d'autenticació i autorització per evitar l'accés no autoritzat als recursos del sistema.
2. **Usabilitat:** el sistema ha de ser fàcil d'utilitzar per als usuaris i ha de disposar d'una interfície gràfica atractiva i intuïtiva que faciliti la seva navegació i realització de les operacions.
3. **Rendiment:** el sistema ha de ser capaç de gestionar un gran volum de dades i transaccions, mantenint al mateix temps uns temps de resposta adequats per als usuaris.
4. **Escalabilitat:** el sistema ha de ser escalable per a poder adaptar-se a noves necessitats i creixement del negoci sense perdre eficiència.
5. **Fiabilitat:** el sistema ha de ser robust i tolerant a falles, mantenint la integritat de les dades i els serveis reservats en tot moment.
6. **Sostenibilitat:** el sistema ha de ser fàcilment sostenible per als desenvolupadors, permetent la incorporació de noves funcionalitats i correcció de possibles errors sense afectar el funcionament del sistema en producció.
7. **Internacionalització:** el sistema ha de ser capaç de suportar diferents idiomes i monedes per a facilitar la seva utilització per a usuaris de diferents regions del món.

### 3.3 Històries d'usuari

A continuació s'especificaran totes les històries d'usuari de l'aplicació ViatgesUOC

- **Com a usuari, vull poder cercar allotjament** en una ubicació determinada per a les dates que tinc al cap, **per tal de trobar l'allotjament** que millor s'adapti a les meves necessitats.
- **Com a usuari, vull poder cercar vols** en una ubicació determinada per a les dates que tinc al cap, **per tal de trobar el viatge** que millor s'adapti a les meves necessitats.
- **Com a usuari, vull poder veure informació detallada dels allotjaments, viatges o altres serveis disponibles** (com pot ser imatges, descripcions, tarifes, comoditats, serveis addicionals i valoracions d'altres usuaris) **per tal de poder prendre una decisió informada.**
- **Com a usuari, vull poder realitzar la reserva d'una forma senzilla** (amb pocs clics si és possible) **per estalviar temps.**

Com a (-rol-) Vull poder (-objectiu-) Per (-benefici-)
--

Criteri d'acceptació:
-----------------------

- **Com a usuari, vull poder efectuar el pagament d'una forma senzilla per tal de fer tot el processo d'una tirada i estalviar temps.**
- **Com a usuari, vull poder gestionar les meves reserves per tal de poder-les visualitzar, editar o cancel·lar si es cau.**
- **Com a usuari, vull poder deixar comentaris i valoracions s'obre el servei que he reservat per tal de poder ajudar a altres usuaris.**
- **Com a usuari, vull poder gaudir d'ofertes i descomptes per tal d'estalviar diners a l'hora de reservar diversos serveis.**
- **Com a administrador de l'aplicació, vull permetre als usuaris gaudir d'ofertes i descomptes per tal d'incentivar la utilització de la plataforma.**
- **Com a propietari d'allotjament, vull poder publicar informació detallada sobre el meu allotjament, incloent-hi imatges, descripcions, tarifes, comoditats i serveis addicionals, per atraure més clients.**
- **Com a propietari d'una aerolínia, vull poder publicar informació detallada sobre els meus vols, incloent-hi, descripcions, tarifes, comoditats i serveis addicionals, per atraure més clients.**
- **Com a propietari, vull poder publicar informació detallada sobre els meus serveis, incloent-hi, descripcions, tarifes i comoditats, per atraure més clients.**
- **Com a propietari, vull poder respondre als comentaris i valoracions dels usuaris per mantenir una bona reputació i millorar l'experiència dels meus clients.**
- **Com a administrador de l'aplicació, vull poder monitorar el rendiment i l'ús de l'aplicació, per garantir que l'aplicació sigui atractiva per als usuaris i als diversos propietaris.**

### 3.4 Diagrama de casos d'ús



**Flux bàsic:**

1. L'usuari accedeix a la pàgina principal i selecciona l'opció «allotjament»
2. L'usuari introdueix la ubicació i les dates desitjades per a la cerca.
3. L'usuari selecciona els filtres que desitja utilitzar per a la cerca.
  - El preu
  - El tipus
  - Les comoditats
  - Els serveis addicionals
  - La valoració mínima
4. L'usuari prem el botó de cercar.
5. El sistema mostra una llista d'allotjaments que coincideixen amb els criteris seleccionats per l'usuari (inicialment el sistema ordenarà els resultats pel preu, seguidament per la valoració mitjana).
6. L'usuari pot seleccionar qualsevol resultat per tal de veure els seus detalls complets.
7. Si l'usuari decideix realitzar una reserva, pot seleccionar el nombre de persones i el tipus d'allotjament i procedir amb el procés de reserva.

**Postcondicions:**

1. L'usuari ha vist una llista d'allotjaments disponibles que coincideixen amb els criteris de cerca especificats.
2. L'usuari pot seleccionar un allotjament per visualitzar la seva informació detallada.
3. El sistema permet a l'usuari seleccionar el nombre de persones i el tipus d'allotjament correctament i continuar amb el procés de reserva.
4. L'usuari pot tornar a la pàgina de cerca per realitzar una nova cerca.

**Flux alternatiu:**

1. Si no hi ha cap allotjament que coincideixi amb els criteris cercats, el sistema mostrarà un missatge indicant que no hi ha cap allotjament disponible amb aquests criteris i permetrà a l'usuari realitzar una nova cerca.

**Requisits especials:**

- **Cercar vols:** Aquest cas d'ús permet a l'usuari cercar vols i viatges per ubicació, dates, preu etc...

**Nom: Cercar viatges i vols**

**Descripció:** L'usuari pot cercar vols i viatges per ubicació, dates, preu, comoditats i altres criteris.

**Actor principal:** Usuari

**Actor secundari:** Sistema

**Precondicions:**

1. L'usuari té una ubicació i un rang de dates específics en ment per a la cerca.

**Flux bàsic:**

1. L'usuari accedeix a la pàgina principal i selecciona l'opció «vols i viatges»
2. L'usuari introdueix la ubicació i les dates desitjades per a la cerca.
3. L'usuari selecciona els filtres que desitja utilitzar per a la cerca.
  - El preu
  - El tipus
  - Les comoditats
  - Els serveis addicionals
4. L'usuari prem el botó de cercar.
5. El sistema mostra una llista de vols que coincideixen amb els criteris seleccionats per l'usuari (inicialment el sistema ordenarà els resultats pel preu).

6. L'usuari pot seleccionar qualsevol resultat per tal de veure més informació.
7. Si l'usuari decideix realitzar una reserva, pot seleccionar el nombre de passatgers i la classe i procedir amb el procés de reserva.
<b>Postcondicions:</b>
1. L'usuari ha vist una llista de vols i viatges disponibles que coincideixen amb els criteris de cerca especificats.
2. L'usuari pot seleccionar un resultat per visualitzar la seva informació detallada.
3. El sistema permet a l'usuari seleccionar el nombre de passatgers i la classe (si hi ha més d'una) correctament i continuar amb el procés de reserva.
4. L'usuari pot tornar a la pàgina de cerca per realitzar una nova cerca.
<b>Flux alternatiu:</b>
1. Si no hi ha cap vol o viatge que coincideixi amb els criteris cercats, el sistema mostrarà un missatge indicant que no hi ha cap resultat disponible amb aquests criteris i permetrà a l'usuari realitzar una nova cerca.
<b>Requisits especials:</b>

- **Visualitzar informació detallada:** Aquest cas d'ús permet a l'usuari visualitzar informació detallada dels allotjaments, viatges o altres serveis disponibles

<b>Nom:</b> Visualitzar informació detallada
<b>Descripció:</b> L'usuari pot visualitzar informació detallada dels allotjaments, viatges i/o altres serveis.
<b>Actor principal:</b> Usuari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b>
- L'usuari ha cercat un allotjament, un vol o viatge o algun altre servei.
<b>Flux bàsic:</b>
1. L'usuari selecciona un resultat de la llista.
2. El sistema mostrarà informació detallada sobre el resultat .
3. L'usuari pot prendre una millor decisió amb la informació presentada pel sistema.
<b>Postcondicions:</b>
1. L'usuari ha pogut visualitzar la informació detallada del resultat cercat prèviament.
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Reservar servei:** Aquest cas d'ús permet a l'usuari realitzar una reserva d'un servei cercat prèviament.

<b>Nom:</b> Reservar servei
<b>Descripció:</b> L'usuari pot realitzar una reserva d'un servei cercat.
<b>Actor principal:</b> Usuari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b>
- L'usuari ha iniciat sessió.
- L'usuari ha clicat al botó reservar del servei desitjat.
<b>Flux bàsic:</b>
1. L'usuari selecciona l'opció "Reservar" des de la pàgina de visualització de detalls del servei.

<ol style="list-style-type: none"> <li>2. El sistema presenta una pàgina de reserva on l'usuari pot seleccionar les opcions del servei</li> <li>3. L'usuari introdueix les opcions i prem «Endavant».</li> <li>4. El sistema presenta una pàgina de confirmació de la reserva amb les dades de la reserva i el preu total.</li> <li>5. L'usuari confirma la reserva.</li> <li>6. El sistema redirigeix a l'usuari a la plataforma de pagament.</li> <li>7. L'usuari efectua el pagament (Aquest cas d'usuari és causat pel servei de pagament desitjat)</li> <li>8. El sistema confirma la reserva i envia per correu un comprovant que simultàniament mostra a l'usuari.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari ha pogut reservar el servei cercat prèviament.</li> <li>2. L'usuari ha rebut el comprovant de la reserva via email i se li ha mostrat per pantalla.</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Efectuar el pagament:** Aquest cas d'ús permet a l'usuari efectuar el pagament del servei.

<b>Nom: Pagament Senzill</b>
<b>Descripció:</b> L'usuari ha de poder efectuar el pagament de la forma més senzilla possible.
<b>Actor principal:</b> Usuari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'usuari ha iniciat sessió.</li> <li>- L'usuari ha realitzat una reserva d'un servei concret.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció "Pagar" des de la pàgina de confirmació de reserva.</li> <li>2. El sistema redirigeix a l'usuari a la plataforma de pagament.</li> <li>3. La plataforma de pagament, encripta les dades per assegurar que la transmissió sigui segura.</li> <li>4. L'usuari efectua el pagament.</li> <li>5. El sistema confirma el pagament i envia per correu un comprovant que simultàniament mostra a l'usuari.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari ha pogut pagar la reserva.</li> <li>2. L'usuari ha rebut el comprovant del pagament via email i se li ha mostrat per pantalla.</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Gestionar reserves:** Aquest cas d'ús permet a l'usuari gestionar les seves reserves.

<b>Nom: Gestionar reserves</b>
<b>Descripció:</b> L'usuari pot gestionar les seves reserves incloent la visualització, edició o cancel·lació.
<b>Actor principal:</b> Usuari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b>

<ul style="list-style-type: none"> <li>- L'usuari ha iniciat sessió.</li> <li>- L'usuari te com a mínim una reserva realitzada.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció "Les meves reserves" des de la pàgina principal de l'aplicació.</li> <li>2. L'aplicació presenta una pàgina amb la llista de totes les reserves de l'usuari.</li> <li>3. L'usuari selecciona la reserva que vol gestionar i prem l'opció "Visualitzar", "Editar" o "Cancel·lar", segons el que vol fer.</li> <li>4. Si l'usuari selecciona "Visualitzar", l'aplicació mostra la informació detallada de la reserva.</li> <li>5. Si l'usuari selecciona "Editar", l'aplicació permet a l'usuari modificar les dades de la reserva com ara les dates, el nombre de persones, etc.</li> <li>6. Si l'usuari selecciona "Cancel·lar", l'aplicació confirma la cancel·lació de la reserva i processa el reemborsament si s'escau.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari ha pogut gestionar la reserva de forma satisfactòria.</li> <li>2. Si l'usuari ha cancel·lat una reserva, s'ha processat el reemborsament si s'escau.</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Sistema de valoracions:** Aquest cas d'ús permet a l'usuari deixar comentaris i valoracions sobre els serveis reservats amb la finalitat d'ajudar a altres usuaris a prendre una decisió informada.

<b>Nom: Comentaris i Valoracions</b>
<b>Descripció:</b> L'usuari pot deixar comentaris i valoracions sobre els serveis reservats amb la finalitat d'ajudar a altres usuaris.
<b>Actor principal:</b> Usuari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'usuari ha iniciat sessió.</li> <li>- L'usuari ha utilitzat un servei almenys una vegada</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció "Les meves reserves" des de la pàgina principal de l'aplicació.</li> <li>2. L'aplicació presenta una pàgina amb la llista de totes les reserves de l'usuari.</li> <li>3. L'usuari selecciona la reserva que vol valorar i prem l'opció "Valorar".</li> <li>4. L'aplicació presenta un formulari on l'usuari pot introduir un comentari i una puntuació per al servei utilitzat.</li> <li>5. L'usuari introdueix el comentari i la puntuació i prem el botó "Enviar".</li> <li>6. L'aplicació processa el comentari i la puntuació.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari ha deixat un comentari i una valoració</li> <li>2. Els altres usuaris poden veure el comentari i la valoració.</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Ofertes i descomptes:** Aquest cas d'ús permet a l'usuari gaudir de descomptes i ofertes.



<b>Nom: Gaudir d'ofertes i descomptes</b>
<b>Descripció: L'usuari ha de poder gaudir d'ofertes i descomptes per tal de poder estalviar diners i consolidar la fidelitat del client.</b>
<b>Actor principal: Usuari</b>
<b>Actor secundari: Sistema</b>
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'usuari ha d'estar registrat a l'aplicació.</li> <li>- L'usuari ha d'haver utilitzat el servei unes quantes vegades.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció "Fidelitat" des de la pàgina principal de l'aplicació.</li> <li>2. L'aplicació mostra les diferents ofertes i descomptes disponibles.</li> <li>3. L'usuari selecciona l'oferta o descompte que li interessa.</li> <li>4. L'aplicació aplica automàticament l'oferta o descompte a la reserva que l'usuari realitza.</li> <li>5. L'usuari finalitza la reserva de forma normal.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari ha pogut gaudir de l'oferta o descompte seleccionat.</li> <li>2. L'usuari ha completat la reserva del servei amb l'oferta o descompte aplicat.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si no hi ha cap oferta o descompte disponible, el sistema mostrarà un missatge indicant que no hi ha cap oferta disponible en aquest moment.</li> </ol>
<b>Requisits especials:</b>

- **Gestió d'ofertes i descomptes per als usuaris:** Aquest cas d'ús permet a l'administrador de l'aplicació gestionar les ofertes i descomptes disponibles per als usuaris. L'objectiu és incentivar la utilització de la plataforma i augmentar el nombre de reserves.

<b>Nom: Gestió d'ofertes i descomptes.</b>
<b>Descripció: L'administrador ha de poder gestionar les ofertes i descomptes disponibles per als usuaris.</b>
<b>Actor principal: Administrador</b>
<b>Actor secundari: Sistema</b>
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'administrador ha d'estar autenticat en el sistema.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'administrador accedeix a la secció de gestió d'ofertes i descomptes.</li> <li>2. L'administrador crea una nova oferta o descompte, especificant el tipus de servei, les condicions, les dates de vigència i el descompte ofert.</li> <li>3. L'administrador pot editar o eliminar les ofertes i descomptes existents.</li> <li>4. Si l'administrador crea una nova oferta o l'edita, el sistema enviarà una notificació a tots els usuaris que puguin aprofitar-la.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. Les ofertes i descomptes estan disponibles per als usuaris de la plataforma.</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

- **Publicar informació d'allotjament:** Aquest cas d'ús permet als propietaris d'allotjament publicar informació detallada sobre els seus allotjaments.

<b>Nom: Publicar informació d'allotjament</b>
---

<b>Descripció:</b> El propietari ha de poder publicar informació detallada sobre els seus allotjaments.
<b>Actor principal:</b> Propietari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> - El propietari ha d'estar autènticat en el sistema.
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. El propietari accedeix a la secció de gestió d'allotjament</li> <li>2. El propietari selecciona l'opció "Publicar allotjament" o "Editar allotjament" segons si ja ha publicat informació anteriorment o no.</li> <li>3. El propietari introdueix la informació detallada de l'allotjament.</li> <li>4. El propietari selecciona l'opció "Publicar" per publicar la informació o "Guardar esborrany" per guardar la informació sense publicar-la.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. La informació detallada de l'allotjament és publicada a l'aplicació i visible per als usuaris que cerquen allotjament.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si el propietari decideix "Guardar esborrany" per guardar la informació sense publicar-la, aquesta apareixerà com a una possible importació en crear un nou allotjament.</li> </ol>
<b>Requisits especials:</b>

- **Publicar informació sobre vols:** Aquest cas d'ús permet als propietaris publicar informació detallada sobre els seus vols.

<b>Nom:</b> Publicar informació de vol
<b>Descripció:</b> El propietari ha de poder publicar informació detallada sobre els seus vols.
<b>Actor principal:</b> Propietari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> - El propietari ha d'estar autènticat en el sistema.
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. El propietari accedeix a la secció de gestió de vols.</li> <li>2. El propietari selecciona el vol que vol actualitzar o publicar.</li> <li>3. El propietari introdueix o actualitza la informació del vol, incloent-hi les descripcions, les tarifes, les comoditats i els serveis addicionals.</li> <li>5. El propietari selecciona "Publicar" per publicar la informació del vol a la plataforma o "Guardar esborrany" per guardar la informació sense publicar-la.</li> <li>4. La informació del vol és actualitzada o publicada a la plataforma.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. La informació detallada de l'allotjament és publicada a l'aplicació i visible per als usuaris que cerquen allotjament.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si el propietari decideix "Guardar esborrany" per guardar la informació sense publicar-la, aquesta apareixerà com a una possible importació en crear un nou vol.</li> </ol>
<b>Requisits especials:</b>

- **Publicar informació sobre altres serveis:** Aquest cas d'ús permet al propietari publicar informació detallada sobre altres serveis.

<b>Nom:</b> Publicar informació d'altres serveis
<b>Descripció:</b> Els propietaris han de poder publicar informació detallada sobre altres serveis.
<b>Actor principal:</b> Propietari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- El propietari ha d'estar autenticat en el sistema.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. El propietari selecciona l'opció "Publicar Serveis" en el menú principal.</li> <li>2. La plataforma mostra un formulari per afegir informació del servei.</li> <li>3. El propietari omple el formulari amb les dades del servei.</li> <li>4. El propietari selecciona l'opció "Publicar" per publicar la informació del servei o "Guardar esborrany" per guardar la informació sense publicar-la.</li> <li>5. La plataforma mostra una confirmació que el servei ha estat publicat correctament.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. La informació del servei és visible per als usuaris de la plataforma i pot ser consultada i reservada.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si el propietari decideix "Guardar esborrany" per guardar la informació sense publicar-la, aquesta apareixerà com a una possible importació en crear un nou allotjament.</li> </ol>
<b>Requisits especials:</b>

- **Respondre comentaris i valoracions:** Aquest cas d'ús permet als propietaris de serveis respondre als comentaris i valoracions dels usuaris.

