



HITAWOD

El negoci de CrossFit, al següent nivell  
Memòria Treball de Fi de Grau d'Enginyeria Informàtica

Un projecte de:


*bisual*



Ivan Aguilar Nieto  
Pol Ribas Pibernat

1. Introducció, motivacions, propòsit i objectius del projecte	5
2. Estudi de viabilitat	6
3. Metodologia	9
3.1. Metodologies Agile	9
3.1.1. Fases de les metodologies Agile	9
3.2. Metodologia incremental	9
4. Planificació	10
4.1. Paquets de treball	11
4.2. Camí crític	15
4.3. Diagrama de Gantt	16
5. Marc de treball i conceptes previs	18
6. Requisits del sistema	19
7. Estudis i decisions	30
7.1. Backend	31
7.2. Frontend Web	31
7.3. Frontend mòbil	31
7.4. Pagaments automàtics	32
7.5. Notificacions push i correu electrònic	32
8. Anàlisi i disseny del sistema	33
8.1. Diagrames de casos d'ús	33
8.2. Diagrama de base de dades	36
8.3. Entitats	37
8.4. Taules relacionals	40
8.5. Procediments més importants del sistema	40
8.5.1. Alta nova companyia	41
8.5.2. Atleta configura mètode de pagament	42
8.5.3. Autenticació i Autorització API Rest	43
8.5.4. Angular Dashboard Business Intelligence Api Call	44
8.5.5. Generar sessions d'entrenament a partir d'una plantilla	45
8.5.6. Apuntar un usuari a una sessió	46
8.5.7. Nova subscripció d'atleta	47
8.5.8. Atleta canvia de tarifa	48
8.5.9. Empresa canvia de tarifa	49
9. Implementació i proves	50
9.1. Frameworks utilitzats	50
9.1.1. Laravel	50

9.1.1.1. Middleware	51
9.1.1.2. Migrations	51
9.1.1.3. Notifications	51
9.1.1.4. Task Scheduling	51
9.1.1.5. Laravel Cashier	52
9.1.2. Angular	52
9.1.2.1. Serveis	53
9.1.2.2. Interceptors	53
9.1.2.3. Route Guards	54
9.1.3. Flutter	54
9.1.3.1. Providers	54
9.1.3.2. Blocs	55
9.2. Solucions als procediments més complicats	56
9.2.1. Alta nova companyia	56
9.2.2. Atleta configura mètode de pagament	57
9.2.3. Autenticació i Autorització de l'API Rest	58
9.2.4. Angular Dashboard Business Intelligence Api Call	60
9.2.5. Generar sessions d'entrenament a partir d'una plantilla	61
9.2.6. Apuntar un usuari a una sessió d'entrenament	62
9.2.7. Llegir webhooks de Stripe	63
9.2.8. Nova subscripció d'atleta	64
9.2.9. Atleta canvia de tarifa	65
9.2.10. Empresa canvia de tarifa	66
9.2.11. Càlcul de la puntuació d'un benchmark	66
9.3. Procediments en background	67
9.4. Altres qüestions interessants	68
10. Implantació i resultats	69
10.1. Pantalles	69
10.1.1. Client Web Angular	69
10.1.1.1. Pàgina de login	69
10.1.1.2. Pàgina de registre	70
10.1.1.3. Dashboard	71
10.1.1.4. Selector de subscripcions	72
10.1.1.5. Pàgina de calendari	73
10.1.1.6. Pàgina de notícies	75
10.1.1.7. Pàgina de subscripcions	76
10.1.1.8. Pàgina de workouts	79
10.1.1.9. Pàgina de facturació	81
10.1.1.10. Pàgina de negoci	82
10.1.2. Client Mòbil Flutter	86
10.1.2.1. Pàgina de login	86



10.1.2.2. Pantalla d'inici	86
10.1.2.3. Selector de subscripcions i historial	87
10.1.2.4. Calendari i sessions	88
10.1.2.5. Usuari	89
10.1.2.6. Facturació, subscripcions i box orders	89
10.1.2.7. Workouts	91
10.1.2.8. Exercicis	92
10.1.2.9. Benchmarks d'un workout	93
10.1.2.10. Notícies	94
10.2. Producció i deploy	94
10.3. Legalitat	95
11. Conclusions	97
11.1. Resultat i comparativa amb la competència	97
12. Repartiment de feina	100
13. Bibliografia	102
14. Treball futur	103
14.1. Continuous Integration	103
14.2. Afegir noves característiques	103
14.3. Llançament al mercat i venda	104



# 1. Introducció, motivacions, propòsit i objectius del projecte

Aquest és un projecte que està creat per a millorar la gestió del negoci del Crossfit, tant des de la vessant de l'empresari com de l'atleta. El Crossfit és una disciplina d'entrenament físic compost d'exercicis funcionals, executats a alta intensitat i constantment substituïts.

L'objectiu que tenim amb aquest projecte és el de desenvolupar el millor software de gestió del negoci del Crossfit del mercat i fer-ne una posterior campanya de màrqueting i venda.

Després d'uns mesos practicant Crossfit i assistint a un centre certificat de forma regular, vam fixar-nos, tot parlant amb alguns propietaris, que havien de fer malabarismes entre les diferents eines de gestió que utilitzaven. Per un costat, una aplicació per gestionar l'aforament de les sessions, per l'altre un programa de facturació i comptabilitat, cada atleta utilitza la seva pròpia aplicació per fer un seguiment de la seva evolució física, utilitzen xarxes socials per comunicar-se amb els atletes inscrits... I tot, sense una unificació i centralització que vam detectar que podria millorar, i molt, la gestió del seu negoci simplificant molts processos repetitius que feien dia a dia.

Vam informar-nos sobre les diferents solucions en cada àmbit que hi havia disponibles al mercat, ja que podien existir alternatives que funcionessin millor que la solució del negoci que vam consultar. La conclusió que en vam treure, és que hi havia bones solucions però que es centraven en una àrea del negoci molt concreta (p.e. facturació) i que a més, no estaven adaptades ni pensades per al Crossfit, sinó més pel negoci del gimnàs, que tot i que ho puguin semblar, no són el mateix.

El Crossfit és una disciplina que està guanyant molta popularitat i vam decidir crear un software que aglutinés solucions per a totes les necessitats que hi podien haver, tant per a administradors del negoci com per a atletes, que vam decidir anomenar Hit a Wod.

Volíem un software que no requerís d'instal·lació, que es pagués en mensualitats com un servei, que per als administradors els permetés gestionar la facturació, la tarificació amb totes les seves variants, sistemes de pagament automàtics, calendari de sessions i entrenaments per als entrenadors, als atletes apuntar-se a les diferents classes, rebre alertes, portar un seguiment de la seva evolució i rendiment sense haver d'estar lligat a un centre, entre altres.

Aquest projecte es crea des de zero sota un altre projecte que tenim en marxa anomenat Bisual, una plataforma que ens permet crear-ne d'altres i així poder-se retroalimentar econòmicament entre ells.

Aquest és un projecte desenvolupat en un equip de dues persones, on el treball fet per cadascun quedarà detallat en una secció extra que hem afegit al final del document.

## 2. Estudi de viabilitat

Per dur a terme el projecte no es precisa cap requeriment tecnològic especial, però sí tenir a disposició una arquitectura de servidors web que es puguin contractar.

En quant a la viabilitat econòmica del projecte, durant la part del desenvolupament no hem necessitat recursos econòmics ja que tot s'ha realitzat en producció local.

També hem realitzat una planificació econòmica per a diferents casos per quan l'aplicació estigui en producció, així com la quantitat d'empreses que hauríem de tenir per tal que el projecte fos viable econòmicament.

Hem considerat diverses fases en quant al desenvolupament de l'economia del projecte:

- **Fase inicial:** Útil per poder veure les despeses essencials. Hem considerat una sola empresa subscripta al nostre producte.
- **Fase mitjana:** Representa un moment de bona expansió a nivell de territori nacional. Hem agafat de referència l'aplicació més estesa a Espanya en aquest sector, que té 680 empreses subscrietes al seu producte. Intentant tocar amb els peus a terra, però sense abandonar la idea de bona expansió a nivell de territori nacional, hem considerat 350 empreses apuntades.
- **Fase final:** Representa un moment de bona expansió a nivell internacional. Hem agafat de referència l'aplicació més estesa a nivell mundial en aquest sector (principalment a Europa i Estats Units), que té 3.700 empreses apuntades. Amb la mateixa filosofia que en l'apartat anterior, hem considerat 2.000 empreses apuntades al nostre producte.

Hem tingut les següents consideracions a nivell de quantitats, que ens ajudarà a fer el càlcul de viabilitat: Es consideraran 200 atletes per cada empresa registrada de mitjana, unes 30 notificacions al mes per cada empresa, amb el qual ens sortiran un número de notificacions (de correu i mòbils) mensuals que seran un cost per nosaltres.

Quantitats			
	Inici (mensual)	Mig (mensual)	Final (mensual)
Número companyies	1	350	2.000
Atletes/companyia (mitjana)	200	200	200
Notificacions/companyia	30	30	30
Numero notificacions mensuals	6.000	2.100.000	12.000.000

Observant els preus de la competència, fent càlculs de despeses i informant-nos degudament sobre teoria de preus, vam considerar els següents plans a oferir a les empreses de Crossfit:

	Tarifes		
	Starter	Scaled	RX
	49,99 €	75,99 €	129,99 €
Número de companyies apuntades inici	0	1	0
Número de companyies apuntades mig	100	200	50
Número de companyies apuntades final	500	1200	300

On “Starter”, “Scaled” i “RX” són els noms dels plans que oferirem a les empreses, els quals tindran les següents restriccions:

Càlcul tarifes	Started	Scaled	RX
<b>Aletes</b>	50	200	Il·limitats
<b>Coaches</b>	2	4	Il·limitats
<b>Admins</b>	1	3	Il·limitats
<b>Boxes</b>	1	2	Il·limitats
Preu/mes	49,99 €	75,99 €	129,99 €

Tal i com hem comentat a l'apartat d'introducció, aquest projecte es desenvolupa sota Bisual, però com que és el primer projecte amb una certa envergadura econòmica, considerarem totes les despeses de Bisual com a empresa en aquest projecte per ser realistes (com ara autònoms, comptabilitats, gestories, etc) i no es tindran en compte costos compartits amb altres projectes (tot i que en un futur puguin arribar a ser-ho).

Les despeses considerades són les següents:

Despeses			
	Inici (mensual)	Mig (mensual)	Final (mensual)
Autònoms	60,00 €	100,00 €	375,00 €
Gestor	30,00 €	30,00 €	75,00 €
Software Comptabilitat	12,00 €	12,00 €	20,00 €
Servidor Correu electrònic	4,80 €	1.680,00 €	9.600,00 €
Servidor SMS	0,00 €	0,00 €	0,00 €
Servidor Preproducció	5,50 €	5,50 €	5,50 €
Servidor Producció	10,00 €	25,00 €	150,00 €

Publicitat	100,00 €	500,00 €	1.000,00 €
Nòmines	0,00 €	5.000,00 €	10.000,00 €
Oficina, llum i internet	0,00 €	175,00 €	1.000,00 €

Recopilant tota la informació que hem declarat anteriorment, tenim el següent balanç mensual en cadascuna de les tres fases (veure fig. 1):

	Inici	Mig	Final
<b>Despeses</b> (mensual)	222,30 €	7.527,50 €	22.225,50 €
<b>Ingressos</b> (mensual)	75,99 €	26.696,50 €	155.180,00 €
<b>Beneficis</b> (mensual)	-146,31 €	19.169,00 €	132.954,50 €

### Viabilitat econòmica

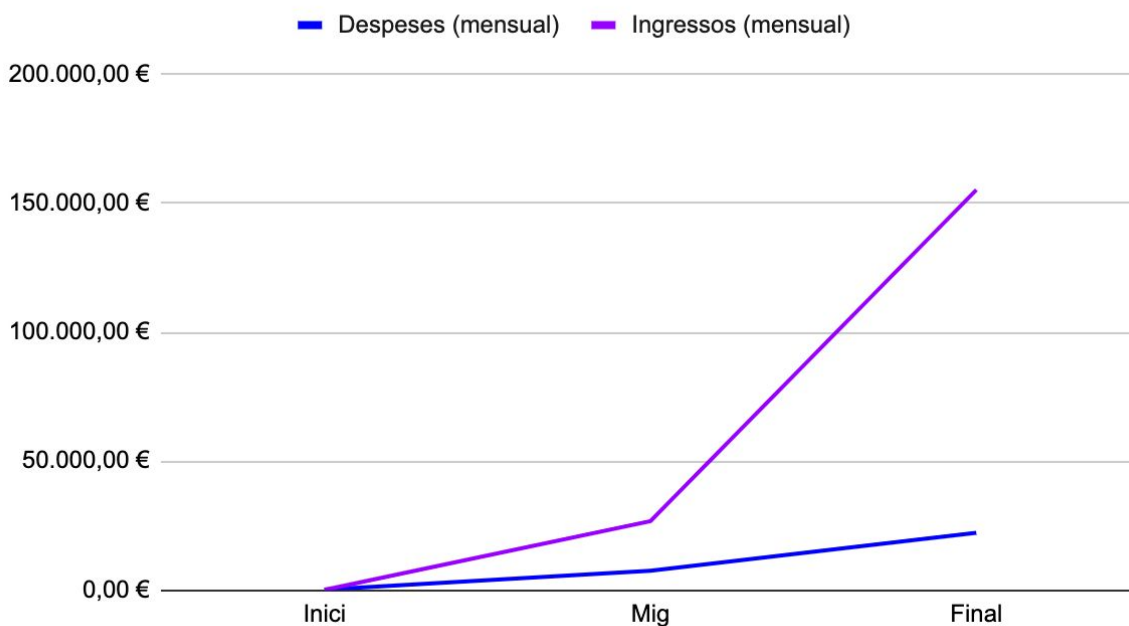


fig.1: Evolució econòmica

Podem veure que és un projecte que amb poques empreses subscrietes pot esdevenir viable i té una escalabilitat econòmica molt bona si es gestiona correctament.



## 3. Metodologia

### 3.1. Metodologies Agile

La gestió Agile és un conjunt de metodologies pel desenvolupament de projectes, es caracteritza per la velocitat d'entrega i la flexibilitat que aporta. Els equips Agile treballen en petites entregues durant tot el projecte que són ensenyades com a demo al client « Divide y vencerás ». Algunes metodologies conegudes són Scrum, Kanban i Lean degut a la facilitat aportada per flexibilitzar un projecte complexe.

#### 3.1.1. Fases de les metodologies Agile

1. **Estudi de requeriments:** Donat una idea, s'anoten tots els requeriments que haurà de complir el producte final.
2. **Definició de tasques:** Donats tots els requeriments, es tradueixen a tasques que assegurin que tots els requeriments seran aconseguits i s'agrupen en petits grups de tasques per poder portar a terme durant un període de temps.
3. **Iteracions:** Un cop agrupats, es decideix quantes iteracions s'han de fer i quines tasques es faran en cada una d'elles.
4. **Preparació de la iteració:** Les tasques de la iteració són assignades a les persones que conformen l'equip per començar a desenvolupar-les.
5. **Desenvolupament de la iteració:** Es desenvolupen les tasques de la iteració actual.
6. **Validació de les tasques:** Un cop acabada la iteració, es revisen les tasques, es mostra l'evolució del projecte al client i es torna al punt "Preparació de la iteració" fins que no queda cap altra iteració a fer.
7. **Entrega del producte:** Quan el producte compleix amb els requeriments inicials, es fa l'entrega final.

### 3.2. Metodologia incremental

El model incremental de gestió de projectes té com a objectiu un creixement progressiu de la funcionalitat del producte.

Durant les entregues parcials del producte, hi ha d'haver una evolució en el producte respecte l'anterior.

Aquesta metodologia té característiques com:

- Els increments són petits.
- Permet una fàcil administració de les tasques en cada iteració.
- La inversió es materialitza a curt termini.
- És un model propici a canvis o modificacions.
- S'adapta a les necessitats que sorgeixin.



## 4. Planificació

Per dur a terme el nostre projecte hem dividit el treball fet durant el transcurs aproximat d'un any en diferents sprints d'entre dues i quatre setmanes, depenent de la quantitat de feina i/o dificultat que requereixen les tasques i el temps disponible.

En alguns casos, sobretot treballant en el backend, vam preferir treballar amb sprints llargs perquè les tasques no eren realment difícils, però el nostre nivell era baix, i tenint un marge ampli ens assegurarem de tenir temps a trobar les solucions més eficients per cadascuna de les tasques.

Amb la finalitat de poder mantenir-nos comunicats, hem fet mínim una reunió diària de pocs minuts per posar-nos al dia, conèixer les dificultats amb les quals es trobava l'altre i consells que podíem compartir entre nosaltres; sense tenir en compte el temps que hem estat treballant junts de manera remota i física, per tal de garantir que realment tots dos integrants de l'equip sabien com funcionaven totes les parts.

Un cop finalitzat cadascun dels sprints, ens reuníem durant un mínim de trenta minuts per comentar la situació actual, decidir quines de les tasques restants incloïem a l'sprint següent i repartir-les assignant un temps màxim pel conjunt d'aquestes.

No hem seguit cap sistema de puntuació entre nosaltres perquè no ho hem considerat necessari en la primera fase del nostre producte, ja que en molts dels casos era un primer contacte amb les eines que hem fet servir i no podíem tenir realment una visió objectiva del treball necessari.

Hem intentat seguir els principis de comunicació entre l'equip de les metodologies Agile, afegint la flexibilitat en el temps dels sprints i assumint nosaltres tots els rols contemplats en aquestes metodologies.

Vam dividir tota la feina que comportava el projecte en petites tasques el més independent possible les unes de les altres. Un cop fet això, les agrupem en paquets de treball, més tard utilitzats pels sprints.

Un paquet de treball és una descripció d'una operació que s'ha de dur a terme, format per una o més tasques i un període de temps.

## 4.1. Paquets de treball

<b>Nom del paquet</b>	Reunió amb el tutor
<b>Temporització</b>	10h (des de Setembre del 2019, una reunió mínima al mes)
<b>Tasques</b>	<ul style="list-style-type: none"><li>- Presentació de la proposta</li><li>- Aclariment de dubtes</li><li>- Preparació de la documentació inicial</li></ul>

<b>Nom del paquet</b>	Reunions internes
<b>Temporització</b>	Durant tot el transcurs del projecte
<b>Tasques</b>	<ul style="list-style-type: none"><li>- Dubtes</li><li>- Planificació</li><li>- Preparació de material</li></ul>

<b>Nom del paquet</b>	Documentació
<b>Temporització</b>	Durant tot el transcurs del projecte
<b>Tasques</b>	<ul style="list-style-type: none"><li>- Documentació del TFG</li><li>- Documentació interna</li></ul>

<b>Nom del paquet</b>	Backend, estructura principal
<b>Temporització</b>	110h
<b>Tasques</b>	<ul style="list-style-type: none"><li>- Manteniment de companyies</li><li>- Manteniment de boxes</li><li>- Manteniment d'usuaris</li><li>- Manteniment de Training Workouts</li></ul>

<b>Nom del paquet</b>	Backend, estructura secundària
<b>Temporització</b>	110h
<b>Tasques</b>	<ul style="list-style-type: none"><li>- Manteniment de plans per companyies</li><li>- Manteniment de training rooms</li><li>- Manteniment de descomptes</li><li>- Manteniment de notícies</li><li>- Manteniment de mètodes de pagament</li></ul>

<b>Nom del paquet</b>	Backend, sessions d'entrenament
<b>Temporització</b>	110h
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Generar calendari de sessions d'entrenament</li> <li>- Manteniment de sessions d'entrenament</li> <li>- Apuntar-se i desapuntar-se d'una sessió d'entrenament</li> <li>- Manteniment de tipus d'entrenament</li> <li>- Manteniment de training rooms</li> <li>- Manteniment de resultats d'entrenament d'usuari</li> <li>- Manteniment de plantilles de calendari</li> </ul>

<b>Nom del paquet</b>	Backend, subscripcions d'usuari
<b>Temporització</b>	110h
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Manteniment de subscripcions d'usuaris</li> <li>- Manteniment de factures</li> <li>- Manteniment de comandes de box</li> <li>- Generar i renovar una subscripció d'usuari</li> </ul>

<b>Nom del paquet</b>	Backend, subscripció de la companyia
<b>Temporització</b>	110h
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Manteniment de tarifes de companyies</li> <li>- Manteniment de comandes de companyia</li> <li>- Manteniment de subscripcions de companyia</li> <li>- Generar, pagar i cancel·lar una subscripció de companyia</li> </ul>

<b>Nom del paquet</b>	Backend, manteniment paràmetres de l'app
<b>Temporització</b>	86h
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Manteniment de preferències d'aplicació</li> <li>- Manteniment d'arxius d'aplicació</li> </ul>

<b>Nom del paquet</b>	Frontend - Web
<b>Temporització</b>	148h ( De Març a Abril, Sprints de dues setmanes, mitjana 3/4h diàries cadascun)
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Autenticació</li> <li>- Pàgina inicial</li> <li>- Pàgina calendari</li> <li>- Pàgina entrenaments</li> <li>- Pàgina notícies</li> <li>- Pàgina de subscripció d'usuari</li> <li>- Pàgina facturació</li> <li>- Pàgina "mi negocio"</li> </ul>

<b>Nom del paquet</b>	Frontend - Mòbil
<b>Temporització</b>	160h (D'Abril a Juny, Sprints de dues setmanes, mitjana 3/4h diàries cadascun)
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Autenticació</li> <li>- Pantalla d'inici</li> <li>- Menú inferior</li> <li>- Pantalla de calendari de box</li> <li>- Pantalla d'estadístiques i historial</li> <li>- Pantalla de facturació</li> <li>- Pantalla del meu perfil</li> </ul>

<b>Nom del paquet</b>	Disseny Gràfic - Web
<b>Temporització</b>	30h (durant una setmana de març)
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Layout + Basic configurations</li> <li>- Api Services + Models</li> <li>- Login/Register/Contraseña olvidada</li> <li>- Dashboard</li> <li>- TV Box</li> <li>- Box Calendar</li> <li>- Workouts</li> <li>- Subscriptions</li> <li>- News</li> <li>- Facturación</li> <li>- Mi negocio</li> </ul>

<b>Nom del paquet</b>	Disseny Gràfic - Mòbil
<b>Temporització</b>	20h (durant una setmana de març)
<b>Tasques</b>	<ul style="list-style-type: none"> <li>- Login/Contraseña olvidada</li> <li>- Inici</li> <li>- Canvi de subscripció</li> <li>- Box Calendar</li> <li>- Stats</li> <li>- Facturació</li> <li>- Detall training session</li> <li>- Pop up usuari</li> <li>- Detall usuari</li> </ul>

Un cop tenim tots els paquets de treball, farem servir el diagrama de Gantt per visualitzar el transcurs del projecte i estudiarem el camí crític d'aquest.

## 4.2. Camí crític

El camí crític ens diu la durada que comporta realitzar el projecte posicionant les tasques de manera seqüencial tenint en compte de quines altres depenen (veure fig. 2).

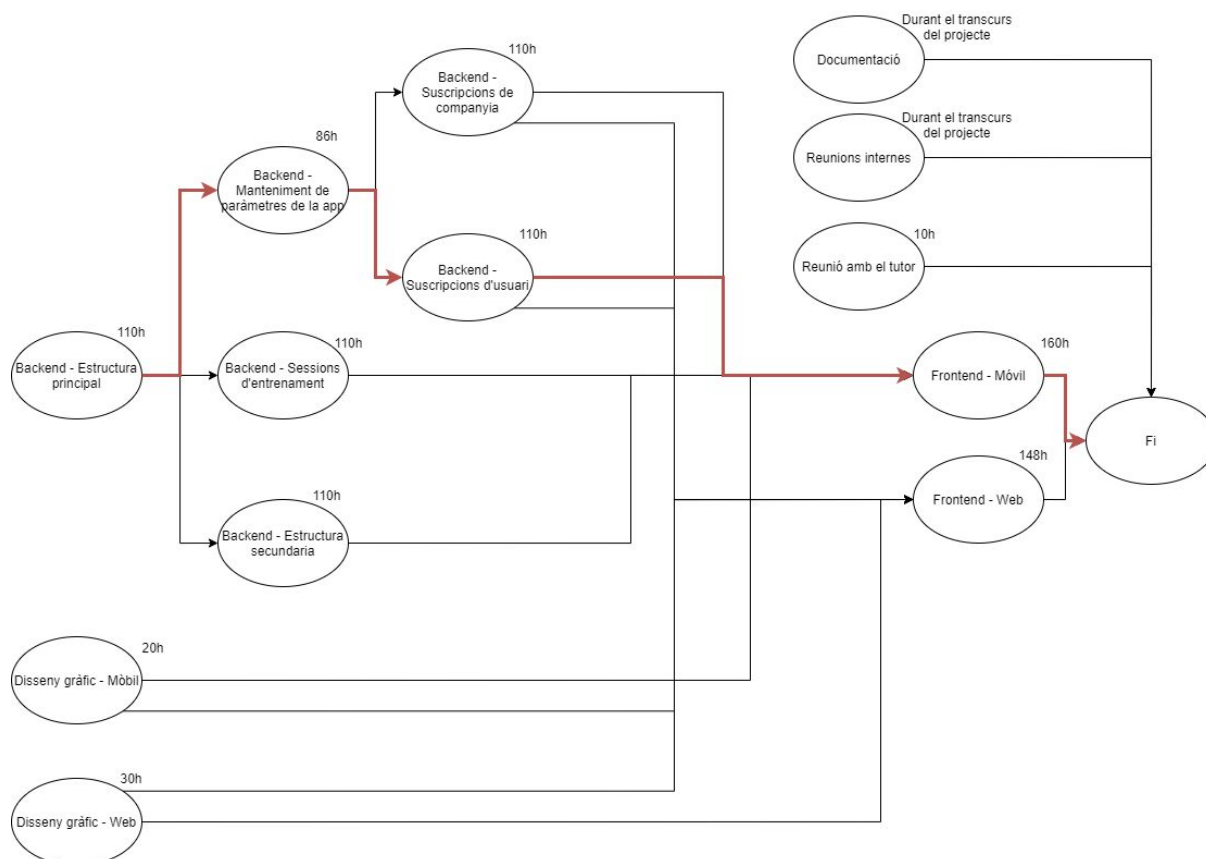


fig.2: Camí crític

En el nostre cas, el camí crític és de 446 hores.

### 4.3. Diagrama de Gantt

El diagrama de Gantt ens donarà una visió global del projecte separat pels paquets de treball, donant-nos informació sobre quines tasques han estat realitzades durant cada període de temps.

Tasca	Setembre				Octubre				Novembre			
	1/4	2/4	3/4	4/4	1/4	2/4	3/4	4/4	1/4	2/4	3/4	4/4
<b>Administració i Gestió</b>												
Reunió amb el tutor												
Reunions internes												
Documentació												
<b>Desenvolupament</b>												
Backend, estructura principal												
Backend, manteniment paràmetres de l'app												
Backend, sessions d'entrenament												
Backend, subscripcions d'usuari												
Backend, subscripció de la companyia												
Backend, estructura secundaria												
Disseny Gràfic - Mòbil												
Disseny Gràfic - Web												
Frontend - Web												
Frontend - Mòbil												





## 5. Marc de treball i conceptes previs

En el món del Crossfit els conceptes utilitzats són en anglès americà. En aquest document, per tal d'explicar i apropar-nos més, podrem veure termes com:

- **Crossfit:** Marca registrada, empresa i nom que obté la disciplina que ens ocupa. El Crossfit és una disciplina d'entrenament compost d'exercicis funcionals, executats a alta intensitat i constantment substituïts. És una disciplina molt més social, intensa i majoritàriament més satisfactòria que anar a un gimnàs tradicional. Es realitzen exercicis de flexibilitat, mobilitat, gimnàstica, halterofília, resistència, força, agilitat, entre altres.
- **Box:** Centre físic de Crossfit on els atletes practiquen l'esport.
- **Wod:** Abreviatura de "Workout of day" (entrenament del dia). Representa la planificació d'exercici que hi ha per el dia en qüestió. Cada dia es fa un wod diferent, vagis a l'hora que vagis, preparat per el coach, i en grups (en funció de l'espai disponible).
- **OpenBox:** Sessió d'entrenament individual on cadascú fa el Wod que considera per tal de millorar tècnica d'exercicis que a un l'interessi.
- **Coach:** Entrenador certificat per Crossfit.
- **Crossfit Games:** Esdeveniment mundial anual que organitza Crossfit per seleccionar Fittest Man/Woman on Earth.
- **Categories de Wods:** Hi ha una sèrie de Wods en Crossfit que són universals per a tothom. Aquests es classifiquen en categories, com per exemple **Hero**, **Girls**, o els mateixos **Wods dels Crossfit Games**.
- **Tipus d'entrenaments:** En el Crossfit hi ha diferents tipus d'entrenaments segons la seva naturalesa (per temps, per rondes, etc). Ens trobem conceptes com **AMRAP** (As many repetitions as possible) que vol dir fer tantes repeticions com es pugui en un temps establert, **EMOM** (every minute on a minute) que vol dir entrar unes repeticions establertes en un minut, i descansar el que resta de minut, i repetir-ho cada minut fins arribar al temps establert; o per exemple **For Time**, que vol dir fer les rondes d'exercicis que s'han proposat en el menor temps possible. Entre altres.
- **RX i Scaled:** Com que els nivells entre tots els atletes poden ser molt variats, s'estableixen normalment als wods diferents nivells per tal que els més avançats puguin competir entre ells, així com també els més novells. Aquests nivells són RX i Scaled respectivament. Scaled respecte RX, normalment representa menys repeticions, menys pes o menys temps.

*Nota: Aquests conceptes s'utilitzaran al llarg del document.*

Com hem comentat a l'inici del document, els negocis de Crossfit tenen diferències amb els negocis de gimnasos tradicionals. Qüestions com ara:

- Les tarifes solen contemplar per separat número de WODs setmanals i d'OpenBox.
- Les sessions de WOD són sempre en grup.
- Poden haver-hi sessions concretes al llarg del dia d'una disciplina en concret (gimnàstics, halterofília, mobilitat, etc).

- Les tarifes no es solen condicionar per l'hora en la que s'assisteix.
- La implicació dels coaches amb els atletes és molt superior. Sol conèixer a tots els seus atletes, així com les mancances i virtuts de cadascun.
- El ventall de varietat d'exercicis que s'executen en els Wods és molt més ample que qualsevol altra disciplina.

Per crear el nostre projecte hem tingut en compte opinions tant d'atletes com de propietaris de centres Crossfit de Girona i Blanes per tal d'assegurar-nos que el nostre anàlisi inicial era correcte. Han participat de forma puntual donant opinions tan en la fase de planificació com en la de desenvolupament.

Un dels temes que és important comprendre i cal deixar clar, entrant en temes més tècnics, és que quan es parli d'integració de Stripe i pagaments automàtics, es parla d'una integració en dos sentits:

- De les empreses com a client de Hit a Wod. Es facturarà a les empreses **sempre** de forma automàtica i usant les claus API del compte de Stripe de Hit a Wod.
- Dels atletes com a clients de les empreses. Les empreses, **opcionalment** podran configurar un client de Stripe per elles i així poder facturar als atletes. Als atletes sempre se'ls podrà oferir el pagament automàtic amb targeta o bé el pagament en efectiu.

## 6. Requisits del sistema

El nostre sistema contempla els següents actors:

- **Usuari no registrat en el sistema:** Un potencial client d'empresa que pot executar la funció de registrar-se.
- **Atleta:** Usuari que té permisos per entrar a l'aplicació mòbil i utilitzar les funcionalitats que ofereix.
- **Coach:** Entrenador que tindrà accés a l'aplicació mòbil, i accés a la pàgina web d'administració per gestionar el calendari i els entrenaments del box.
- **Box Admin:** Responsable d'un dels boxes d'una companyia. Tindrà el mateix que un coach, i també als apartats de facturació i paràmetres de configuració del box. No podrà veure la facturació global de l'empresa.
- **Company Admin:** Responsable de la companyia. Tindrà el mateix accés que el Box Admin però podrà veure tots els boxes, la facturació global de l'empresa i la configuració d'empresa, així com els apartats de pagament i subscripció mensual. En una empresa amb un sol box, el company admin i el box admin seran la mateixa figura.
- **Super Admin:** Tècnics de Hit a Wod. Per poder tenir accés a tot.