<b>Nom:</b> Respondre comentaris i valoracions
<b>Descripció:</b> Els diversos propietaris han de poder respondre als comentaris i valoracions dels usuaris,
<b>Actor principal:</b> Propietari
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- El propietari ha d'estar autenticat en el sistema.</li> <li>- El propietari del servei ha de tenir una reserva activa.</li> <li>- La reserva ha d'haver rebut comentaris o valoracions d'usuaris.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. El propietari accedeix a la secció de comentaris i valoracions de l'aplicació.</li> <li>2. El propietari cerca la reserva activa per la qual vol respondre als comentaris o valoracions.</li> <li>3. El propietari selecciona el comentari o valoració específic al qual vol respondre.</li> <li>4. L'aplicació presenta un formulari on el propietari pot introduir una resposta</li> <li>5. El propietari redacta la resposta adequada per a l'usuari.</li> <li>6. El propietari envia la resposta al comentari o valoració de l'usuari.</li> <li>7. El sistema processa la resposta i aquesta apareixerà com a resposta al comentari/valoració de l'usuari.</li> <li>8. El sistema enviarà un email a l'usuari per notificar-lo que té una nova resposta en un comentari.</li> </ol>

<b>Postcondicions:</b>
<ol style="list-style-type: none"> <li>1. El propietari del servei ha enviat una resposta al comentari o valoració de l'usuari.</li> <li>2. El sistema ha enviat una notificació a l'usuari que té una nova resposta.</li> </ol>
<b>Flux alternatiu:</b>
<ol style="list-style-type: none"> <li>1. Si el propietari del servei no té cap reserva activa o comentari/valoració d'un usuari, l'aplicació mostrarà un missatge informatiu.</li> <li>2. Si el propietari del servei intenta respondre a un comentari/valoració que ja ha estat respost per un altre propietari, l'aplicació mostrarà un missatge d'error i no permetrà enviar una resposta addicional.</li> </ol>
<b>Requisits especials:</b>

- **Monitorar l'ús i rendiment de l'aplicació:** Aquest cas d'ús permet a l'administrador de l'aplicació monitorar el rendiment i l'ús de l'aplicació.

<b>Nom: Monitorar l'ús i el rendiment</b>
<b>Descripció:</b> L'administrador ha de poder monitorar el rendiment i l'ús de l'aplicació.
<b>Actor principal:</b> Administrador
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b>
<ul style="list-style-type: none"> <li>- L'administrador ha d'estar autenticat en el sistema.</li> <li>- L'administrador té accés a l'eina de monitoratge</li> <li>- L'aplicació està en funcionament i accessible</li> </ul>
<b>Flux bàsic:</b>
<ol style="list-style-type: none"> <li>1. L'administrador accedeix l'eina de monitoratge</li> <li>2. L'eina proporciona estadístiques sobre l'ús de l'aplicació.</li> <li>3. L'administrador analitza les dades per identificar àrees d'èxit i problemes potencials</li> <li>4. Si es detecten problemes, l'administrador treballa amb l'equip per identificar i resoldre el problema</li> <li>5. L'administrador utilitza aquesta informació per fer recomanacions als propietaris i millorar l'experiència dels usuaris</li> </ol>
<b>Postcondicions:</b>
<ol style="list-style-type: none"> <li>1. L'administrador té un coneixement més profund de l'ús i rendiment de l'aplicació</li> <li>2. Els problemes detectats s'han abordat o estan en procés de resolució</li> <li>3. Els propietaris i usuaris poden beneficiar-se de les recomanacions de l'administrador per millorar la seva experiència en l'aplicació</li> </ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

Els següents casos d'ús que han anat sorgint dinàmicament i no s'han extret dels requisits o objectius principals:

- **Perfil d'usuari:** Aquest cas d'ús permet a l'usuari poder crear-se i poder mantenir un perfil d'usuari per tal de poder gaudir d'una experiència personalitzada.

<b>Nom: Crear i mantenir perfil d'usuari</b>
<b>Descripció:</b> L'usuari ha de poder crear-se i gestionar el seu perfil d'usuari.
<b>Actor principal:</b> Usuari

<b>Actor secundari: Sistema</b>
<b>Precondicions:</b>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona l'opció de crear un nou perfil d'usuari.</li> <li>2. El sistema demana a l'usuari que introdueixi la seva informació personal.</li> <li>3. L'usuari proporciona la informació necessària per a crear el seu perfil.</li> <li>4. El sistema verifica que la informació proporcionada sigui vàlida i completa.</li> <li>5. El sistema crea un nou perfil d'usuari amb la informació proporcionada.</li> <li>6. L'usuari pot modificar el seu perfil en qualsevol moment, afegint o eliminant informació o preferències.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. El perfil d'usuari ha estat creat amb èxit i es pot utilitzar per buscar i reservar allotjament i altres serveis.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si la informació proporcionada per l'usuari no està completa, el sistema mostrarà un missatge d'error i demanarà a l'usuari que introdueixi la informació restant.</li> </ol>
<b>Requisits especials:</b>

- **Comunicació entre usuaris i proveïdors de serveis:** Aquest cas d'ús proporciona una plataforma de comunicació.

<b>Nom: Comunicació</b>
<b>Descripció:</b> Tant els usuaris com els propietaris han de poder gaudir d'una plataforma on es puguin comunicar de forma fàcil i ràpida.
<b>Actor principal:</b> <ul style="list-style-type: none"> <li>• Usuari.</li> <li>• Propietari.</li> </ul>
<b>Actor secundari: Sistema</b>
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'usuari i el propietari han d'estar registrats en la plataforma.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari selecciona la reserva per a la qual vol comunicar-se amb el propietari.</li> <li>2. L'usuari redacta el missatge que vol enviar al propietari i fa clic en enviar.</li> <li>3. La plataforma envia el missatge al propietari.</li> <li>4. El propietari rep el missatge i pot respondre a través de la plataforma.</li> <li>5. La plataforma envia la resposta al missatge de tornada a l'usuari.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. L'usuari i el propietari han establert una comunicació per resoldre dubtes, solucionar problemes o acordar detalls de la reserva.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si el propietari no respon als missatges, l'usuari pot contactar amb el servei d'assistència al client de la plataforma.</li> <li>2. Si el propietari infringeix les normes de la plataforma, l'usuari pot presentar una queixa.</li> </ol>
<b>Requisits especials:</b>

- **Gestionar queixes:** Aquest cas d'ús permet als usuaris presentar queixes o reclamacions sobre serveis reservats

<b>Nom: Gestió de queixes</b>
<b>Descripció:</b> L'usuari ha de poder presentar queixes o reclamacions.
<b>Actor principal: Usuari</b>
<b>Actor secundari: Sistema</b>

<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'usuari ha d'estar registrat en la plataforma.</li> <li>- L'usuari ha d'haver reservat com a mínim un servei.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'usuari accedeix a la secció de "Suport" de la plataforma.</li> <li>2. L'usuari accedeix a la secció de "Queixes i reclamacions" dins de la secció «Suport»</li> <li>3. L'usuari selecciona el servei pel qual vol presentar una queixa o reclamació.</li> <li>4. L'usuari escriu una descripció detallada del problema i afegeix qualsevol prova que pugui tenir.</li> <li>5. L'usuari envia la queixa o reclamació a la plataforma.</li> <li>6. La plataforma notifica al propietari sobre la queixa o reclamació presentada per l'usuari.</li> <li>7. El propietari del servei es comunica amb l'usuari per aclarir la situació.</li> <li>8. L'usuari rep la resolució del problema i, si és satisfactòria, la queixa o reclamació es dona per tancada.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. La queixa o reclamació de l'usuari s'ha registrat a la plataforma i s'ha pres la mesura adequada per resoldre el problema.</li> </ol>
<b>Flux alternatiu:</b> <ol style="list-style-type: none"> <li>1. Si el proveïdor del servei no resol el problema de manera satisfactòria, l'usuari pot presentar la queixa o reclamació a un nivell superior del suport al client de la plataforma.</li> </ol>
<b>Requisits especials:</b>

- **Garantir la seguretat i privacitat de la informació dels usuaris:**  
Aquest cas d'ús permet a la plataforma garantir que totes les dades personals i financeres dels usuaris estan protegides i mantenen les normes de seguretat i privacitat establertes en la jurisdicció pertinent.

<b>Nom:</b> Seguretat i privacitat de la informació
<b>Descripció:</b> La plataforma ha de garantir que les dades de caràcter personal i financer estan protegides per prevenir qualsevol mena d'accés no autoritzat.
<b>Actor principal:</b> Administrador
<b>Actor secundari:</b> Sistema
<b>Precondicions:</b> <ul style="list-style-type: none"> <li>- L'administrador ha d'estar autenticat en el sistema.</li> </ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"> <li>1. L'administrador de seguretat revisa regularment les polítiques de seguretat i privacitat del sistema per assegurar-se que es compleixen els estàndards adequats.</li> <li>2. Es realitzen auditories de seguretat per detectar possibles vulnerabilitats en el sistema i es prenen mesures per corregir-les.</li> <li>3. És monitora l'activitat dels usuaris per detectar possibles anomalies o intents d'accés no autoritzats.</li> <li>4. Es realitzen còpies de seguretat periòdiques per assegurar la integritat de les dades i la seva recuperació en cas de pèrdua o corrupció.</li> </ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"> <li>1. Les dades dels usuaris són segures i protegides d'acord amb les polítiques de privacitat i seguretat del sistema.</li> <li>2. Es prenen mesures per prevenir i gestionar qualsevol incident de seguretat o privacitat.</li> </ol>
<b>Flux alternatiu:</b>

<b>Requisits especials:</b>
-----------------------------

- **Gestionar el contingut:** Aquest cas d'ús permet a l'administrador de l'aplicació gestionar el contingut publicat pels diversos propietaris.

<b>Nom: Gestió de contingut</b>
<b>Descripció: L'administrador ha de poder gestionar el contingut de l'aplicació.</b>
<b>Actor principal: Administrador</b>
<b>Actor secundari: Sistema</b>
<b>Precondicions:</b> <ul style="list-style-type: none"><li>- <b>L'administrador ha d'estar autenticat en el sistema.</b></li></ul>
<b>Flux bàsic:</b> <ol style="list-style-type: none"><li>1. <b>L'administrador accedeix a la secció de gestió de contingut de l'aplicació.</b></li><li>2. <b>L'aplicació mostra una llista de contingut existent.</b></li><li>3. <b>L'administrador selecciona una opció per, editar o eliminar contingut.</b></li><li>4. <b>Si L'administrador selecciona "Editar contingut", l'aplicació li permet seleccionar un contingut existent per editar-lo.</b></li><li>5. <b>Si L'administrador selecciona "Eliminar contingut", l'aplicació li permet seleccionar un contingut existent per eliminar-lo del sistema.</b></li><li>6. <b>Després de realitzar l'acció desitjada, l'aplicació actualitza la llista de continguts.</b></li></ol>
<b>Postcondicions:</b> <ol style="list-style-type: none"><li>1. <b>La llista de contingut s'haurà actualitzat amb les modificacions realitzades per l'administrador.</b></li></ol>
<b>Flux alternatiu:</b>
<b>Requisits especials:</b>

### 3.5.1 Dependències entre casos d'ús.

En el sistema s'han observat les següents dependències entre els casos d'ús.

1. El cas d'ús "Cercar allotjament en una ubicació determinada per a les dates seleccionades" està relacionat amb el cas d'ús "Cercar viatges en una ubicació determinada per a les dates seleccionades", ja que són funcionalitats de cerca similars només canvia el resultat (allotjament o vol).
2. El cas d'ús "Visualitzar informació detallada dels allotjaments, viatges i altres serveis disponibles" és una dependència per a les funcionalitats de cerca, reserva i valoració.
3. El cas d'ús "Realitzar la reserva depèn del cas d'ús "Visualitzar informació detallada dels allotjaments, viatges i altres serveis disponibles", ja que el primer requereix que el segon s'hagi completat prèviament.
4. El cas d'ús "Efectuar el pagament" depèn dels casos d'ús "Realitzar la " i "Garantir la seguretat i privacitat de la informació dels usuaris".
5. El cas d'ús "Gestionar les reserves" depèn del cas d'ús "Realitzar la reserva".

6. El cas d'ús "Deixar comentaris i valoracions" és una dependència del cas d'ús "Realitzar la reserva".

### 3.6 Model d'actors

A continuació es pot observar el procés per generar un model d'actors:

1. Identificar els usuaris del sistema:

<b>Usuaris identificats:</b>
<b>Clients</b>
<b>Propietaris d'allotjaments</b>
<b>Propietaris d'aerolínies</b>
<b>Propietari d'altres serveis</b>
<b>Administradors</b>
<b>Analista</b>

2. Classificar els usuaris: Aquest pas implica classificar els diversos usuaris en grups per tal de poder generar patrons amb les seves característiques o funcions.

<b>Usuaris identificats:</b>	<b>Funcionalitat</b>
<b>Clients</b>	Persona que utilitza el sistema per a les seves funcions específiques, com cercar, reservar, pagar els diversos serveis oferts.
<b>Propietaris d'allotjaments</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels allotjaments.
<b>Propietaris d'aerolínies</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels vols.
<b>Propietari d'altres serveis</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades de la resta de serveis oferts.
<b>Administradors</b>	Persona amb accés a totes les funcions del sistema, incloent-hi la gestió d'usuaris, la configuració del sistema i la supervisió del rendiment.
<b>Analista</b>	Persona amb accés a algunes funcions del sistema, incloent-hi la supervisió del rendiment i l'anàlisi.

3. Identificar les accions del sistema: Aquest pas implica classificar les diverses accions que realitzarà el sistema per cada usuari.

<b>Usuaris identificats:</b>	<b>Funcionalitat</b>	<b>Accions del sistema</b>
<b>Clients</b>	Persona que utilitza el sistema per a les seves funcions específiques,	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> </ul>



	com cercar, reservar, pagar els diversos serveis oferts.	<ul style="list-style-type: none"> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar accions automàtiques</li> <li>• Realitzar càlculs</li> </ul>
<b>Propietaris d'allotjaments</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels allotjaments.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>
<b>Propietaris d'aerolínies</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels vols.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>
<b>Propietari d'altres serveis</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades de la resta de serveis oferts.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>
<b>Administradors</b>	Persona amb accés a totes les funcions del sistema, incloent-hi la gestió d'usuaris, la configuració del sistema i la supervisió del rendiment.	<ul style="list-style-type: none"> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> <li>• Actualitzar dades</li> <li>• Generar informes</li> </ul>

<b>Analista</b>	Persona amb accés a algunes funcions del sistema, incloent-hi la supervisió del rendiment i l'anàlisi.	<ul style="list-style-type: none"> <li>• Realitzar càlculs</li> <li>• Generar informes</li> <li>• Realitzar accions automàtiques</li> </ul>
-----------------	--	---

4. Identificar les relacions entre usuaris i el sistema: Aquest pas explica com es relacionen els diferents usuaris amb el sistema, aquest pas generarà rols pels diferents usuaris.

<b>Usuaris identificats:</b>	<b>Funcionalitat</b>	<b>Accions del sistema</b>	<b>Rols</b>
<b>Clients</b>	Persona que utilitza el sistema per a les seves funcions específiques, com cercar, reservar, pagar els diversos serveis oferts.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar accions automàtiques</li> <li>• Realitzar càlculs</li> </ul>	<b>Usuari estàndard</b> també anomenat <b>client</b>
<b>Propietaris d'allotjaments</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels allotjaments.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>	<b>Propietari</b>
<b>Propietaris d'aerolínies</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades dels vols.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>	<b>Propietari</b>
<b>Propietari d'altres serveis</b>	Persona que té el control sobre el contingut que es publica al sistema, en aquest cas, les dades de la resta de serveis oferts.	<ul style="list-style-type: none"> <li>• Interactuar amb l'usuari</li> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> </ul>	<b>Propietari</b>

<b>Administradors</b>	Persona amb accés a totes les funcions del sistema, incloent-hi la gestió d'usuaris, la configuració del sistema i la supervisió del rendiment.	<ul style="list-style-type: none"> <li>• Processar dades d'entrada</li> <li>• Emmagatzemar dades</li> <li>• Enviar dades a altres sistemes</li> <li>• Realitzar càlculs</li> <li>• Actualitzar dades</li> <li>• Generar informes</li> </ul>	<b>Administrador</b>
<b>Analista</b>	Persona amb accés a algunes funcions del sistema, incloent-hi la supervisió del rendiment i l'anàlisi.	<ul style="list-style-type: none"> <li>• Realitzar càlculs</li> <li>• Generar informes</li> <li>• Realitzar accions automàtiques</li> </ul>	<b>Analista</b>

### 3.7 Disseny de la interfície d'usuari

Pel disseny de la interfície d'usuari s'ha utilitzat Figma.

Es pot accedir a la versió «Interactiva» del prototip utilitzant la següent URL:

<https://www.figma.com/proto/YGPluvZMIFiYEzHeWKasYi/ViatgesUOC?type=design&node-id=3-19368&scaling=min-zoom&page-id=0%3A1&starting-point-node-id=3%3A19368>

### 3.8 Limitacions observades

En la realització dels apartats anteriors s'han observat una sèrie de limitacions i restriccions que podrien comportar problemes a l'hora de la implementació de l'aplicació, aquestes són les següents:

1. **Limitacions de temps:** El sistema ha de respondre en un temps raonable per satisfer les expectatives dels usuaris i evitar la frustració. El temps de resposta depèn de la complexitat de les operacions i la capacitat del servidor. Les limitacions de temps poden afectar el rendiment de l'aplicació si aquesta no avalua de forma correcta i s'optimitza en conseqüència.
2. **Limitacions d'espai:** El sistema ha de tenir una capacitat **d'emmagatzematge** suficient per gestionar tota la informació dels usuaris i dels proveïdors de serveis. Les limitacions d'espai també

poden afectar la quantitat de dades que es poden visualitzar a la vegada en la interfície del sistema.

3. **Limitacions de seguretat:** És important garantir la seguretat de les dades dels usuaris i dels proveïdors de serveis per evitar la pèrdua o el robatori de la informació. El **sistema** ha de complir les normatives vigents en matèria de protecció de dades personals. Les limitacions de seguretat poden afectar en el fet que l'aplicació no compleixi amb les normatives vigents i de cap manera les dades dels usuaris es poden veure vulnerades.
4. **Limitacions de dispositius:** El sistema ha de ser compatible amb una ampla gamma de dispositius, com ara telèfons mòbils, tauletes i ordinadors de sobretaula. Les limitacions de dispositius poden afectar la facilitat d'ús del sistema en dispositius amb pantalles més petites o amb diferents resolucions.
5. **Limitacions de coneixement dels usuaris:** El sistema ha de ser intuïtiu i fàcil d'utilitzar per als usuaris amb diferents nivells de coneixement tecnològic. Les limitacions de coneixement dels usuaris poden afectar la facilitat d'ús del sistema si no s'ofereix una interfície clara i senzilla per als usuaris inexperts.
6. **Limitacions d'idioma:** El sistema ha de ser accessible per a usuaris que parlen diferents idiomes. Les limitacions d'idioma poden afectar la capacitat del sistema per atraure i satisfer una audiència global.