Nota: El sistema de permisos de Hit a Wod és un dels punts forts de l'aplicació, ja que les funcionalitats no han estat només restringides a nivell de frontend (app mòbil o web) sinó que també han estat restringits a nivell d'Api. Per cada funcionalitat s'han establert uns permisos per cada rol. El funcionament d'autorització REST està detallat en el seu respectiu apartat en aquest document.

A continuació es detallen els requisits per aplicació i per actor:

#	Descripció	Not logged in	Athlete	Coach	B. Adm	C. Adm	S. Adm
<b>Backend</b>							
<b>Manteniment de companyies</b>							
<b>HAWR001</b>	Pot obtenir les companyies	No	No	No	No	No	Sí
<b>HAWR002</b>	Pot obtenir una companyia	No	No	No	No	Sí	Sí
<b>HAWR003</b>	Pot crear una companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR004</b>	Pot modificar una companyia	No	No	No	No	Sí	Sí
<b>HAWR005</b>	Pot esborrar una companyia	No	No	No	No	Sí	Sí
<b>HAWR006</b>	Pot obtenir els boxes d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR007</b>	Pot obtenir les tarifes d'usuari d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR008</b>	Pot obtenir els usuaris d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR009</b>	Pot obtenir els descomptes d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR010</b>	Pot obtenir les comandes d'un box d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR011</b>	Pot obtenir els tipus de pagament d'una companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR012</b>	Pot obtenir les tarifes d'usuari d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR013</b>	Pot obtenir les comandes d'una companyia	No	No	No	No	Sí	Sí
<b>HAWR014</b>	Pot obtenir els notícies d'una companyia	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de mètodes de pagament</b>							
<b>HAWR015</b>	Pot modificar el tipus de pagament de la companyia	No	No	No	No	Sí	Sí
<b>HAWR016</b>	Pot modificar el tipus de pagament de la companyia per un usuari	No	Sí	No	No	Sí	Sí
<b>Manteniment de boxes</b>							
<b>HAWR017</b>	Pot obtenir els boxes	No	No	No	No	No	Sí
<b>HAWR018</b>	Pot obtenir un box	No	Sí	Sí	Sí	Sí	Sí

<b>HAWR019</b>	Pot crear un box	No	No	No	No	Sí	Sí
<b>HAWR020</b>	Pot modificar un box	No	No	No	Sí	Sí	Sí
<b>HAWR021</b>	Pot eliminar un box	No	No	No	No	Sí	Sí
<b>HAWR022</b>	Pot obtenir els usuaris d'un box	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR023</b>	Pot obtenir les subscripcions d'usuari d'un box	No	No	No	Sí	Sí	Sí
<b>HAWR024</b>	Pot obtenir els usuaris d'un box	No	No	Sí	Sí	Sí	Sí
<b>HAWR025</b>	Pot obtenir els calendar templates d'un box	No	No	Sí	Sí	Sí	Sí
<b>HAWR026</b>	Pot obtenir les sessions d'entrenament d'un box	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR027</b>	Pot obtenir la companyia d'un box	No	No	No	Sí	Sí	Sí
<b>HAWR028</b>	Pot obtenir els coaches d'un box	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR029</b>	Pot obtenir els entrenaments d'un box	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR030</b>	Pot obtenir les notícies d'un box	No	Sí	Sí	Sí	Sí	Sí
<b>Generar calendari de sessions d'entrenament</b>							
<b>HAWR031</b>	Pot generar un calendar template d'un box	No	No	Sí	Sí	Sí	Sí
<b>Manteniment d'usuaris</b>							
<b>HAWR033</b>	Pot obtenir tots els usuaris	No	No	No	No	No	Sí
<b>HAWR034</b>	Pot obtenir un usuari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR035</b>	Pot crear/registrar un usuari	No	No	Sí	Sí	Sí	Sí
<b>HAWR036</b>	Pot modificar un usuari	No	No	No	No	No	Sí
<b>HAWR037</b>	Pot eliminar un usuari	No	No	No	No	No	Sí
<b>HAWR038</b>	Pot obtenir les sessions d'entrenament d'un usuari	No	No	No	No	Sí	Sí
<b>HAWR039</b>	Pot obtenir les puntuacions d'un usuari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR040</b>	Pot obtenir la subscripció d'usuari d'un usuari	No	No	No	No	No	Sí
<b>HAWR041</b>	Pot obtenir les comandes al box d'un usuari	No	No	No	No	No	Sí
<b>HAWR042</b>	Pot obtenir els entrenaments preferits d'un usuari	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de plans per companyies</b>							

<b>HAWR043</b>	Pot obtenir totes les tarifes de companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR044</b>	Pot obtenir una tarifa de companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR045</b>	Pot crear una tarifes de companyia	No	No	No	No	No	Sí
<b>HAWR046</b>	Pot modificar una tarifes de companyia	No	No	No	No	No	Sí
<b>HAWR047</b>	Pot eliminar la tarifa de companyia	No	No	No	No	No	Sí
<b>Manteniment de Training Rooms</b>							
<b>HAWR048</b>	Pot obtenir totes les sales d'entrenament	No	No	No	No	No	Sí
<b>HAWR049</b>	Pot obtenir una sala d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR050</b>	Pot crear una sala d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR051</b>	Pot modificar una sala d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR052</b>	Pot eliminar una sala d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR053</b>	Pot obtenir les sessions d'entrenament d'una sala d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR054</b>	Pot obtenir el box d'una sala d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de Sessions d'entrenament</b>							
<b>HAWR055</b>	Pot obtenir totes les sessions d'entrenament	No	No	No	No	No	Sí
<b>HAWR056</b>	Pot obtenir una sessió d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR057</b>	Pot crear una sessió d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR058</b>	Pot modificar una sessió d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR059</b>	Pot eliminar una sessió d'entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR060</b>	Pot obtenir tots els usuaris d'una sessió d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR061</b>	Pot obtenir les puntuacions d'una sessió d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>Apuntar-se i desapuntar-se d'una sessió d'entrenament</b>							

<b>HAWR062</b>	Pot apuntar-se a una sessió d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR063</b>	Pot desapuntar-se de una sessió d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de Training Workouts</b>							
<b>HAWR064</b>	Pot obtenir tots els entrenaments	No	No	No	No	No	Sí
<b>HAWR065</b>	Pot obtenir un entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR066</b>	Pot crear un entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR067</b>	Pot modificar un entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR068</b>	Pot eliminar un entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR069</b>	Pot obtenir el box d'un entrenament	No	No	Sí	Sí	Sí	Sí
<b>Manteniment de Tipus d'entrenament</b>							
<b>HAWR070</b>	Pot obtenir tots els tipus d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR071</b>	Pot obtenir un tipus d'entrenament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR072</b>	Pot crear un tipus d'entrenament	No	No	No	No	No	Sí
<b>HAWR073</b>	Pot modificar un tipus d'entrenament	No	No	No	No	No	Sí
<b>HAWR074</b>	Pot eliminar un tipus d'entrenament	No	No	No	No	No	Sí
<b>Manteniment de Training Rooms</b>							
<b>HAWR075</b>	Pot obtenir tots el exercicis	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR076</b>	Pot obtenir un exercici	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR077</b>	Pot crear un exercici	No	No	No	No	No	Sí
<b>HAWR078</b>	Pot modificar un exercici	No	No	No	No	No	Sí
<b>HAWR079</b>	Pot eliminar un exercici	No	No	No	No	No	Sí
<b>Manteniment de preferències de box: No es podrà via aplicació</b>							
<b>Manteniment de resultats d'entrenament d'usuari</b>							
<b>HAWR080</b>	Pot obtenir totes les puntuacions	No	No	No	No	No	Sí
<b>HAWR081</b>	Pot obtenir una puntuació	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR082</b>	Pot crear una puntuació	No	No	No	No	No	Sí
<b>HAWR083</b>	Pot obtenir l'usuari d'una puntuació	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de plantilles de calendari</b>							

<b>HAWR084</b>	Pot obtenir tots els calendar templates	No	No	No	No	No	Sí
<b>HAWR085</b>	Pot obtenir un calendar template	No	No	Sí	Sí	Sí	Sí
<b>HAWR086</b>	Pot crear un calendar template	No	No	Sí	Sí	Sí	Sí
<b>HAWR087</b>	Pot modificar un calendar template	No	No	Sí	Sí	Sí	Sí
<b>HAWR088</b>	Pot eliminar un calendar template	No	No	Sí	Sí	Sí	Sí
<b>HAWR089</b>	Pot obtenir el box d'un calendar template	No	No	Sí	Sí	Sí	Sí
<b>Manteniment de tarifes de companyies</b>							
<b>HAWR090</b>	Pot obtenir totes les tarifes de companyia	No	No	No	No	No	Sí
<b>HAWR091</b>	Pot obtenir una tarifa de companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR092</b>	Pot crear una tarifa de companyia	No	No	No	No	Sí	Sí
<b>HAWR093</b>	Pot modificar una tarifa de companyia	No	No	No	No	Sí	Sí
<b>HAWR094</b>	Pot eliminar una tarifa de companyia	No	No	No	No	Sí	Sí
<b>HAWR095</b>	Pot obtenir la subscripció d'una tarifa de companyia	No	No	No	Sí	Sí	Sí
<b>Manteniment de subscripcions d'usuaris</b>							
<b>HAWR096</b>	Pot obtenir totes les subscripcions d'usuari	No	No	No	No	No	Sí
<b>HAWR097</b>	Pot obtenir una subscripció d'usuari	No	Sí	No	Sí	Sí	Sí
<b>HAWR098</b>	Pot crear una subscripció d'usuari	No	No	No	Sí	Sí	Sí
<b>HAWR099</b>	Pot modificar una subscripció d'usuari	No	Sí	No	No	No	Sí
<b>HAWR100</b>	Pot pagar una subscripció d'usuari	No	Sí	No	No	No	Sí
<b>HAWR101</b>	Pot eliminar una subscripció d'usuari	No	No	No	No	No	Sí
<b>Generar i renovar una subscripció d'usuari</b>							
<b>HAWR102</b>	GET BoxOrders	No	Sí	No	Sí	Sí	Sí
<b>HAWR103</b>	PUT Renew	No	Sí	No	Sí	Sí	Sí
<b>Manteniment de pagaments</b>							



<b>HAWR104</b>	Pot obtenir totes les factures	No	No	No	No	No	Sí
<b>HAWR105</b>	Pot obtenir una factura	No	Sí	No	Sí	Sí	Sí
<b>HAWR106</b>	Pot crear una factura	No	No	No	Sí	Sí	Sí
<b>HAWR107</b>	Pot modificar una factura	No	No	No	Sí	Sí	Sí
<b>HAWR108</b>	Pot eliminar una factura	No	No	No	Sí	Sí	Sí
<b>Manteniment de mètodes de pagament</b>							
<b>HAWR109</b>	Pot obtenir tots els mètodes de pagament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR110</b>	Pot obtenir un mètode de pagament	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR111</b>	Pot crear un mètode de pagament	No	No	No	No	No	Sí
<b>HAWR112</b>	Pot modificar un mètode de pagament	No	No	No	No	No	Sí
<b>HAWR113</b>	Pot eliminar un mètode de pagament	No	No	No	No	No	Sí
<b>Manteniment de comandes de box</b>							
<b>HAWR114</b>	Pot obtenir totes les comandes de box	No	No	No	No	Sí	
<b>HAWR115</b>	Pot obtenir una comanda de box	No	No	Sí	Sí	Sí	
<b>HAWR116</b>	Pot crear una comanda de box	No	No	Sí	Sí	Sí	
<b>HAWR117</b>	Pot actualitzar una comanda de box	No	No	Sí	Sí	Sí	
<b>HAWR118</b>	Pot eliminar una comanda de box	No	No	Sí	Sí	Sí	
<b>HAWR119</b>	Pot obtenir el pagament d'una comanda de box	No	No	Sí	Sí	Sí	
<b>Box Product, Variant &amp; Images... De moment no hi ha res a nivell d'aplicació</b>							
<b>Mateniment de descomptes</b>							
<b>HAWR120</b>	Pot obtenir tots els descomptes	No	No	No	No	No	Sí
<b>HAWR121</b>	Pot obtenir un descompte	No	No	No	No	Sí	Sí
<b>HAWR122</b>	Pot crear un descompte	No	No	No	No	Sí	Sí
<b>HAWR123</b>	Pot modificar un descompte	No	No	No	No	Sí	Sí
<b>HAWR124</b>	Pot eliminar un descompte	No	No	No	No	Sí	Sí
<b>Manteniment de preferències de l'aplicació</b>							
<b>HAWR125</b>	Pot obtenir totes les preferències	No	Sí	Sí	Sí	Sí	Sí

<b>HAWR126</b>	Pot obtenir una preferència	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR127</b>	Pot actualitzar una preferència	No	No	No	No	No	Sí
<b>Manteniment de factures</b>							
<b>HAWR128</b>	Pot obtenir totes les factures	No	No	No	No	No	Sí
<b>HAWR129</b>	Pot obtenir una factura	No	No	No	Sí	Sí	Sí
<b>HAWR130</b>	Pot actualitzar una factura	No	No	No	Sí	Sí	Sí
<b>HAWR131</b>	Pot eliminar una factura	No	No	No	Sí	Sí	Sí
<b>HAWR132</b>	Pot generar el pdf d'una factura	No	No	No	Sí	Sí	Sí
<b>Manteniment de comandes de companyia</b>							
<b>HAWR133</b>	Pot obtenir totes les comandes de companyia	No	No	No	No	No	Sí
<b>HAWR134</b>	Pot obtenir una comanda de companyia	No	No	No	No	Sí	Sí
<b>HAWR135</b>	Pot crear una comanda de companyia	No	No	No	No	No	Sí
<b>HAWR136</b>	Pot actualitzar una comanda de companyia	No	No	No	No	No	Sí
<b>HAWR137</b>	Pot eliminar una comanda de companyia	No	No	No	No	No	Sí
<b>Manteniment de subscripcions de companyia</b>							
<b>HAWR138</b>	Pot obtenir totes les subscripcions de companyia	No	No	No	No	No	Sí
<b>HAWR139</b>	Pot obtenir una subscripció de companyia	No	No	No	No	Sí	Sí
<b>HAWR140</b>	Pot crear una nova subscripció de companyia	No	No	No	No	No	Sí
<b>HAWR141</b>	Pot actualitzar una subscripció de companyia	No	No	No	No	No	Sí
<b>HAWR142</b>	Pot eliminar una subscripció de companyia	No	No	No	No	No	Sí
<b>Generar, pagar i cancel·lar una subscripció</b>							
<b>HAWR143</b>	Pot generar una nova subscripció de companyia	No	No	No	No	Sí	Sí
<b>HAWR144</b>	Pot pagar una subscripció nova o renovar-la	No	No	No	No	Sí	Sí
<b>HAWR145</b>	Pot cancel·lar una subscripció	No	No	No	No	Sí	Sí
<b>Manteniment d'arxius d'aplicació</b>							

<b>HAWR146</b>	Pot obtenir un arxiu de l'aplicació	No	Sí	Sí	Sí	Sí	Sí
<b>Manteniment de notícies</b>							
<b>HAWR147</b>	Pot obtenir totes les notícies	No	No	No	No	No	Sí
<b>HAWR148</b>	Pot obtenir una notícia	No	No	No	Sí	Sí	Sí
<b>HAWR149</b>	Pot crear una noticia	No	No	No	Sí	Sí	Sí
<b>HAWR150</b>	Pot actualitzar una noticia	No	No	No	Sí	Sí	Sí
<b>HAWR151</b>	Pot eliminar una noticia	No	No	No	Sí	Sí	Sí
<b>Frontend - Web</b>							
<b>Autenticació</b>							
<b>HAWR152</b>	Pot registrar una companyia	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR153</b>	Pot entrar	No	Sí	Sí	Sí	Sí	Sí
<b>Pàgina inicial</b>							
<b>HAWR154</b>	Pot seleccionar un box	No	No	Sí	Sí	Sí	Sí
<b>HAWR155</b>	Pot accedir al menú d'accions ràpides	No	No	Sí	Sí	Sí	Sí
<b>HAWR156</b>	Pot accedir al xat d'ajuda	No	No	Sí	Sí	Sí	Sí
<b>HAWR157</b>	Pot veure les estadístiques	No	No	Sí	Sí	Sí	Sí
<b>Pàgina calendari</b>							
<b>HAWR158</b>	Pot crear una nova sessió	No	No	Sí	Sí	Sí	Sí
<b>HAWR159</b>	Pot crear una plantilla de calendari	No	No	Sí	Sí	Sí	Sí
<b>HAWR160</b>	Pot veure el calendari actual	No	No	Sí	Sí	Sí	Sí
<b>Pàgina entrenaments</b>							
<b>HAWR161</b>	Pot llistar els entrenaments	No	No	Sí	Sí	Sí	Sí
<b>HAWR162</b>	Pot crear un entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR163</b>	Pot modificar un entrenament	No	No	Sí	Sí	Sí	Sí
<b>HAWR164</b>	Pot eliminar un entrenament	No	No	Sí	Sí	Sí	Sí
<b>Pàgina notícies</b>							
<b>HAWR165</b>	Pot llistar les notícies	No	No	No	Sí	Sí	Sí
<b>HAWR166</b>	Pot crear una noticia	No	No	No	Sí	Sí	Sí
<b>HAWR167</b>	Pot modificar una noticia	No	No	No	Sí	Sí	Sí
<b>HAWR168</b>	Pot eliminar una noticia	No	No	No	Sí	Sí	Sí
<b>Pàgina subscripció d'usuari</b>							
<b>HAWR169</b>	Pot llistar les subscripcions d'usuari	No	No	No	Sí	Sí	Sí

<b>HAWR170</b>	Pot crear un subscripció d'usuari	No	No	No	Sí	Sí	Sí
<b>HAWR171</b>	Pot veure al usuari de la subscripció d'usuari	No	No	No	Sí	Sí	Sí
<b>Pàgina facturació</b>							
<b>HAWR172</b>	Pot veure les comandes de box	No	No	No	Sí	Sí	Sí
<b>HAWR173</b>	Pot veure les comandes de companyia	No	No	No	No	Sí	Sí
<b>Pàgina 'mi negocio'</b>							
<b>HAWR174</b>	Pot veure la companyia	No	No	No	Sí	Sí	Sí
<b>HAWR175</b>	Pot editar la companyia	No	No	No	No	Sí	Sí
<b>HAWR176</b>	Pot actualitzar la tarifa de la companyia	No	No	No	No	Sí	Sí
<b>HAWR177</b>	Pot gestionar descomptes	No	No	No	Sí	Sí	Sí
<b>HAWR178</b>	Pot gestionar tarifes de subscripció d'usuaris	No	No	No	Sí	Sí	Sí
<b>HAWR179</b>	Pot mantenir els boxes	No	No	No	No	Sí	Sí
<b>Frontend - Mòbil</b>							
<b>Autenticació</b>							
<b>HAWR180</b>	Pot entrar	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR181</b>	Pot recuperar la contrasenya	No	Sí	Sí	Sí	Sí	Sí
<b>Pantalla d'inici</b>							
<b>HAWR182</b>	Pot veure notícies del box	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR183</b>	Pot veure estadístiques de l'usuari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR184</b>	Pot veure la proxima sessió	No	Sí	Sí	Sí	Sí	Sí
<b>Menú inferior</b>							
<b>HAWR185</b>	Pot navegar entre les pantalles principals	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR186</b>	Pot canviar de subscripció d'usuari	No	Sí	Sí	Sí	Sí	Sí
<b>Pantalla de calendari de box</b>							
<b>HAWR187</b>	Pot navegar pel calendari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR188</b>	Pot apuntar-se a una sessió	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR189</b>	Pot accedir a la sessió	No	Sí	Sí	Sí	Sí	Sí
<b>Pantalla d'estadístiques i historial</b>							
<b>HAWR190</b>	Pot veure workouts preferits	No	Sí	Sí	Sí	Sí	Sí

<b>HAWR191</b>	Pot veure el rendiment i historial d'un exercici concret	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR192</b>	Pot entrar estadístiques d'un exercici	No	Sí	Sí	Sí	Sí	Sí
<b>Pantalla de facturació</b>							
<b>HAWR193</b>	Pot renovar la subscripció d'usuari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR194</b>	Pot veure les comandes	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR195</b>	Pot configurar una targeta per pagar	No	Sí	Sí	Sí	Sí	Sí
<b>Pantalla del meu perfil</b>							
<b>HAWR196</b>	Veure el meu usuari	No	Sí	Sí	Sí	Sí	Sí
<b>HAWR197</b>	Actualitzar el meu usuari	No	Sí	Sí	Sí	Sí	Sí

## 7. Estudis i decisions

L'estructura, i les seves respectives tecnologies utilitzades, a nivell general del nostre sistema és el següent (veure fig. 3):

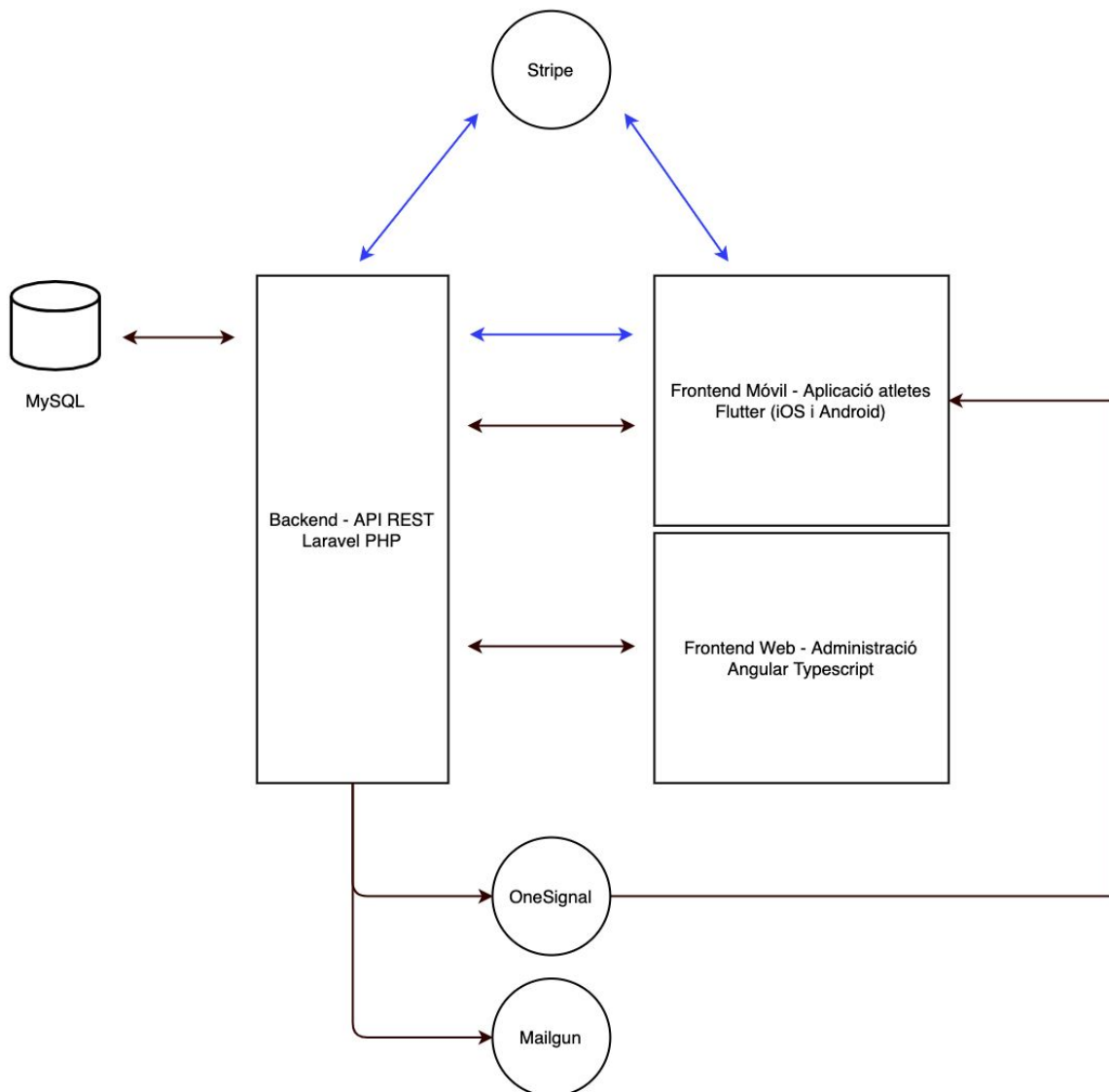


fig. 3: Diagrama de comunicació de les tecnologies utilitzades

És un sistema que implementa la filosofia de separar la lògica de negoci (backend) de les vistes (frontend).

## 7.1. Backend

Per un costat tenim el backend, on s'hi concentra tota la lògica de negoci, que es comunica amb els frontend a través d'una API Rest que ofereix com a servei. Està desenvolupat amb PHP 7, concretament amb el framework Laravel, una solució que per als nostres interessos ha resultat ser molt elegant a l'hora de desenvolupar. Com que serà una aplicació que calculem que, per naturalesa, tindrà poques peticions, no ha sigut problema el triar PHP com a llenguatge de desenvolupament del mateix tot i sabent que no és un llenguatge que destaquï per la seva velocitat. Ara bé, coneixem PHP i Laravel de l'entorn laboral i sabem que per a un projecte que tindrà una base de dades relacional (MySQL en aquest cas), notificaciones, integració amb pagaments, etc, seria ideal degut al coneixement que en teníem i el gran suport que hi ha a les xarxes.

Pàgina oficial de Laravel: <https://laravel.com/>

## 7.2. Frontend Web

Per la part de la web d'administració que utilitzaran els negocis per gestionar-se, utilitzarem Angular, un framework bastant recent de Google en TypeScript (JavaScript tipat) del qual ja tenim experiència programant i funcionaria perfecte per a un software que volíem que fos SPA (Single Page Application), donant així la impressió d'un petit ERP Web.

Pàgina oficial d'Angular: <https://angular.io/>

## 7.3. Frontend mòbil

La nostra aposta més arriscada ha estat utilitzar Flutter com a framework per desenvolupar les aplicacions mòbils. És un framework de Google en Dart, bastant recent que ens permet tenir un sol codi per Android i iOS, i que compila a codi natiu, així que no renunciem a l'experiència nativa de les aplicacions mòbils, com si fan altres frameworks més populars. Flutter és un framework que tot i les bones crítiques que té i que la seva evolució en popularitat és molt bona, té poc suport per part de la comunitat (tot i que en nou mesos, hem notat un augment d'aquesta bastant considerable).

Pàgina oficial de Flutter: <https://flutter.dev/>

## 7.4. Pagaments automàtics

Un aspecte que volíem implementar sí o sí eren pagaments automàtics, tant per la part de pagaments dels negocis de Crossfit cap a nosaltres, com per part dels atletes cap als negocis. Així que, per tal d'evitar implementar passarel·les de pagament, vam decidir utilitzar la solució més implementada i més popular utilitzada avui en dia que és Stripe i aprofitar tota la seva lògica de pagaments recurrents. També cal destacar que Laravel té una integració inicial (que hem hagut de personalitzar) amb Stripe.

Pàgina oficial de Stripe: <https://stripe.com/>

Documentació d'integració de Laravel amb Stripe: <https://laravel.com/docs/master/billing>

## 7.5. Notificacions push i correu electrònic

També volíem poder enviar notificacions ja sigui per correu o push als atletes i empreses. Així que hem utilitzat Mailgun i OneSignal (que a la vegada utilitza Firebase) respectivament, dos serveis molt utilitzats, estesos i estables que ens permetran tenir una gran fiabilitat a l'hora d'executar aquests procediments externs al nostre sistema. També cal destacar altre cop, el paper rellevant de Laravel ja que també ofereix integració amb aquests serveis. Sobre els correus, volíem utilitzar servidors de mail que tingués reputació, ja que per experiència, sabem que utilitzar servidors IMAP i SMTP propis has de treballar bé la reputació del mateix per tal que no arribin a spam. Així que vam decidir externalitzar-ho.