### 3.9 Conclusió

En conclusió, aquest document ha establert els requisits del sistema. S'han definit tant els requisits funcionals com els no funcionals per garantir una interfície amigable per a l'usuari.

Tanmateix, també s'han descrit les històries d'usuari així com el model d'actor del sistema.

En definitiva, aquest document ha establert una base sòlida per començar amb la implementació del sistema proporcionant una guia clara dels requisits que s'han de complir per a una experiència d'usuari satisfactòria i segura.

## 4. Disseny

### 4.1 Disseny de la arquitectura del sistema

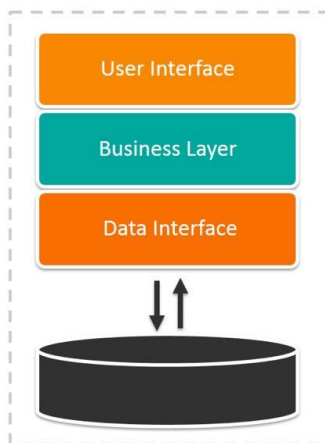
De paradigmes d'arquitectura n'hi ha molts, en aquest punt és important parar-se un temps a pensar una bona arquitectura, per aquest projecte en concret, hi ha varies aproximacions que podrien valer perfectament, es classificaran alguns de forma que sigui més fàcil poder decidir quina pot ser la millor opció. [4]

### **Arquitectura Monolítica:**

L'arquitectura monolítica és un tipus d'arquitectura de software en què tots els components d'una aplicació es desenvolupen, despleguen i s'executen com una sola entitat cohesiva.

En aquesta arquitectura, tot el codi i les funcionalitats de l'aplicació resideixen en un sol procés o monòlit, que pot ser una aplicació web, una aplicació de servidor o una aplicació empresarial. [5]

#### **Monolithic Architecture**



**Il·lustració 3: Arquitectura Monolítica**

#### 1. Avantatges:

- a. Fàcil desenvolupament i desplegament, ja que tots els components resideixen en un sol sistema.
- b. Alt rendiment gràcies a l'accés directe a la base de dades.
- c. Simplicitat en el manteniment i solució de problemes, ja que tot està contingut en una única aplicació.
- d. Integració simplificada amb tecnologies heretades.
- e. Menor latència en les comunicacions internes.

#### 2. Desavantatges:

- a. Manca d'escalabilitat, ja que l'aplicació s'ha d'escalar en la seva totalitat.
- b. Complexitat creixent a mesura que el sistema es fa més gran i més complex.
- c. Dificultat per a la implementació de tecnologies i frameworks moderns.
- d. Major risc de fallades, ja que un error en un component pot afectar tot el sistema.

- e. Limitacions en la distribució geogràfica i la disponibilitat global.

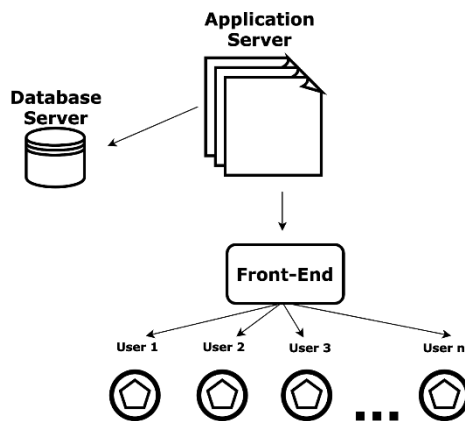
### **Arquitectura Client-Servidor:**

L'arquitectura client-servidor és un model arquitectònic de programari en què les funcionalitats i les tasques són distribuïdes entre dues parts principals: el client i el servidor.

El client és l'aplicació o la interfície d'usuari que interactua amb l'usuari final.

El servidor és responsable de gestionar i processar les peticions del client, així com de mantenir i gestionar les dades i la lògica de negoci. [6]

### **Client-Server Architecture High-Level Diagram**



**Il·lustració 4: Arquitectura Client-Servidor**

#### 1. Avantatges:

- a. Separació clara entre la interfície d'usuari i la lògica de negoci, facilitant el manteniment i la modificació de cada part per separat.
- b. Escalabilitat mitjançant l'ús de múltiples servidors per gestionar les peticions dels clients.
- c. Facilitat per a l'actualització i desplegament de nous clients i servidors de manera independent.
- d. Millor seguretat mitjançant el control d'accés centralitzat i autenticació dels clients.
- e. Possibilitat de reutilitzar components de servidor en diferents aplicacions de client.

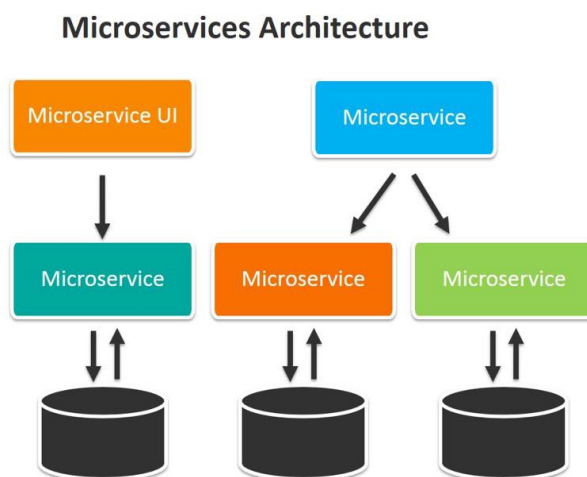
#### 2. Desavantatges:

- a. Dependència de la infraestructura de xarxa, amb el risc de rendiment reduït o pèrdua de connexió.
- b. Complexitat en la gestió i manteniment dels servidors.
- c. Sobrecàrrega de comunicació entre el client i el servidor en aplicacions amb moltes peticions.

- d. Costos d'infraestructura i manteniment més elevats per gestionar els servidors.
- e. Requereix un control rigorós per evitar l'acoblament excessiu entre el client i el servidor.

### **Arquitectura de Microserveis:**

L'arquitectura de microserveis és un model arquitectònic en què una aplicació es divideix en diversos serveis petits, autònoms i independents, cada un d'ells centrat en una única funcionalitat específica. Cada microservei s'executa com un procés o una aplicació separada i pot comunicar-se amb altres microserveis mitjançant protocols de comunicació com HTTP o RPC. [8,9]



**Il·lustració 5: Arquitectura de Microserveis**

#### 1. Avantatges:

- a. Modularitat i desacoblament, ja que l'aplicació es divideix en petits serveis independents.
- b. Escalabilitat granular, ja que cada servei pot ser escalat de manera independent segons la demanda.
- c. Facilitat per a l'adopció de tecnologies i frameworks moderns en cada servei.
- d. Major agilitat en el desenvolupament i desplegament, ja que els serveis poden ser implementats, provats i actualitzats de manera independent.
- e. Millor tolerància a fallades, ja que un error en un servei no afecta la resta.

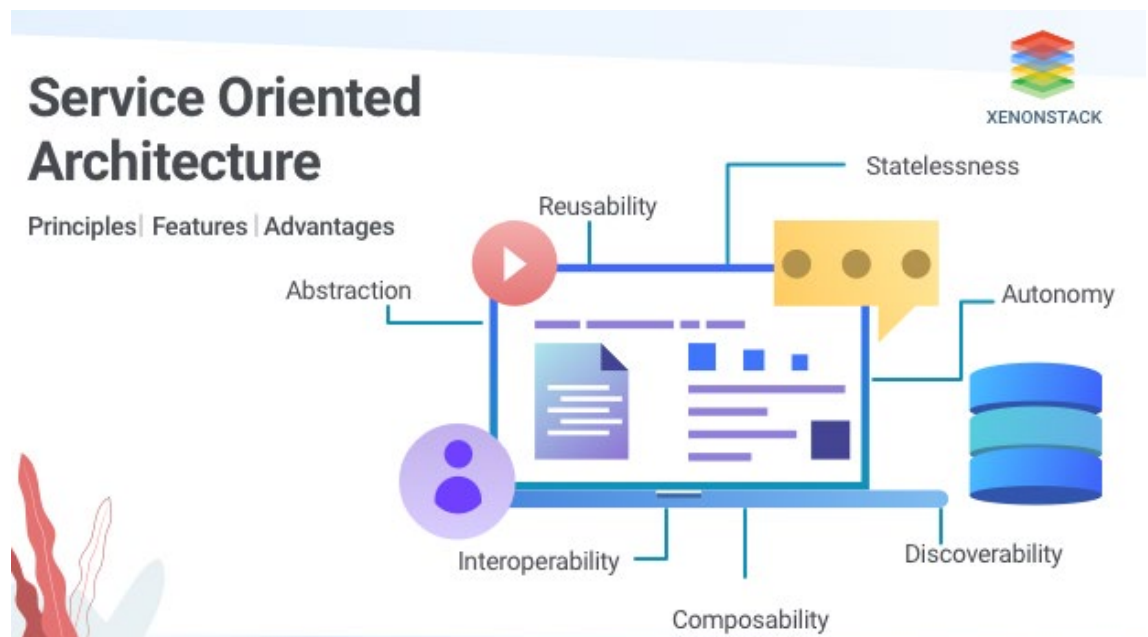
#### 2. Desavantatges:

- a. Complexitat operativa més gran, ja que s'han de gestionar múltiples serveis.
- b. Necessitat de comunicació eficient entre els serveis per mantenir la coherència de les dades.
- c. Requeriments de més recursos i capacitat de càlcul per gestionar la comunicació entre els serveis.

- d. Augment de la complexitat de les proves i la depuració en un entorn distribuït.
- e. Requereix una bona arquitectura de control i supervisió per gestionar i monitorar els microserveis.

### **Arquitectura Orientada a Serveis (SOA):**

L'arquitectura Orientada a Serveis (SOA, per les seves sigles en anglès) és un model arquitectònic en el qual les funcionalitats de l'aplicació s'organitzen com a serveis independents que poden ser utilitzats per diverses aplicacions i processos. En una arquitectura SOA, els serveis són components autònoms que ofereixen funcionalitats específiques i es poden invocar mitjançant interfícies estàndard.[11]



**Il·lustració 6: Arquitectura Orientada a Serveis (SOA)**

1. Avantatges:
  - a. Reutilització de serveis en diferents aplicacions, permetent un desenvolupament més eficient.
  - b. Facilitat per a la integració de sistemes i aplicacions existents.
  - c. Flexibilitat per canviar o actualitzar components sense afectar les altres parts del sistema.
  - d. Simplificació de les interaccions entre les aplicacions mitjançant interfaces ben definides.
  - e. Millor visibilitat i control dels serveis empresarials mitjançant eines de gestió específiques.
2. Desavantatges:
  - a. Complexitat en la planificació i el disseny inicial del sistema.
  - b. Costos inicials més elevats per a la implementació d'infraestructura de suport.

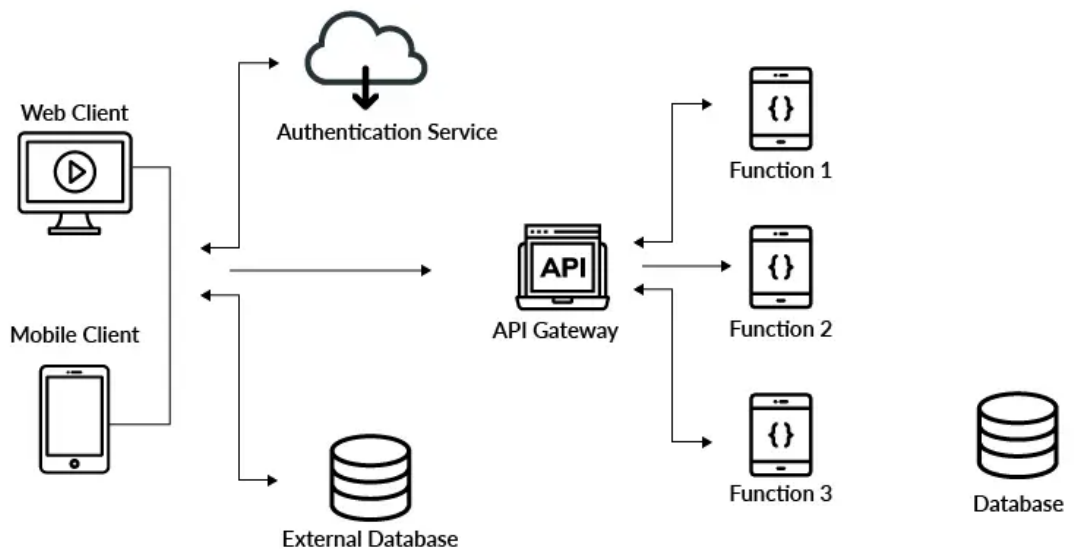


- c. Dependència de la fiabilitat de la xarxa per a la comunicació entre els serveis.
- d. Possibilitat d'acoblament fort entre els serveis, dificultant la seva modificació o substitució.
- e. Necessitat de coordinació i governança per mantenir la coherència i la compatibilitat entre els serveis.

### **Arquitectura sense servidor (Serverless):**

L'arquitectura sense servidor (Serverless) és un model arquitectònic en el qual el proveïdor de serveis en núvol és responsable de l'execució i el dimensionament automàtics dels serveis, mentre que els desenvolupadors se centren únicament en el codi de les funcions o les aplicacions sense haver de gestionar la infraestructura subjacent.[13]

#### **Working Of Serverless Architecture**



#### **Il·lustració 7: Arquitectura sense servidor (Serverless)**

##### 1. Avantatges:

- a. Desplegament i escalabilitat automàtics segons la demanda, sense la necessitat de gestionar infraestructura.
- b. Optimització de recursos, ja que només es paga pel temps d'execució efectiu.
- c. Facilitat per al desenvolupament ràpid de prototips i proves de concepte.
- d. Millora de la fiabilitat, ja que els serveis sense servidor tenen mecanismes de recuperació automàtics.
- e. Integració senzilla amb serveis i APIs en el núvol.

##### 2. Desavantatges:

- a. Limitacions en termes de temps d'execució, recursos i polítiques de seguretat.
- b. Complexitat en la gestió de les funcions i la seva coordinació en aplicacions més grans.
- c. Dificultat per a la depuració i el seguiment en un entorn sense servidor.
- d. Requeriments específics per al disseny i la implementació de les funcions.
- e. Possibles costos inesperats si no es gestiona adequadament el temps d'execució i els recursos.

## 4.2 Arquitectura hexagonal: Microserveis (Perquè?)

El motiu de per què s'ha escollit microserveis per representar l'arquitectura hexagonal, és primer de tot perquè és l'arquitectura que es va estudiar durant el grau (un dels objectius era aplicar tot el coneixement obtingut), aleshores, juntament parlant, amb el consultor, s'ha considerat utilitzar-la com a model per poder dissenyar el sistema.

A més a més, si s'observa en l'anàlisi, l'arquitectura hexagonal amb microserveis separats pot ser una bona opció per aquest sistema per diverses raons. A continuació, s'enumeren els avantatges i inconvenients d'aquesta arquitectura.

---

Avantatges:

---

1. **Desacoblament i modularitat:** L'ús d'una arquitectura hexagonal amb microserveis permet separar clarament les diferents funcionalitats del sistema en capes. Això facilita el desenvolupament, el manteniment i l'evolució independent de cadascuna d'elles.
  - a. Per exemple, un microservei per a la gestió de cerca d'allotjament, un altre per a la gestió de reserva de, i així successivament.
2. **Escalabilitat:** Amb microserveis separats, es té la capacitat d'escalar individualment cada capa o servei en funció de les necessitats de càrrega.
  - a. Per exemple, si la funcionalitat de cerca és la més utilitzada, es pot escalar només aquest microservei per gestionar un augment en el nombre de sol·licituds sense afectar els altres components del sistema.
3. **Flexibilitat tecnològica:** L'arquitectura hexagonal permet utilitzar diferents tecnologies i llenguatges de programació per als diferents

microserveis. Això ofereix la flexibilitat de triar les millors eines per a cada funcionalitat específica.

4. **Facilitat de manteniment i evolució:** Amb les capes ben separades i els microserveis, es té la possibilitat de realitzar canvis en una capa o servei sense afectar la resta del sistema. Això facilita les tasques de manteniment, correcció d'errors i introducció de noves funcionalitats.
  - a. Per exemple, es poden actualitzar o afegir nous microserveis sense impactar negativament en la integritat global del sistema.

---

Inconvenients:

---

1. **Complexitat inicial:** L'arquitectura hexagonal amb microserveis implica una complexitat inicial més gran. Cal definir les interfícies i comunicacions entre els microserveis i assegurar-se que tots els components estiguin adequadament connectats. Això requereix un temps i una planificació addicional per configurar el sistema inicialment.
2. **Sobrecàrrega de comunicació:** Amb microserveis separats, cal gestionar les comunicacions i les peticions entre els diferents components del sistema. Això pot introduir una certa sobrecàrrega de xarxa, especialment si no s'optimitzen adequadament els mecanismes de comunicació. Cal garantir que les comunicacions entre els microserveis siguin eficients per evitar retards innecessaris.
3. **Complexitat de desplegament i gestió:** Amb múltiples microserveis, el desplegament i la gestió del sistema poden ser més complexos. S'han de tenir en compte aspectes com el control de versions, la monitorització i la gestió dels diferents components del sistema. Cal assegurar-se que el desplegament i la gestió dels microserveis siguin gestionats de manera eficient.
4. **Dependència de la xarxa:** Amb microserveis separats, el sistema depèn de la disponibilitat i estabilitat de la xarxa perquè els diferents components puguin comunicar-se entre ells. En cas de fallades en la xarxa, això pot afectar el funcionament global del sistema. Cal garantir una infraestructura de xarxa adequada per a un bon funcionament del sistema.

Tot i que es tenen certs inconvenients, els avantatges que comporta la utilització d'aquesta arquitectura juntament amb el coneixement adquirit han sigut els factors determinants per escollir una arquitectura hexagonal amb microserveis.

### 4.3 Disseny del sistema

El primer pas **consisteix a dissenyar l'esquelet de tota l'aplicació**, aquest contindrà **la llista de comandes** per executar les diferents funcionalitats de l'aplicació (cada comanda contindrà la següent informació: Actor, Cas d'ús, Nom de la comanda en format LCC Lower Camel Case en anglès, descripció de la comanda), a continuació **l'especificació de la comanda** amb (paràmetres, el seu retorn, les precondicions i les seves postcondicions), un **diagrama de classes** d'alt nivell, **especificació dels hexàgons** o capes del sistema, **mapatge funcional del sistema** en cada hexàgon o capa i finalment **un mapa de l'arquitectura**.