Pàgina oficial de Mailgun: <https://www.mailgun.com/>

Pàgina oficial de OneSignal: <https://onesignal.com/>

Documentació de Laravel amb notificacions: <https://laravel.com/docs/master/notifications>



# 8. Anàlisi i disseny del sistema

## 8.1. Diagrames de casos d'ús

Per tal de simplificar els diagrames, els hem separat per els diferents actors que inclou el nostre sistema:

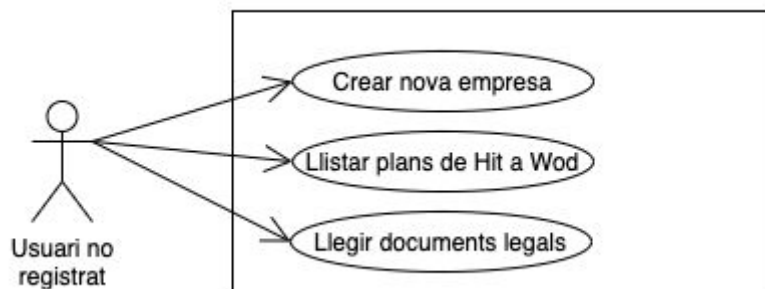


fig. 4: Cas d'ús - Usuari no registrat

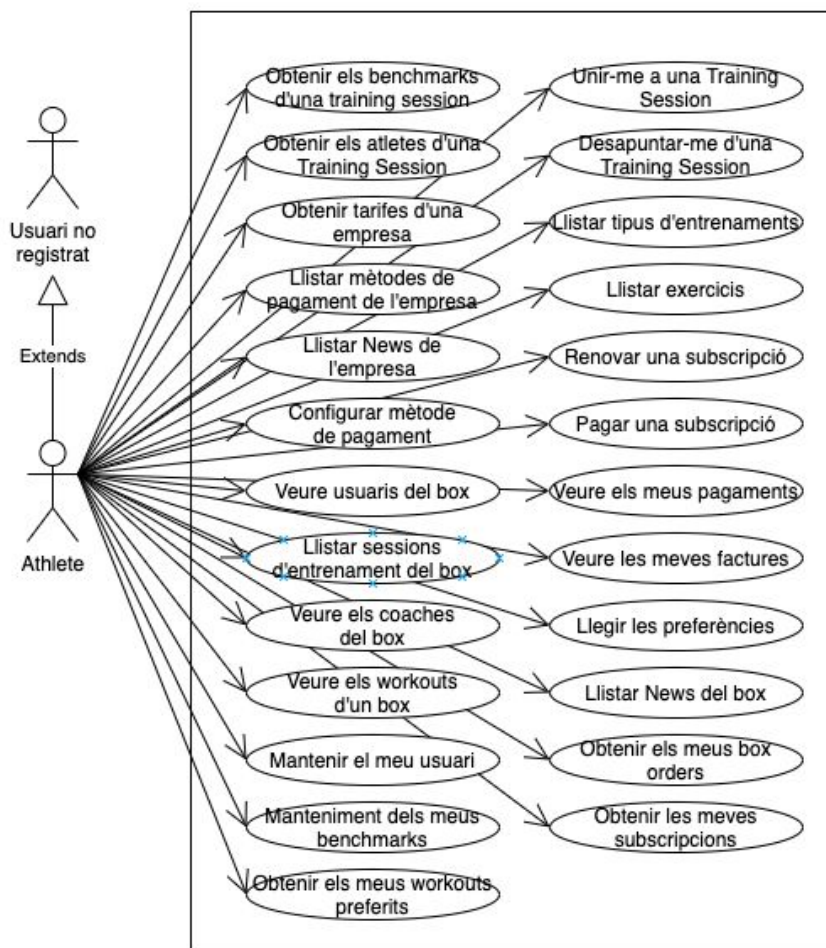


fig. 5: Cas d'ús - Atleta

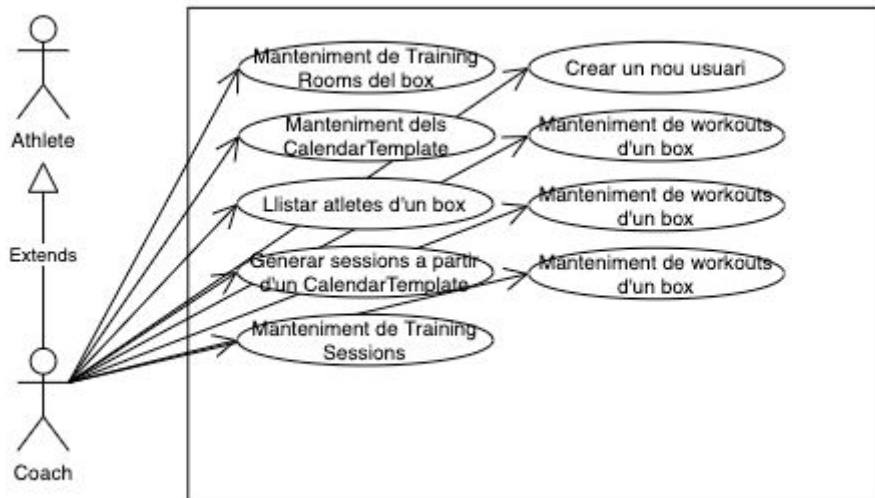


fig. 6: Cas d'ús - Coach

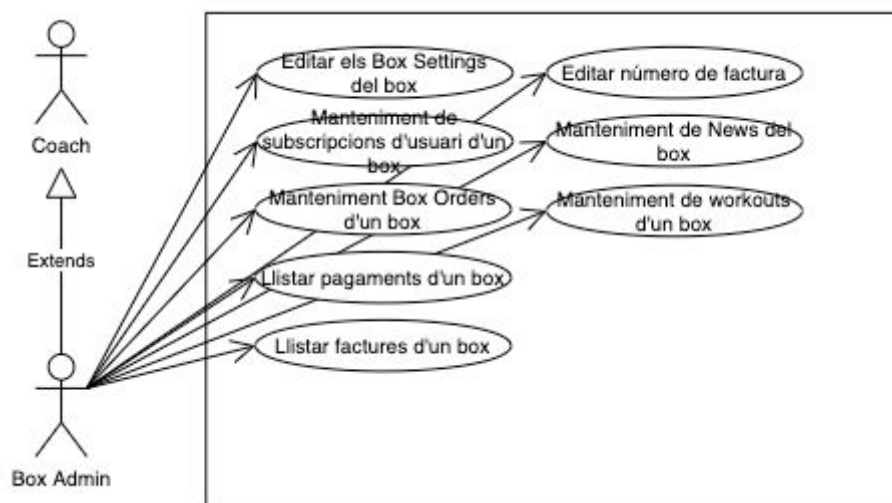


fig. 7: Cas d'ús - Box Admin

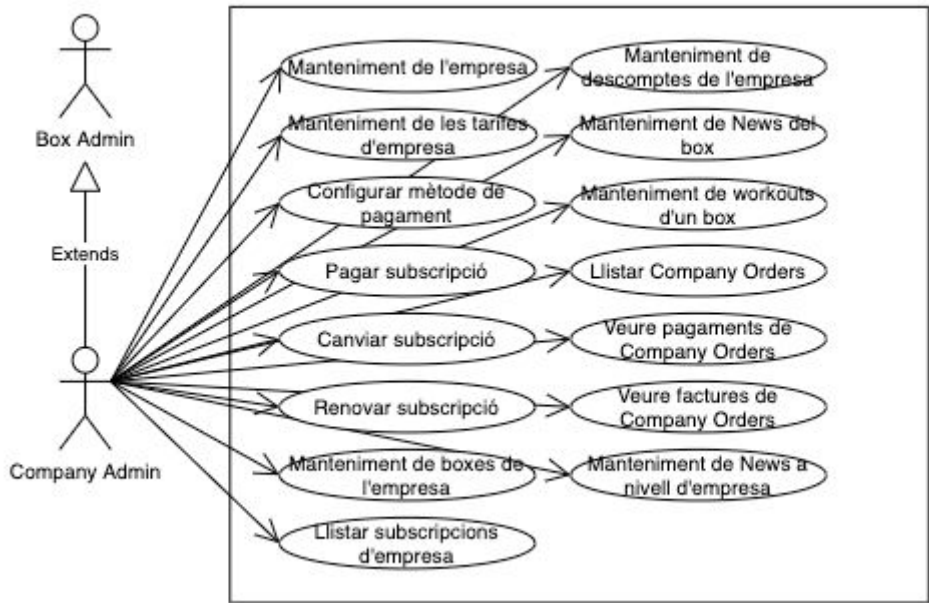


fig. 8: Cas d'ús - Company Admin

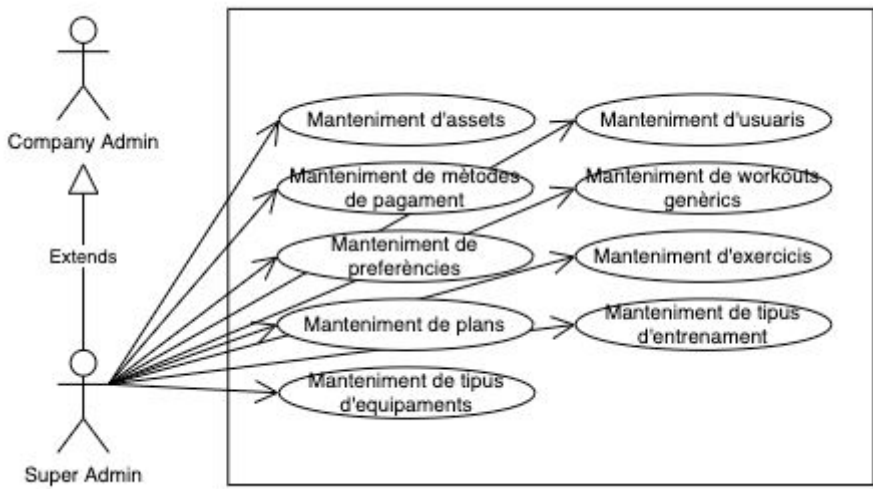


fig. 9: Cas d'ús - Super Admin



## 8.3. Entitats

Farem una breu explicació de cada entitat i mencionarem els seus atributs més importants i quina funció tenen:

- **Boxes:** Representa un centre físic. Cada empresa pot tenir varis.
- **Box Setting:** Representa les diferents opcions que estan associats a cada box i el seu valor. Contenen un descripció. Exemple de box setting: Minuts d'antelació amb els que un atleta es pot desapuntar d'una sessió.
- **Exercises:** Un exercici que es farà servir en els entrenaments, i dels que un atleta podrà tindrà un seguiment.
- **Box Order:** Comanda de subscripció o de la botiga (en futures versions). Té un pagament i una factura associada.
- **Box Order Line:** Línia de comanda, pertany a una comanda. Pot contenir descomptes i estar associat a una subscripció. Sempre representa una comanda de l'atleta al box.
- **Calendar Template:** Plantilles de cada box que modelitzen una plantilla setmanal d'entrenaments, en quant a horaris. Conté temps de duració per defecte d'una sessió (default\_duration), temps entre sessions (time\_between\_wods), l'hora que comença i acaba el dia (start\_day\_time, end\_day\_time), i els dies de la setmana que aplica (days, un array d'integers).
- **Calendar Template Exclusion:** Pertany a un calendar template i representa aturades en un dia, per exemple, per l'hora de dinar.
- **Company:** Empresa que té centres de Crossfit, el nostre client directe. Conté, entre altres, el nom, logo, informació comercial, de facturació i de contacte. També conté l'identificador de Stripe de client i de pagament (stripe\_id i stripe\_payment\_method\_id) que utilitzarem nosaltres per facturar-lo. També pot configurar les seves API Keys del seu Stripe per facturar als seus atletes (stripe\_public\_key i stripe\_private\_key).
- **Company Orders:** Representa una comanda de les companyies a Hit a Wod. Està estrictament lligat al pagament mensual de la seva tarifa. A l'igual que Box Order, té un pagament i una factura.
- **Discounts:** Es poden aplicar als box orders. Poden ser aplicats a valor absolut (en euros) o a nivell relatiu (en percentatge). Poden estar actius o no. Per exemple: hi ha centres que ofereixen descomptes a estudiants universitaris.
- **Invoice:** Una factura, amb totes les dades, tant de l'emissor com del receptor. S'utilitza tan en Box Orders com en Company Orders. Com a detall curiós: conté la numeració que es genera de forma automàtica per cada companyia, acceptant número, sèrie i prefix.
- **News:** Comunicats de notícies que poden emetre els negocis, tan a nivell de box com de companyia, per informar quan vulguin als atletes. Provoca notificacions, ja sigui correus o push.
- **Payment Type:** Tipus de pagament suportats. De moment tenim Credit Card (Stripe) i Cash (en efectiu), així podem saber si un pagament, i per tant una comanda, com ha estat pagada i si accepta recurrència (atribut auto).

- **Payment:** Pagament, relacionat sempre amb un Box Order o Company Order. Associat a un tipus de pagament, la quantitat total, i si ha estat cobrat o no.
- **Plan:** Tarifes que ofereix Hit a Wod a les empreses. De moment només hi ha Starter, Scaled i RX. Conté les restriccions de cada pla (n\_athletes, n\_admins, n\_coaches) i els preus (tax\_prcnt i total). I molt important, contenen l'identificador de tarifa a l'Stripe de Hit a Wod.
- **Preferences:** Paràmetres variables del sistema de Hit a Wod. Són parells de clau => valor, que utilitzen els diferents procediments. Contenen informació com la de facturació del negoci de Hit a Wod, correus de contacte de Hit a Wod, polítiques de privacitat, protecció de dades, cookies.
- **Pricing:** Tarifa que ofereixen les empreses als seus atletes. Una tarifa pot estar associada a un o varis boxes. Conté un identificador de Stripe per si es vol poder facturar de forma recurrent. Conté també el preu, el número de wods i open setmanals que ofereix i que poden fer els atletes de forma setmanal. També poden ser bonos (recharge\_weekly == false), que implica que no es recarreguen de forma setmanal. Les tarifes duren un cert temps (mensual, trimestral) i poden establir-ho amb l'atribut days. També pot ser que vulguin restringir la tarifa per hores del dia (start\_time i end\_time).
- **Role:** Sempre associat a una subscripció. Pot ser que un usuari sigui atleta, coach, box admin, company admin o super admin (en el cas d'usuaris de Hit a Wod). S'utilitza per restringir crides al API Rest, tant a nivell de frontend com de backend.
- **Stripe User:** És una entitat que neix de la relació many to many entre una empresa i un usuari. Serveix per guardar l'identificador de mètodes de pagament de Stripe per tal que puguem fer que una empresa cobri de forma recurrent a un usuari. Això implica que un atleta només estarà donant la targeta de crèdit a l'empresa en qüestió.
- **Subscription:** Determina si una empresa està en actiu o no en el sistema de Hit a Wod gràcies a les dates (start\_date i end\_date). Es guarda l'id de la subscripció de Stripe.
- **Training Room:** Un box pot tenir diferents sales d'entrenament. És on s'imparteixen les sessions. Té una capacitat per defecte.
- **Training Session:** És una sessió concreta que té dia i hora (day, start\_time i end\_time). Té un entrenador associat (coach\_id) i una sala d'entrenament (training\_room\_id). I pot tenir un entrenament establert (training\_workout\_id). També té un tipus d'entrenament (training\_type\_id). Té una capacitat (capacity), normalment igual a la del Training Room. credits és l'atribut que et diu per quantes classes conta aquella sessió (normalment 1).
- **Training Type:** Tipus d'entrenament d'una sessió, normalment serà WOD o Open Box. Però hi ha altres alternatives com Gymnastics, Olympics, Endurance, Power, etc.
- **Training Workout:** Representa un entrenament. Pot pertànyer a un box o no. Per exemple, els Wods genèrics (dels Crossfit Games, Hero, Girls, etc) sempre tindran box\_id == null, que vol dir que tothom els podrà veure, però no editar. En canvi els Training Workouts que tinguin box\_id != null, voldrà dir que només poden ser visualitzats per aquell box, i sí que seran editables. Pot tenir un o varis Training Workout Section

- **Training Workout Section:** Entitat feble de Training Workout, ja que pot tenir una o varies seccions. Una secció representa un AMRAP, un EMOM, etc, ja que un Workout pot estar format per un AMRAP seguit d'un EMOM a la vegada. Aquí la lògica ja es comença a complicar, ja que en funció de la tipologia de la secció (type) utilitzarà uns atributs o altres (seconds, rounds). L'atribut description s'utilitza com a notes aclaratòries de la secció. Pot tenir varies línies que representen els exercicis a realitzar.
- **Training Workout Line:** Entitat feble de Training Workout Section. Està directament relacionat amb un exercici (exercise\_id). Pot ser que et demanin que facis unes repeticions concretes (reps), les màximes que puguis en un temps (seconds), pot ser que siguin metres, pounds o feet (units), i la mesura es diferencia entre homes i dones (maleVal i femaleVal). També es pot proporcionar una alternativa Scaled per a cadascun d'aquests atributs.
- **User Benchmark:** Representa una valoració d'un usuari per una sessió (i per tant, workout).
- **User Benchmark Section:** Entitat feble de User Benchmark, relacionada amb una Training Workout Section. Si és un EMOM utilitza completed, si és un AMRAP utilitza rounds i reps, i si és un For Time utilitza time. Diu si has fet RX o Scaled amb l'atribut level.
- **User Benchmark Line:** Entitat feble de User Benchmark Section, relacionada amb una Training Workout Line. S'escriuen els atributs de weight, reps i sets, que poden ser o no redundants, depenent de si t'has regit al pes RX/Scaled especificat per la Training Workout Line.
- **User Subscriptions:** Una de les entitats més importants per entendre el funcionament del sistema. És la relació entre un usuari, una empresa i el rol que executa dins aquesta. Aquesta entitat ens permet que un usuari pugui ser coach al centre de Girona sense pagar res, i ser atleta amb tarificació automàtica al centre de Blanes. I per tant que un usuari i les seves dades no estiguin lligades a un centre en concret, més enllà de les de pagament que són exclusives de cada centre. Pot ser una subscripció il·limitada (end\_date == null), perfecte per coaches i administradors. Aquí també guardem informació de comptadors setmanals d'assistència per tal de saber si un atleta ha assistit a totes les sessions setmanals que li pertocuen per la seva tarifa. Es restableixen setmanalment. Hi ha l'identificador de subscripció de Stripe per tal de fer el cobrament recurrent. També l'atribut renew et diu si es renova de forma automàtica o no.
- **User:** Entitat que representa un usuari i les seves dades com nom, cognoms, informació de contacte, contrasenya, email, etc.

Nota: Les entitats que pengen de BoxProduct van estar contemplades inicialment però finalment per qüestions de temps no han estat inclosos en aquesta versió del sistema. Es contemplarà més endavant incloure una botiga per cada box on BoxProduct hi tingui protagonisme.

## 8.4. Taules relacionals

Farem una breu explicació sobre les diferents taules relacionals que existeixen i la seva funcionalitat principal.

- `box_box_settings`: Estableix, per cadascun dels settings que hi ha, el valor per cada box.
- `notify_user_training_sessions`: Serveix perquè un usuari que es vulgui apuntar a una sessió que està plena, sigui avisat via notificació (push o correu) quan un usuari es desapunti.
- `pricing_box`: Una relació many to many simple, que permet associar varis boxes a una tarifa d'empresa.
- `training_workout_user`: Un usuari podrà guardar els seus Training Workout preferits.
- `user_training_sessions`: Diu si un usuari està apuntat a una sessió.

## 8.5. Procediments més importants del sistema

En aquest apartat s'han detallat amb diagrames els procediments més complicats del sistema de Hit a Wod. Per als procediments en els que el flux de peticions era més complex i sobretot s'utilitzen serveis externs (com les API de Stripe, OneSignal o Mailgun) ens hem decantat per fer un diagrama de seqüència, ja que hem cregut que és el que més bé representa la complexitat del procediment, obviant llavors molts condicionals i excepcions que hi ha en els mateixos, els quals es poden veure detallats en el codi del programa si es desitja. Per als altres procediments, on l'algorísmica i els condicionals agafen un rol important, ens hem decantat per il·lustrar-ho amb diagrames d'activitat.