### 4.3.1 Llista de comandes

Com s'ha indicat, el primer que contindrà el disseny serà una llista de comandes, amb la informació recollida en la fase d'anàlisi, és molt senzill realitzar aquesta llista (aprofitant i anomenant cada comanda en format LLC):

Comanda en format LLC	Actor	Descripció	Cas d'ús
<hr/> <i>Client</i> <hr/>			
cercarAllotjament	Client	Cercar un allotjament introduint certs paràmetres de cerca	Cas d'ús: Cercar allotjament
cercarVol	Client	Cercar un vol introduint certs paràmetres de cerca	Cas d'ús: Cercar vols i viatges
veureDetallsDeAllotjament	Client	Visualitzar els detalls d'un allotjament	Visualitzar informació detallada
veureDetallsDeVols	Client	Visualitzar els detalls d'un vol	Visualitzar informació detallada
reservarAllotjament	Client	Reservar un allotjament	Reservar servei de forma segura
reservarVol	Client	Reservar un vol	Reservar servei de forma segura
reservarServei	Client	Reservar un servei	Reservar servei de

			forma segura
realitzarPagament	Client	Efectuar el pagament d'un servei	Pagament senzill
editarReserva	Client	Modificar una reserva realitzada	Gestionar reserves
cancelarReserva	Client	Cancel·lar una reserva realitzada	Gestionar reserves
afegirComentari	Client	Crear un comentari nou sobre una reserva	Comentaris i valoracions
afegirValoracio	Client	Afegir una valoració a una reserva	Comentaris i valoracions
aplicarDescompte	Client	Utilitzar un descompte disponible	Gaudir d'ofertes i descomptes
registrarse	Client	Registrar-se com a usuari client	Crear i mantenir perfil d'usuari
editarUsuari	Client	Modificar els camps de l'usuari	Crear i mantenir perfil d'usuari
donarseDeBaixa	Client	Eliminar el perfil d'usuari del sistema	Crear i mantenir perfil d'usuari
afegirQueixa	Client	Crear una nova queixa	Gestió de queixes
editarQueixa	Client	Modificar una queixa creada prèviament	Gestió de queixes
<hr/> <i>Propietari</i> <hr/>			
publicarAllotjament	Propietari	Publicar un nou allotjament	Publicar informació d'allotjament
editarAllotjament	Propietari	Editar un allotjament	Editar informació d'allotjament

			(no hi ha cas d'ús)
eliminarAllotjament	Propietari	Eliminar un allotjament	Eliminar allotjament (no hi ha cas d'ús)
publicarVol	Propietari	Publicar un nou vol	Publicar informació de vol
editarVol	Propietari	Editar un vol	Editar informació d'allotjament (no hi ha cas d'ús)
eliminarVol	Propietari	Eliminar un vol	Eliminar vol (no hi ha cas d'ús)
publicarAltreServei	Propietari	Publicar un altre servei	Publicar informació d'altres serveis
editarAltreServei	Propietari	Editar un altre servei	Editar informació d'allotjament (no hi ha cas d'ús)
eliminarAltreServei	Propietari	Eliminar un altre servei	Eliminar servei (no hi ha cas d'ús)
respondreQueixa	Propietari	Respondre queixa creada per un usuari	Comunicació
editarResposta	Propietari	Modificar la resposta de la queixa	Comunicació
eliminarResposta	Propietari	Eliminar la resposta	Comunicació
respondreComentari	Propietari	Respondre un comentari	Respondre comentari o valoració
respondreValoracio	Propietari	Respondre un valoracio	Respondre comentari o valoració
<hr/> <i>Administrador</i> <hr/>			

crearOferta	Administrador	Crear una oferta nova	Gestió d'ofertes i descomptes.
editarOferta	Administrador	Editar una oferta existent	Gestió d'ofertes i descomptes
eliminarOferta	Administrador	Eliminar una oferta existent	Gestió d'ofertes i descomptes
editarAllotjament	Administrador	Editar un allotjament	Gestió de contingut
editarVol	Administrador	Editar un vol	Gestió de contingut
borrarAllotjament	Administrador	Eliminar un allotjament	Gestió de contingut
borrarVol	Administrador	Eliminar un vol	Gestió de contingut
<hr/> <i>Analista</i> <hr/>			
calcularMitjana	Analista	Calcular la mitjana de valoracions de qualsevol servei	Monitorar l'ús i el rendiment
calcularRendiment	Analista	Calcular el rendiment de l'aplicació	Monitorar l'ús i el rendiment
generarReportsDeRendiment	Analista	Generar un informe de rendiment	Monitorar l'ús i el rendiment

### 4.3.2 Especificació de les comandes

Per cada comanda s'especificaran: paràmetres, el retorn, les precondicions, les postcondicions

---

*Usuari*

---

Comanda	Paràmetres	Retorn	Precondicions	Postcondicions
cercarAllotjament	Preu, tipus, comoditats, serveis, valoració (Tots els paràmetres	Allotjament o llista d'allotjaments	Ubicació, rang de dates (la data d'entrada és obligatòria, no	Llista d'allotjaments, el sistema permet visualitzar

	són opcionals, si no s'envien cercarà tots els allotjaments.		és necessària la data de sortida).	detalls i seguir amb el procés de reserva, el sistema permet a l'usuari tornar endarrere.
cercarVol	Preu, tipus, comoditats, serveis, valoració (Tots els paràmetres són opcionals, si no s'envien cercarà tots els allotjaments).	Vol o llista de vols	Ubicació, rang de dates (la data d'entrada és obligatòria, no és necessària la data de sortida).	Llista d'allotjaments, el sistema permet visualitzar detalls i seguir amb el procés de reserva, el sistema permet a l'usuari tornar endarrere.
veureDetalls DeAllotjament	Allotjament	Nul	Allotjament	El sistema permet a l'usuari tornar endarrere
veureDetalls DeVols	Vol	Nul	Vol	El sistema permet a l'usuari tornar endarrere
reservarAllotjament	Allotjament	Correu amb el comprovant de la reserva	Allotjament	El comprovant es mostra per pantalla
reservarVol	Vol	Correu amb el comprovant de la reserva	Vol	El comprovant es mostra per pantalla
reservarServei	Servei	Correu amb el comprovant de la reserva	Servei	El comprovant es mostra per pantalla
realitzarPagament	Reserva	Factura i comprovant de pagament enviat a l'email	Iniciar sessió, una reserva	Reserva modificada
cancelarReserva	Reserva	Reserva eliminada	Iniciar sessió, una reserva	Reserva eliminada, rebolcament si escau



afegirComentari	Reserva, allotjament, vol o un servei	Nou comentari	Iniciar sessió, una reserva	Servei associat a la reserva conte un nou comentari
afegirValoracio	Reserva, allotjament, vol o un servei	Nova valoració	Iniciar sessió, una reserva	Servei associat a la reserva conte una nova valoració
aplicarDescompte	Reserva	Preu de la reserva es redueix aplicant el descompte al preu final	Iniciar sessió, una reserva, descomptes disponibles	L'usuari podrà efectuar el pagament amb el preu reduït o no aplicar el descompte i tornar endarrere
registrarse	Nom, cognoms, adreça de correu electrònic, DNI, número de telèfon, contrasenya	Usuari nou	L'usuari no existeix	L'email queda registrat i no pot ser utilitzat de nou, l'usuari podrà iniciar sessió amb l'usuari creat.
editarUsuari	Usuari, el nom, cognoms, adreça de correu electrònic, DNI, número de telèfon, contrasenya (tots els camps són opcionals, tret de l'usuari).	L'usuari veu modificat el camp o camps pertinents	L'usuari existeix al sistema	L'usuari veu el camp o camps modificats. Si es canvia la contrasenya l'usuari haurà d'iniciar sessió novament amb la nova contrasenya (el sistema farà fora a l'usuari).
donarseDeBaixa	Usuari	Usuari eliminat	L'usuari existeix al sistema	Una vegada utilitzat el sistema donarà 48 hores per tornar a activar l'usuari, passat aquest temps l'usuari es barrarà del sistema i no es podrà tornar a utilitzar.

afegirQueixa	Allotjament, vol, servei, una reserva	La nova queixa s'afegeix al servei pertinent	Iniciar sessió, tenir una reserva	La queixa s'afegeix a la llista del servei pertinent, s'envia un email al propietari per informar-lo de la nova queixa afegida.
editarQueixa	Queixa	La queixa es veurà modificada	Iniciar sessió, tenir una reserva, una queixa prèviament	La queixa es veu modificada, s'envia un email al propietari per informar dels canvis.

---

*Propietari:*

---

publicarAllotjament	Imatges, descripció, tarifa, comoditats, serveis addicionals	Un allotjament	Propietari autenticat	El nou allotjament es publica a la llista d'allotjaments
editarAllotjament	Allotjament, imatges, descripció, tarifa, comoditats, serveis addicionals (tots els paràmetres són opcionals menys l'allotjament, aquest és obligatori).	L'allotjament veu el seu/s camps modificat/s	Propietari autenticat	L'allotjament veu els seus camps modificats.
eliminarAllotjament	Allotjament	Allotjament eliminat	Propietari autenticat	L'allotjament s'elimina de la llista d'allotjaments

publicarVol	Descripció, tarifa, comoditats, serveis addicionals, ruta del vol, horari de sortida i arribada, la durada del vol, el nombre de parades.	Un vol	Propietari autenticat	El nou vol es publica a la llista de vols
editarVol	Vol, descripció, tarifa, comoditats, serveis addicionals, ruta del vol, horari de sortida i arribada, la durada del vol, el nombre de parades (tots els paràmetres són opcionals menys l'allotjament, aquest és obligatori).	El vol veu el seu/s camps modificat/s	Propietari autenticat	El vol veu els seus camps modificats.
eliminarVol	Vol	Vol eliminat	Propietari autenticat	El vol s'elimina de la llista de vols
publicarAltreservei	Nom del servei, descripció, imatges, tarifa i comoditats	Un servei	Propietari autenticat	El nou servei es publica a la llista de serveis
editarAltreservei	Nom del servei, descripció, imatges, tarifa i comoditats	El servei veu el seu/s camps modificat/s	Propietari autenticat	El servei veu els seus camps modificats.
eliminarAltreservei	Servei	Servei eliminat	Propietari autenticat	El servei s'elimina de la llista de serveis
respondreQueixa	Queixa	Queixa amb resposta	Propietari autenticat	La resposta queda afegida a la queixa, l'usuari rep

				un email indicant que s'ha modificat la queixa.
editarResposta	Queixa amb resposta	La resposta de la queixa és modificada	Propietari autenticat	La nova resposta es veu modificada, l'usuari rep un email indicant que s'ha modificat la queixa.
eliminarResposta	Resposta	Resposta eliminada	Propietari autenticat	La resposta és eliminada i ja no es pot visualitzar, l'usuari rep un email indicant que s'ha modificat la queixa.
respondreComentari	Un comentari	El comentari conte la nova resposta	Propietari autenticat	El comentari conte la nova resposta, l'usuari rep un email indicant que el seu comentari conte una nova resposta.
respondreValoracio	Una valoració	La valoració conte la nova resposta	Propietari autenticat	La valoració conte la nova resposta, l'usuari rep un email indicant que la seva valoració conte una nova resposta.

---

Administrador

---

crearOferta	Servei, condicions, dates de vigència	Oferta	Administrador autenticat	Oferta disponible a la llista d'ofertes, a més a més els usuaris que poden gaudir d'aquesta oferta rebran un correu indicant que hi ha una nova oferta disponible.
editarOferta	Servei, condicions, dates de vigència, oferta (tots els camps són opcionals menys l'oferta)	Oferta	Administrador autenticat	L'oferta es veu modificada
eliminarOferta	Oferta	Oferta eliminada	Administrador autenticat	L'oferta deixa d'estar disponible a la llista d'ofertes
editarAllotjament	Allotjament, imatges, descripció, tarifa, comoditats, serveis addicionals (tots els paràmetres són opcionals menys l'allotjament, aquest és obligatori).	L'allotjament veu el seu/s camps modificat/s	Administrador autenticat	L'allotjament es veu modificat
editarVol	Vol, descripció, tarifa, comoditats, serveis addicionals, ruta del vol, horari de	El vol veu el seu/s camps modificat/s	Administrador autenticat	El vol es veu modificat

	sortida i arribada, la durada del vol, el nombre de parades (tots els paràmetres són opcionals menys l'allotjament, aquest és obligatori).			
editarAltreServei	Nom del servei, descripció, imatges, tarifa i comoditats	El servei veu el seu/s camps modificat/s	Administrador autenticat	El servei es veu modificat
eliminarAllotjament	Un allotjament	Allotjament eliminat	Administrador autenticat	L'allotjament s'elimina de la llista
eliminarVol	Un vol	Vol eliminat	Administrador autenticat	El vol s'elimina de la llista
eliminarAltreServei	Un servei	Servei eliminat	Administrador autenticat	El servei s'elimina de la llista

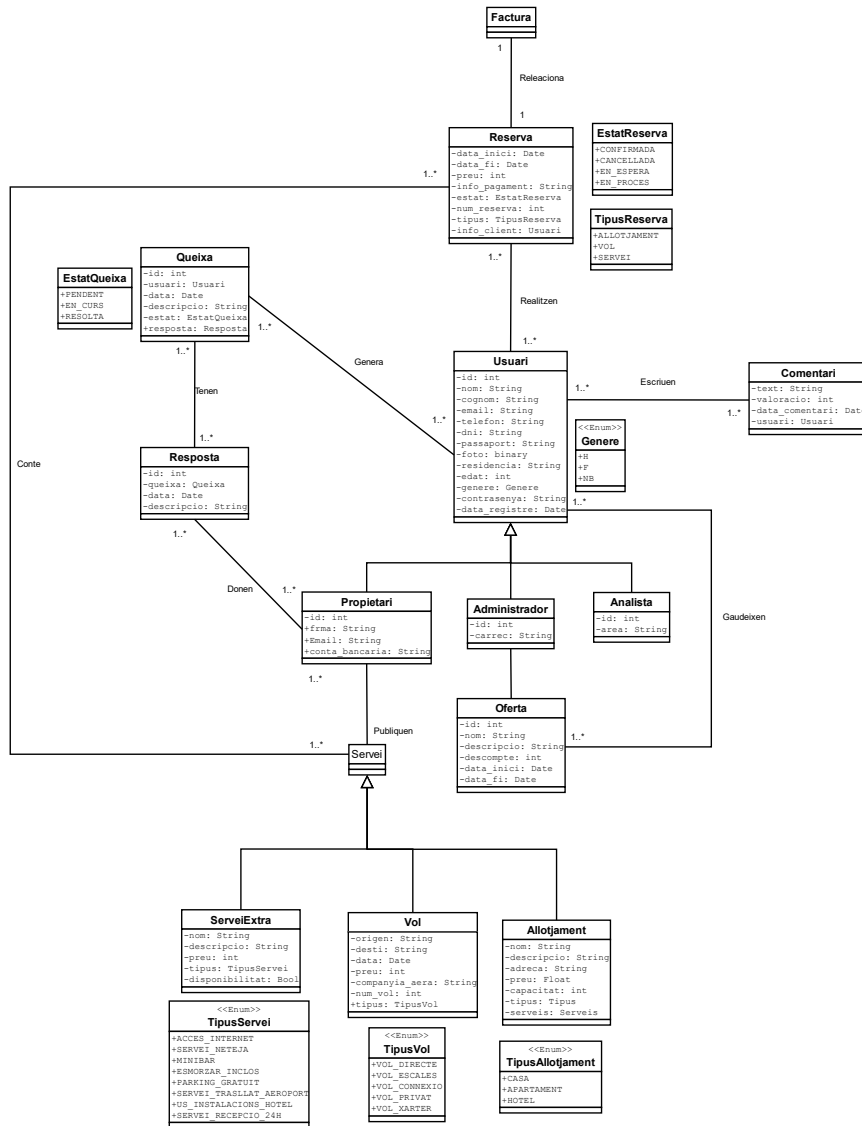
---

### *Analista*

---

calcularMitjana	Allotjament, vol o altre servei, llista de valoracions	Mitjana aritmètica de les valoracions d'un allotjament, vol o altre servei	Analista autenticat	L'administrador rebrà un email amb les valoracions mitjanes.
calcularRendiment	Nul	Estimació del rendiment del sistema	Analista autenticat	L'analista podrà generar un report amb la informació extreta
generarReportsDeRendiment	Estimació del rendiment	Informe de rendiment	Analista autenticat	L'administrador rebrà un email amb

### 4.3.3 Diagrama de classes



Il·lustració 8: Diagrama de classes

El diagrama **anterior** correspon al model del domini.

Un **model de domini** es un model conceptual amb tota la informació per resoldre un problema concret. [16]

### 4.3.4 Subdominis

La idea és, generar 5 microserveis separats, a continuació, s'explicaran els diferents subdominis.

#### SUBDOMINIS

#### EXPLICACIÓ DEL SUBDOMINI

<b>CATÀLEG DE SERVEIS</b>	Aquest és el subdomini principal, funcionarà de catàleg o llistat de serveis.
<b>PERFIL D'USUARI</b>	Aquest subdomini és l'encarregat de gestionar tot el sistema de perfils d'usuari.
<b>SISTEMA DE RESERVA I PASSAREL·LA DE PAGAMENT</b>	Aquest subdomini contindrà totes les accions que tenen com a finalitat generar una reserva i reencaminar a l'usuari a la passarel·la de pagament.
<b>RENDIMENT</b>	Subdomini estrictament lligat al càlcul del rendiment de l'aplicació.
<b>SISTEMA DE COMUNICACIÓ I FEEDBACK</b>	Finalment, l'últim subdomini encarregat de tota l'experiència d'usuari.