## 8.5.1. Alta nova companyia

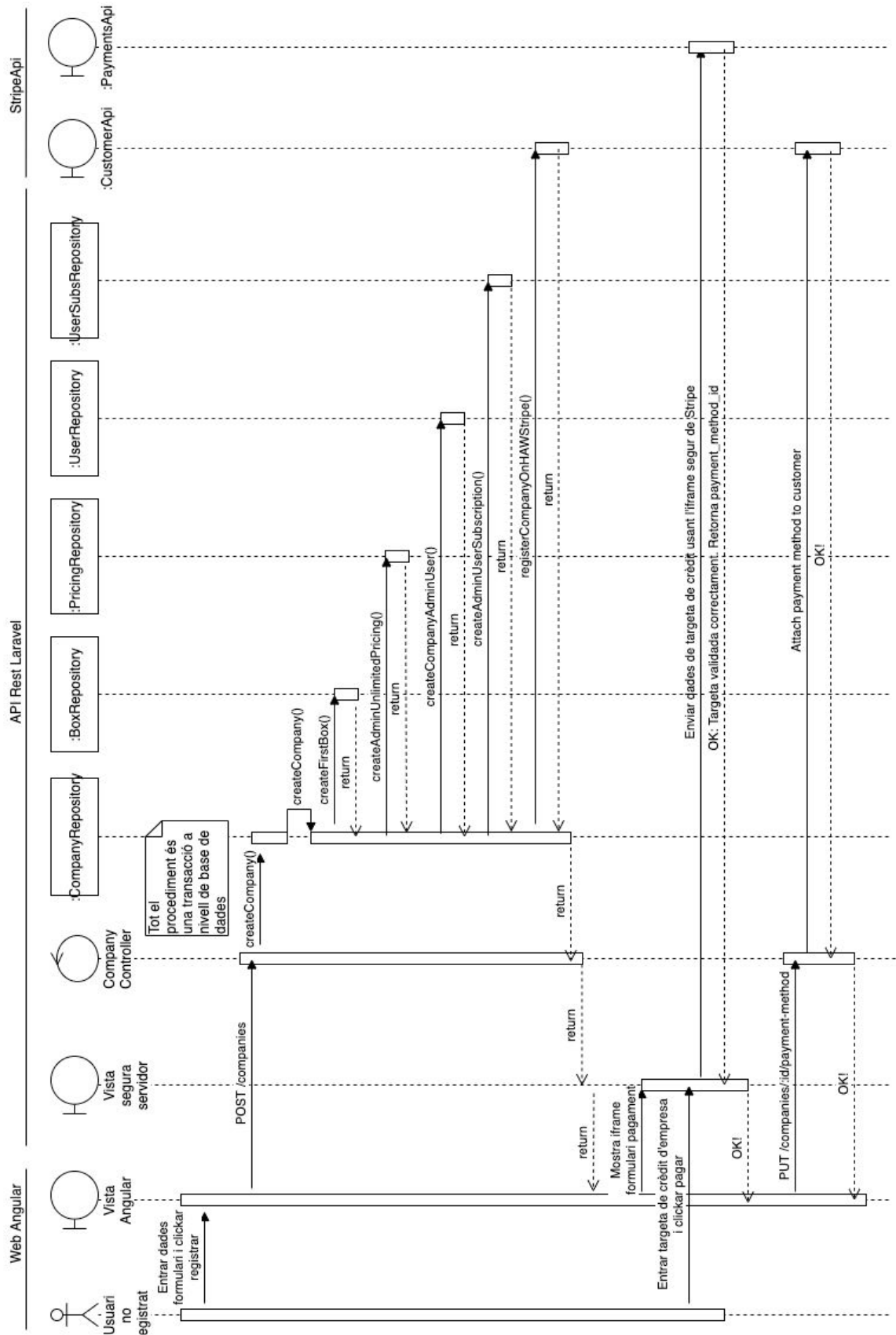


fig.11: Alta nova companyia



### 8.5.3. Autenticació i Autorització API Rest

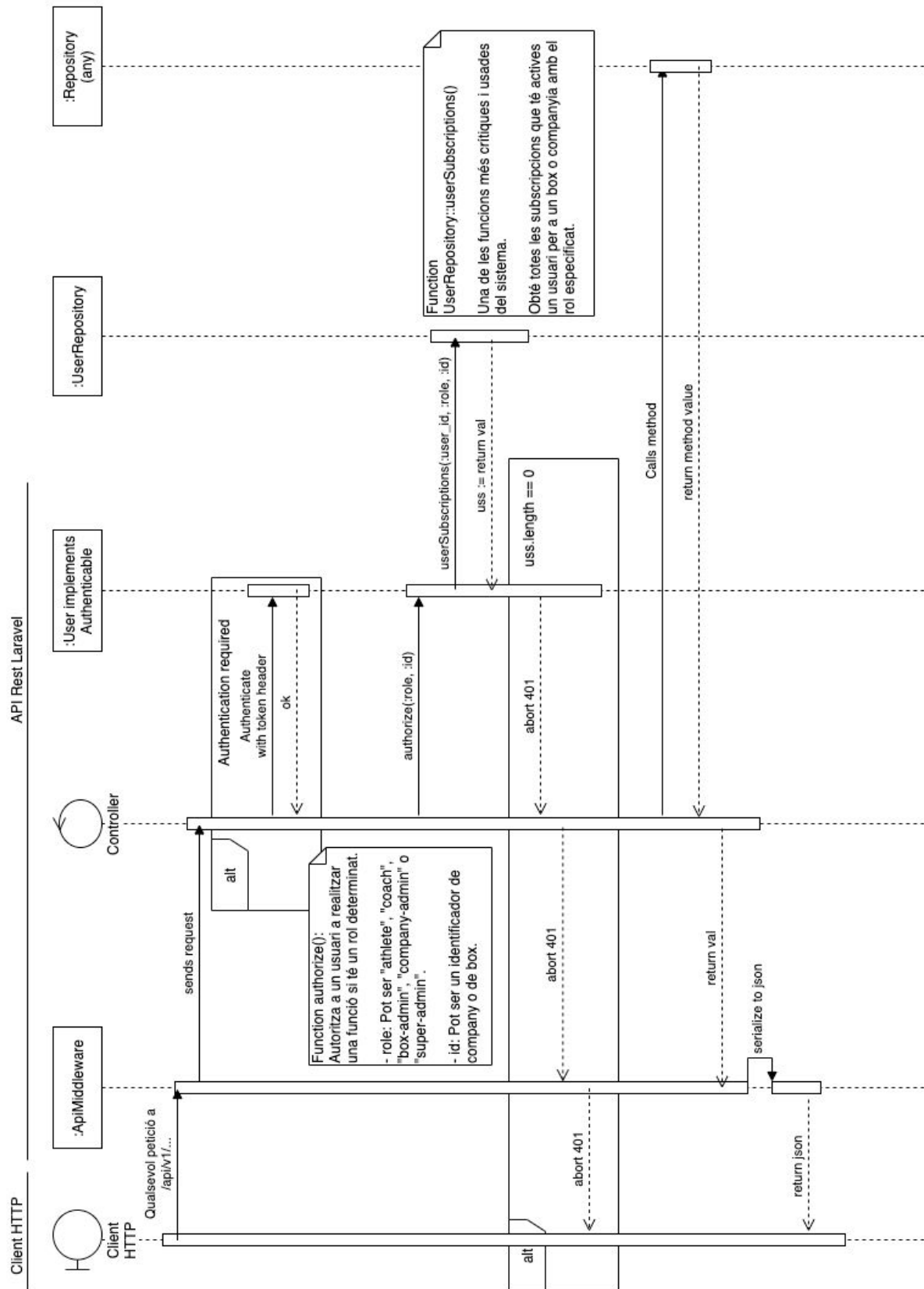


fig.13: Autenticació i Autorització API Rest

## 8.5.4. Angular Dashboard Business Intelligence Api Call

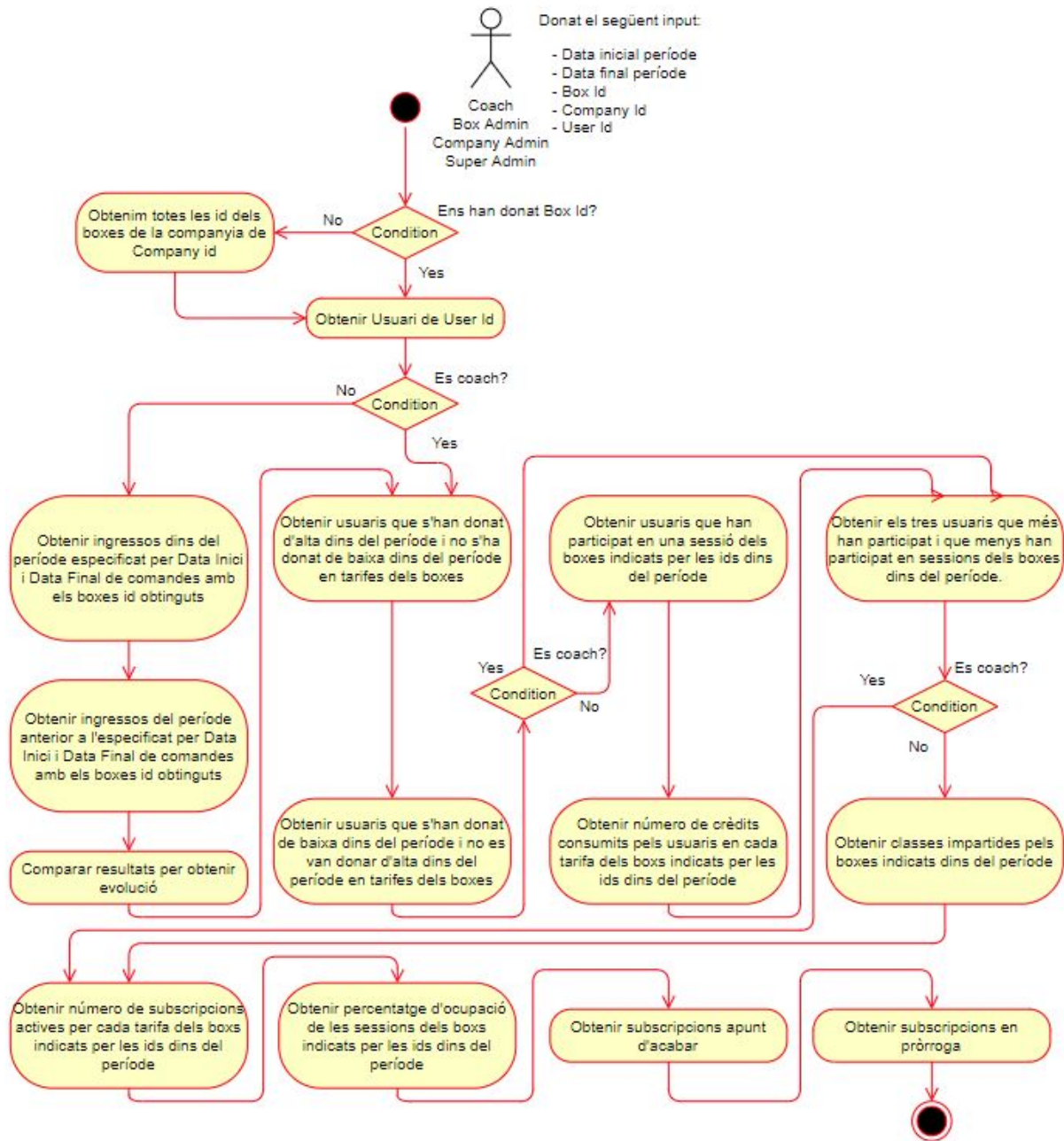


fig.14: Angular Dashboard

### 8.5.5. Generar sessions d'entrenament a partir d'una plantilla

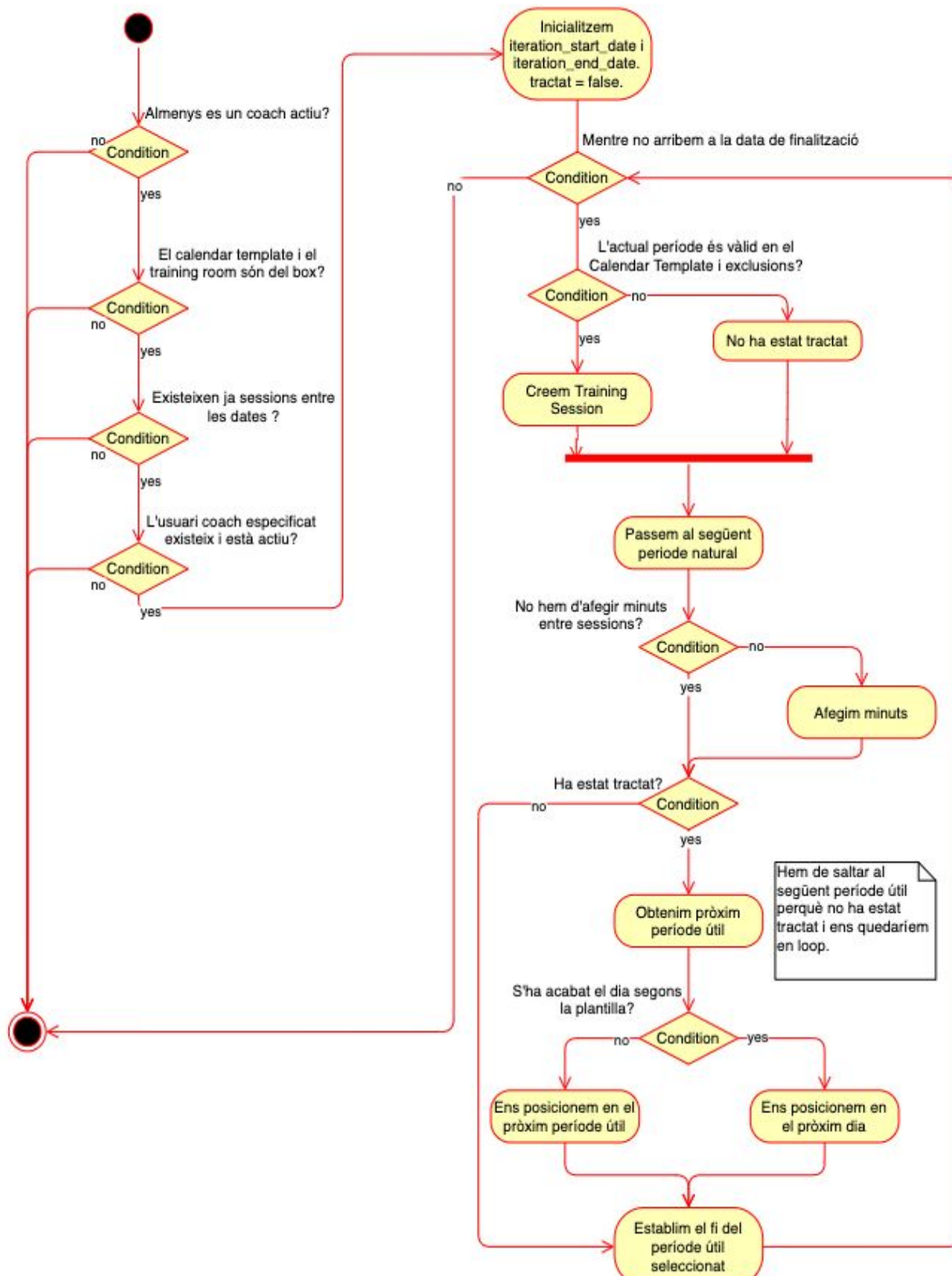


fig.15: Generar sessions d'entrenament a partir d'una plantilla

## 8.5.6. Apuntar un usuari a una sessió

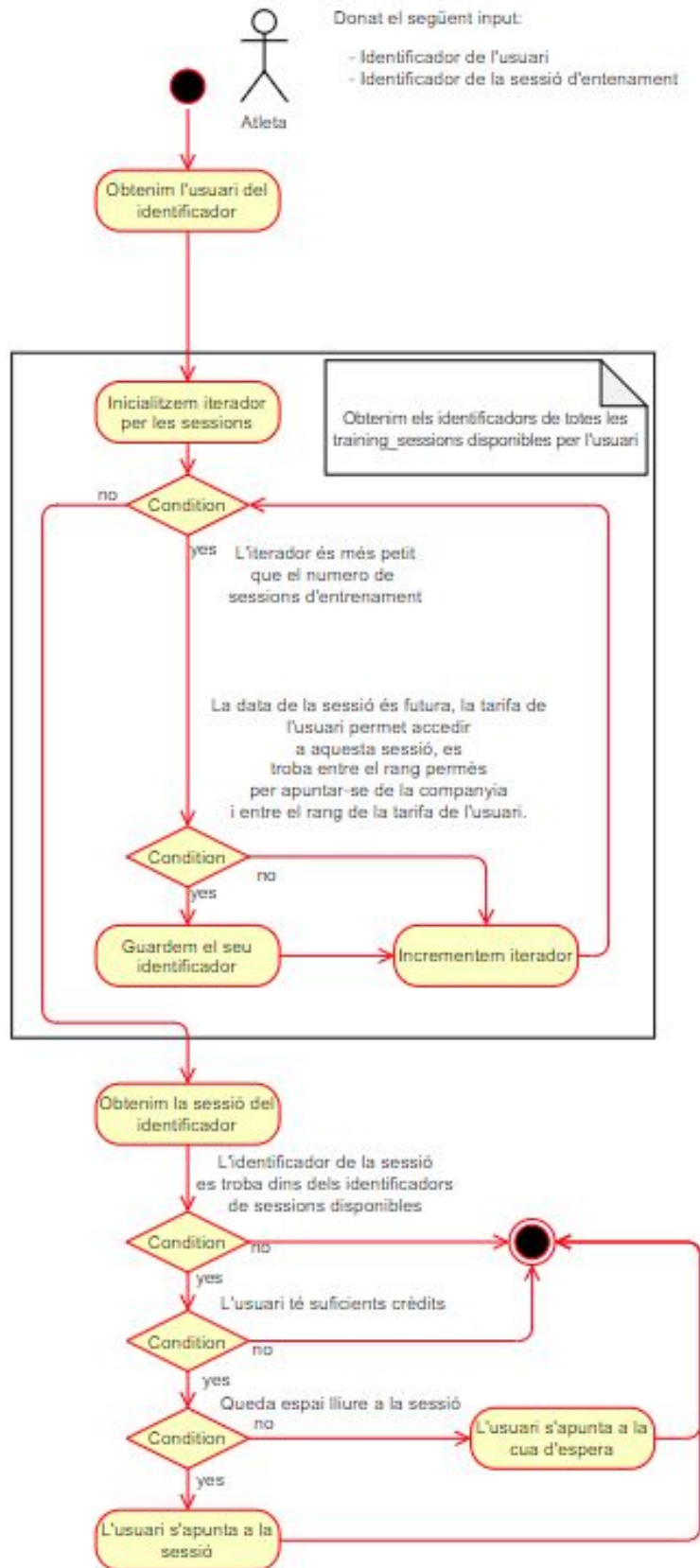


fig.16: Apuntar a un usuari a una sessió

## 8.5.7. Nova subscripció d'atleta

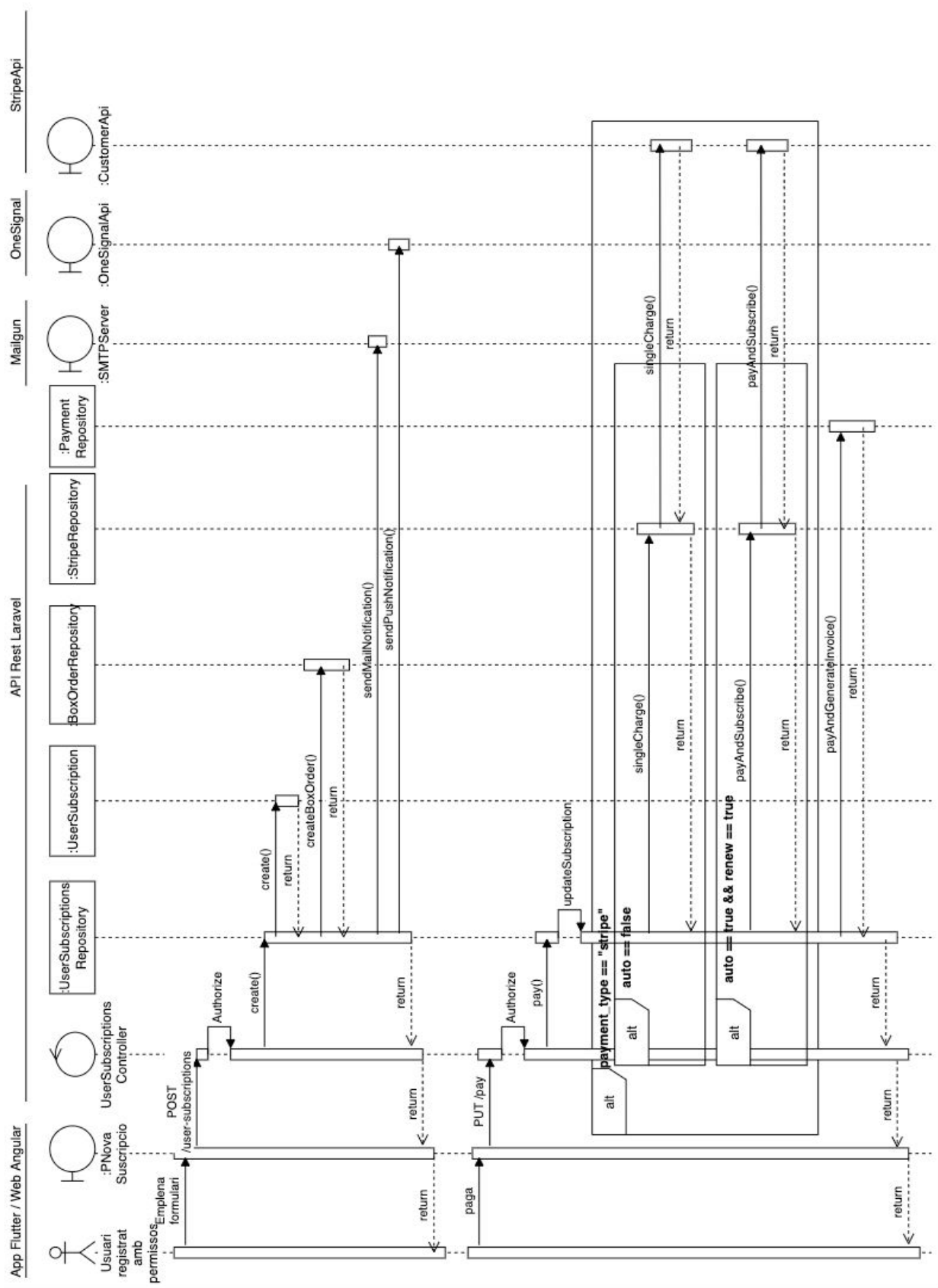


fig.16: Nova subscripció d'atleta

## 8.5.8. Atleta canvia de tarifa

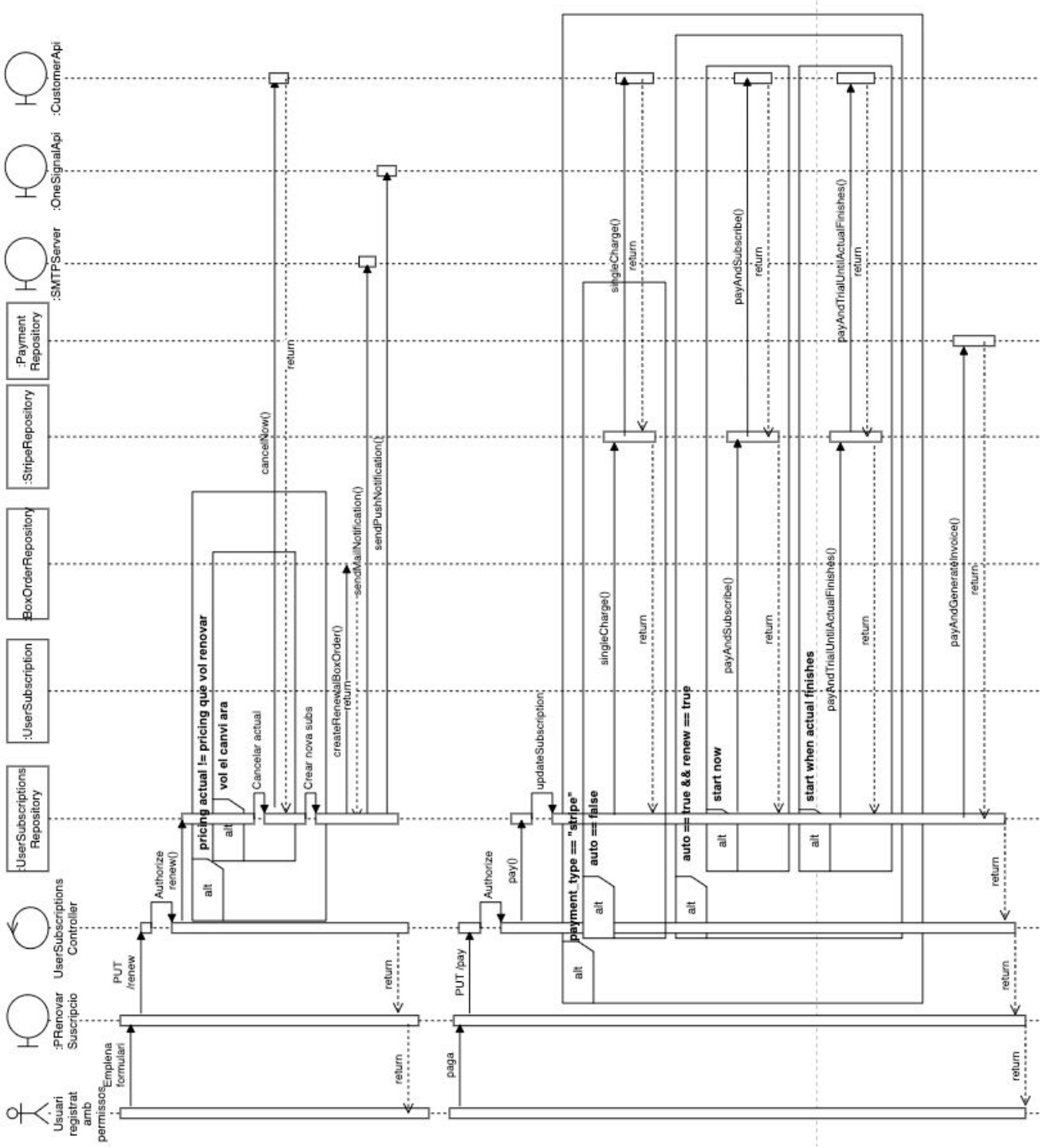


fig.17: Atleta canvia de tarifa



## 8.5.9. Empresa canvia de tarifa

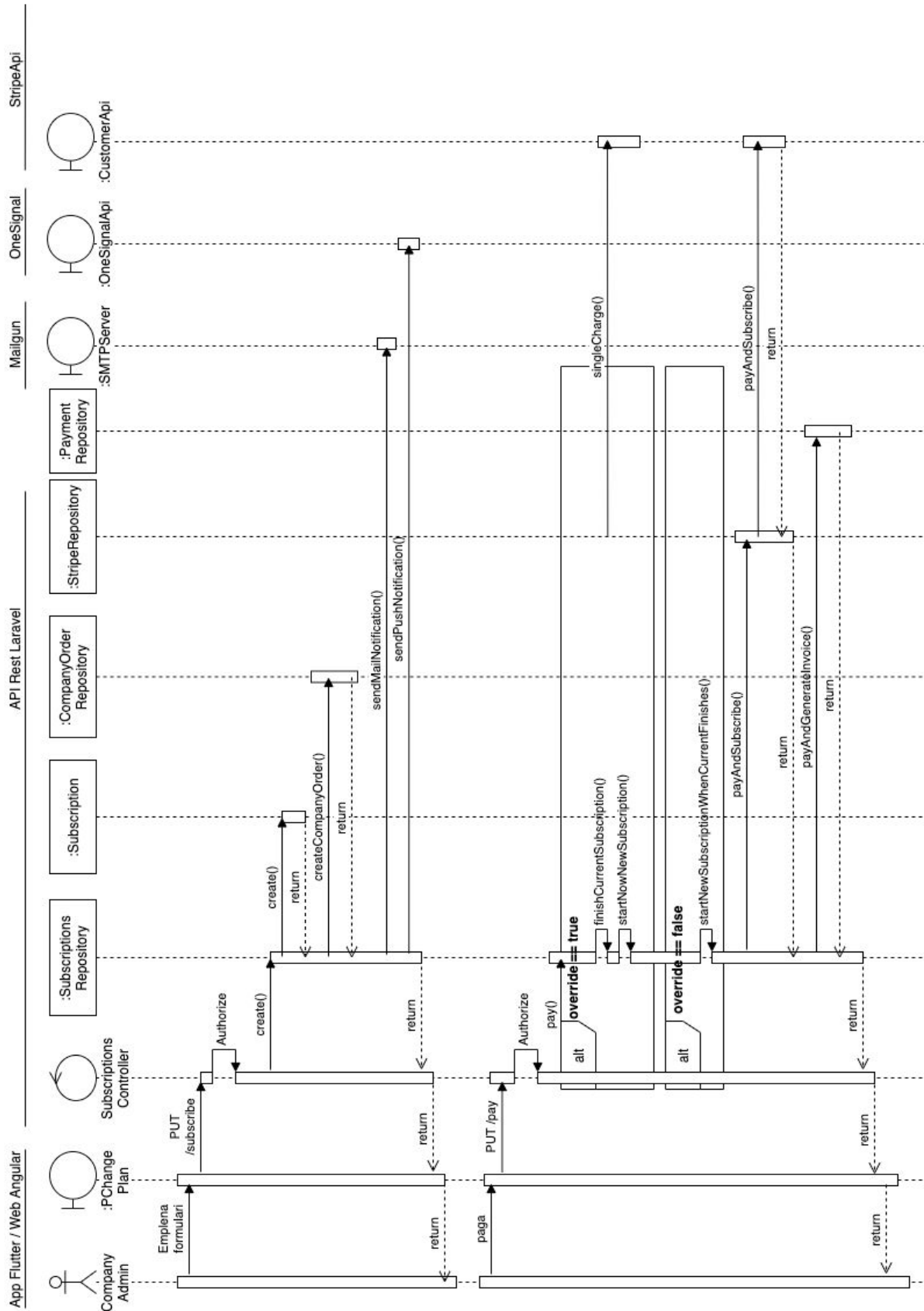


fig.18: Empresa canvia de tarifa

# 9. Implementació i proves

## 9.1. Frameworks utilitzats

### 9.1.1. Laravel

Explicar les diferents capes de Laravel que hi ha amb un diagrama de capes: Middleware, Controller, Repository, Model. I les vistes. I la capa de llibreries que ja porta Laravel en si com Cashier, Notifications, etc. Migrations, Seeder i Factories.

Laravel és un framework basat amb PHP molt popular i usat en la indústria informàtica. Tot i que és un framework tant per frontend com per backend, nosaltres l'hem utilitzat exclusivament de API Rest, excepte per les vistes de configuració de mètodes de pagament, ja que així hem pogut utilitzar la integració que té amb Stripe anomenada Laravel Cashier. Així que podem simplificar l'estructura de la nostra aplicació de Laravel de la següent manera (veure fig. 19):

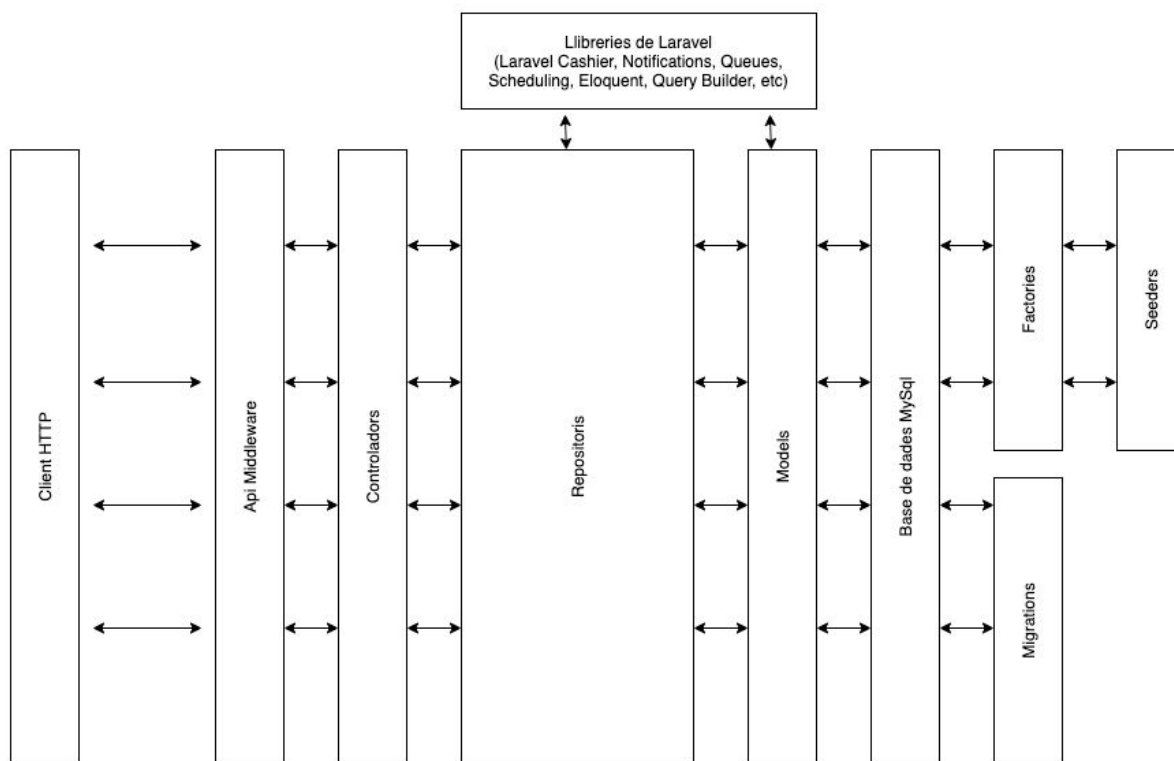


fig. 19: Estructura Laravel

### 9.1.1.1. Middleware

És la capa prèvia als controladors, que en moltes ocasions no es toca. En el nostre cas hem creat tres middlewares:

- Api: Encarregat de la serialització en json i modelitzar (ve per defecte amb Laravel)
- Paginació: Encarregat de paginar i afegir metadades a les crides en les que se'ls passa `pageSize` i `pageNumber` com a paràmetres query i la resposta és un array.
- Autenticació: Hem utilitzat la llibreria JWT per gestionar l'autenticació, tot i que ha estat personalitzada adaptant-se a les nostres necessitats. Bàsicament llegeix el paràmetre `token` o el header `Authorization` per identificar l'usuari. I si la crida requeria d'autenticació, el valida, i sinó no el deixa passar.

### 9.1.1.2. Migrations

Molt important en el període de desenvolupament, però més importants en el període de producció. Ens permet fer evolucionar la base de dades i tirar enrere amb cada deploy sense escriure cap línia de SQL, tot programat amb PHP. A més, ens permet reconstruir la base de dades, netejar-la, evolucionar-la o fer rollback amb una comanda, molt útil.

### 9.1.1.3. Notifications

Un dels punts forts de Laravel és que realment utilitza molts patrons ben considerats en l'enginyeria del software. El cas que ens ocupa n'és un exemple, ja que quan vols notificar un usuari (que extén `Notifiable`) tan sols has de cridar el mètode `notify()` amb els seus corresponents paràmetres, i ja s'encarrega d'enviar-li notificacions a partir dels canals configurats prèviament per l'usuari, ja sigui mail, push, sms o una llibreria de tercers que pots implementar com a canal de notificació. A cadascuna de les notificacions (`app/Notifications`) pots configurar el format del correu, el missatge de la notificació push, i els canals per els quals s'enviarà aquella notificació concreta. Una forma molt elegant d'abstreure l'enviament per diferents canals i de comprovar cada usuari quins canals té configurats.

### 9.1.1.4. Task Scheduling

Una característica molt interessant de Laravel és que pots encuar tasques per fer-les asíncrones. Per exemple, quan crees una empresa has de fer les següents tasques:

- Enviar una notificació a l'empresa conforme t'has donat d'alta
- Notificació a l'empresa conforme el pagament s'ha realitzat correctament
- Enviar les dades d'usuari al nou usuari creat

Fer scheduling d'aquestes tasques implica un augment de 2 a 5 segons aproximadament d'espera en la resposta de la crida que s'estava fent. Llavors, encuant aquests Jobs (així és com s'anomenen a Laravel, tot i que en el nostre projecte eren `Notifications`) guanyàvem molta rapidesa en tasques que al final l'usuari no tenia perquè esperar-se.

També ofereix fer scheduling ajudant-se del cron de linux per executar tasques cada cert temps.

### 9.1.1.5. Laravel Cashier

Laravel Cashier ens ha anat molt bé per aprofitar les crides a les diferents APIs que ofereix Stripe (principalment Payments API i Customers API). Però nosaltres necessitàvem una integració a dues bandes que no està contemplada per Laravel:

- Per una part, poder facturar a les empreses com a clients de Hit a Wod, és a dir, amb les claus de Stripe de Hit a Wod. Si només fos això podríem utilitzar Laravel Cashier per defecte tal i com ve.
- I per una altra part, poder facturar als atletes de les empreses per part de les empreses, és a dir, amb les claus de Stripe de les empreses.

Per tant, hem hagut de personalitzar bona part de Laravel Cashier per tal de poder-ho dur a terme. Petits exemples, per a la classe abstracta Billable, hem hagut de crear CustomStripeUserBillable i CustomCompanyBillable que l'extenen, i per el SubscriptionBuilder, hem hagut de crear CustomCashierUserSubscription i CustomSubscriptionBuilder. Bàsicament el que fan és agafar les claus de Stripe d'on toca i afegir les metadades que volem. Llavors, també va tocar que les entitats User, StripeUser, Subscription i UserSubscription extenessin aquestes classes abstractes. El codi es pot veure en el repositori de Laravel, a app/Extensions.

### 9.1.2. Angular

Explicar les diferents capes: Components, serveis, middlewares, interceptors, pipes, etc.

Angular és un framework de frontend basat en Typescript (Javascript tipat) desenvolupat i mantingut de la mà de Google i amb una creixent popularitat tot i ser un framework jove. L'estructura d'una aplicació Angular sol ser la següent (veure fig.20):

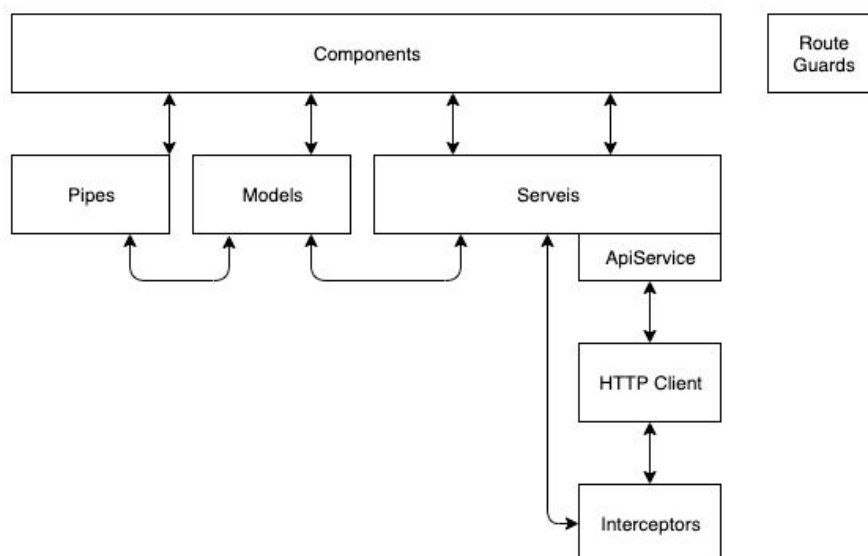


fig.20: Estructura Angular

### 9.1.2.1. Serveis

Capes d'aplicació accessibles des de tots els components. Molt utilitzats per crear una capa de servei d'interacció amb les API que utilitza l'aplicació, etc. En concret nosaltres tenim, com a més importants:

- **Api Service:** Conté la definició de totes les crides que necessita l'aplicació per al backend i serialitza les peticions transformant-les en models. Utilitza el client Http d'angular.
- **AuthenticationService:** Encarregat de dir si hi ha una sessió activa o s'ha de tornar a loguejar. També és l'encarregat de renovar les sessions per si un usuari no ha sortit. També, quan es fa login, és l'encarregat d'enviar el token al TokenService, l'usuari al UserService i d'avisar a l'aplicació que està tot okay. I fa tot el contrari quan es fa un logout.
- **SubscriptionService:** Encarregat de mantenir la llista de subscripcions de l'usuari loguejat i de mantenir i recordar (amb LocalStorage) la última subscripció activa, així quan tornem, no en triarà una qualsevol. També ofereix un observable amb el que els components es poden subscriure per si canvia la subscripció activa.
- **TokenService:** Encarregat de guardar, carregar i desxifrar el token que se li passa. Treballa amb LocalStorage també per guardar-lo. Cedeix el token als interceptors per col·locar-los a les crides a la API.
- **UserService:** Encarregat de mantenir l'usuari amb persistència (LocalStorage) en l'aplicació. També ofereix un observable perquè els components puguin obtenir l'usuari loguejat.
- **UtilsService:** Conté variables diverses. És un calaix de sastre. A destacar:
  - Controla les notificacions de succés, d'error i d'alerta que es mostren a l'aplicació. Ofereix un observable en la que els components s'hi poden subscriure.
  - Manté la paleta de colors de l'aplicació.
  - Controla i ofereix un observable per la barra de Loading de l'aplicació.

### 9.1.2.2. Interceptors

És la capa que es col·loca per on passen totes les crides HTTP de l'aplicació Angular.

Nosaltres hem creat un interceptor per:

- En les requests de la api, col·locar-hi el token del TokenService de forma automàtica, així no ho hem de fer una a una.
- També en les requests de la api activar el loading de forma automàtica fins que se n'obté una resposta, utilitzant el UtilsService.
- Un cop rebuda les respostes de la api, en cas de ser un error, mostra un error, i en cas de succés i de ser de tipus POST, PUT o DELETE, mostra un missatge de creació/actualització/esborrat satisfactòriament de forma automàtica.

### 9.1.2.3. Route Guards

Són els encarregats de restringir les rutes a certs rols d'usuari. Per exemple, els coach no poden accedir a la facturació o accedir a les variables d'empresa. Els route guards són els encarregats que això no passi.

### 9.1.3. Flutter

Flutter és un framework en Dart de frontend mòbil i web (només mòbil per nosaltres) desenvolupat per Google, molt jove (maig de 2017) i amb una creixent popularitat de forma accentuada. Els principals punts forts de Flutter és que pots mantenir un sol codi i aquest compila a codi natiu (Swift i Java/Kotlin) utilitzant llibreries en C++ i no és una solució híbrida que no compila a codi natiu com ara Xamarín, Ionic o React Native que utilitzen ponts per utilitzar mòduls nadius (si és que n'utilitzen).

Un dels punts fluixos de Flutter és que al ser tan jove, no hi ha un patró de programació i d'organització de fitxers, així que després de documentar-nos molt, hem establert la següent estructura que ens ha funcionat molt bé (veure fig.21):

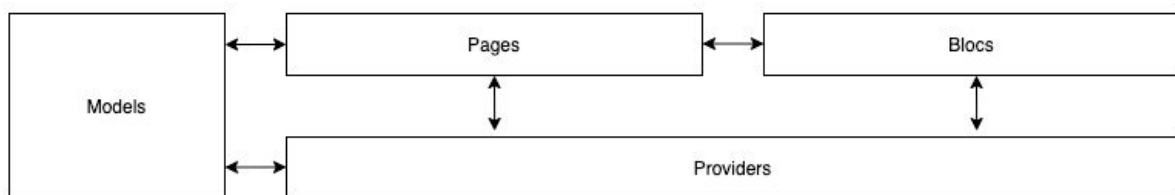



fig.21: Estructura Flutter

#### 9.1.3.1. Providers

Utilitzant el widget InheritedWidget de Laravel, es poden oferir capes superiors de forma eficient i que serveixen a les pàgines i a l'aplicació en general. Estan programats en forma de Singleton. Normalment ofereixen observables (Streams a flutter). Ve a ser el símil dels serveis en Angular. Nosaltres els hem usat per:

- ApiProvider: Una capa d'accés a la API de Laravel. Implementa un "interceptor" dins seu que s'encarrega de col·locar el token a les peticions per autenticar-les i de cridar els mètodes de serialització dels models. També aquest interceptor avisa al Snackbar provider per mostrar missatges de succés per els POST, PUT o DELETE o mostrar missatges d'error si és que retorna un codi de resposta  $\geq 400$ .
- AuthProvider: Encarregat de mantenir l'autenticació a l'aplicació. Conté el token i l'usuari en qüestió que guarda dins el dispositiu per mantenir persistència si l'usuari tanca l'aplicació. Ofereix observables per obtenir l'usuari. Encrypta la informació i té la possibilitat de protegir-se per autenticació biomètrica (Google Authentication, Touch ID o Face ID).
- MainScaffoldProvider: Encarregat de la navegació entre les cinc pàgines principals. Quan es clica un ítem del menú, crida a MainScaffoldProvider perquè redibuixi la pantalla.

- 
- `SnackBarProvider`: Encarregat de mostrar a l'scaffold els missatges de succès, alerta o error a l'scaffold.
  - `UserSubscriptionsProvider`: Encarregat de mantenir totes les subscripcions de l'usuari i de mantenir la subscripció activa. Ofereix observables perquè les pàgines s'hi apuntin. Així, quan un usuari passa del centre de Girona a Blanes, tota l'aplicació se n'adona i mostra la informació adient.

### **9.1.3.2. Blocs**

Sempre associats a una pàgina per una crida a l'api o per un formulari, són els pares de l'asíncronicitat per la que tant destaca Flutter. Contenen la lògica de negoci que pertoca al frontend, contenen validacions i sobretot ofereixen observables que permeten a les pàgines saber quan han de mostrar l'efecte loading (en el nostre cas, hem utilitzat skeleton) o mostrar la informació que ja ha arribat del servidor. Aquests observables són utilitzats per les pàgines amb un `StreamBuilder`.

## 9.2. Solucions als procediments més complicats

### 9.2.1. Alta nova companyia

Aquest procediment és un dels més importants ja que no només crea una nova empresa, sinó que ho deixa tot preparat a nivell de base de dades perquè l'empresa pugui començar a operar amb Hit a Wod. Com que toca Company, Box, Pricing, Plan, Subscription, UserSubscription i fins i tot treballa amb serveis externs (com Stripe), hem hagut de mantenir la consistència de les dades ajudant-nos de les transaccions de base de dades i fent un rollback en cas que hi hagués algun problema (veure fig.22).