#### 4.3.5 Mapatge de funcions amb el seu subdomini

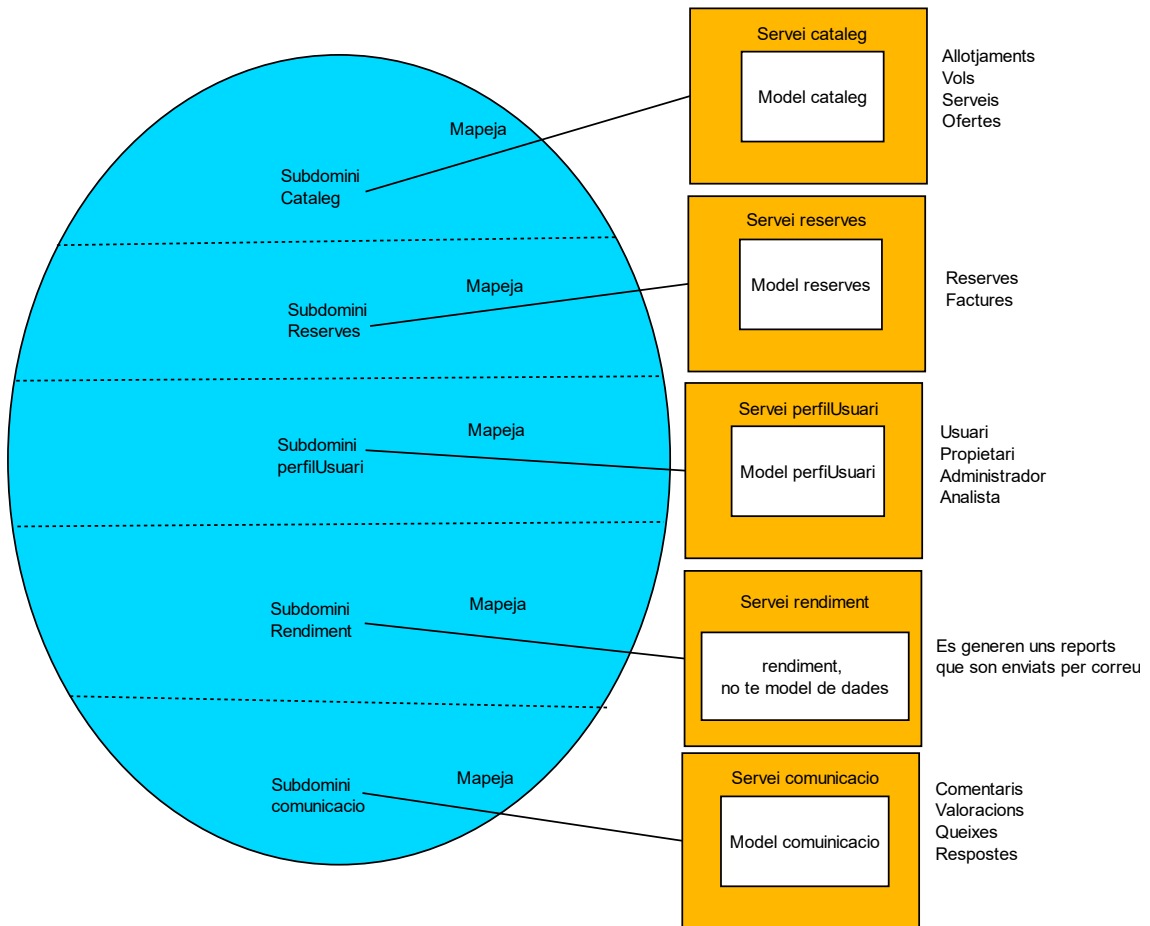
Seguidament, es definirà cada subdomini amb les seves funcionalitats a més a més s'especificaran si seran necessàries col·laboracions externes.

<b>Serveis identificats</b>	<b>Operacions del sistema</b>	<b>Col·laboracions externes</b>
<b>Catàleg de serveis</b>	CRUD d'allotjaments CRUD de vols CRUD de serveis CRUD d'ofertes cercarAllotjament cercarVol veureDetallsDeAllotjament veureDetallsDeVols	Es connecta amb el sistema de reserva i també amb el de comunicació.
<b>Perfil d'usuari</b>	registrarse editarUsuari donarseDeBaixa	
<b>Sistema de reserva i passarel·la de pagament</b>	reservarAllotjament reservarVol reservarServei realitzarPagament editarReserva cancelarReserva aplicarDescompte	La part estrictament associada al pagament es farà mitjançant una passarel·la de pagaments externa.
<b>Rendiment</b>	calcularRendiment generarReportsDeRendiment	Per calcular el rendiment, s'accedeix al catàleg.
<b>Sistema de comunicació i feedback</b>	afegirComentari afegirValoracio afegirQueixa editarQueixa respondreQueixa	Hi ha una col·laboració externa entre els subdominis de Sistema de comunicació i catàleg.



	editarResposta eliminarResposta respondreComentari respondreValoracio calcularMltjana	
--	---	--

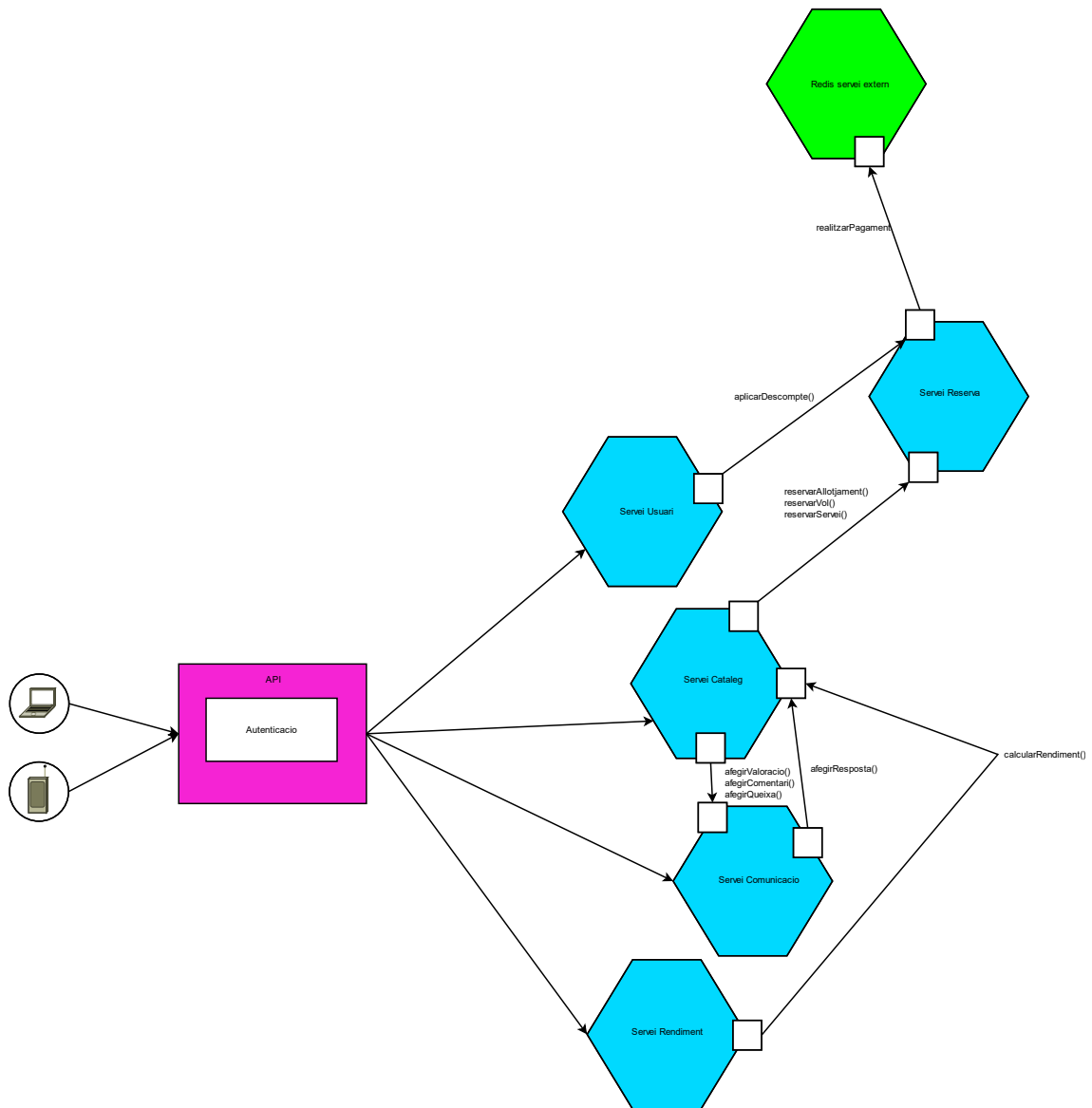
Per sintetitzar tota aquesta informació, s'ha adjuntat el següent diagrama on es pot veure tot el mapatge complet dels diferents subdominis.



**Il·lustració 9: Mapatge complet dels subdominis**

### 4.3.6 Mapa hexagonal

El següent pas és observar un diagrama general amb tots els microserveis i les col·laboracions entre ells.



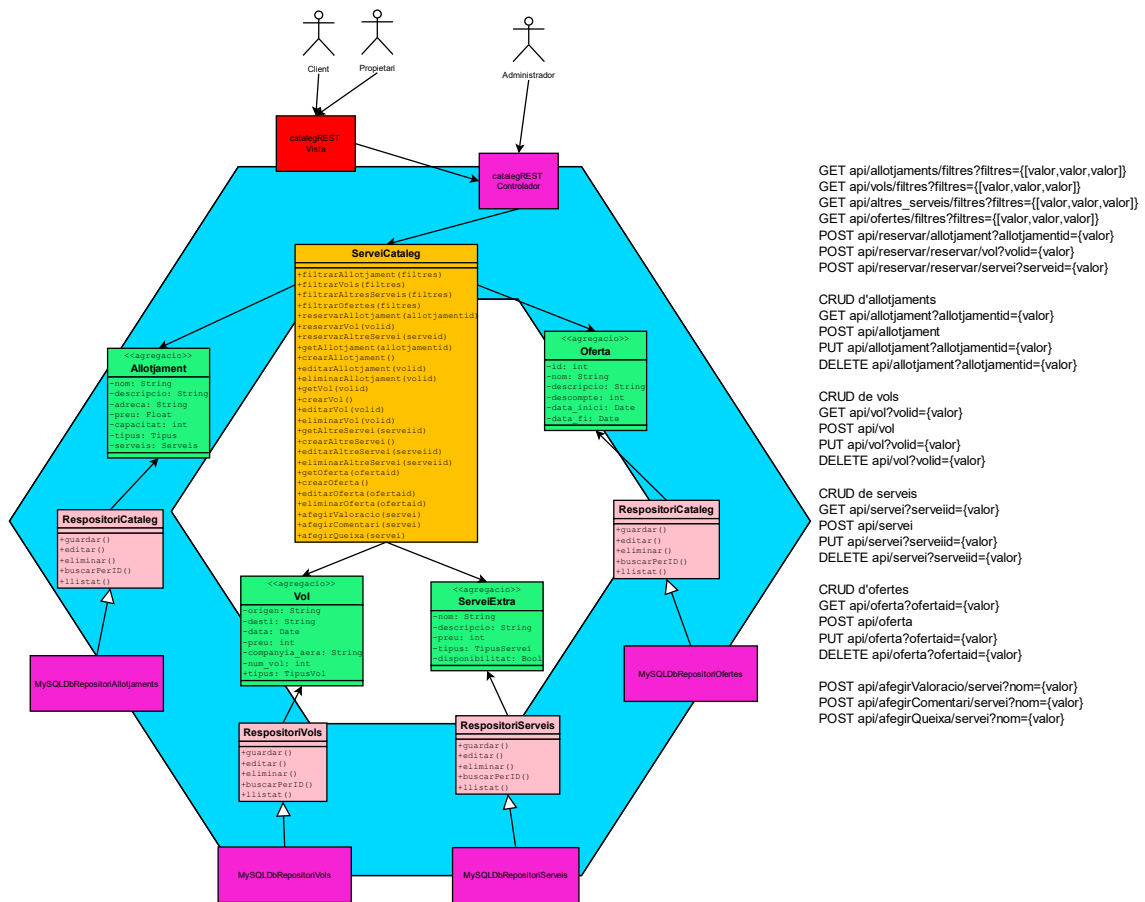
**Il·lustració 10: Diagrama general**

S'ha generat un mapa hexagonal per cada microservei de forma que es pugui veure en general la seva distribució.

---

*Catàleg de serveis:*

---



II-lustració 11: Catàleg de serveis

Les crides són les següents:

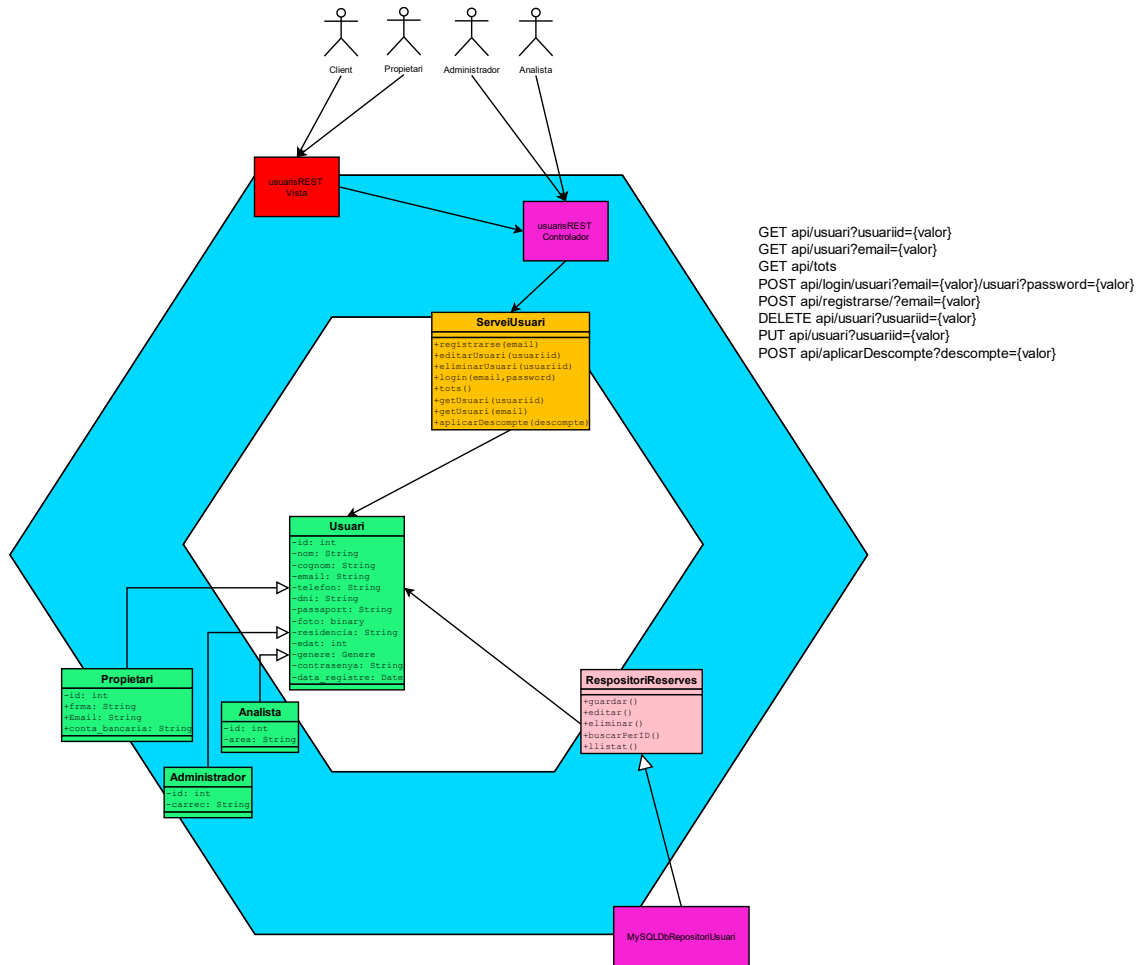
- GET api/allotjament?allotjamentid={valor}, si la crida funciona correctament, retorna un allotjament
- POST api/allotjament, si la crida funciona correctament, crea un allotjament
- PUT api/allotjament?allotjamentid={valor}, si la crida funciona correctament, modifica l'allotjament
- DELETE api/allotjament?allotjamentid={valor}, si la crida funciona correctament, elimina l'allotjament
- GET api/vol?volid={valor}, si la crida funciona correctament, retorna un vol
- POST api/vol, si la crida funciona correctament, crea un vol
- PUT api/vol?volid={valor}, si la crida funciona correctament, modifica el vol
- DELETE api/vol?volid={valor}, si la crida funciona correctament, elimina el vol
- GET api/servei?serveid={valor}, si la crida funciona correctament, retorna un servei
- POST api/servei, si la crida funciona correctament, crea un servei

- PUT api/servei?serveid={valor}, si la crida funciona correctament, modifica el servei
- DELETE api/servei?serveid={valor}, si la crida funciona correctament, elimina el servei
- GET api/oferta?ofertaid={valor}, si la crida funciona correctament, retorna una oferta
- POST api/oferta, si la crida funciona correctament, crea una oferta
- PUT api/oferta?ofertaid={valor}, si la crida funciona correctament, modifica l'oferta
- DELETE api/oferta?ofertaid={valor}, si la crida funciona correctament, elimina l'oferta
- GET api/allotjaments/filtres?filtres={[valor,valor,valor]}, si la crida funciona correctament, el sistema mostrarà una llista d'allotjaments
- GET api/vols/filtres?filtres={[valor,valor,valor]}, si la crida funciona correctament, el sistema mostrarà una llista de vols.
- GET api/altres\_serveis/filtres?filtres={[valor,valor,valor]}, si la crida funciona correctament, el sistema mostrarà una llista de serveis.
- GET api/ofertes/filtres?filtres={[valor,valor,valor]}, si la crida funciona correctament, el sistema mostrarà una llista d'ofertes
- POST api/reservar/allotjament?allotjamentid={valor}, si la crida funciona correctament, envia l'usuari a la passarel·la de reserva.
- POST api/reservar/reservar/vol?volid={valor}, si la crida funciona correctament, envia l'usuari a la passarel·la de reserva.
- POST api/reservar/reservar/servei?serveid={valor}, si la crida funciona correctament, envia l'usuari a la passarel·la de reserva.
- POST api/afegirComentari/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix el nou comentari.
- POST api/afegirValoracio/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix la nova valoració.
- POST api/afegirQueixa/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix la nova queixa.