Crear nou Customer a Stripe: <https://stripe.com/docs/api/customers/create>

Iniciar una nova subscripció a Stripe: <https://stripe.com/docs/api/subscriptions/create>

```
CompanyRepository.php
28
29 public static function create($c, $u, $b, $plan_id) {
30     DB::beginTransaction();
31     try {
32         //1. crear companyia
33         $company = Company::create($c);
34
35         //2. Crear box
36         $b["company_id"] = $company->id;
37         $box = BoxRepository::create($b);
38
39         //3. Crear pricing ilimitat
40         PricingRepository::create([
41             "company_id" => $company->id,
42             "name" => "Tarifa gratuïta ilimitada",
43             "description" => "Tarifa gratuïta ilimitada para administradores y coaches",
44             "price" => null,
45             "vat_included" => null,
46             "wod_counter" => 100,
47             "open_counter" => 100,
48             "total_counter" => 100,
49             "is_public" => false
50         ], [$box->id]);
51
52         //4. crear usuari administrador
53         $user = UserRepository::create($u);
54
55         //5. crear subscripció per el plan de la companyia
56         $sub = SubscriptionRepository::subscribe($company->id, $plan_id);
57
58         //6. crear subscripció per administrador
59         $user_sub = UserSubscriptionRepository::createCompanyFirstAdminUserSubscription($company->id, $user->id, $
60
61         //7. Registrar a Stripe
62         $stripe_data = $company->createAsStripeCustomer(StripeRepository::companyToStripeFormat($company));
63
64         DB::commit();
65     }
66     catch(\Exception $e) {
67         //gestionar transaccions
68         DB::rollBack();
69
70         throw $e;
71     }
72
73     //8. Enviar notificació
74     $company->notify((new NewCompanyNotification($company, PlanRepository::find($plan_id)))->delay(Carbon::now()->
75
76     $user_sub->refresh();
77     $user->refresh();
78
79     return [
```

fig.22: Codi repositori de Companyia, crida create



## 9.2.2. Atleta configura mètode de pagament

És el procediment amb el flux de peticions més complexe del sistema. Era la primera vegada per nosaltres que implementàvem un flux de peticions com aquest i al començament no va ser gens fàcil, ja que t'imagines que funciona d'una manera segons com s'explica a la documentació de Stripe i llavors és d'una altra. La màxima que té Stripe a l'hora de funcionar, és que els identificadors de mètodes de pagament no viatgin mai al client frontend i tot quedi guardat al backend, d'aquí neixen els **SetupIntent**. Quan comprens això, és quan entens el perquè de la complexitat. En el respectiu diagrama de seqüència que hi ha en aquest document (veure fig.23) es pot veure el flux detallat perfectament.

Més informació sobre els SetupIntents i el workflow de Stripe:

<https://stripe.com/docs/payments/save-and-reuse>

Exemple sobre com es configura des del client web Angular o des de Flutter:

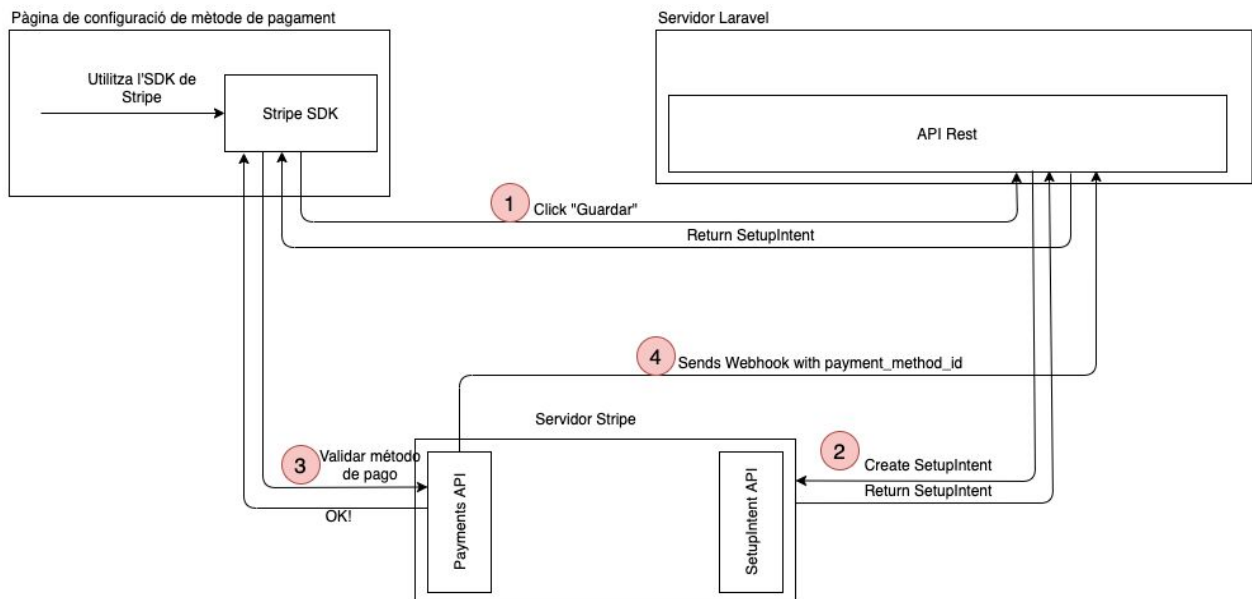


fig.23: Estructura configurar mètode de pagament

### 9.2.3. Autenticació i Autorització de l'API Rest

Treballant en l'entorn laboral ens hem fixat que moltes de les aplicacions a nivell de backend sí que fan autenticació, però no fan autorització de peticions. Per exemple, portant-ho al nostre terreny, podríem dir que un coach pot renovar les subscripcions d'una empresa o canviar de tarifa. En el nostre sistema això no passa. Donat un usuari i una empresa, el sistema, per cada petició, comprova que l'usuari tinguin permissos (els permissos estan detallats a l'apartat de requisits). No només ho capem a nivell de frontend (angular i flutter) sinó que també ho capem a nivell de backend.

La funció que realitza això és la de `User::authorize(string $role, int id, array params)`.

L'explicació del seu funcionament és:

- **role:** Els seus valors poden ser "athlete", "coach", "box\_admin", "company\_admin" i "super\_admin". Si es proporciona qualsevol altra string, s'aborta la petició. Si l'usuari és super admin, no mira cap subscripció, directament permet continuar la petició.
- **id:** Si role és "athlete", "coach" o "box\_admin", aquest és un identificador de box. Altrament, si és "company\_admin", és un identificador d'empresa.
- **array params:**
  - ["active"] => Per defecte és true. Estableix si es necessita que la subscripció d'usuari segons la qual s'acceptarà l'autorització hagi de ser activa o no.
  - ["company\_active"] => Per defecte és true. Estableix si es necessita que l'empresa segons la qual s'està demanant l'autorització ha de ser activa o no.
  - ["level"] => Valors disponibles: "default" i "all". Per defecte és "default". Indica si volem que el comportament del paràmetre id sigui l'especificat a dalt, o en canvi, si és "all", impliqui que "id" és a nivell de companyia donat un "athlete", "coach" o "box\_admin". És un paràmetre que s'ha afegit perquè hi havia peticions que donat un box\_id, necessitava autoritzar a un company\_admin.

Aquest mètode utilitza el mètode `UserRepository::userSubscriptions` que li retorna totes les subscripcions que coincideixen amb els paràmetres especificats anteriorment. Si n'existeix alguna, vol dir que té permissos, sinó, s'aborta la petició.

Un exemple real sobre autorització (veure fig.24):

```
NewsController.php
43  /**
44   * @param Request $request
45   * @return \Illuminate\Http\JsonResponse
46   *
47   * S'ha d'especificar sí o sí box_id, per temes d'autorització. Payment no té referència directa amb box o company
48   */
49  public function create(Request $request) {
50      //1. Params
51      $box_id = $request->box_id;
52      $company_id = $request->company_id;
53
54      //Authorize
55      if($company_id != null) auth()->user()->authorize("company_admin", $company_id);
56      else if($box_id != null) auth()->user()->authorize("box_admin", $box_id);
57      else return response()->json(["error" => "Esta noticia no tiene ni company_id ni box_id."], status: 400);
58
59      //3. Create
60      try {
61          $new = NewsRepository::create($request->all());
62      }
63      catch(\Exception $e) { return response()->json(["error" => $e->getMessage()], status: 400); }
64
65      return response()->json($new, status: 200);
66  }
```

fig.24: Codi repositori de Notícies, crida create

Com que una New pot ser a nivell d'empresa o de box, es mira en el body de la request a quin nivell pertany. Un cop es té, es mira si es té permisos per aquell box / companyia per procedir a cridar el mètode en el Repositori. Tots els mètodes de controlador tenen aquesta estructura:

1. Obtenir paràmetres (ja sigui del body o del request, encara que siguin implícits)
2. Autoritzar (si és que s'ha de fer)
3. Crida al mètode de repositori que conté la lògica de negoci amb tractament d'errors (try, catch).

## 9.2.4. Angular Dashboard Business Intelligence Api Call


L'objectiu que teníem creant un dashboard és fer que el client pugui tenir una primera impressió de l'estat del negoci amb una primera ullada, però, depenent del rol de l'usuari que estigui visualitzant el dashboard, tindrà unes limitacions o unes altres.

Tota la pantalla del dashboard té filtre de dates per estudiar un període en concret,

El "company admin", si no té cap box seleccionat veurà les dades respecte a tots els box que té, el "box admin", només veurà informació respecte al seu box igual que el "coach", que aquest últim tindrà les següents limitacions:

Nom	Descripció	Informació donada	Compan y Admin	Box Admin	Coach
<b>Ingressos</b>	Resum de l'evolució dels ingressos	Número i gràfic lineal d'evolució durant dues dates	✓	✓	X
<b>Activitat d'usuaris</b>	Usuaris nous i usuaris que s'han donat de baixa	Número i percentatge respecte del període anterior	✓	✓	✓
<b>Activitat al Box</b>	Número d'usuaris que han visitat el box	Gràfic de barres	✓	✓	✓
<b>Consum de crèdits</b>	Percentatge de consum de crèdits respecte a cada tarifa	Gràfic de barres	✓	✓	X
<b>Màx. i Mín. usuaris</b>	Usuaris que han vingut més i usuaris que han vingut menys	Llista d'usuaris	✓	✓	X
<b>Classes impartides</b>	Classes impartides	Gràfic lineal d'evolució durant dues dates	✓	✓	✓
<b>Subscripcions a tarifa</b>	Número d'usuaris inscrits en cada tarifa	Gràfic circular	✓	✓	X
<b>Ocupació del box</b>	Percentatge d'ocupació de les classes en cada box	Gràfic de barres	✓	✓	✓
<b>Subscripcions a punt d'acabar</b>	Usuaris amb subscripcions a punt d'acabar	Llista d'usuaris - tarifa	✓	✓	✓
<b>Subscripcions a pròrroga</b>	Usuaris amb subscripcions en pròrroga	Llista d'usuaris - tarifa	✓	✓	✓

Per culpa del treball que comporta per al servidor, aquesta crida pot trigar un temps, és per això que vam decidir crear un servei dins la web que recupera aquestes dades en background inclús quan estàs a una altra pàgina per estalviar el temps d'espera quan



l'usuari vol consultar la seva informació, com també no repeteix la crida si no ha passat un període de temps on han pogut veure's afectades les dades o l'usuari no fa ús d'un botó per refrescar les dades.

### 9.2.5. Generar sessions d'entrenament a partir d'una plantilla

Un dels objectius que teníem amb Hit a Wod era acostumar als coaches dels centres de Crossfit a utilitzar el nostre software per planificar els entrenaments i els horaris de la setmana. Això implica estar-s'hi el menor temps possible. Així que una de les funcionalitats que hem implementat en aquest sentit, són les plantilles d'entrenament que permeten generar amb un sol click una setmana d'entrenaments. Aquest procés, on el frontend especifica:

- data inici
- data fi (normalment una setmana)
- Identificador de la plantilla a utilitzar per la generació
- Identificador de la sala d'entrenament a utilitzar
- Tipus d'entrenament que es vol usar en les sessions generades (WOD, Open, etc)
- Coach que realitzarà les sessions.

en el backend es realitzen totes les comprovacions de lògica de negoci com ara:

- Mirar que no hi hagi sessions entre mig del període per evitar solapaments.
- Que el coach estigui en actiu en el box especificat
- Que data inici sigui inferior a data fi
- Llavors, per cada possible sessió es comprova:
  - Que el dia de la setmana sigui correcte en funció de l'atribut days del CalendarTemplate
  - Que estigui dins l'horari especificat per el CalendarTemplate
  - No estigui exclòs per algún CalendarTemplateExclusion del CalendarTemplate, com per exemple una exclusió per l'hora de dinar.

Llavors, els coaches ja podran començar a especificar mitjançant l'aplicació Angular els workouts que pertiquen a cada dia (veure fig.25).

```

CompanyRepository.php | BoxRepository.php
287 public static function generateSessions($id, Carbon $start_date, Carbon $end_date, $calendar_template_id, $training_room_id, $training_type_id = null, $coach_id = null) {
288     //0. Checks inicials
289     $calendar_template = CalendarTemplateRepository::find($calendar_template_id);
290     $training_room = TrainingRoomRepository::find($training_room_id);
291     if($calendar_template->box_id != $id || $training_room->box_id != $id) throw new \Exception("Calendar Template and Training Room must be of the box provided.");
292     if(sizeof(TrainingSession::where(["box_id" => $id, "day" => $start_date, "day" => $end_date, "training_room_id" => $training_room_id])->get()) > 0){
293         throw new \Exception("There are already training sessions.");
294     }
295     if($coach_id && !UserRepository::find($coach_id)->isCoach($id)) {
296         throw new \Exception("Please provide an active coach or unset the coach_id property.");
297     }
298
299     //1. Check if end_date is greater than start_date. Avoid infinite loops!
300     if($start_date >= $end_date) throw new \Exception("start_date is greater than end_date. Please provide correct dates.");
301
302     //2. Generar -- Ho fem transactionalment
303     DB::beginTransaction();
304     try {
305         $res = [];
306         $iteration_start_date = $start_date->copy();
307         $iteration_end_date = $iteration_start_date->copy()->addMinutes($calendar_template->default_duration);
308         $tractat = false;
309         while($iteration_end_date <= $end_date) {
310             $corrected_start_date = $iteration_start_date->copy()->setTimeFromTimeString($calendar_template->start_day_time);
311             $corrected_end_date = $iteration_start_date->copy()->setTimeFromTimeString($calendar_template->end_day_time);
312
313             $bool1 = in_array($iteration_start_date->dayOfWeekIso, $calendar_template->days) ? 1 : 0; //Mirem que el dia de la setmana sigui acceptat
314             $bool2 = in_array($iteration_end_date->dayOfWeekIso, $calendar_template->days) ? 1 : 0;
315             $bool3 = ($corrected_start_date <= $iteration_start_date && $corrected_end_date > $iteration_start_date) ? 1 : 0; //Mirem que estigui dins els horaris normals de calendar template
316             $bool4 = ($corrected_start_date <= $iteration_end_date && $corrected_end_date > $iteration_end_date) ? 1 : 0;
317             $bool5 = !CalendarTemplateRepository::isExcluded($calendar_template, $iteration_start_date) ? 1 : 0; //Mirem que no estigui exclos
318             $bool6 = !CalendarTemplateRepository::isExcluded($calendar_template, $iteration_end_date) ? 1 : 0;
319             //var_dump("Bools: ", $bool1, $bool2, $bool3, $bool4, $bool5, $bool6);
320             if( $bool1 && $bool2 && $bool3 && $bool4 && $bool5 && $bool6 ) {
321                 //Podem crear la sessió
322                 $data = [
323                     "box_id" => $id,
324                     "training_room_id" => $training_room_id,
325                     "day" => $iteration_start_date,
326                     "start_time" => $iteration_start_date,
327                     "end_time" => $iteration_end_date,
328                     "capacity" => $training_room->capacity
329                 ];
330                 if($training_type_id) $data["training_type_id"] = $training_type_id;
331                 if($coach_id) $data["coach_id"] = $coach_id;
332                 $session = TrainingSessionRepository::create($data);
333                 array_push($res, $session);
334                 $tractat = true;
335             }
336             else $tractat = false;

```

fig.25: Codi repositori de Box, crida generateSessions

## 9.2.6. Apuntar un usuari a una sessió d'entrenament

Quan el nostre frontend recupera les dades de les sessions, el backend adjunta un atribut fictici a la sessió anomenada "stat".

Aquesta propietat de l'objecte és un String que ens dona informació de què pot fer l'usuari amb aquesta sessió, si pot apuntar-se, pot sortir, pot ingressar resultats, ...

Si l'usuari pot apuntar-se a la sessió, el botó del frontend mostrarà un "Apúntame".

El backend, per saber que realment un usuari pot apuntar-se a la sessió comprova diferents coses:

- La sessió encara no s'ha realitzat.
- El marge per apuntar-se segons els paràmetres de la companyia és més petit que la diferència entre l'hora de la sessió i l'hora actual.
- L'horari de la tarifa de l'usuari permet apuntar-se a la sessió
- L'usuari té crèdits suficients per apuntar-se
- La capacitat de la sessió és major al número d'usuaris apuntats
- L'usuari té una tarifa activa

Encara que el frontend tingui un atribut del backend que permet apuntar a l'usuari, el backend en el moment d'apuntar-lo, tornarà a comprovar si és possible fer-ho, de cara a evitar problemes per si el frontend tenia informació desactualitzada.

En cas que l'única condició que no compleix l'usuari sigui la de capacitat, permetem a l'usuari apuntar-se a la "cua", on si en algun moment queda algun lloc lliure perquè algú marxa, aquest serà notificat mitjançant una notificació push.

## 9.2.7. Llegir webhooks de Stripe

El nostre servidor ha de saber d'alguna manera si les renovacions han estat satisfactòries o no per part d'Stripe. Pot ser per múltiples motius que s'escapen del nostre control: insuficiència de saldo, requerir una acció extra per temes legals d'Europa amb 3DSecure i SCA, mètode de pagament eliminat, etc. El mètode que proposa Stripe és configurar un endpoint en el nostre servidor on ells enviaran les notificacions. Així doncs, usant unes guies que generarem per les empreses, hauran de configurar els Webhooks de Stripe per tal que apuntin als nostres servidors i nosaltres poder processar les renovacions automàtiques i seguir renovant les subscripcions dels atletes i empreses. En el nostre cas, l'endpoint és el POST /api/v1/secure/stripe/webhook. El que fa aquest endpoint és dexifrar els payload que envia Stripe per saber de quin tipus d'actualització es tracta.

Per facilitar-nos la feina, quan creem un nou customer (ja sigui un Company o un User), afegim a les metadades un user\_id o un company\_id per identificar posteriorment en els webhooks per a qui està dirigit aquell webhook. I també amb les subscripcions (ja sigui Subscription o UserSubscription), afegim a les metadades un subscription\_id o user\_subscription\_id per identificar-les quan arribin els webhooks (veure fig.26).

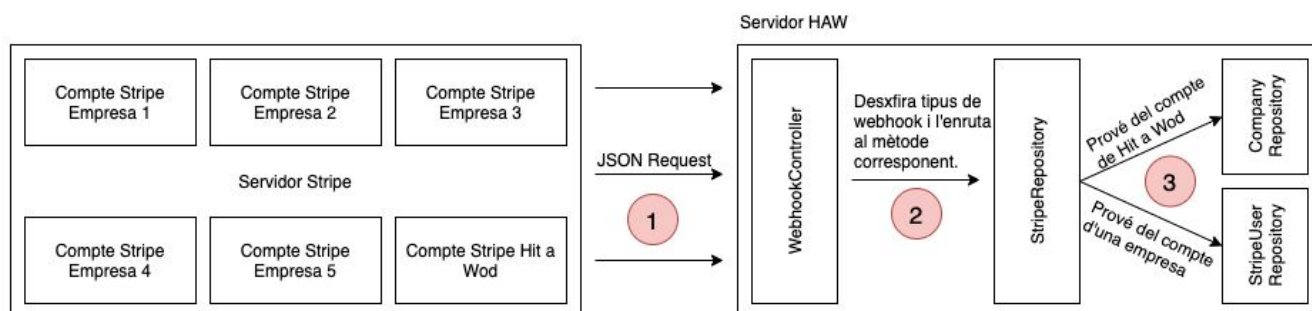


fig.26: Estructura Stripe contra el servidor de HAW

### Punt 1

S'ha produït un esdeveniment en els servidors de Stripe (p.e., la renovació automàtica d'una subscripció d'una empresa a la tarifa RX de Hit a Wod, o p.e., la renovació automàtica d'una subscripció d'un usuari a l'empresa 1 a la tarifa Mensual 3). Stripe ens notificarà, ja que prèviament s'hauran configurat els webhooks en el compte corresponent apuntant al servidor de Hit a Wod.

### Punt 2

Rebem l'esdeveniment. Es processarà i es mirarà quin tipus de webhooks és en el WebhookController. Concretament, els webhooks que tractem són ([https://stripe.com/docs/api/webhook\\_endpoints](https://stripe.com/docs/api/webhook_endpoints)):

- **invoicePaymentSucceed:** Per quan un pagament s'ha realitzat correctament i s'ha de processar la renovació d'una subscripció.
- **invoicePaymentFailed:** Un pagament ha fallat i s'ha d'avisar a l'empresa i a l'usuari en qüestió per tal que puguin prendre accions.
- **invoicePaymentActionRequired:** Quan es necessita una acció extra per la verificació del pagament. Sobretot passa a Europa amb la normativa SCA mitjançant 3DSecure. Tot i així, nosaltres no ens hi hauríem de trobar mai ja que quan es configura el mètode de pagament, és quan es passa el 3DSecure i es verifica el mètode de pagament.
- **CustomerSubscriptionDeleted:** Quan s'elimina una subscripció des de Stripe. Això no hauria de passar mai, però hem considerat que pot haver-hi un error humà i notifiquem a l'empresa i a l'atleta, amb una posterior inhabilitació de la subscripció.
- **CustomerUpdated:** Quan s'actualitza un usuari des de Stripe. En principi tampoc hauria de passar, ja que els usuaris són els únics que es poden actualitzar. Però s'envien notificacions d'alerta.
- **CustomerDeleted:** El mateix que l'apartat anterior.
- **PaymentMethodAttached:** Quan un usuari associa un mètode de pagament al seu perfil. En aquest procediment guardem l'identificador per fer pagaments posteriorment, així com la marca de la targeta i els últims quatre dígitos de la targeta.
- **PaymentMethodDetached:** Tot i que no s'hauria de fer mai des de Stripe, altre cop, tenim en compte el factor d'error humà. Si es desvincula un mètode de pagament des de Stripe, el nostre sistema hauria d'estar al corrent. Així que actualitzem la informació rebuda.

Informació bàsica sobre com Laravel Cashier tracta els Stripe Webhooks:

<https://laravel.com/docs/master/billing#handling-stripe-webhooks>

### Punt 3

Un cop sabem quin tipus d'esdeveniment és i som a l'StripeRepository, hem de desxifrar si es tracta d'un esdeveniment d'empresa o d'usuari. Per això, Hit a Wod quan configura i dóna d'alta empreses i usuaris els afegeix metadata perquè aquest procediment que ens ocupa sigui possible. Gràcies a aquesta metadata, sabrem si hem de tractar una empresa o un usuari. I llavors, executarem els mètodes del repositori que ens ocupa (normalment CompanyRepository, StripeUserRepository, SubscriptionRepository o UserSubscriptionRepository).

## 9.2.8. Nova subscripció d'atleta

Un atleta no podrà descarregar-se l'aplicació de Hit a Wod i registrar-se. A Hit a Wod confiem amb què anar al centre sigui la forma d'iniciar una subscripció. Llavors ja podran renovar, però per iniciar, aquesta és la forma.

El centre podrà registrar l'usuari i associar-li una nova subscripció no pagada, que llavors l'atleta podrà decidir si pagar a través del mòbil amb l'aplicació amb targeta o pagar allà mateix al moment amb efectiu.

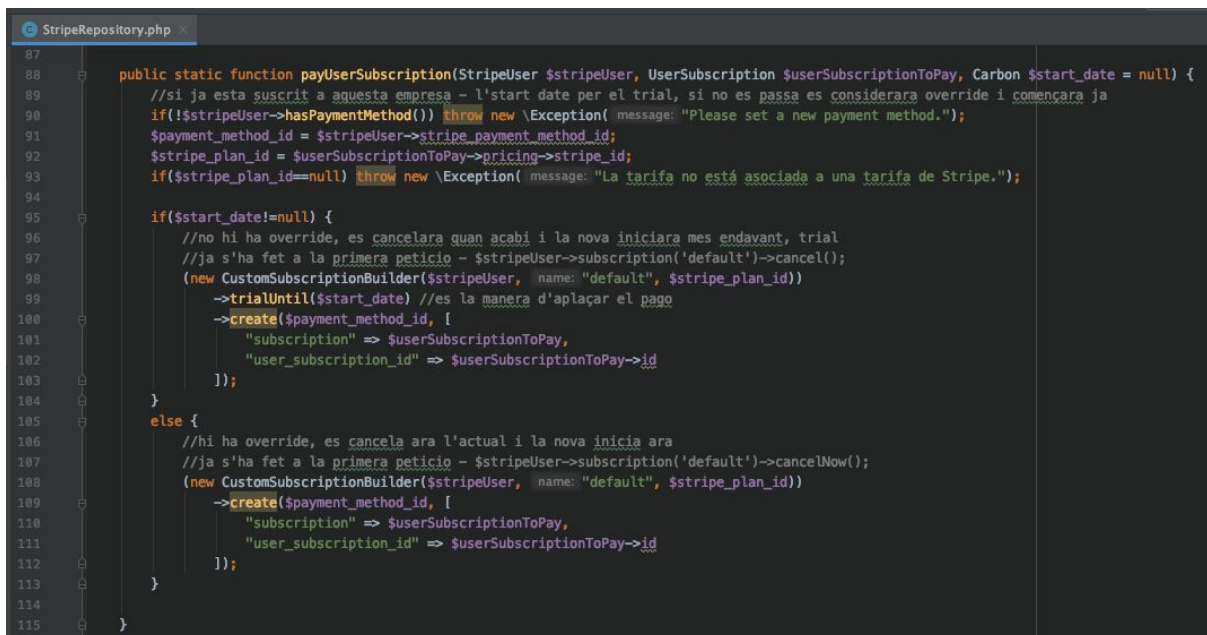


També, a nivell curiós, necessitarem que cada empresa, abans de donar-se d'alta, hagi configurat prèviament el seu panell de Stripe per tal de poder processar pagaments dels seus atletes.

### 9.2.9. Atleta canvia de tarifa

Quan un atleta vol augmentar o disminuir la seva tarifa, és a dir, canviar-la i generar una nova subscripció amb tarifa nova, té dues opcions: o bé que la nova s'iniciï quan l'actual acabi, o bé sobrescriure la subscripció actual i començar al moment la nova. Si a més la subscripció s'està renovant amb Stripe, també hem d'avisar a Stripe dels canvis que es volen fer.

A Stripe no es pot iniciar una subscripció i posposar-la perquè comenci al cap de 18 dies quan la nostra subscripció actual acabi. A nivell conceptual no existeix. El que sí que existeixen a Stripe són els períodes de prova. Així que ens hem aprofitat d'això, quan volem que el nostre canvi de subscripció comenci quan acaba l'actual, s'activa un període de prova fins que acaba l'actual i així es manté tot sincronitzat (veure fig.27).



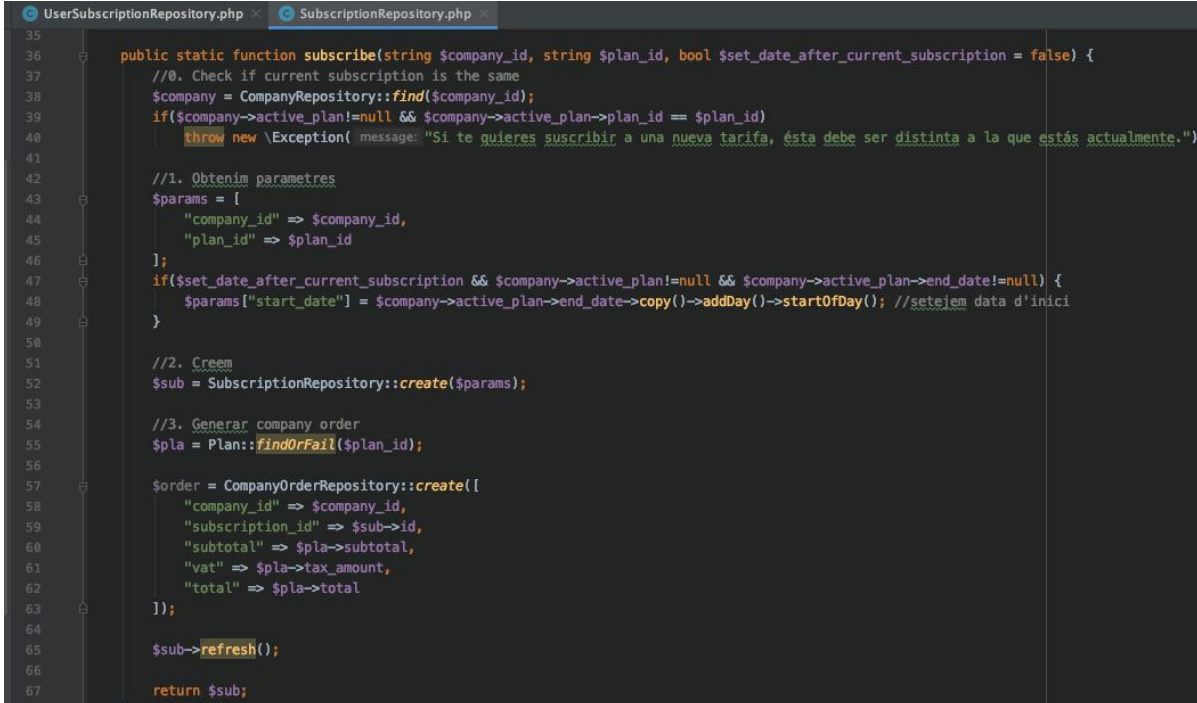
```
87
88 public static function payUserSubscription(StripeUser $stripeUser, UserSubscription $userSubscriptionToPay, Carbon $start_date = null) {
89     //si ja esta suscrit a aquesta empresa - l'start date per el trial, si no es passa es considerara override i començara ja
90     if(!$stripeUser->hasPaymentMethod()) throw new \Exception( 'message: "Please set a new payment method."');
91     $payment_method_id = $stripeUser->stripe_payment_method_id;
92     $stripe_plan_id = $userSubscriptionToPay->pricing->stripe_id;
93     if($stripe_plan_id==null) throw new \Exception( 'message: "La tarifa no està associada a una tarifa de Stripe."');
94
95     if($start_date!=null) {
96         //no hi ha override, es cancelara quan acabi i la nova iniciara mes endavant, trial
97         //ja s'ha fet a la primera peticio - $stripeUser->subscription('default')->cancel();
98         (new CustomSubscriptionBuilder($stripeUser, name: "default", $stripe_plan_id)
99             ->trialUntil($start_date) //es la manera d'aplaçar el pago
100             ->create($payment_method_id, [
101                 "subscription" => $userSubscriptionToPay,
102                 "user_subscription_id" => $userSubscriptionToPay->id
103             ]));
104     }
105     else {
106         //hi ha override, es cancela ara l'actual i la nova inicia ara
107         //ja s'ha fet a la primera peticio - $stripeUser->subscription('default')->cancelNow();
108         (new CustomSubscriptionBuilder($stripeUser, name: "default", $stripe_plan_id)
109             ->create($payment_method_id, [
110                 "subscription" => $userSubscriptionToPay,
111                 "user_subscription_id" => $userSubscriptionToPay->id
112             ]));
113     }
114 }
115 }
```

fig. 27: Codi repositori de Stripe, crida payUserSubscription

Aquí podem veure com es fa el període de prova (mètode trialUntil) si s'especifica una data a començar.