---

*Perfil d'usuari:*

---

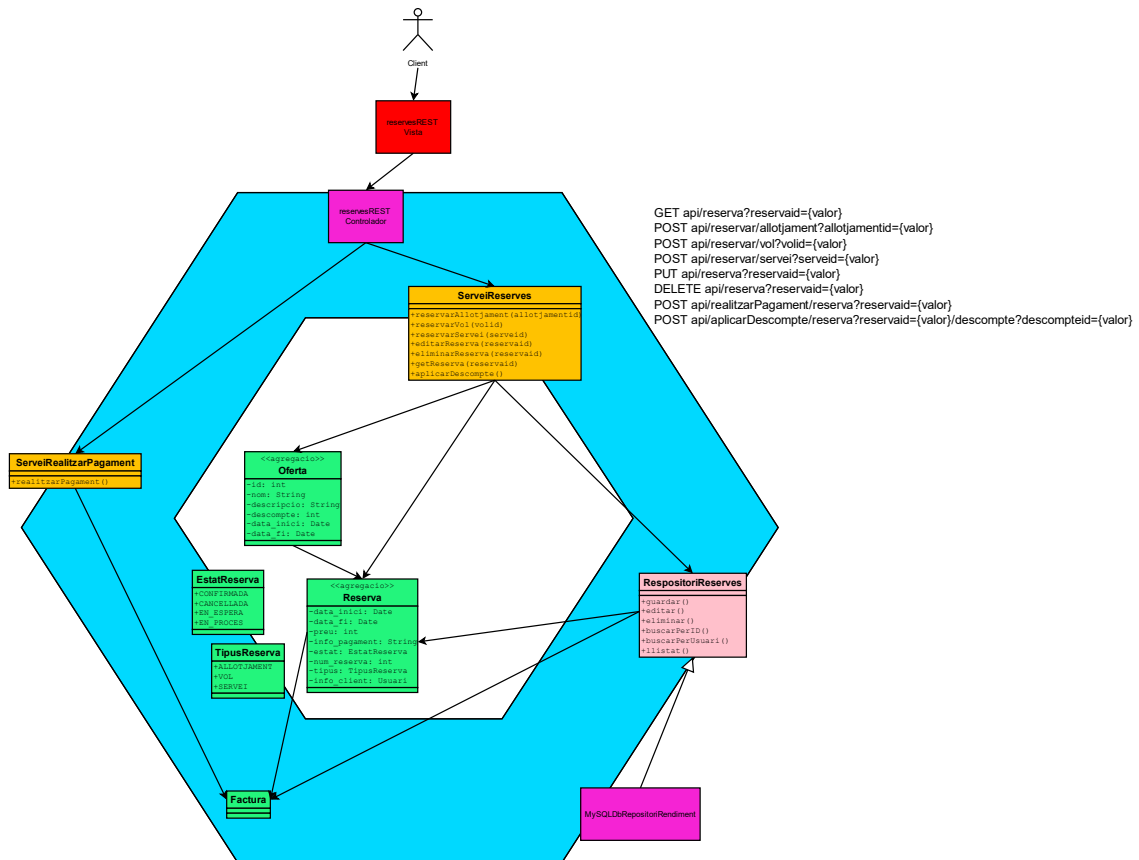


II-lustració 12: Perfil d'usuari

Les crides són les següents:

- GET api/usuari?usuariid={valor}, si la crida funciona correctament, retorna un usuari
- GET api/usuari?email={valor}, si la crida funciona correctament, retorna un usuari
- GET api/tots, si la crida funciona correctament, retorna tots els usuaris
- POST api/login/usuari?email={valor}/usuari?password={valor}, si la crida funciona correctament, permet a l'usuari fer iniciar sessió
- POST api/registrarse/?email={valor}, si la crida funciona correctament, registra a l'usuari al sistema.
- DELETE api/usuari?usuariid={valor}, si la crida funciona correctament, elimina a l'usuari del sistema.
- PUT api/usuari?usuariid={valor}, si la crida funciona correctament, modifica l'usuari
- POST api/aplicarDescompte?descompte={valor}, si la crida funciona correctament, la pròxima reserva tindrà aplicada el descompte.

## Sistema de reserva i passarel·la de pagament:



Il·lustració 13: Sistema de reserva i passarel·la de pagament

Les crides són les següents:

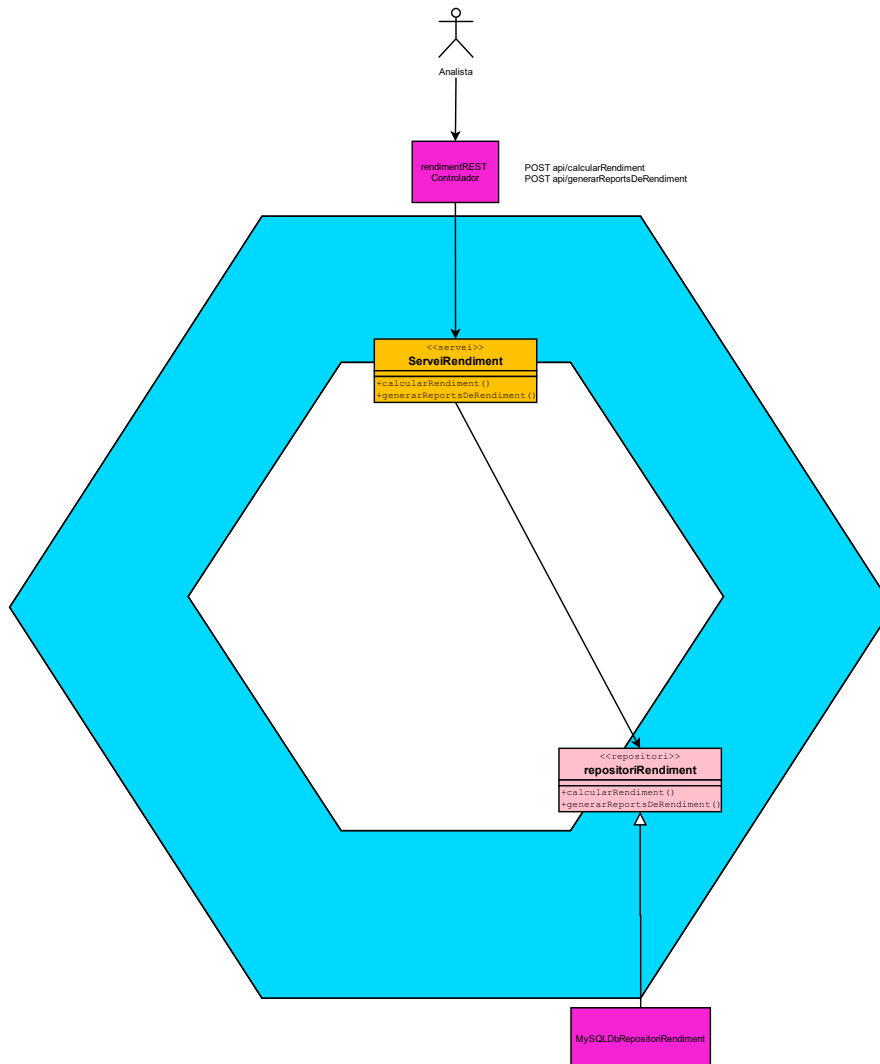
- GET api/reserva?reservaid={valor} si la crida funciona correctament, retorna una reserva
- POST api/reservar/allotjament?allotjamentid={valor}, si la crida funciona correctament, es crea una nova reserva per l'allotjament.
- POST api/reservar/vol?volid={valor}, si la crida funciona correctament, es crea una nova reserva pel vol.
- POST api/reservar/servei?serveid={valor}, si la crida funciona correctament, es crea una nova reserva pel servei.
- PUT api/reserva?reservaid={valor}, si la crida funciona correctament, es modifica la reserva
- DELETE api/reserva?reservaid={valor}, si la crida funciona correctament, s'elimina la reserva
- POST api/realitzarPagament/reserva?reservaid={valor}, si la crida funciona correctament, es transfereix a l'usuari a la passarel·la de pagament.

- POST  
api/aplicarDescompte/reserva?reservaid={valor}/descompte?desc  
ompteid={valor}, si la crida funciona correctament, el preu de la  
reserva es veu modificat.

---

*Rendiment:*

---



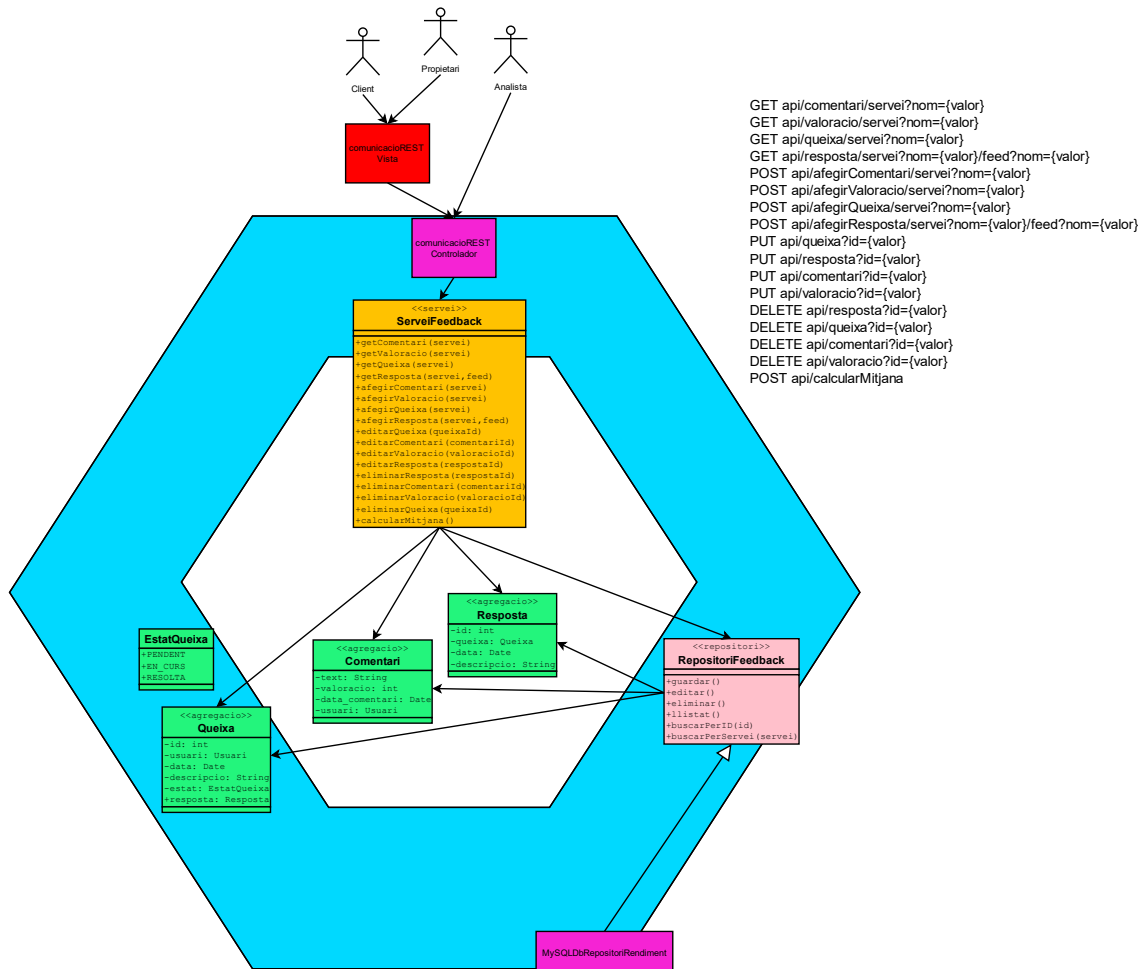
**II·lustració 14: Rendiment**

Les crides són les següents:

- POST api/calcularRendiment, si la crida funciona correctament, retorna el rendiment
- POST api/generarReportsDeRendiment, si la crida funciona correctament, retorna el report de rendiment i l'envia a l'administrador via email.

Tot i que el sistema no té model de dades per el rendiment, l'informe de rendiment es guarda en format JSON a un repositori.

### Sistema de comunicació i feedback:



II-lustració 15: Sistema de comunicació i feedback

Les crides són les següents:

- GET api/comentari/servei?nom={valor}, si la crida funciona correctament, retorna el comentari.
- GET api/valoracio/servei?nom={valor}, si la crida funciona correctament, retorna la valoració.
- GET api/queixa/servei?nom={valor}, si la crida funciona correctament, retorna la queixa.
- GET api/reposta/servei?nom={valor}, si la crida funciona correctament, retorna la resposta.
- POST api/afegirComentari/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix el nou comentari.



- POST api/afegirValoracio/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix la nova valoració.
- POST api/afegirQueixa/servei?nom={valor}, si la crida funciona correctament, el sistema afegeix la nova queixa.
- POST api/afegirResposta/servei?nom={valor}/feed?nom={valor}, si la crida funciona correctament, el sistema afegeix la nova resposta.
- PUT api/queixa?id={valor}, si la crida funciona correctament, la queixa es modifica
- PUT api/resposta?id={valor}, si la crida funciona correctament, la resposta es modifica
- PUT api/comentari?id={valor}, si la crida funciona correctament, el comentari es modifica
- PUT api/valoracio?id={valor}, si la crida funciona correctament, la valoració es modifica
- DELETE api/queixa?id={valor}, si la crida funciona correctament, la queixa s'elimina del sistema.
- DELETE api/resposta?id={valor}, si la crida funciona correctament, la resposta s'elimina del sistema.
- DELETE api/valoracio?id={valor}, si la crida funciona correctament, la valoració s'elimina del sistema.
- DELETE api/comentari?id={valor}, si la crida funciona correctament, el comentari s'elimina del sistema.
- POST api/calcularMitjana, si la crida funciona correctament, es modifica la mitjana aritmètica de valoracions

#### 4.4 Disseny relacional de la base de dades

Avui en dia hi ha dos grans paradigmes de bases de dades, **relacionals i no relacionals**.

Per (coneixement i facilitat) s'utilitzarà una base de dades relacional del tipus SQL, tot sigui dit però és indiferent quin paradigma s'utilitzi la implementació es pot aconseguir indiferentment ja sigui utilitzant un model relacional com un de no relacional.

El disseny relacional de bases de dades **és un procés que es basa a generar una taula per cada entitat del sistema, aquesta contindrà una clau primària (PK) que normalment serà l'ID o l'identificador de l'entitat**, a més a més aquest model relacional contindrà tots els índexs i característiques.

Per tant, el primer pas esdevé identificar la clau primària de cada entitat:

Classes:

- Usuari: ID usuari
- Propietari: ID propietari
- Administrador: ID administrador

- Analista: ID analista
- Allotjament: ID allotjament
- Vol: ID vol
- Servei: ID servei
- Comentari: ID comentari
- Reserva: Número de reserva (clau primària)
- Oferta: ID oferta
- Queixa: ID queixa
- Resposta: ID resposta

Una vegada identificades les claus de cada entitat podem identificar les relacions entre les entitats:

- Els propietaris publiquen allotjaments, vols i altres serveis. Això significa que les entitats "Allotjaments", "Vols" i "Serveis" tenen una clau forana "ID propietari" que apunta a l'entitat "Propietaris".
- Els usuaris poden fer reserves d'allotjaments, vols i altres serveis. Això significa que l'entitat "Reserves" té una clau forana "ID usuari" que apunta a l'entitat "Usuaris" i una clau forana que apunta a l'entitat corresponent.
- Els usuaris poden fer comentaris i valoracions sobre allotjaments, vols i altres serveis. Això significa que l'entitat "Comentaris" té una clau forana "ID usuari" que apunta a l'entitat "Usuaris" i una clau forana que apunta a l'entitat corresponent.
- Els usuaris poden fer queixes sobre allotjaments, vols i altres serveis. Això significa que l'entitat "Queixes" té una clau forana "ID usuari" que apunta a l'entitat "Usuaris" i una clau forana que apunta a l'entitat corresponent.
- Els propietaris poden respondre queixes. Això significa que l'entitat "Resposta" té una clau forana "ID queixa" i "ID propietari" que apunten a les entitats corresponents.
- Ells allotjaments, vols i altres serveis poden tenir aplicada una oferta. Això significa que l'entitat "Oferta" té una clau forana "ID allotjament/vol/servei" apunta a l'entitat corresponent.

Per tant, el disseny relacional quedaria de la següent forma:

- Taula Usuaris: ID usuari (clau primària), Nom, Cognom, Email, Telèfon, DNI, Passaport, Foto, Residencia, Edat, Gènere, Contrasenya, Data de registre.
- Taula Propietaris: ID propietari (clau primària), ID usuari (clau forana), Firma, Serveis, Email, Número de comptabilitat.
- Taula Administradors: ID administrador (clau primària), ID usuari (clau forana), Càrrec.
- Taula Analistes: ID analista (clau primària), ID usuari (clau forana), Àrea d'anàlisi.

- Taula Allotjaments: ID allotjament (clau primària), Nom de l'allotjament, ID propietari (clau forana), Descripció, Adreça, Preu, Capacitat, Tipus d'allotjament, Serveis inclosos.
- Taula Vols: ID vol (clau primària), Número de vol, ID propietari (clau forana), Origen, Destí, Data, Preu, Companyia aèria, Tipus de vol.
- Taula Serveis: ID servei (clau primària), Nom del servei, ID propietari (clau forana), Descripció de servei, Preu, Tipus de servei, Disponibilitat.
- Taula Reserves: Número de reserva (clau primària), ID usuari (clau forana), ID allotjament/vol/servei (clau forana), Data d'inici, Data de finalització, Estat de la reserva (per exemple, pendent, confirmada, cancel·lada), Preu final.
- Taula Comentaris: ID comentari (clau primària), ID usuari (clau forana), ID allotjament/vol/servei (clau forana), Data del comentari, Text del comentari, Valoració (en una escala d'1 a 5).
- Taula Ofertes: ID oferta (clau primària), Nom, Data d'inici, Data de finalització, ID allotjament/vol/servei (clau forana).
- Taula Queixes: ID queixa (clau primària), ID usuari (clau forana), Data, Descripció, Estat, ID resposta (clau forana).
- Taula Respostes: ID resposta (clau primària), ID queixa (clau forana), Data, Descripció

Hi ha taules que en el diagrama de classes i en la descripció no tenen el camp ID, ja que són herències de les seves superclasses, com passa amb Usuari > Propietari > Administrador > Analista.

## 4.5 Disseny en capes

Aquest punt definirà com seran les diferents capes de l'aplicació, el sistema tindrà 3 capes visiblement separades:

- La capa de presentació,
- La capa de lògica de negoci
- La capa de dades

La **capa de presentació** és la interfície que els usuaris utilitzen per interactuar amb l'aplicació, aquesta capa inclou la interfície d'usuari (UI) i la lògica de presentació.

La interfície d'usuari és la part visible de l'aplicació que permet als usuaris veure i interactuar amb les funcions de l'aplicació.

La lògica de presentació gestiona les dades que es mostren a la interfície d'usuari i proporciona una experiència d'usuari amigable i intuïtiva.

La **capa de lògica de negoci** és la capa que gestiona la lògica del negoci i la lògica de processament, aquesta capa inclou la lògica del negoci, que és la lògica que governa la funcionalitat de l'aplicació, i la lògica de processament, que gestiona les dades que flueixen a través de l'aplicació.

La **capa de dades** és la capa que gestiona les dades de l'aplicació. Aquesta capa inclou la base de dades, que és on es guarden les dades de l'aplicació, i el sistema de gestió de dades, que gestiona l'accés i la manipulació de les dades a la base de dades.

#### 4.5.1 Disseny de les capes del sistema

El que es realitzarà en aquest apartat serà una classificació de quines classes entren en cada una de les diferents capes que tindrà el sistema.

##### La capa de presentació:

A grans trets la capa de presentació és bàsicament la part visual i d'interacció amb l'usuari.

L'usuari no interacciona amb les classes unitàries com a tal, ell, però interacciona amb uns arxius anomenats "vistes" (una vista és un arxiu que conté llenguatge de marques) **el sistema contindrà una vista per hexàgon**, totes i cada una d'elles permetran a l'usuari poder interactuar amb la capa de presentació de cada una dels subdominis de l'aplicació.