## 9.2.10. Empresa canvia de tarifa

Els problemes que ens hem trobat quan una empresa vol canviar de tarifa són els mateixos que quan un atleta canvia de tarifa, oferim les mateixes opcions de sobreescrivre tarifes, però amb la simplificació que en empresa només oferim pagaments via Stripe, no considerem el pagament en efectiu, com és lògic (veure fig.28).



```
35
36 public static function subscribe(string $company_id, string $plan_id, bool $set_date_after_current_subscription = false) {
37     //0. Check if current subscription is the same
38     $company = CompanyRepository::find($company_id);
39     if($company->active_plan!=null && $company->active_plan->plan_id == $plan_id)
40         throw new \Exception( message: "Si te quieres suscribir a una nueva tarifa, ésta debe ser distinta a la que estás actualmente.");
41
42     //1. Obtenim parametres
43     $params = [
44         "company_id" => $company_id,
45         "plan_id" => $plan_id
46     ];
47     if($set_date_after_current_subscription && $company->active_plan!=null && $company->active_plan->end_date!=null) {
48         $params["start_date"] = $company->active_plan->end_date->copy()->addDay()->startOfDay(); //setejem data d'inici
49     }
50
51     //2. Creem
52     $sub = SubscriptionRepository::create($params);
53
54     //3. Generar company order
55     $pla = Plan::findOrFail($plan_id);
56
57     $order = CompanyOrderRepository::create([
58         "company_id" => $company_id,
59         "subscription_id" => $sub->id,
60         "subtotal" => $pla->subtotal,
61         "vat" => $pla->tax_amount,
62         "total" => $pla->total
63     ]);
64
65     $sub->refresh();
66
67     return $sub;

```

fig.28: Codi repositori de Subscripció, crida subscribe

## 9.2.11. Càlcul de la puntuació d'un benchmark

Tenir en compte totes les variables que pot tenir un mateix entrenament, els diferents nivells que poden tenir els atletes, les unitat utilitzades per la mesura d'un exercici, entre d'altres, genera un problema a l'hora de assignar una puntuació justa per cada atleta; és per això que considerem únicament tres nivells de dificultat: RX, Scaled i Custom i valorem únicament els dos primers, ja que son els dos nivells que permeten competir al atleta.

En el cas del nivell "Custom", son per aquells atletes que no poden assumir l'entrenament, en nivell Scaled i molt menys en RX, per aquests, oferim que l'atleta pugui guardar els resultats amb els pesos, distàncies, temps i unitats que ha fet servir, pero la seva puntuació és zero, ja que no te suficient nivell com per poder competir però si com per poder continuar avançar.

Amb aquestes decisions el nostre objectiu és que qualsevol atleta pugui portar un seguiment de la seva evolució i aquells que tinguin un nivell suficient per competir, puguin veure com es valoren els seus resultats.

La puntuació del benchmark ve donada per la suma de la puntuació de cada una de les seccions del benchmark.

En cas que l'entrenament no tingui una manera de puntuació especificat pel creador d'aquesta el càlcul de la puntuació de cada secció té dues variables condicionals que farà que es tingui en compte variables com: les repeticions, les rondes, el temps, ...

Tindrem en compte el nivell amb el que s'ha entrenat i el tipus de secció.

Considerem tres tipus de nivells: RX, Scaled i Custom i sis tipus de seccions: AMRAP, Death By, For Time, Chipper, EMOM i E2MOM.

Un cop tenim aquestes variables:

Si és un AMRAP:

La puntuació ve donada per les rondes i les repeticions

Si és un Death By:

La puntuació ve donada per les rondes

Si és un For Time o un Chipper;

La puntuació ve donada pel temps empleat

Si és un EMOM o E2MOM:

La puntuació ve donada per si s'ha pogut completar o no

Un cop tenim la puntuació decidida per les variables, mirem el nivell amb el que s'ha treballat.

Si s'ha fet en RX:

La puntuació queda igual

Si s'ha fet en Scaled:

La puntuació es dividida entre dos

Si s'ha fet en Custom

La puntuació val zero

Si el creador de l'entrenament té una manera de puntuar específica, la puntuació sera introduïda pel mateix usuari seguint les guies del creador.

## 9.3. Procediments en background

Laravel ofereix Task Scheduling per tal de mantenir processos en background integrats en el sistema de forma fàcil i senzilla. En el nostre cas, hem necessitat dels següents processos en background (app/Console/Commands i l'scheduling es fa a app/Console/Kernel.php):

- ResetSubscriptionCounter: Cada dilluns a la matinada s'executa aquesta tasca per renovar, a les subscripcions actives i que no siguin bons, els comptadors de sessions setmanals i així poder-se apuntar a noves classes.
- GenerateUpcomingRenewalSubscriptionCompanyOrder: Cada dia es revisen les subscripcions d'empreses que estiguin a punt de caducar per tal de generar la futura comanda que hauran de pagar.
- GenerateUpcomingRenewalUserSubscriptionBoxOrder: El símil de l'anterior però per als usuaris.

## 9.4. Altres qüestions interessants

1. Hem implementat paginació automàtica en el servidor si es passen els query paràmetres `pageSize` i `pageNumber` i el format de resposta és un array.
2. Els procediments més complicats estan tots assegurats amb transaccions a base de dades, és a dir, si alguna cosa de tot el procediment falla, es tira tot enrere. Per als procediments que depenen de crides a serveis externs, es deixen per al final perquè així es pot fer `rollback` de forma satisfactòria.
3. Tant a la web com a l'app mòbil, hem apostat per l'efecte `skeleton` com a animació per mostrar els `loading`. És un patró de disseny que cada cop s'utilitza més i permet a l'usuari previsualitzar el que veurà uns segons més tard.

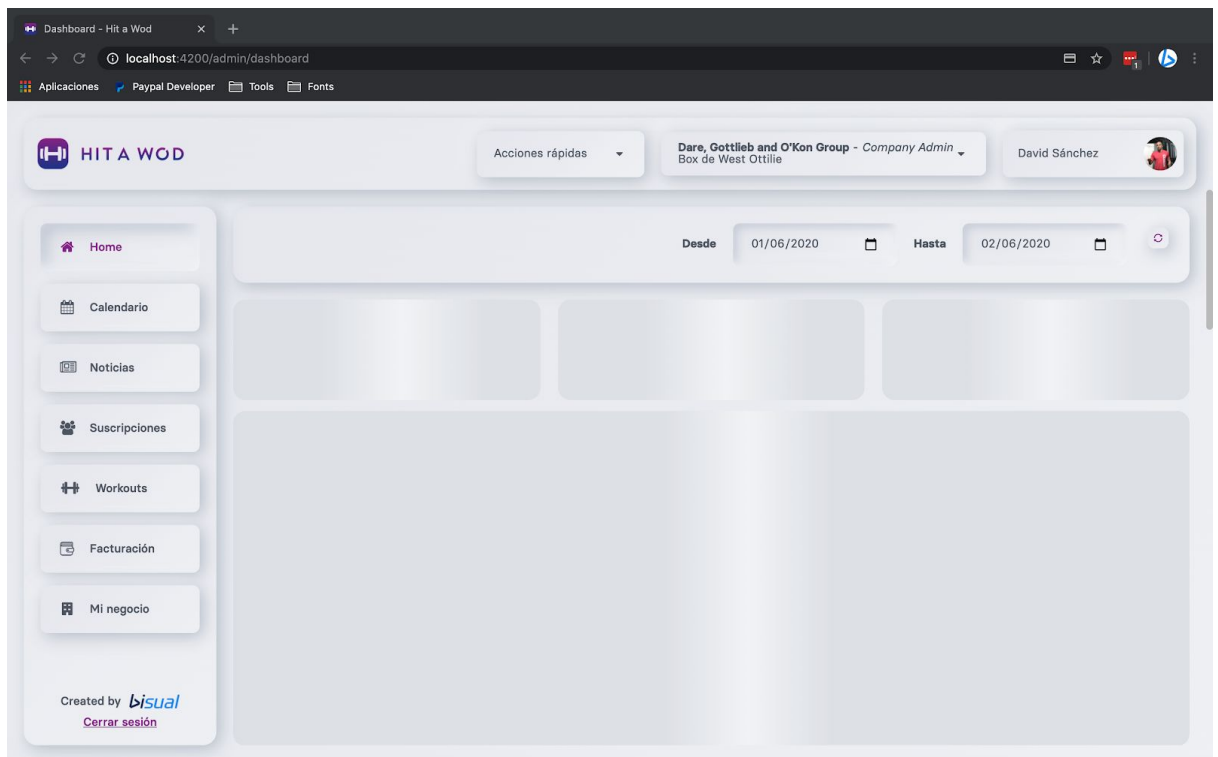


fig.29: Web - Skeleton loader

4. Tots els llistats de qualsevol entitat, sobretot en web, s'han optimitzat per ser filtrades, paginades i ordenades segons com vulgui l'usuari de la web.

# 10. Implantació i resultats

## 10.1. Pantalles

### 10.1.1. Client Web Angular

#### 10.1.1.1. Pàgina de login

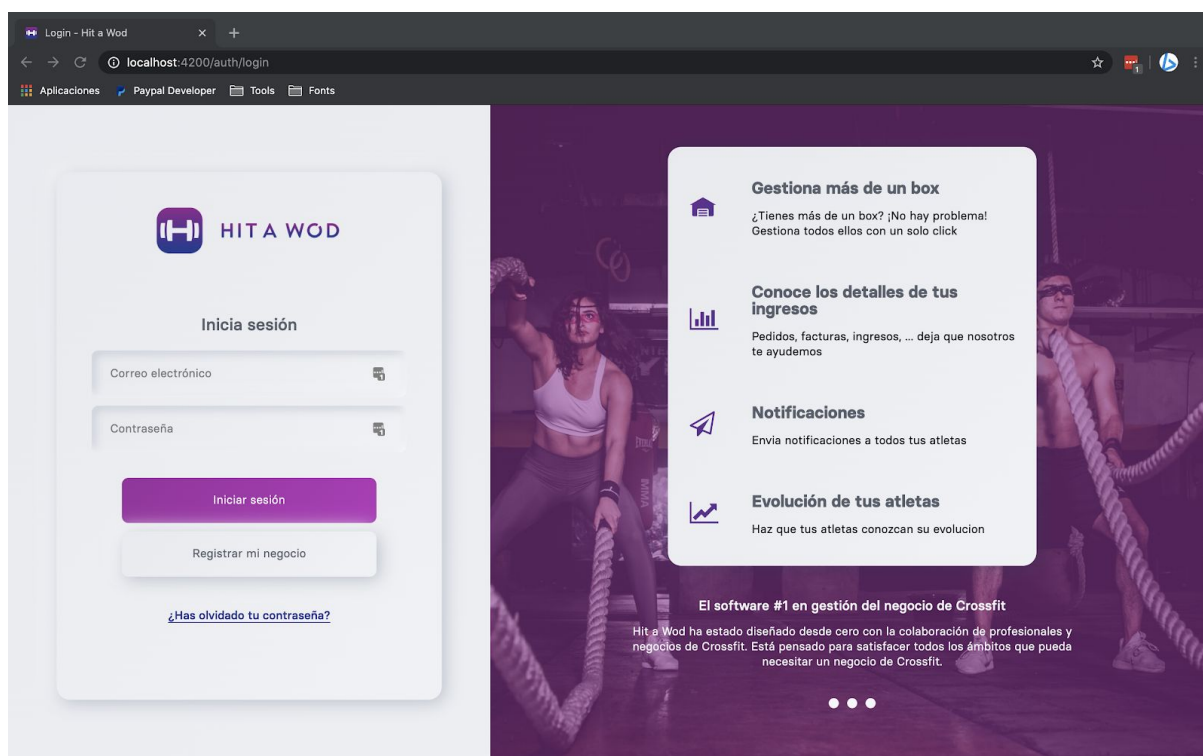


fig.30 - Web, pàgina login

Pantalla d'inici de sessió per a coaches i administradors. Permet iniciar sessió, iniciar el procediment de registre i restaurar la contrasenya.

### 10.1.1.2. Pàgina de registre

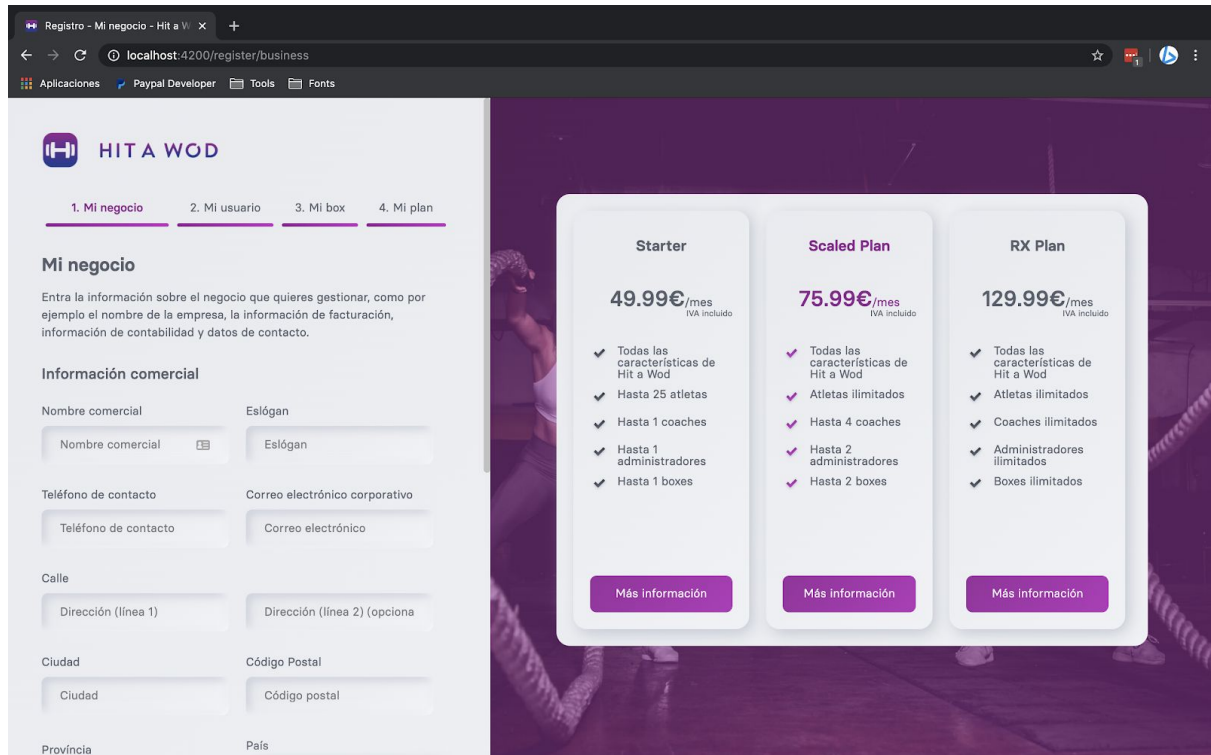


fig.31 - Web, pàgina registre

Pàgina de registre. Aquí es proporciona informació sobre el negoci, el box, l'usuari administrador i el pla al que es subscriurà.

### 10.1.1.3. Dashboard

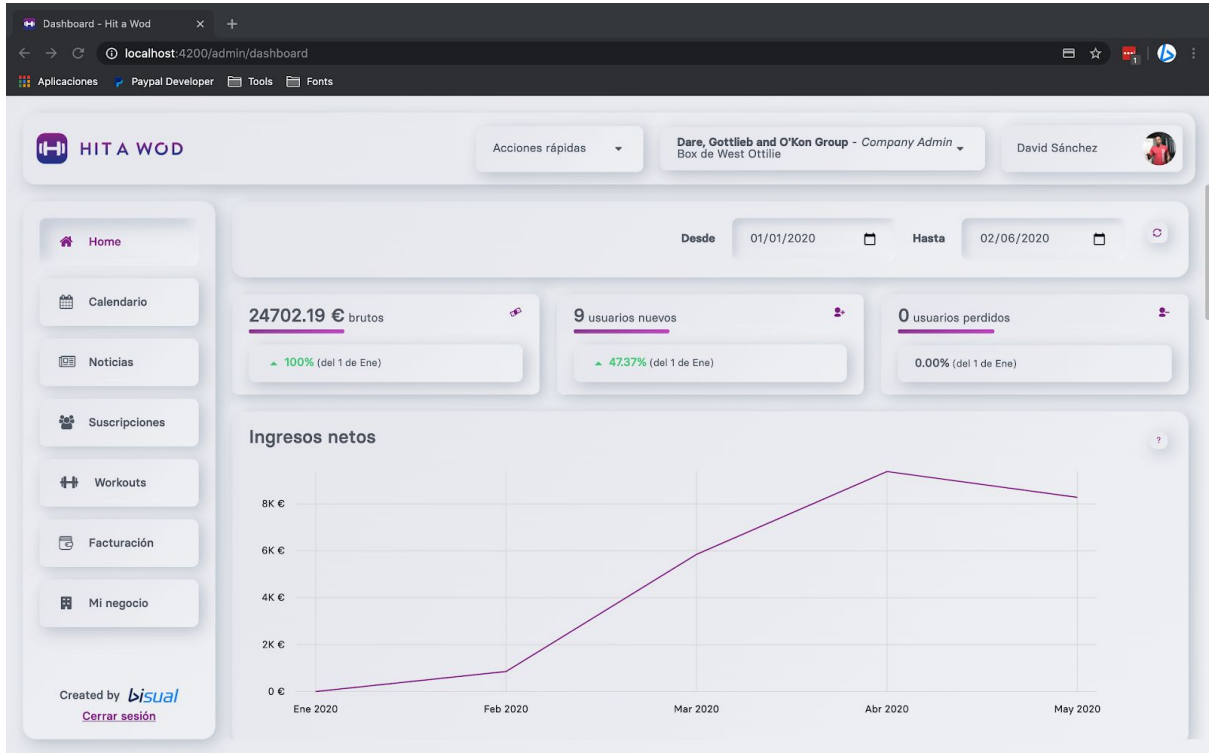


fig.32 - Web, página dashboard part 1

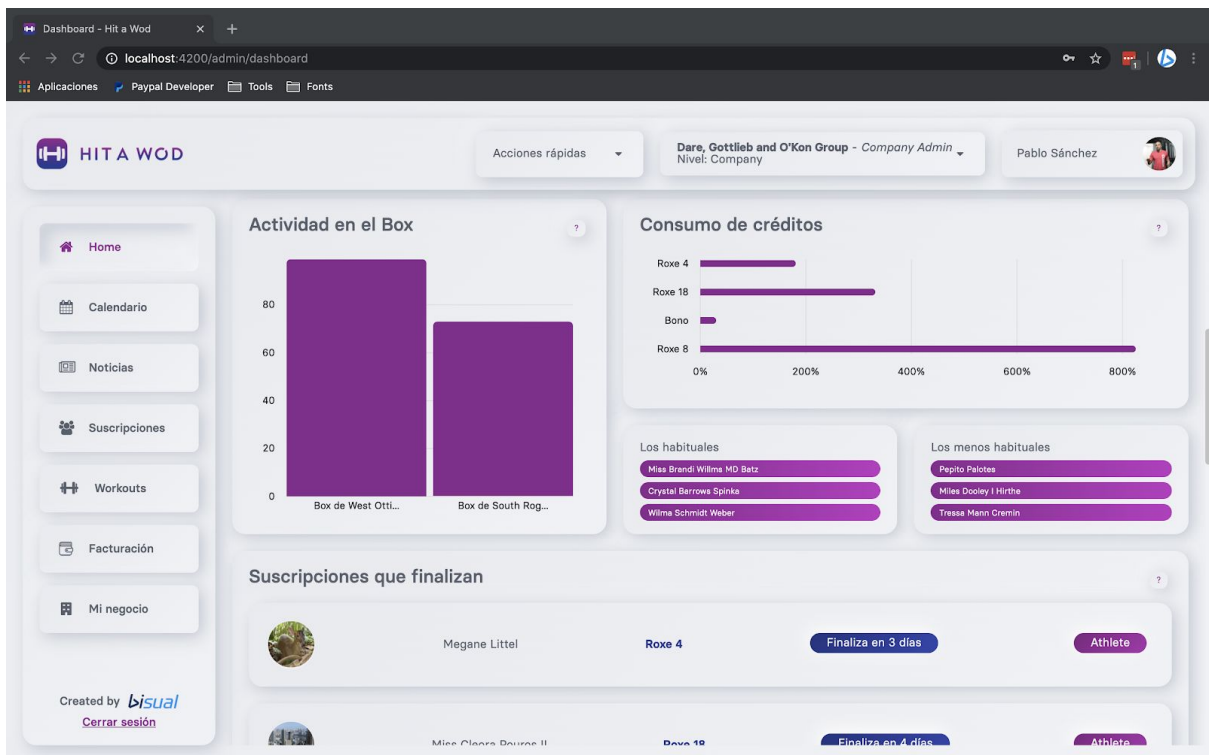


fig.33 - Web, página dashboard, part 2

Una de les pantalles més interessants del client web. Ofereix:

- Resum d'ingressos i despeses, filtrat per dates
- Activitat en el box, per saber els boxes de l'empresa als que s'hi fan més sessions
- Resum d'assistència per tarifa
- Usuaris més i menys habituals
- Usuaris que no tenen la renovació automàtica activada que estan a punt de caducar o que estan en pròrroga.
- Tarifes més utilitzades pels atletes.
- Resum del pla d'empresa cap a Hit a Wod. Per saber si ha d'augmentar o no el pla.

A més, com en totes les pantalles, té en compte el rol de l'usuari loguejat, és a dir, sap si ha de mostrar la informació de facturació, d'assistència, de subscripcions. Per exemple, no mostrarà els ingressos totals al coach.

#### 10.1.1.4. Selector de subscripcions

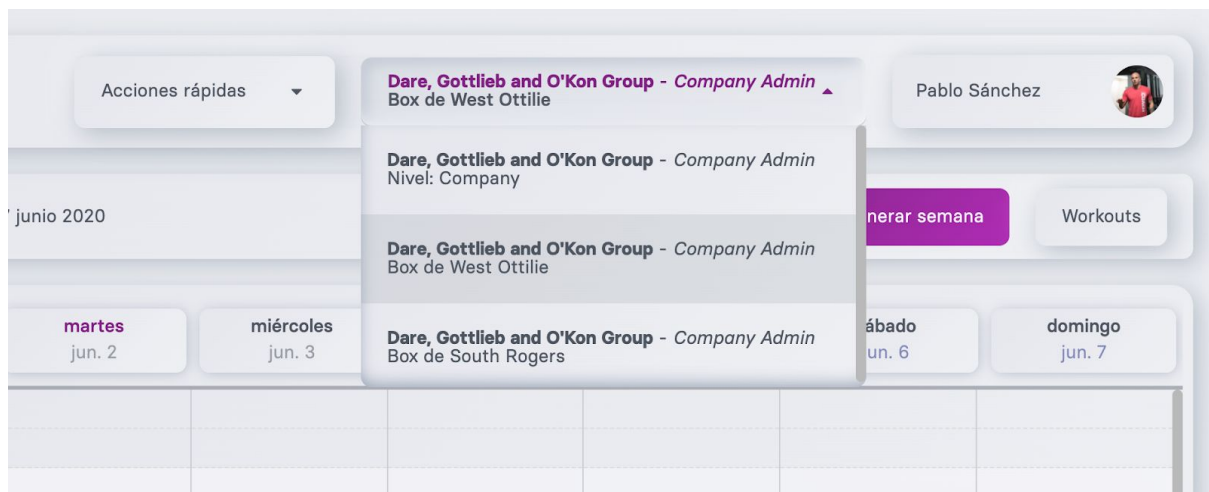


fig.34 - Web, pàgina calendari, selector de subscripcions

Un usuari pot ser que:

1. Estigui en varies empreses a la vegada amb diferents rols. El selector de subscripcions et permet commutar entre ells.
2. Accedir a nivell de box o d'empresa. Quan ets administrador de box, està clar que només tindràs accés a aquell box. Però si ets administrador d'empresa, necessites accés per separat a cada box, per exemple, per la pàgina de calendari. Es contempla aquesta opció. Per exemple, en el dashboard, l'usuari, si és administrador de companyia, podrà obtenir informes a nivell d'empresa en general i a nivell de box si es prefereix.



### 10.1.1.5. Pàgina de calendari

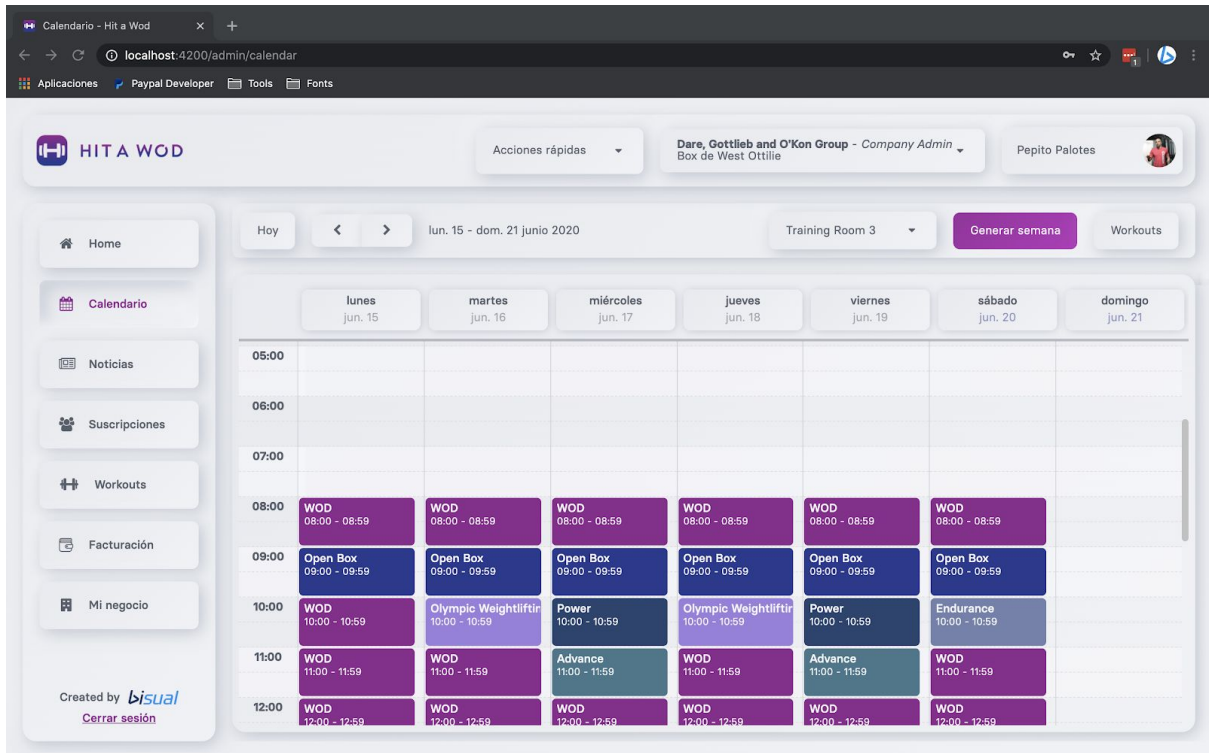


fig.35 - Web, pàgina calendari

És una pàgina que et permet gestionar el calendari de sessions, allà on els coaches passaran la majoria del seu temps. Les funcionalitats que permet:

- Gestió de sessions d'entrenament

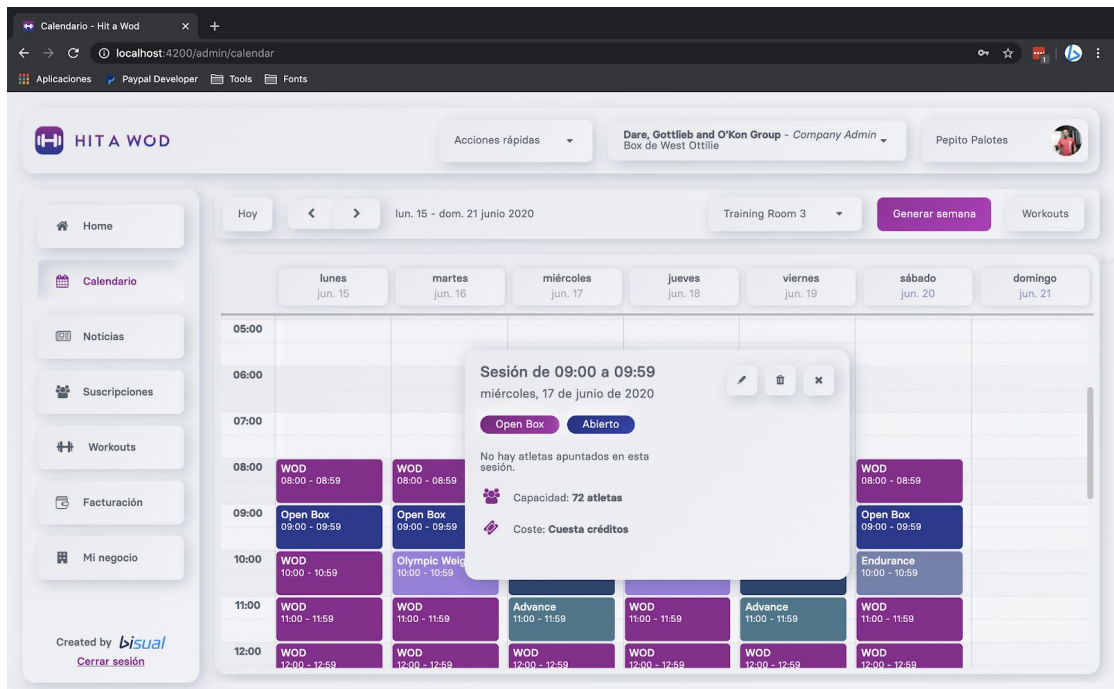


fig.36 - Web, pàgina calendari, detall sessió

- Assignar workouts a sessions

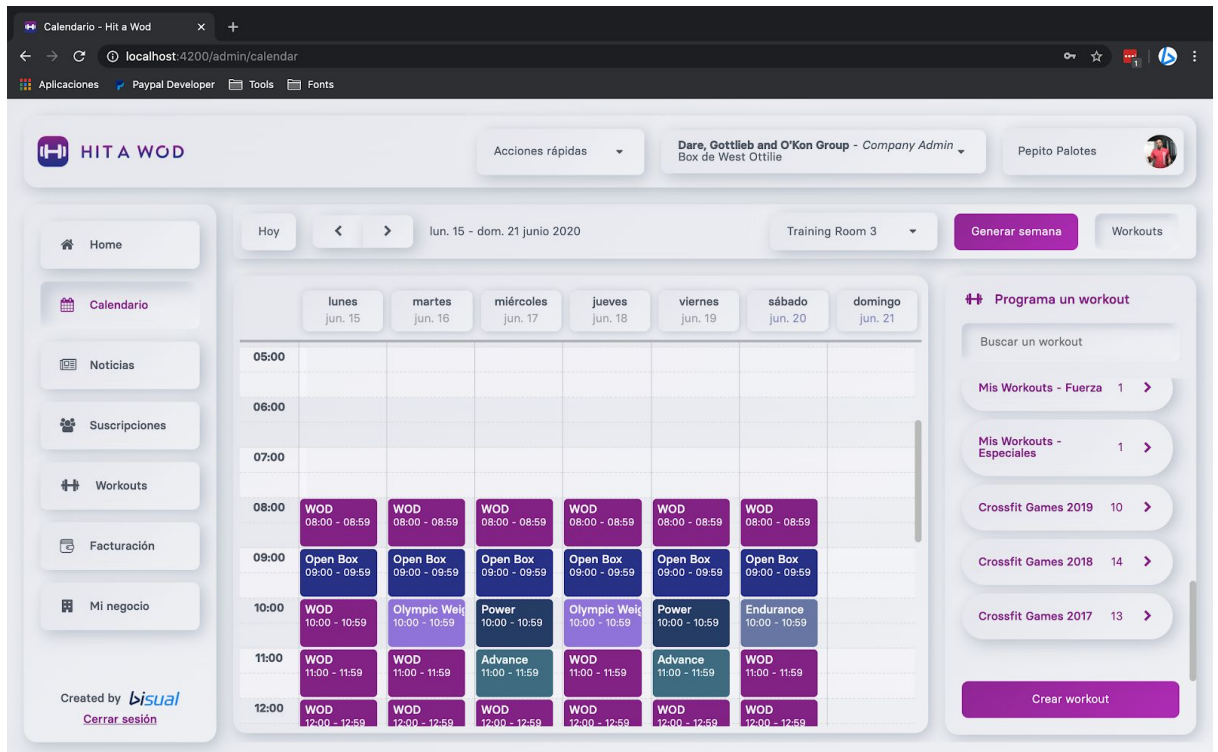


fig.37 - Web, pàgina calendari, selector de workouts

- Manteniment de plantilles de setmanes

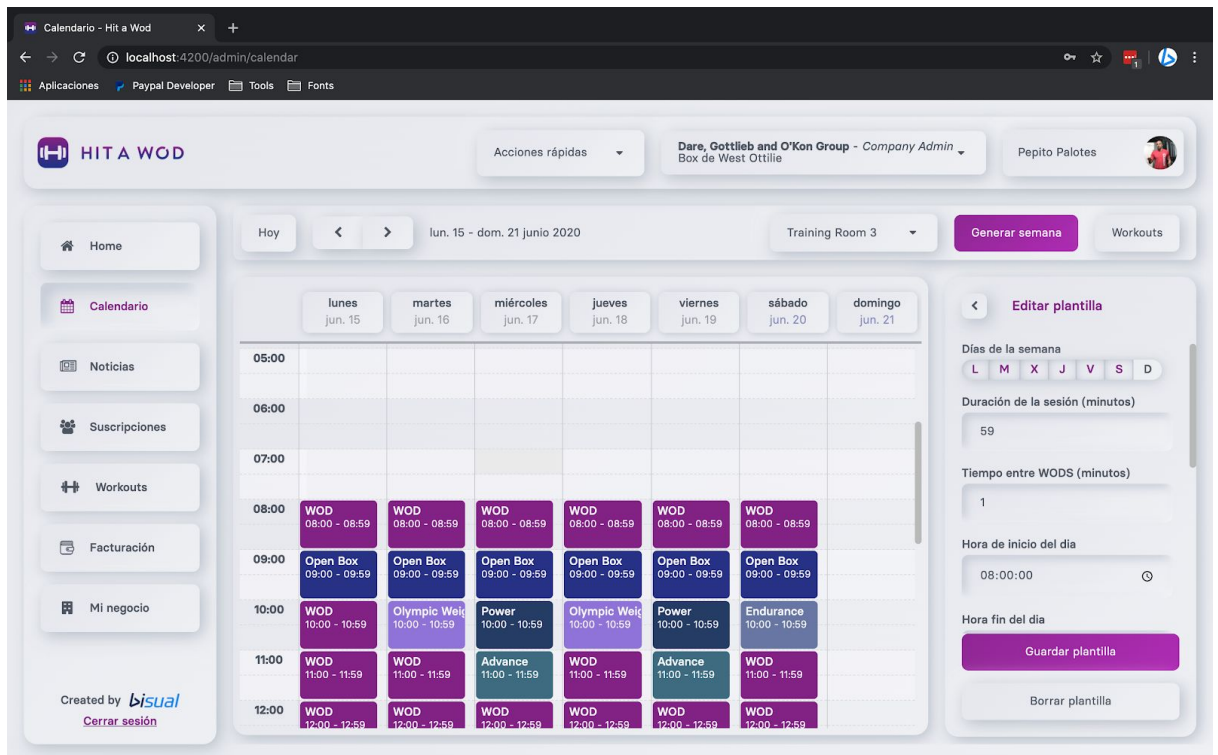


fig.38 - Web, pàgina calendari, editar plantilla

- Generar noves setmanes

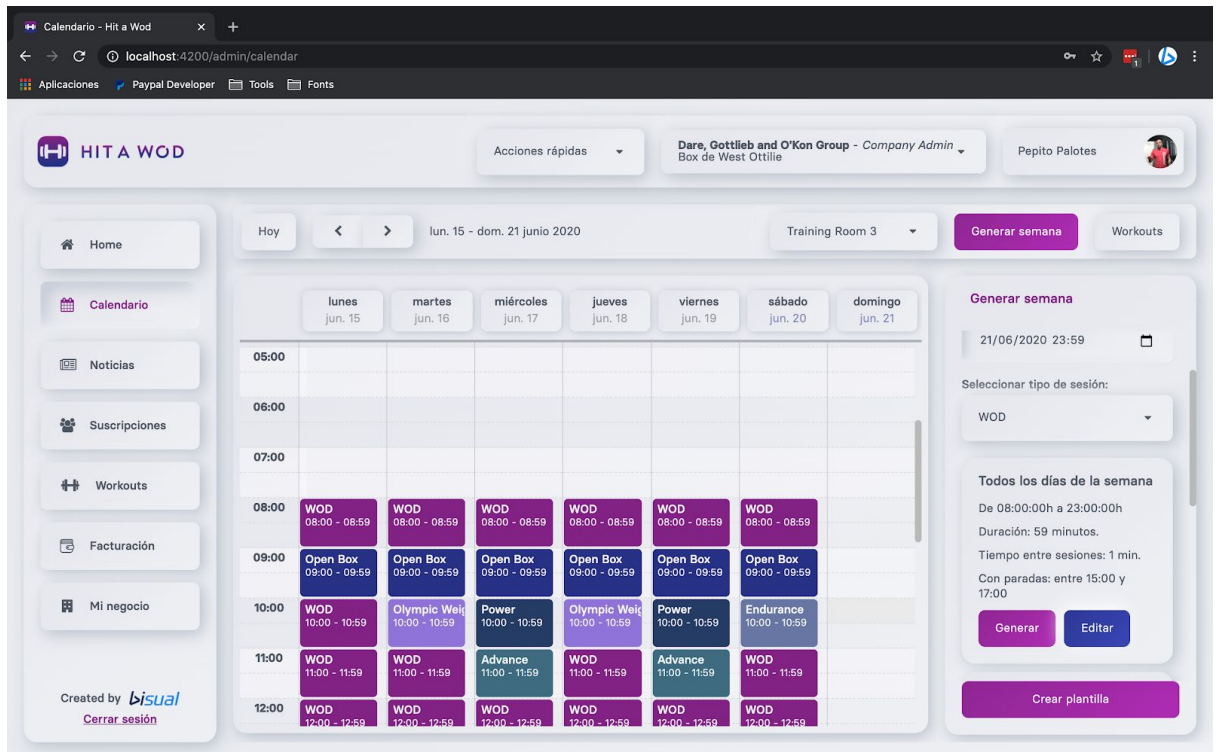


fig.39 - Web, pàgina calendari, generador de setmanes

### 10.1.1.6. Pàgina de notícies

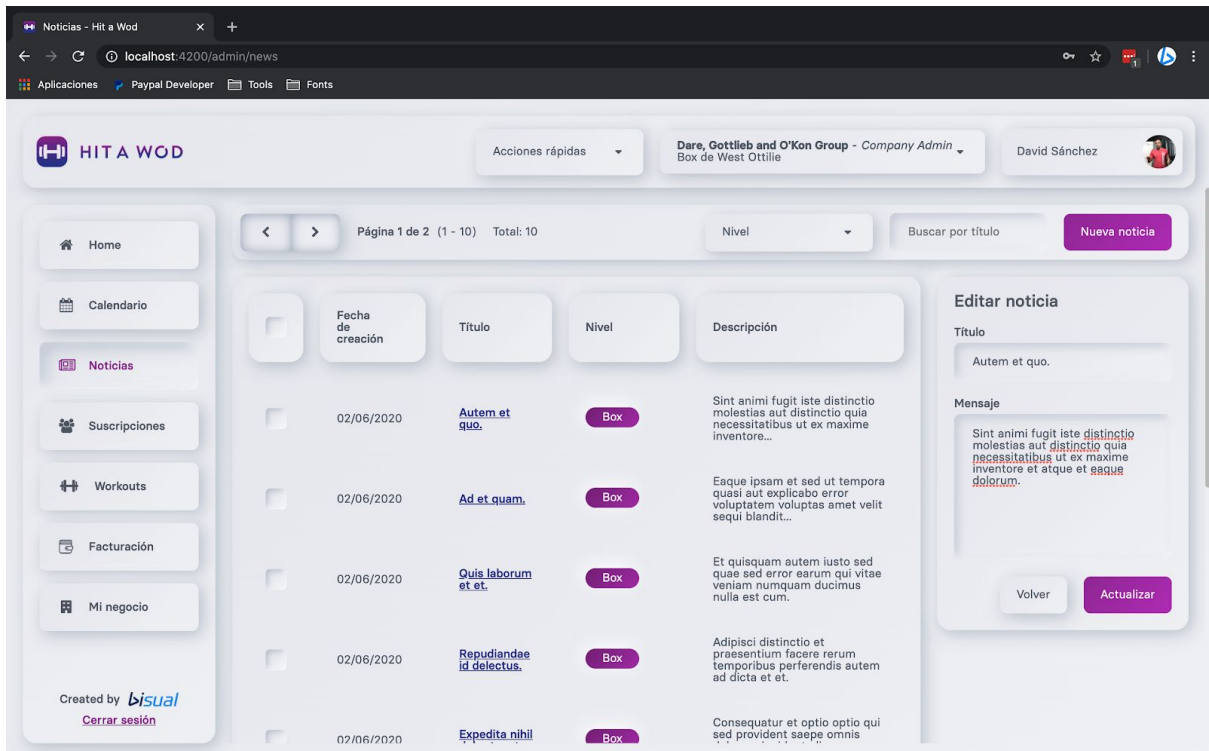


fig.40 - Web, pàgina de notícies

És la pàgina on els administradors podran fer el manteniment de les notícies. És un manteniment bastant bàsic comparat amb altres pantalles.

### 10.1.1.7. Pàgina de subscripcions

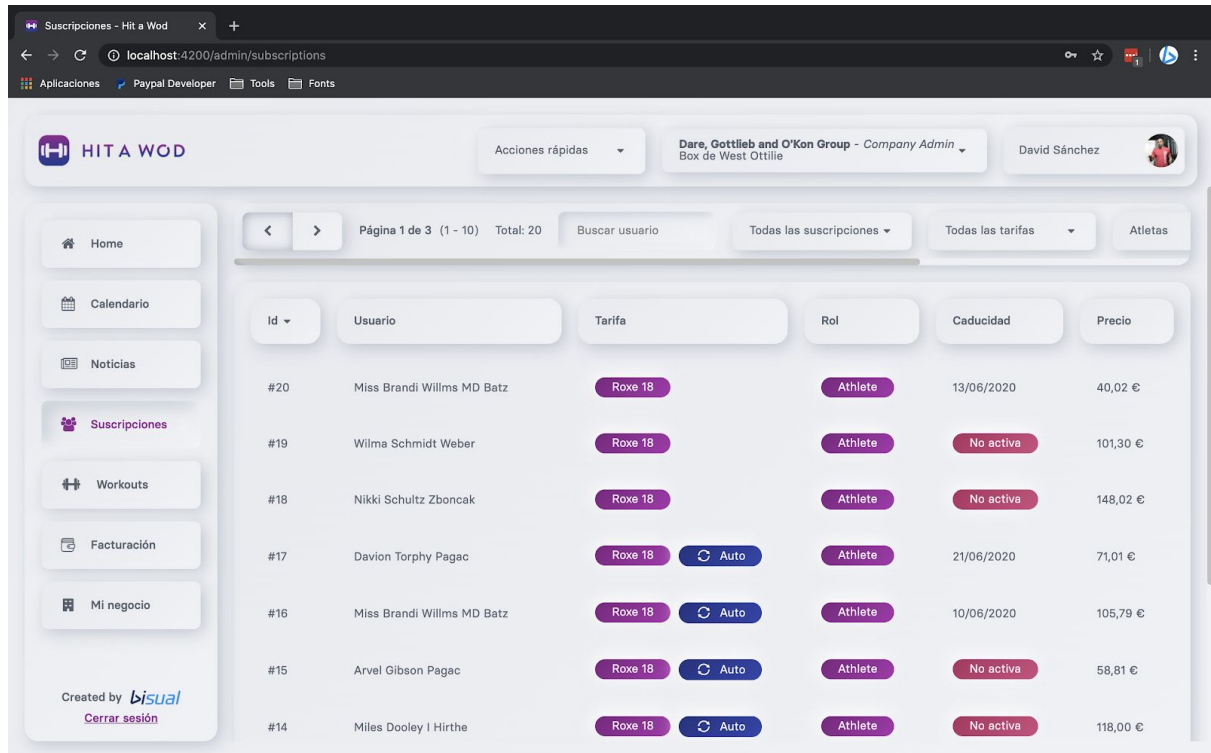


fig.41 - Web, pàgina de subscripcions

Aquesta pantalla et permet:

- Crear subscripcions

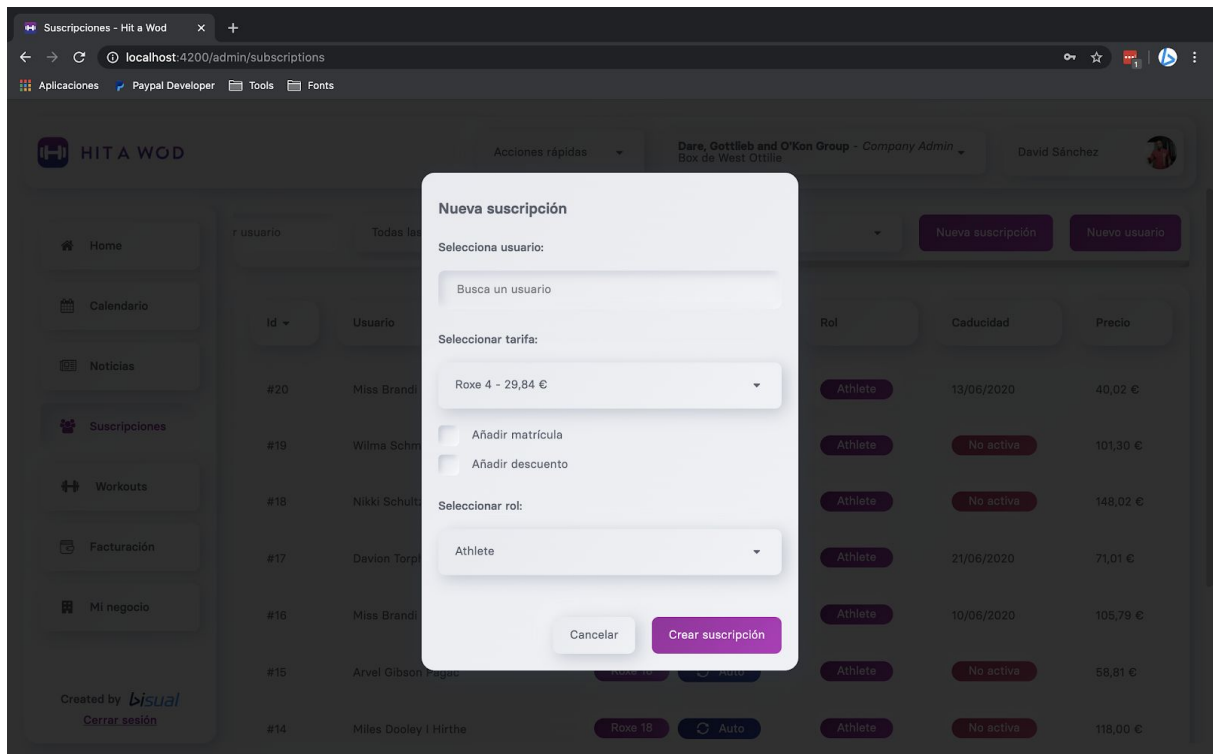


fig.42 - Web, pàgina de subscripcions, nova subscripció

- Detall de subscripcions

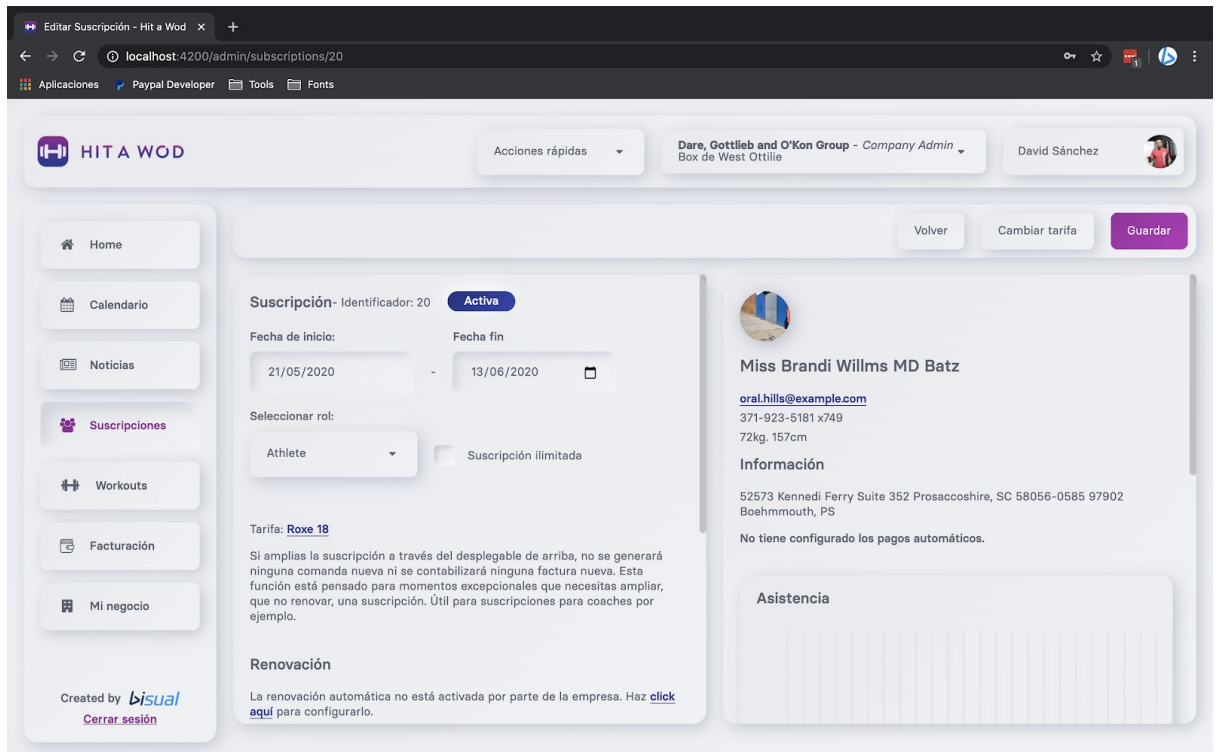
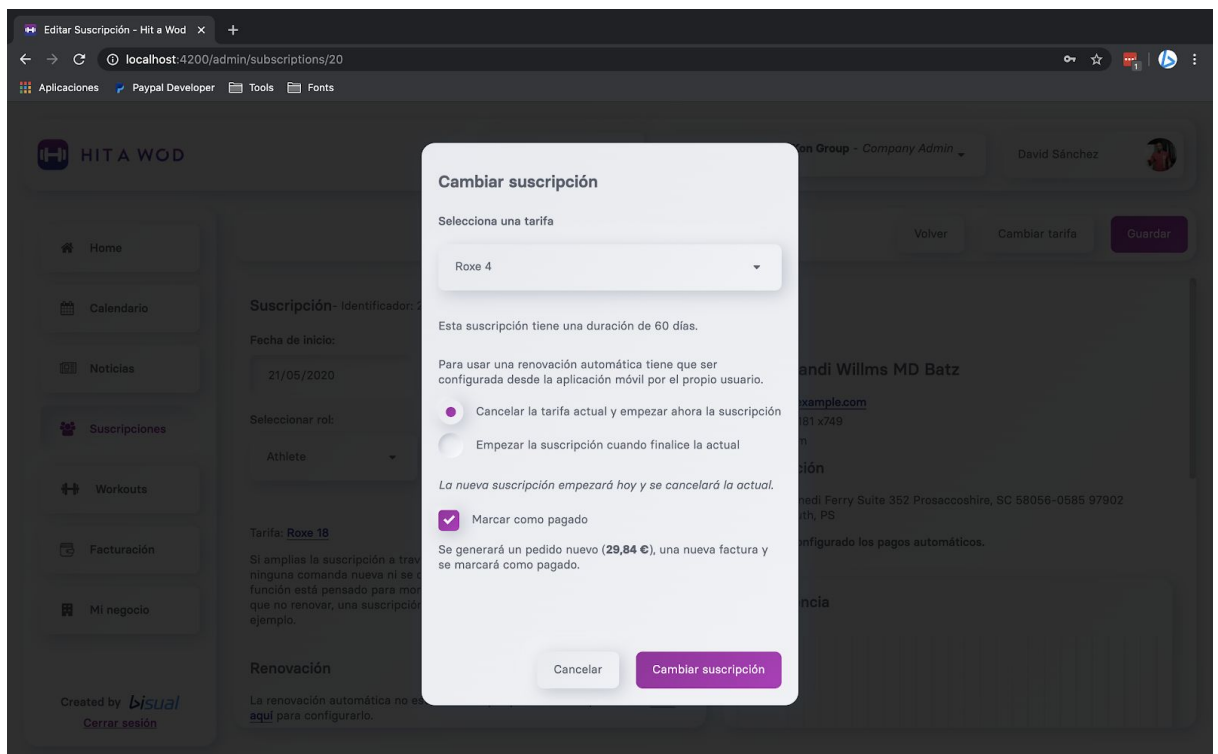


fig.43 - Web, página de subscripcions, detall

- Canvi de tarifa



- fig.44 - Web, página de subscripcions, canvi de tarifa

- Crear usuari

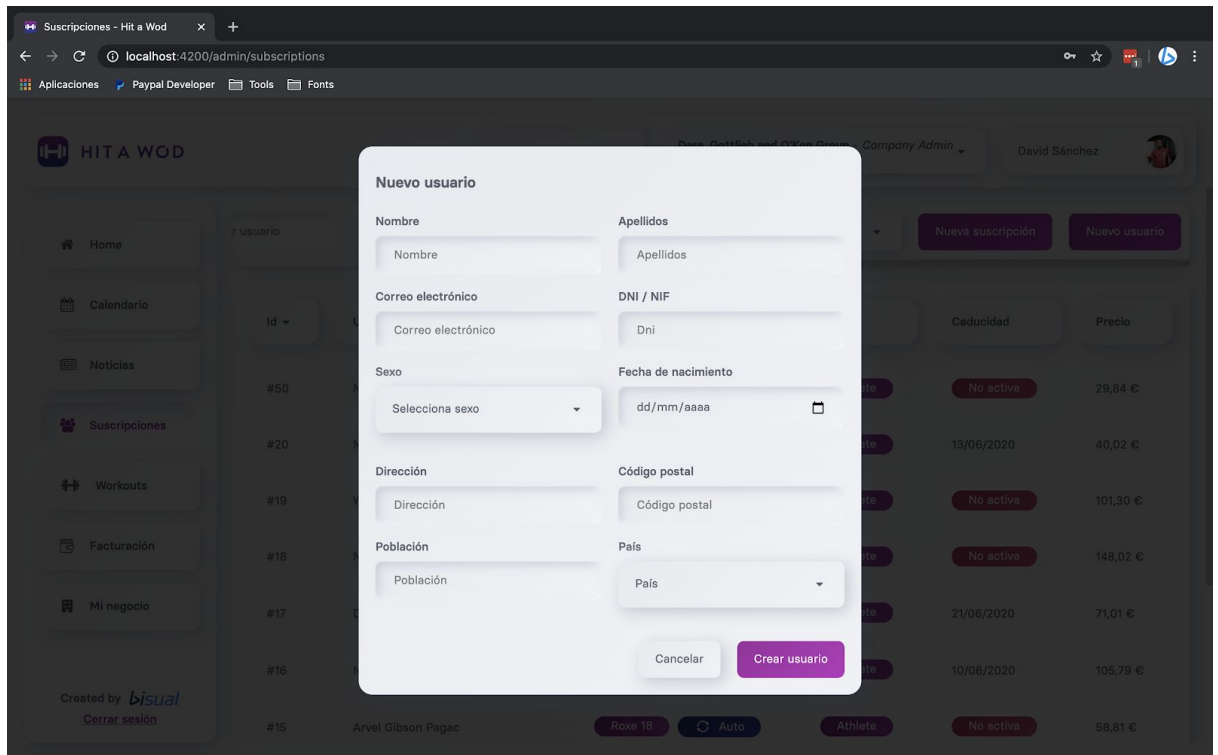


fig.45 - Web, pàgina de subscripcions, crear nou usuari

- Processar pagaments pendents

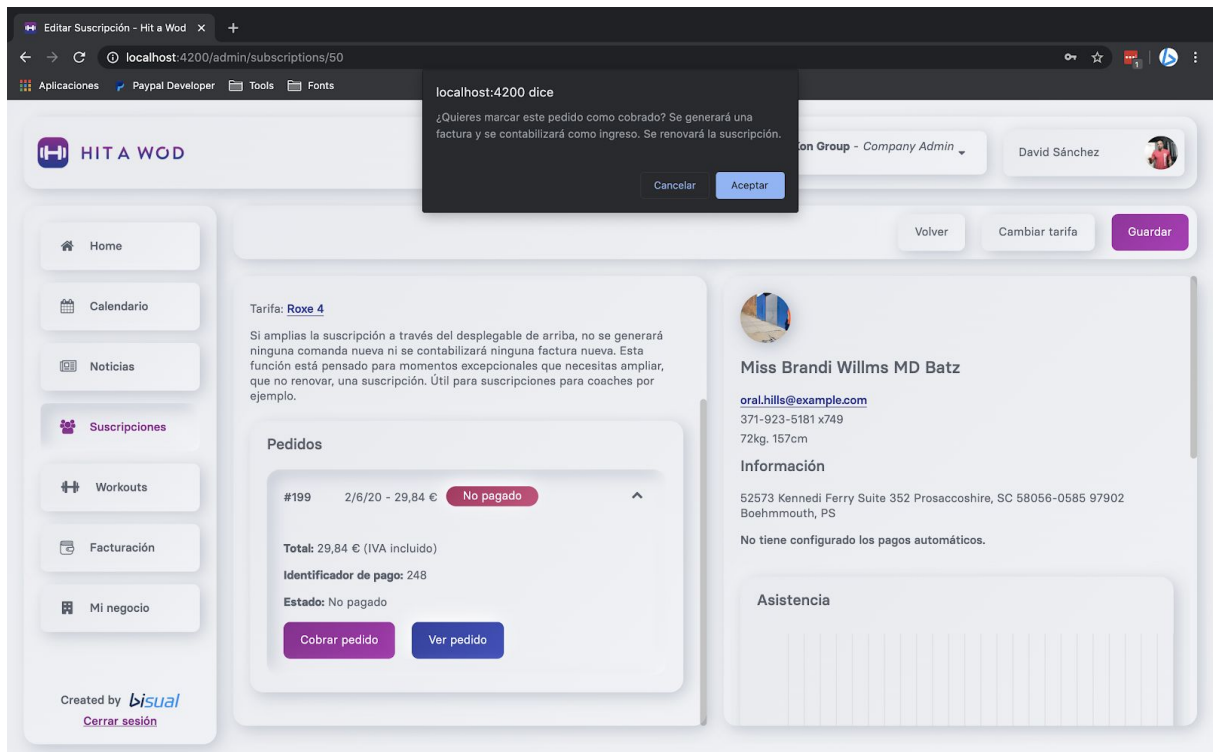


fig.46 - Web, pàgina de subscripcions, pagaments

## 10.1.1.8. Pàgina de workouts

The screenshot shows the HIT A WOD admin interface. The top navigation bar includes the logo, a dropdown for 'Acciones rápidas', the user 'Dare, Gottlieb and O'Kon Group - Company Admin', and the name 'David Sánchez'. The main content area features a sidebar with navigation options: Home, Calendario, Noticias, Suscripciones, Workouts (highlighted), Facturación, and Mi negocio. The main area displays a table of workouts with columns for Nombre, Tipo, Categoría, Propiedad, and Uso en sesiones. A detailed view of the 'Fran' workout is shown on the right, including instructions and a list of exercises: Thruster (21-15-9 reps, 75lb) and Pull-Up (21-15-9 reps).

Nombre	Tipo	Categoría	Propiedad	Uso en sesiones
Fran	Hero	For Time	Común	0
Open 14.4	Open 2014	AMRAP	Común	0
Split Triplet	Crossfit Games 2019	For Time	Común	1
Sprint	Crossfit Games 2019	For Time	Común	0
Sprint Couplet	Crossfit Games 2019	For Time	Común	0
Ruck	Crossfit Games 2019	For Time	Común	0

**Fran**  
Perform the 21 Thrusters and 21 Pull-Ups, then 15 Thrusters and 15 Pull-Ups, then 9 Thrusters and 9 Pull-Ups, as fast as possible.

- For Time
- 3 rondas

Thruster  
21-15-9 reps  
(75lb)

Pull-Up  
21-15-9 reps

Ver workout

fig.47 - Web, pàgina de workouts

Aquesta pantalla permet el manteniment de workouts. Els propis, permet editar-los, i els que són per totes les empreses (com ara els WODs dels Crossfit Games) simplement permet visualitzar-los.