##### La capa de lògica de negoci:

La capa de lògica, contindrà totes les classes que s'utilitzen perquè el sistema pugui funcionar.

Es generaran un seguit de **classes servei** per tal de poder donar accés a les funcionalitats de totes i cada una de les classes unitàries a més a més **aquestes classes servei permeten ajuntar codi que pugui ser reutilitzat**.

Aquests serveis, però, requereixen un organisme superior (comunament anomenat **controlador**) que sol **encarregar-se de la connexió amb la base de dades (mitjançant el repositori) i amb l'usuari (mitjançant la vista)**, que el que fa és redirigir la crida efectuada cap a la classe o servei que ha d'executar la funcionalitat desitjada en moment d'execució.

##### La capa de dades:

Finalment, la **capa de les dades contindrà totes les funcionalitats per emmagatzemar, tractar, i modificar les dades**.

Aquesta capa conte les classes que tenen contacte amb la base de dades o requereixen persistència de dades, típicament hi ha dues nomenclatures que s'utilitzen per contextualitzar la capa de dades: **els models** (els models són les classes primitives de les dades amb els seus atributs i funcions) i **els repositoris** (els repositoris són les classes amb contacte directe a la base de dades).

**Com a mínim hi ha un model per cada classe que volgués tenir persistència de dades**, aquests models són representacions dels objectes de forma primitiva.

Des del punt de vista de la base de dades, es tenen ara si els repositoris, **hi ha mínim un repositori per taula a la base de dades** que estigui associada a un model.

Aquesta informació és visible en cada diagrama d'arquitectura.

## 4.6 Patrons de disseny

Per tal de generar un bon disseny, és important seguir moltes de les pràctiques de patrons de disseny de programari (també apreses durant el grau) i que permetran a l'equip d'implementació generar un producte seguint un mateix format i fàcilment comprensible.

Aquests patrons ajuden sobretot a abordar problemes comuns i aporten solucions que han estat provades i refinades al llarg del temps, tanmateix, els patrons de disseny promouen la reutilització del codi, la modularitat i l'eficiència, tot mantenint la coherència i la comprensibilitat d'aquest.

Alguns dels patrons de disseny inclouen el patró factoria, el patró observador i el DAO o objecte d'accés a dades, entre altres.

---

### *Patró Factoria*

---

(Factory Method (patrón de diseño), 2023)

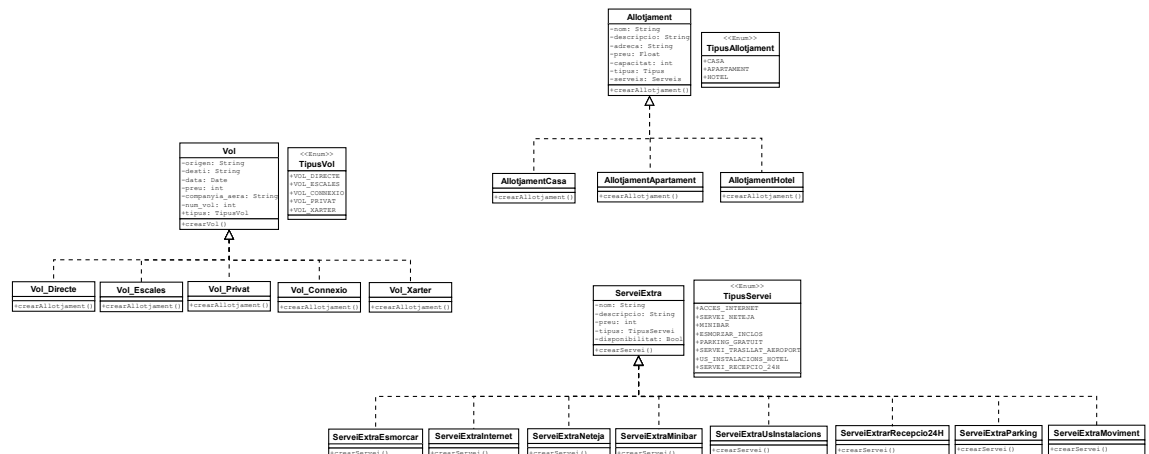
Com el seu nom indica, el patró factoria serveix principalment per gestionar la creació d'objectes dins l'aplicació (factoria d'objectes), sense necessitat que el codi client conegui els detalls concrets de la seva creació.

Es basa en el principi que el codi client, que necessita utilitzar objectes com allotjaments, vols, serveis, ofertes, etc, no ha de conèixer els detalls específics de com aquests objectes es creen i s'inicialitzen.

Aquest patró, també conegut com a "Factory Method" o "Abstract Factory", encapsula tot el codi de creació d'objectes en un sol punt. Això crea una capa d'abstracció que separa la lògica de creació d'objectes de la lògica principal de l'aplicació. D'aquesta forma, la lògica de l'aplicació es pot concentrar en la seva funcionalitat principal sense haver de gestionar la complexitat de la creació d'objectes.

La interfície proporcionada per aquest patró factoria estandarditza la creació d'objectes per a totes les classes. També permet una extensió específica per a cada tipus d'objecte en cas que sigui necessari programar alguna funcionalitat de creació específica. No obstant això, es recomana que la majoria dels mètodes de creació d'objectes siguin comuns, permetent així que totes les parts de l'aplicació utilitzin els mateixos mètodes i homogeneïtzar encara més la implementació.

## Aplicació del patró:



II-lustració 16: Aplicació del patró factory

A tall de resum, s'han creat classes per cada objecte concret dins de la seva interfície, delegant la lògica de la creació a la subclasse pertinent per tal que qualsevol classe que necessita accés pogués crear objectes de qualsevol mena (accedint a la interfície) però sense haver de conèixer la lògica de la creació, a més a més l'aplicació del patró permet tenir lògiques diferents per exemple per separar els diferents vols, amb les seves característiques pròpies.

---

### Patró Observador

---

(Observer (patrón de diseño), 2023)

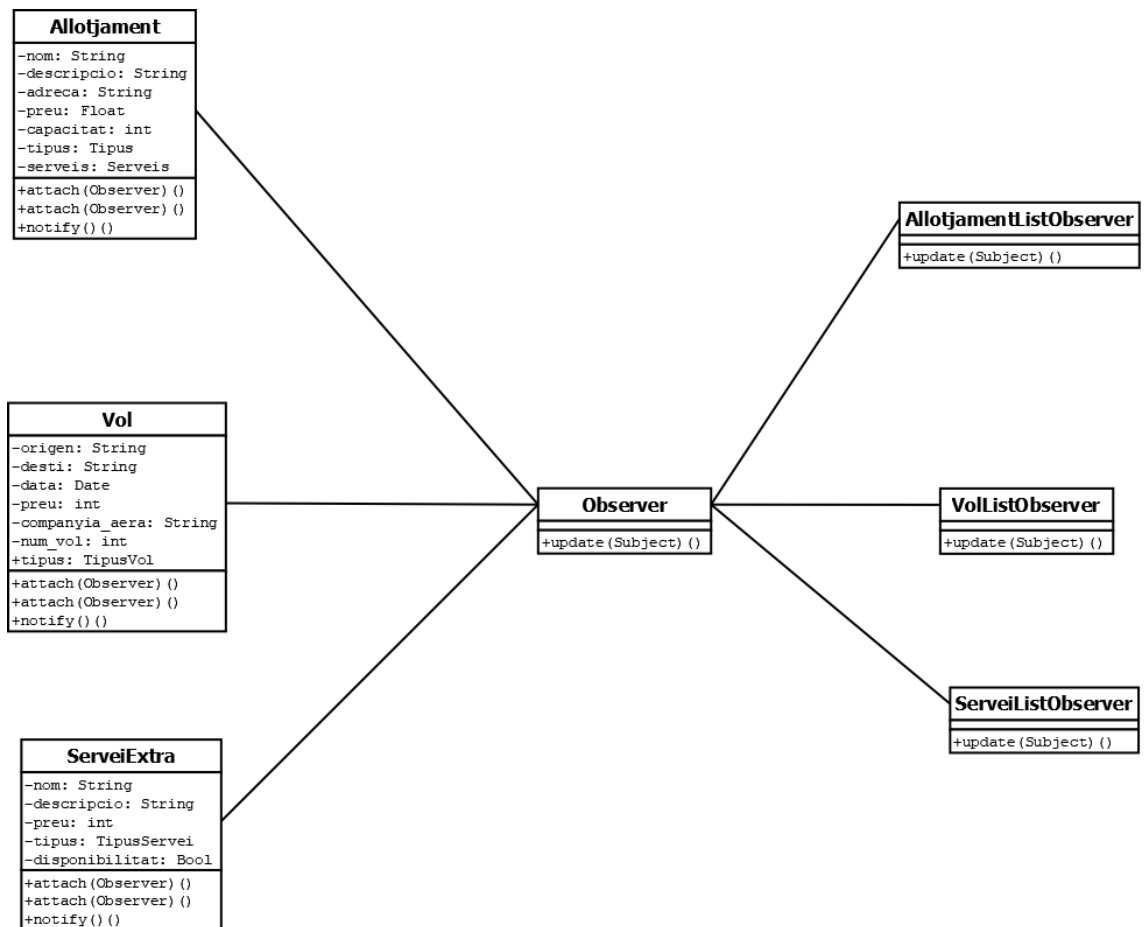
Com el seu nom indica, el patró observador serveix principalment per mantenir a diversos objectes notificats dels canvis que es produeixen en un objecte determinat.

Es basa en el principi que un objecte, conegut com a "subjecte", manté una llista de dependències, conegudes com a "observadors", i els notifica automàticament de qualsevol estat o canvi. Aquests observadors, per la seva part, es poden subscriure i eliminar-se en qualsevol moment.

Aquest patró, també conegut com a "Publisher/Subscriber" en el món del disseny de programari, encapsula tota la lògica de notificació en un sol punt. Això com passa amb el patró factoria genera una capa d'abstracció que separa la lògica de notificació de la lògica de l'aplicació principal. D'aquesta forma, la lògica d'aplicació es pot concentrar en la seva funcionalitat principal sense haver de gestionar la complexitat de la notificació de canvis.

La interfície proporcionada per aquest patró observador estandarditza el mètode de notificació per a tots els objectes.

## Aplicació del patró:



**Il·lustració 17: Aplicació del patró Observer**

Si un allotjament, vol o servei canvia el seu estat el que fa és cridar al seu mètode «notify», que al seu torn crida el mètode “update” de cada “observador” en la seva llista, informant sobre el canvi d'estat. D'aquesta forma s'aconsegueix actualitzar les llistes d'apartaments, vols i serveis de forma automàtica quan per exemple algun deixa d'estar disponible, a més a més d'encapsular el sistema de canvis d'estat de les llistes permetent l'ocultació de la complexitat de la gestió de les subscripcions i notificacions.

---

*Patró DAO o objecte d'accés a dades.*

---

(Objeto de acceso a datos, 2023)

Com el seu nom indica, el patró connector de base de dades serveix principalment per assegurar la comunicació entre la base de dades i l'aplicació.

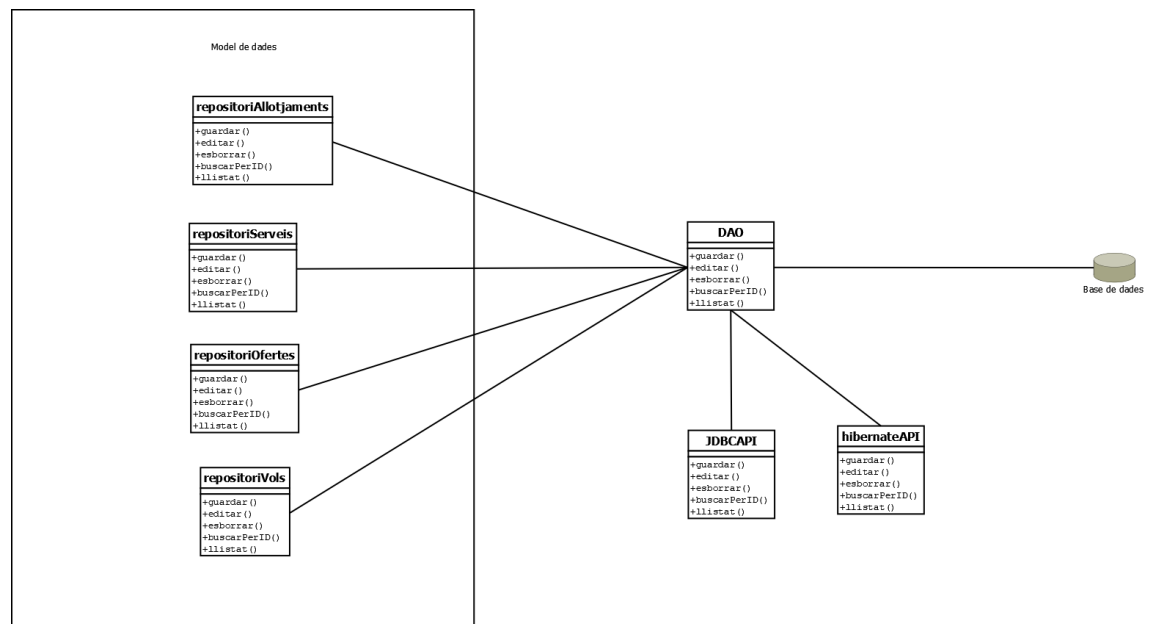
Es basa en el principi que el model de dades, on es creen entitats com allotjaments, vols, ofertes, etc., no ha de conèixer les funcionalitats que

interaccionen amb la base de dades, com ara guardar, editar, esborrar, i llistar.

Aquest patró, també conegut com a "Data Accés Object" (DAO) en el món del disseny de programari, encapsula tot el codi d'accés a la base de dades en un sol punt. Això genera una capa d'abstracció que separa la lògica d'accés a les dades de la lògica de l'aplicació principal.

La interfície proporcionada per aquest patró connector estandarditza l'accés a la base de dades per a totes les classes, permetent tenir un accés molt simple a l'hora d'haver d'emmagatzemar dades en la base de dades, tanmateix, d'aquesta manera, en cas de canvi a la base de dades o al sistema de gestió de bases de dades, l'impacte sobre la resta de l'aplicació serà mínim, ja que el codi d'accés a la base de dades està centralitzat el DAO.

### Aplicació del patró.



**Il·lustració 18: Aplicació del patró DAO**

Com es pot apreciar, s'aïlla la lògica del model de dades de la interfície que connecta amb la base de dades. D'aquesta manera, el model mai depèn del tipus de base de dades que es té per sota; aquest és indiferent del sistema que s'utilitza, ja que farà servir la interfície DAO per connectar-se utilitzant l'API que correspongui, mantenint el sistema separat, un altre punt a tenir en compte, el DAO encapsula les consultes a la base de dades, de manera que el codi que utilitza el DAO no necessita saber com es realitza la consulta a la base de dades, només necessita conèixer les operacions que proporciona el DAO.

## 5. Conclusions



En aquest treball s'ha aconseguit una comprensió profunda del procés de creació d'un projecte d'enginyeria de programari passant exhaustivament per les seves fases principals d'anàlisi i disseny.

**L'experiència ha proporcionat una visió global del procés, des de la planificació inicial fins al disseny, on s'ha tingut l'oportunitat de treballar en profunditat en totes les àrees de desenvolupament de programari de forma continuada.**

L'aplicació pràctica de tot el coneixement adquirit durant el grau d'enginyeria ha estat una part vital d'aquest projecte on s'ha hagut d'aplicar conceptes teòrics en un context real, obtenint així una millor comprensió d'aquests.

En el llarg d'aquest treball s'ha intentat dissenyar una aplicació que en aquest punt podria ser altament implementada de forma força eficaç sense haver de perdre més temps en investigar ni haver d'analitzar o dissenyar més parts del sistema, juntament amb el treball fet, un equip podria agafar aquest document i implementar l'aplicació ViatgesUOC d'una forma que es considera prou eficaç perquè aquesta pogués tenir un racó en el món dels metacercadors.

En conclusió, **una vegada fet el treball, en retrospectiva la seva realització ha estat una oportunitat valuosa per a l'aplicació de les habilitats i coneixements adquirits al llarg del grau** a l'hora, tanmateix, de tenir una millor visió de com esdevé implementar les diferents fases de principi a fi d'un projecte amb aquest domini.

**És possible afirmar, que s'han assolit tots els objectius plantejats a l'inici del projecte, l'aplicació** resultant -interactuant amb el prototip-, sembla que **és senzilla d'utilitzar**, és a dir l'usuari per menys temps en realitzar les cerques, això juntament amb el fet que **l'àrea de comunicació és força més visible**, poden generar una millor experiència d'usuari i funcionar com a punt de partida per començar a implementar una aplicació força maca.

De cara als objectius acadèmics, no fa falta explicar-se en massa profunditat, ja que ja s'han comentat amb anterioritat, només caldria afegir que tot i que sembla una part senzilla de realitzar, no ho és pas, **analitzar i dissenyar una aplicació és possiblement el 60% del fet que una aplicació sigui utilitzable o no en el futur** una vegada implementada.

Com caldria esperar amb un projecte d'aquesta magnitud, **malgrat haver assolit tots els objectius, es considera que sempre hi ha aspectes que es poden millorar**, per exemple, es podria haver intentat **implementar alguna part del projecte** per obtenir una experiència totalment completa.

Però, malgrat haver-hi oportunitat per a la millora, hi ha una gran satisfacció personal amb l'assoliment dels objectius i es té una força creença que aquest projecte ha estat **primer, una experiència**

**d'aprenentatge valuosa de cara a l'autosuperació personal a l'hora de fer front als problemes, segon, un camí increïblement autodidacte de consolidació de tots els coneixements obtinguts amb anterioritat, tercer una forma d'entendre millor el futur professional de l'àrea de l'enginyeria del programari** que ha estat l'àrea de l'itinerari escollit en el grau realitzat del qual aquest projecte esdevé la seva culminació.

En termes de seguiment de la planificació i metodologia al llarg del projecte, en línia general sa seguit la planificació inicial, **al començament del projecte, es va establir una visió global que va servir com a guia per a les fases successives del treball**, generant 2 grans blocs un per cada una de les fases, no obstant això, a mesura que el projecte avançava, **es va veure la necessitat de realitzar algunes modificacions** la finalitat de les quals van esdevenir ajustar-se millor als objectius per tal de dissenyar una aplicació tan completa com sigui possible.

Aquests canvis han estat **principalment canvis de nomenclatura d'apartats, el contingut d'alguns apartats s'ha modificat, i petits canvis de millora de lectura** per tal de fer més comprensible el document final, a més a més **la distribució original de l'apartat 3 s'ha reestructurat** perquè era força complexa d'entendre de la forma que s'havia descrit inicialment i amb ajuda del professor consultor va ser possible canviar-la per tal que fos de la forma més comprensible possible a la part que completa per a l'equip que podria implementar l'aplicació utilitzant el document.

Per altra banda, **amb relació a les línies de treball futur**, un aspecte que no s'ha pogut explorar en aquest treball i queda com a tasca pendent seria la **implementació de l'aplicació** seguint tota la informació presentada, com s'ha comentat en les línies anteriors, de cara a un futur immediat, les principals línies d'acció serien entregar aquest document formal a un equip per tal que ells implementin l'aplicació (**intentant no modificar el contingut**) per tal de saber si el formulat anteriorment en aquest treball és suficient per poder implementar l'aplicació Viatges UOC.

Tanmateix, podria ser una bona acció contactar amb un equip de dissenyadors gràfics per tal que ells acabessin de perfilar el «Look and feel» de l'aplicació per tal d'una vegada implementada, poder afegir l'apartat visual i publicar una primera versió real de l'aplicació.

Finalment, per acabar és important dedicar unes línies en dedicació a la implicació que ha fet durant tot el projecte el professor Josep Maria Camps Riba, ha estat tant un suport on poder preguntar dubtes, com una persona tan propera amb la qual s'ha pogut comentar fins i tot l'estat d'ànim respecte al projecte en certs punts, a grans trets i ara si per acabar, ha estat un plaer realitzar aquest TFG sota la seva tutela.

## 6. Bibliografía

Amb format:

- **Web:** URL i data en que s'ha visitat

- **UML Diagrames de casos d'ús.** (20 de Mar de 2023)  
[http://insjoaquimmir.cat/wiki/index.php/UML\\_Diagrames\\_de\\_casos\\_d%27%C3%BAs](http://insjoaquimmir.cat/wiki/index.php/UML_Diagrames_de_casos_d%27%C3%BAs)
- **Requerimientos Funcionales y No Funcionales** (22 de Mar de 2023). Obtenido de <https://medium.com/@requeridosblog/requerimientos-funcionales-y-no-funcionales-ejemplos-y-tips-aa31cb59b22a>
- **Usabilidad: ¿qué es y cuáles son sus principios?** (26 de Abr de 2023) <https://blogs.uoc.edu/informatica/usabilidad-que-es-y-cuales-son-sus-principios/>
- **Arquitectura de software.** (12 de Mai 2023) Obtenido de [https://es.wikipedia.org/wiki/Arquitectura\\_de\\_software](https://es.wikipedia.org/wiki/Arquitectura_de_software)
- **Aplicación monolítica.** (15 de Mai 2023) [https://es.wikipedia.org/wiki/Aplicaci%C3%B3n\\_monol%C3%ADtica#:~:text=En%20ingenier%C3%ADa%20de%20software%2C%20una,y%20sobre%20una%20misma%20plataforma.](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_monol%C3%ADtica#:~:text=En%20ingenier%C3%ADa%20de%20software%2C%20una,y%20sobre%20una%20misma%20plataforma.)
- **Cliente-servidor.** (15 de Mai 2023) <https://es.wikipedia.org/wiki/Cliente-servidor>
- **Ventajas e inconvenientes de la arquitectura cliente/servidor.** (15 de Mai 2023) <http://somebooks.es/ventajas-e-inconvenientes-de-la-arquitectura-clienteservidor/>
- **Arquitectura de microservicios.** (15 de Mai 2023) [https://es.wikipedia.org/wiki/Arquitectura\\_de\\_microservicios](https://es.wikipedia.org/wiki/Arquitectura_de_microservicios)
- **Arquitectura de microservicios: qué es, ventajas y desventajas.** (15 de Mai 2023) <https://decidesoluciones.es/arquitectura-de-microservicios/>
- **Comparación entre la arquitectura monolítica y la arquitectura de microservicios.** (16 de Mai 2023) <https://www.atlassian.com/es/microservices/microservices-architecture/microservices-vs-monolith>
- **Arquitectura orientada a servicios.** (16 de Mai 2023) [https://es.wikipedia.org/wiki/Arquitectura\\_orientada\\_a\\_servicios](https://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios)
- **SOA Ventajas y Desventajas.** (16 de Mai 2023) <http://soa-fpuna.blogspot.com/2011/11/ventajas-y-desventajas.html>
- **Serverless computing.** (17 de Mai 2023) [https://es.wikipedia.org/wiki/Serverless\\_computing](https://es.wikipedia.org/wiki/Serverless_computing)
- **¿Qué es el Serverless Computing?: ventajas y desventajas** (18 de Mai 2023) <https://blog.nubosperta.com/qu%C3%A9-es-el-serverless-computing>

- **Pros y contras de usar de Serverless para tus desarrollos.** (19 de Mai 2023) Obtenido de <https://bokokode.com/posts/pros-y-contras-de-usar-de-serverless-para-tus-desarrollos>
- **Modelo de dominio** (20 de Mai 2023) [https://es.wikipedia.org/wiki/Modelo\\_de\\_dominio](https://es.wikipedia.org/wiki/Modelo_de_dominio)
- **Abstract Factory.** (22 de Mai 2023) [https://es.wikipedia.org/wiki/Abstract\\_Factory](https://es.wikipedia.org/wiki/Abstract_Factory)
- **Factory Method** (patrón de diseño).(22 de Mai 2023) [https://es.wikipedia.org/wiki/Factory\\_Method\\_\(patr%C3%B3n\\_de\\_dise%C3%B1o\)](https://es.wikipedia.org/wiki/Factory_Method_(patr%C3%B3n_de_dise%C3%B1o))
- **Observer** (patrón de diseño). (24 de Mai 2023) Obtenido de [https://es.wikipedia.org/wiki/Observer\\_\(patr%C3%B3n\\_de\\_dise%C3%B1o\)](https://es.wikipedia.org/wiki/Observer_(patr%C3%B3n_de_dise%C3%B1o))
- **Objeto de acceso a datos.** (24 de Mai 2023) [https://es.wikipedia.org/wiki/Objeto\\_de\\_acceso\\_a\\_datos](https://es.wikipedia.org/wiki/Objeto_de_acceso_a_datos)
- **Fundamento de las bases de datos: Modelo entidad-relación** (25 de Mai 2023) <https://www.genbeta.com/desarrollo/fundamento-de-las-bases-de-datos-modelo-entidad-relacion>
- **Tots els mòduls, PACs i practiques de les diferents assignatures de la UOC** (De principi a fi, no hi ha una data concreta)

## 7. Annexos

### 7.1 Entitats, relacions, classes i funcions

#### Entitats del sistema.

Les entitats són objectes del món real o conceptual que es poden identificar amb un únic nom. Es poden representar per objectes creats a partir de les classes.[21]

Les entitats del sistema són les següents:

Usuaris > representa els usuaris que utilitzen l'aplicació.

- ID usuari: identificador únic de l'usuari.
- Nom: nom de l'usuari.
- Cognom: cognom de l'usuari.
- Email: adreça de correu electrònic de l'usuari.
- Telèfon: número de telèfon de l'usuari.
- DNI: número de document d'identitat de l'usuari.
- Passaport: número de passaport de l'usuari.
- Foto: foto de perfil de l'usuari.
- Residencia: país de residència de l'usuari.
- Edat: edat de l'usuari.
- Gènere: gènere de l'usuari.
- Contrasenya: contrasenya per accedir a l'aplicació.

- Data de registre: data en què l'usuari es va registrar a l'aplicació.

Propietaris > Entitat que representa els propietaris:

- ID propietari: identificador únic del propietari.
- Firma: firma del propietari.
- Serveis: serveis oferts pel propietari.
- Email: adreça de correu electrònic del propietari.
- Número de comptabilitat: número de compte bancari del propietari.

Administradors > Entitat que representa al administradors del sistema.

- ID administrador: identificador únic de l'administrador.
- Càrrec: càrrec de l'administrador dins del sistema.

Analistes > Entitat que representa a l'encarregat de recollir dades genera anàlisis de l'ús del sistema. (Tots els camps de l'usuari més un identificador d'analista)

- ID analista: identificador únic per cada analista.
- Àrea d'anàlisi: àrea de l'aplicació que l'analista supervisa.

Allotjaments > Entitat que representa els llocs i allotjaments disponibles per als clients per reservar.

- Nom: nom de l'allotjament.
- Descripció: descripció de l'allotjament.
- Adreça: adreça de l'allotjament.
- Preu: preu de l'allotjament per nit.
- Capacitat: capacitat de persones de l'allotjament.
- Tipus d'allotjament: tipus d'allotjament (p. ex., apartament, casa, habitació d'hotel, etc.).
- Serveis inclosos: serveis inclosos en l'allotjament (p. ex., accés a internet, servei de neteja, etc.).

Vols > Entitat que representa els viatges disponibles per reservar.

- Origen: lloc de sortida del vol.
- Destí: lloc d'arribada del vol.
- Data: data del vol.
- Preu: preu del vol.
- Companyia aèria: nom de la companyia aèria que ofereix el vol.
- Número de vol: número del vol.
- Tipus de vol: tipus de vol (p. ex., directe, amb escales, etc.).

Serveis > Entitat que representa els serveis addicionals, com ara excursions, transport local, reserva de restaurants, etc., oferts per un propietari.

- Nom: nom del servei.
- Descripció: descripció del servei.
- Preu: preu del servei.
- Tipus de servei: tipus de servei (p. ex., excursió, transport, reserva de restaurant, etc.).
- Disponibilitat: disponibilitat del servei en una data determinada.

Comentaris > Entitat que representa les opinions dels usuaris sobre els allotjaments, vols i altres serveis després de la seva estada o ús.

- Text: text del comentari.
- Valoració: puntuació del comentari (per exemple, d'1 a 5 estrelles).
- Data del comentari: data en què l'usuari va fer el comentari.
- Usuari que ha fet el comentari: nom d'usuari de la persona que ha fet el comentari.

Reserves > Entitat que representa un registre de la sol·licitud d'un usuari per a un allotjament, vol o servei, que inclou les dates i la informació de pagament.

- Data d'inici: data d'inici de la reserva.
- Data de finalització: data de finalització de la reserva.
- Preu: preu total de la reserva.
- Informació de pagament: informació de pagament de la reserva.
- Estat de la reserva: estat actual de la reserva (p. ex., confirmada, cancel·lada, en espera, etc.).
- Número de reserva: número únic que identifica la reserva.
- Tipus de reserva: tipus de reserva (p. ex., allotjament, vol, servei, etc.).
- Informació del client: informació del client que ha fet la reserva (p. ex., nom, adreça de correu electrònic, número de telèfon, etc.).

Ofertes i descomptes: > Entitat que representa les ofertes als usuaris per incentivar la reserva de serveis a través de la plataforma.

- Nom: nom de l'oferta o descompte.
- Descripció: descripció de l'oferta o descompte.
- Percentatge de descompte: percentatge de descompte aplicat a la reserva.
- Data d'inici de l'oferta: data en què comença l'oferta o el descompte.
- Data de finalització de l'oferta: data en què acaba l'oferta o el descompte.

Queixes > Entitat que representa les queixes que els usuaris poden presentar sobre algun aspecte del servei o de l'experiència d'ús de l'aplicació.

- ID: identificador únic de la queixa.
- Usuari: identificador de l'usuari que presenta la queixa.
- Data: data en què s'ha presentat la queixa.
- Descripció: descripció detallada de la queixa.
- Estat: estat actual de la queixa (per exemple, pendent, en curs, resolta).
- Resposta: resposta del sistema o de l'administrador a la queixa presentada.

Resposta > Entitat que representa les respostes associades a les queixes que els usuaris poden presentar sobre algun aspecte del servei o de l'experiència d'ús de l'aplicació.

- ID: identificador únic de la resposta.
- Queixa: Queixa relacionada.
- Data: data en què s'ha enviat la resposta.
- Descripció: descripció detallada de la resposta.

### **Relacions entre entitats.**

Les relacions, són associacions entre entitats que requereixen dependències d'altres entitats, per tant, es genera una relació entre ambdues entitats.

Les relacions entre entitats són les següents:

- Un usuari pot ser un propietari o un client, però també pots ser les dues coses alhora.
- Un usuari pot realitzar una o més d'una reserva i cada reserva només pot estar relacionada amb un usuari.
- Un propietari pot oferir molts serveis diferents, i cada servei pot ser ofert per múltiples propietaris.
- Una reserva està associada a un sol allotjament, vol o servei.
- Un allotjament pot ser publicat per un o diversos propietaris, i un propietari pot publicar diversos allotjaments.
- Cada allotjament té diversos serveis addicionals que es poden oferir als clients.
- Els usuaris poden fer comentaris i valoracions sobre els allotjaments, vols i altres serveis.
- Un allotjament pot tenir molts comentaris diferents, i cada comentari només pot estar associat a un allotjament.
- Els clients poden fer reserves per allotjaments, vols i altres serveis oferts per un propietari. Les reserves estan relacionades amb un usuari i un allotjament, vol o servei.
- Un vol pot ser operat per moltes companyies aèries diferents, i cada companyia aèria pot oferir molts vols diferents.
- Un analista pot supervisar moltes àrees diferents de l'aplicació, i cada àrea només pot ser supervisada per un únic analista.
- Una reserva està relacionada amb una sola factura i una factura només està associada amb una reserva.

### **Classes del sistema**

Les classes són com plantilles que permeten definir les característiques i comportaments d'un grup d'objectes relacionats.

A partir de les entitats del sistema es poden extreure les classes principals d'aquest, el més senzill i el que s'ha realitzat ha sigut una classe per entitat, de forma que el resultat és el següent:

Classes:

1. Usuari
2. Propietari
3. Administrador
4. Analista
5. Allotjament
6. Vol
7. Servei
8. Comentari
9. Reserva
10. Oferta
11. Queixa
12. Resposta

### **Funcions del sistema**

Les funcions són blocs de codi que acompleixen una tasca específica i concreta.

Totes les classes tenen:

- Constructor: per crear una nova instància de la classe, amb els paràmetres necessaris
- Getters i setters: per accedir i modificar les propietats de l'objecte

### **Classe Usuari**

- afegir\_usuari: Mètode que permet afegir usuaris al sistema.
- eliminar\_usuari: Mètode que permet eliminar usuaris del sistema.
- modificar\_usuari: Mètode que permet modificar usuaris del sistema.
- obtenir\_usuari: Mètode que permet obtenir un usuari concret.
- obtenir\_usuaris: Mètode que permet obtenir tots els usuaris en el sistema.

### **Classe Propietari**

- afegir\_propietari: Mètode que permet afegir propietaris al sistema.
- actualitzar\_propietari: Mètode que permet eliminar propietaris del sistema.
- eliminar\_propietari: Mètode que permet modificar propietaris del sistema.



- obtenir\_propietari: Mètode que permet obtenir un propietari concret.
- obtenir\_propietaris: Mètode que permet obtenir tots els propietaris en el sistema.

### **Classe Administrador**

- afegir\_administrador: Mètode que permet afegir administradors al sistema.
- actualitzar\_administrador: Mètode que permet eliminar administradors del sistema.
- eliminar\_administrador: Mètode que permet modificar administradors del sistema.
- obtenir\_administrador: Mètode que permet obtenir un administrador concret.
- obtenir\_administradors: Mètode que permet obtenir tots els administradors en el sistema.

### **Classe Analista**

- afegir\_analista: Mètode que permet afegir analistes al sistema.
- actualitzar\_analista: Mètode que permet eliminar analistes del sistema.
- eliminar\_analista: Mètode que permet modificar analistes del sistema.
- obtenir\_analista: Mètode que permet obtenir un analista concret.
- obtenir\_analistes: Mètode que permet obtenir tots els analistes en el sistema.

### **Classe Allotjament**

- afegir\_allotjament: Mètode que permet afegir allotjaments al sistema.
- actualitzar\_allotjament: Mètode que permet eliminar allotjaments del sistema.
- eliminar\_allotjament: Mètode que permet modificar els allotjaments del sistema.
- obtenir\_allotjament: Mètode que permet obtenir un allotjament concret.
- obtenir\_allotjaments: Mètode que permet obtenir tots els allotjaments en el sistema.

### **Classe Vol**

- afegir\_vol: Mètode que permet afegir vols al sistema.
- actualitzar\_vol: Mètode que permet eliminar vols del sistema.
- eliminar\_vol: Mètode que permet modificar els vols del sistema.
- obtenir\_vol: Mètode que permet obtenir un vol concret.

- obtenir\_vols: Mètode que permet obtenir tots els vols en el sistema.

### **Classe Servei (Per servei s'entén qualsevol servei que no sigui ni allotjament ni vol)**

- afegir\_servei: Mètode que permet afegir serveis al sistema.
- actualitzar\_servei: Mètode que permet eliminar serveis del sistema.
- eliminar\_servei: Mètode que permet modificar els serveis del sistema.
- obtenir\_servei: Mètode que permet obtenir un servei concret.
- obtenir\_serveis: Mètode que permet obtenir tots els serveis en el sistema.

### **Classe Comentari**

- afegir\_comentari: Mètode que permet afegir comentaris al sistema.
- actualitzar\_comentari: Mètode que permet eliminar comentaris del sistema.
- eliminar\_comentari: Mètode que permet modificar els comentaris del sistema.
- obtenir\_comentari: Mètode que permet obtenir un comentari concret.
- obtenir\_comentaris: Mètode que permet obtenir tots els comentaris en el sistema.

### **Classe Oferta**

- afegir\_oferta: Mètode que permet afegir ofertes al sistema.
- actualitzar\_oferta: Mètode que permet eliminar ofertes del sistema.
- eliminar\_oferta: Mètode que permet modificar les ofertes del sistema.
- obtenir\_oferta: Mètode que permet obtenir una oferta concret.
- obtenir\_ofertes: Mètode que permet obtenir totes les ofertes en el sistema.

### **Classe Reserva**

- afegir\_reserva: Mètode que permet afegir reserves al sistema.
- actualitzar\_reserva: Mètode que permet eliminar reserves del sistema.
- eliminar\_reserva: Mètode que permet modificar les reserves del sistema.
- obtenir\_reserva: Mètode que permet obtenir una reserva concreta.
- obtenir\_reserves: Mètode que permet obtenir totes les reserves en el sistema.

### **Classe Queixa**

- afegir\_queixa: Mètode que permet afegir queixes al sistema.

- actualitzar\_queixa: Mètode que permet eliminar queixes del sistema.
- eliminar\_queixa: Mètode que permet modificar les queixes del sistema.
- obtenir\_queixa: Mètode que permet obtenir una queixa concreta.
- obtenir\_queixes: Mètode que permet obtenir totes les queixes en el sistema.

### **Classe Resposta**

- afegir\_resposta: Mètode que permet afegir respostes al sistema.
- actualitzar\_resposta: Mètode que permet eliminar respostes del sistema.
- eliminar\_resposta: Mètode que permet modificar les respostes del sistema.
- obtenir\_resposta: Mètode que permet obtenir una resposta concreta.
- obtenir\_respostes: Mètode que permet obtenir totes les respostes en el sistema.