- Visualitzar:

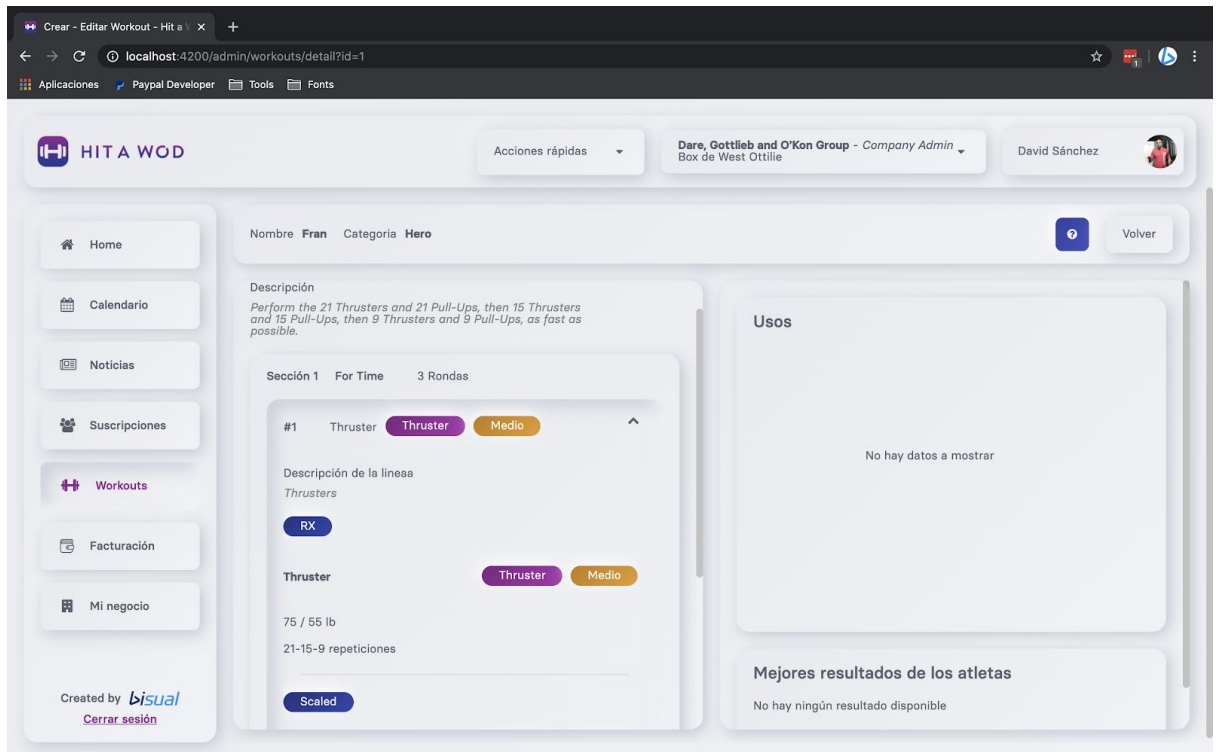


fig.48 - Web, página de workouts, detall

- Editar:

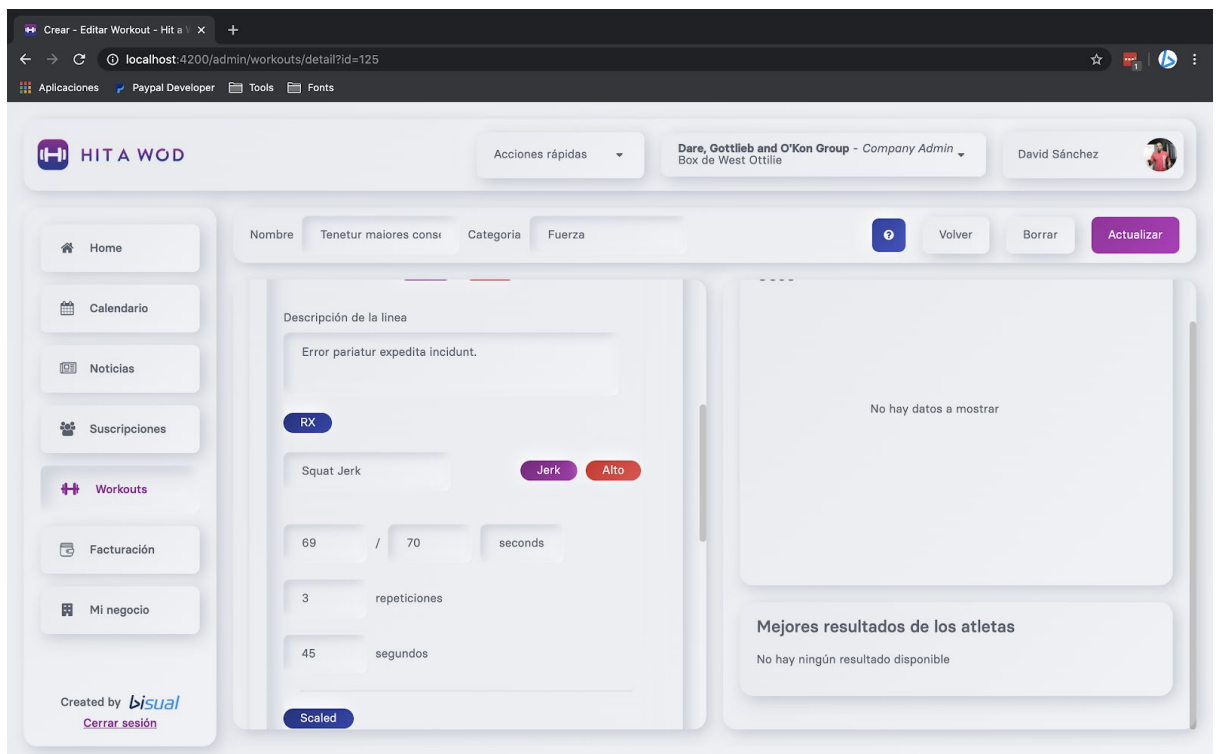


fig.49 - Web, página de workouts, crear



### 10.1.1.9. Pàgina de facturació

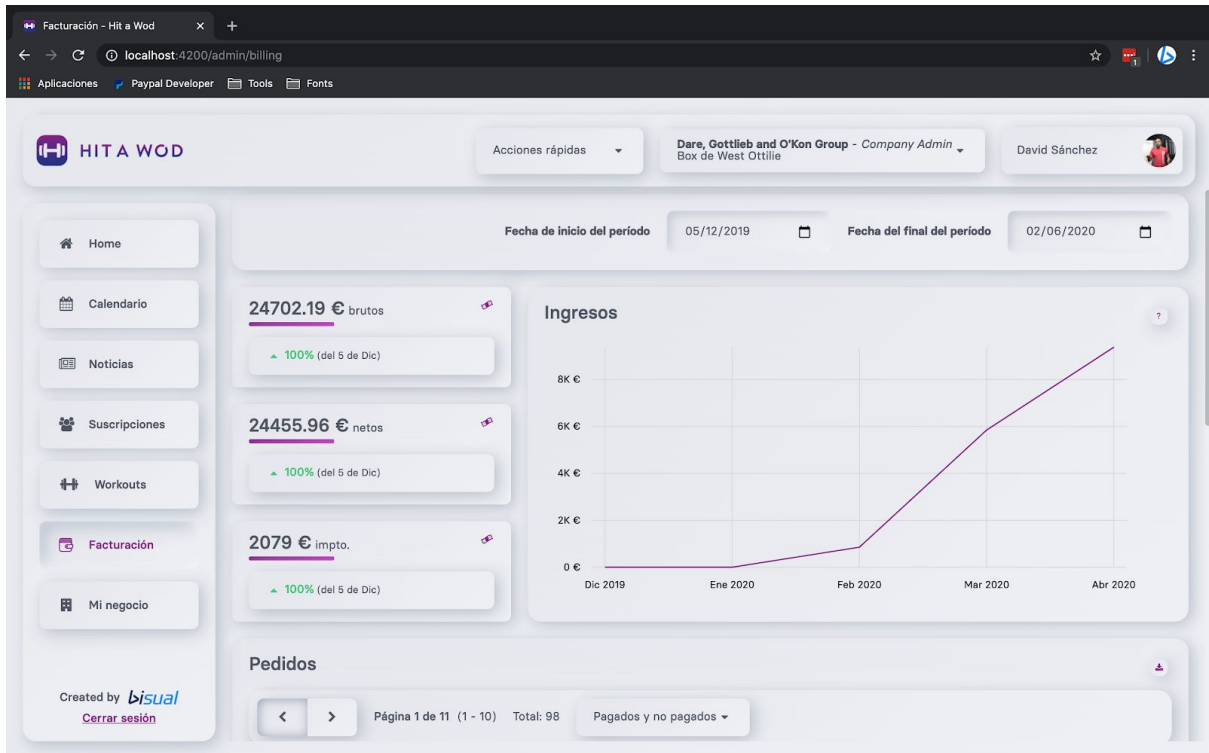


fig.50 - Web, pàgina de facturació

Pantalla on els administradors podran accedir al resum i detall de tots els seus ingressos. Podran veure factures, editar el seu número, veure el detall dels box orders, etc.

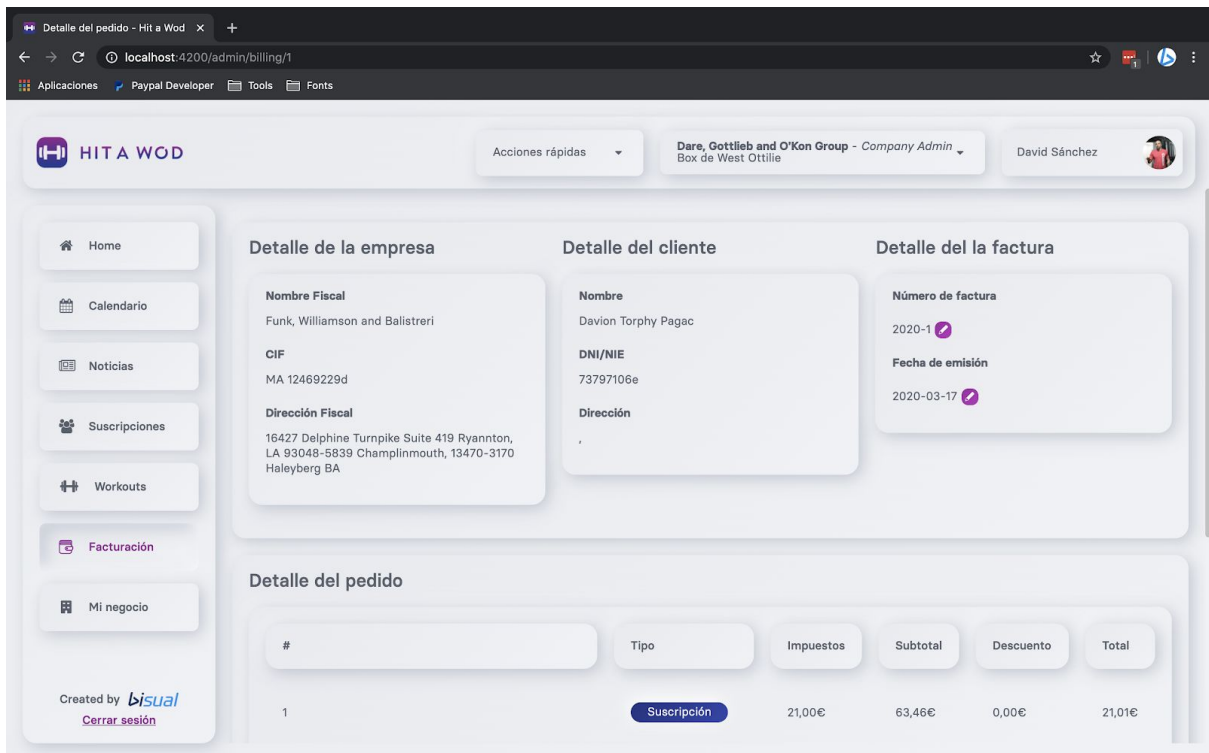


fig.51 - Web, pàgina de facturació, detall factura

### 10.1.1.10. Pàgina de negoci

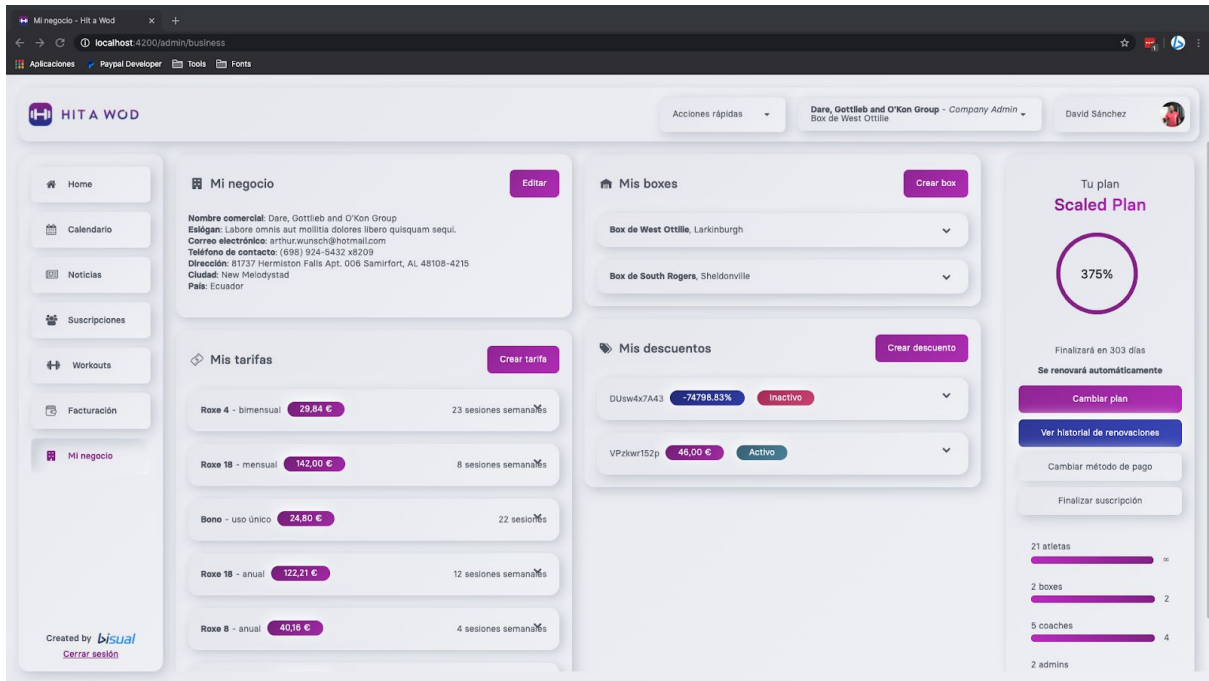


fig.52 - Web, pàgina de negoci

Pàgina on els administradors d'empresa podran:

- Mantener la informació de la seva empresa:

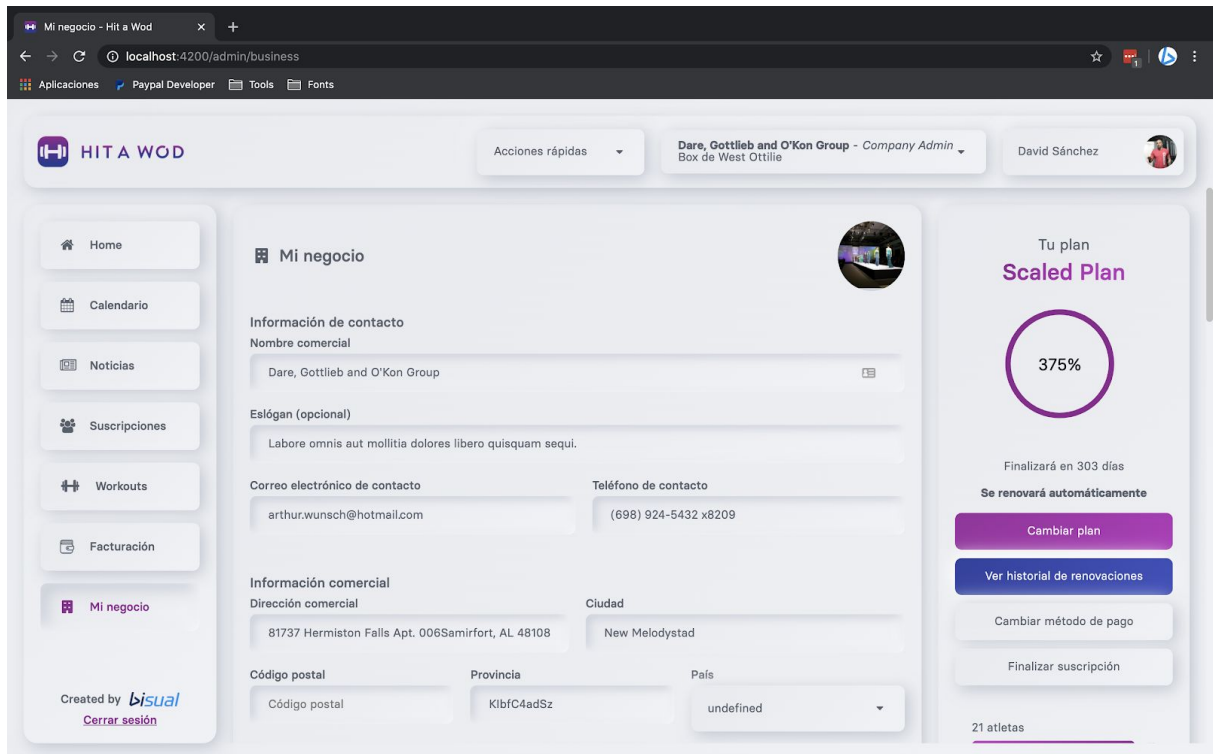


fig.53 - Web, pàgina del negoci

- Mantener les seves tarifes:

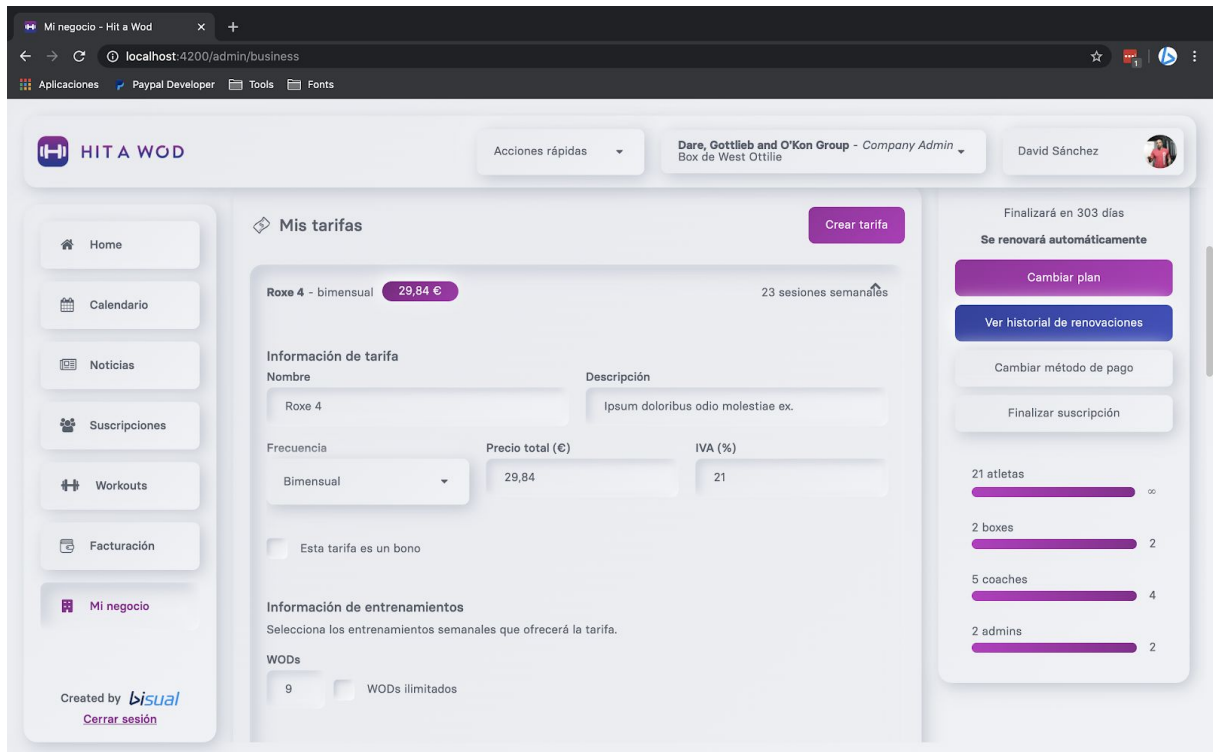


fig.54 - Web, pàgina del negoci, tarifes

- Mantener els seus boxes:

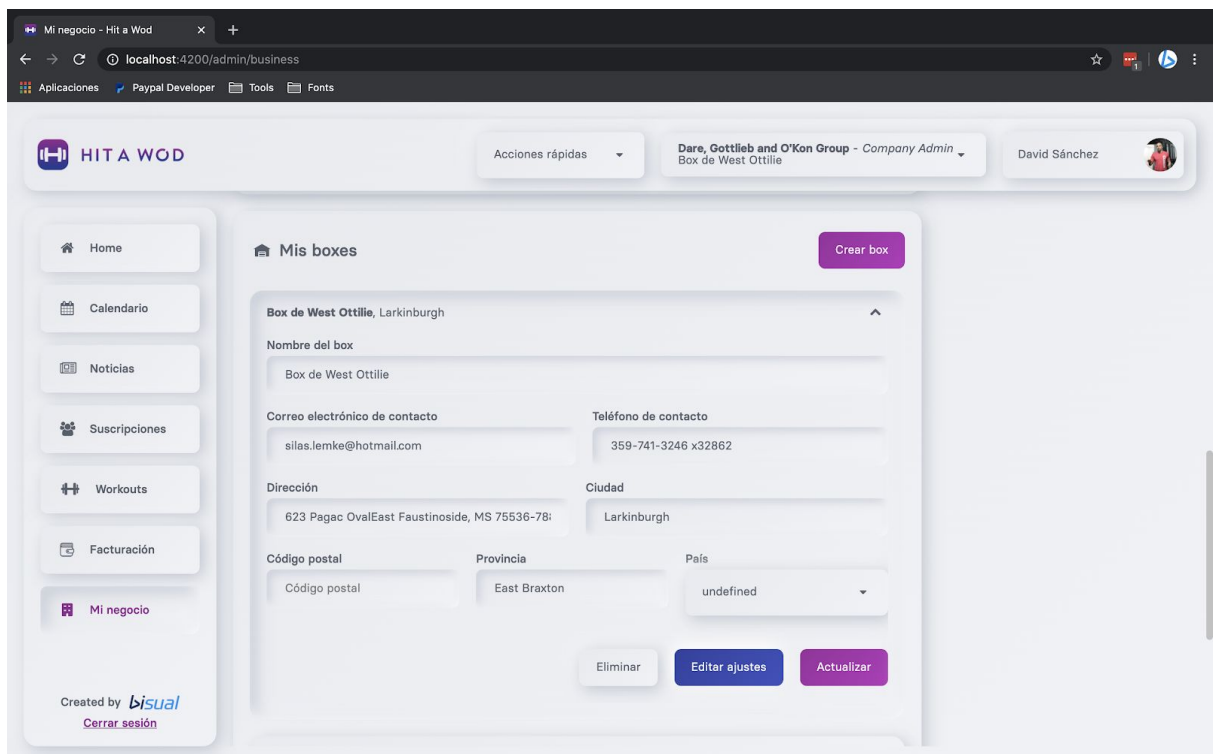


fig.55 - Web, pàgina del negoci, boxes

- Mantener els seus descomptes:

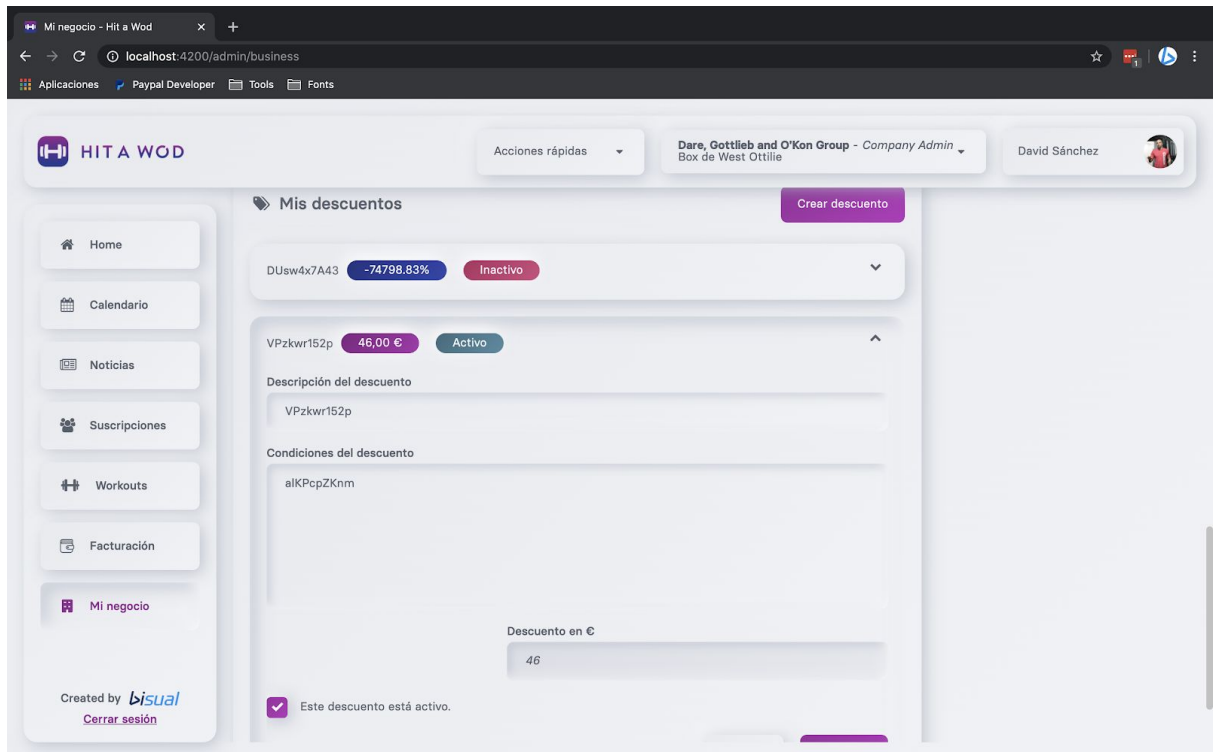


fig.56 - Web, pàgina del negoci, descomptes

- Mantener la subscripció amb Hit a Wod
  - Canviar la tarifa

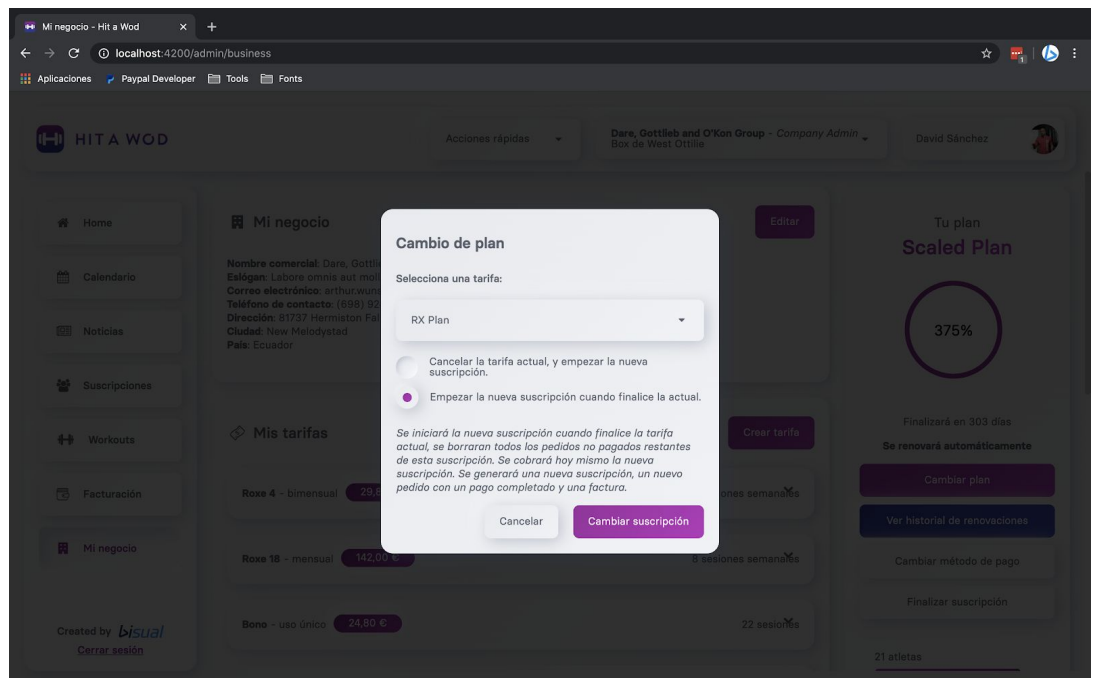


fig.57 - Web, pàgina del negoci, tarifes amb Hit a Wod

- Veure historial de renovacions i Company Orders

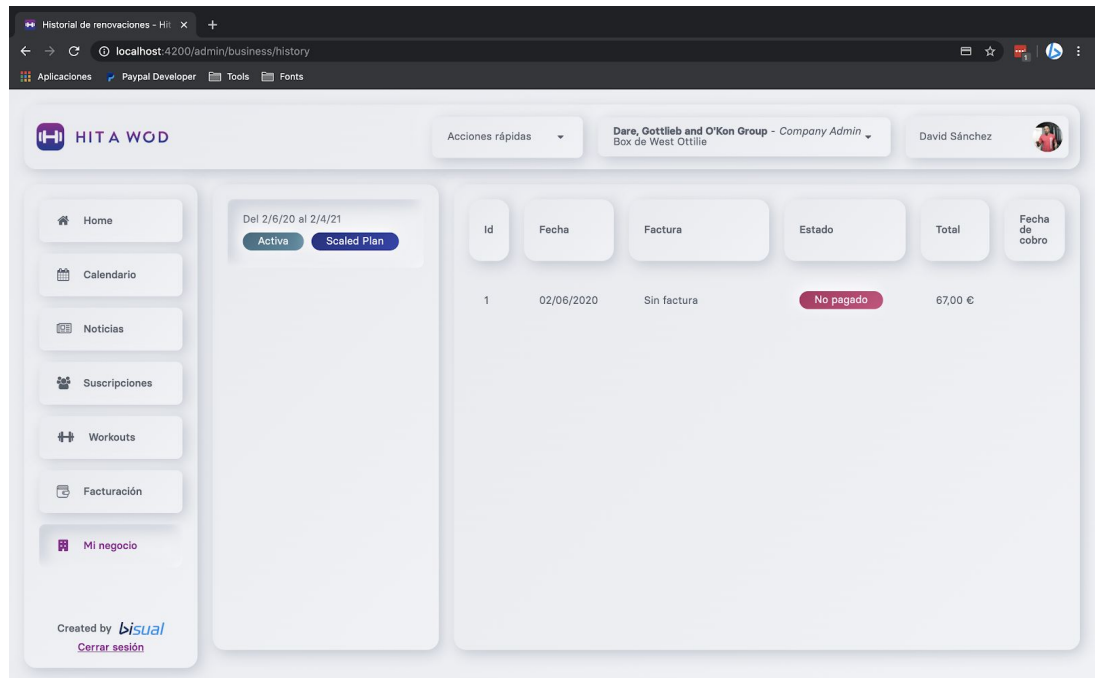


fig.58 - Web, pàgina del negoci, historial de comandes

- Canviar mètode de pagament

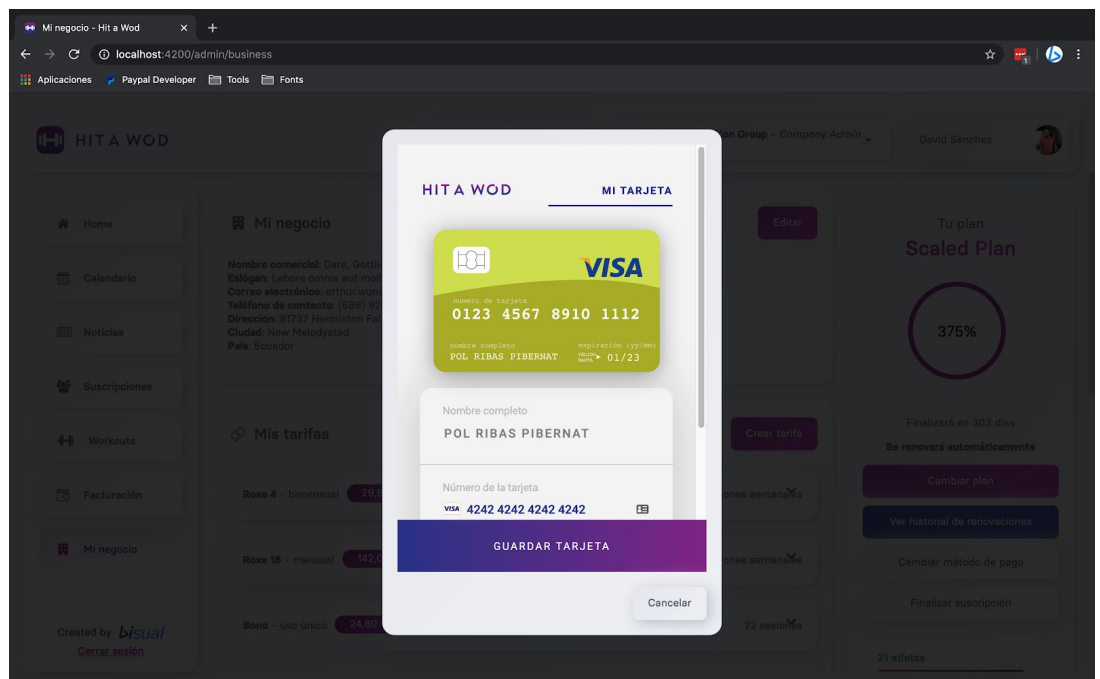


fig.59 - Web, pàgina del negoci, mètode de pagament

## 10.1.2. Client Mòbil Flutter

### 10.1.2.1. Pàgina de login

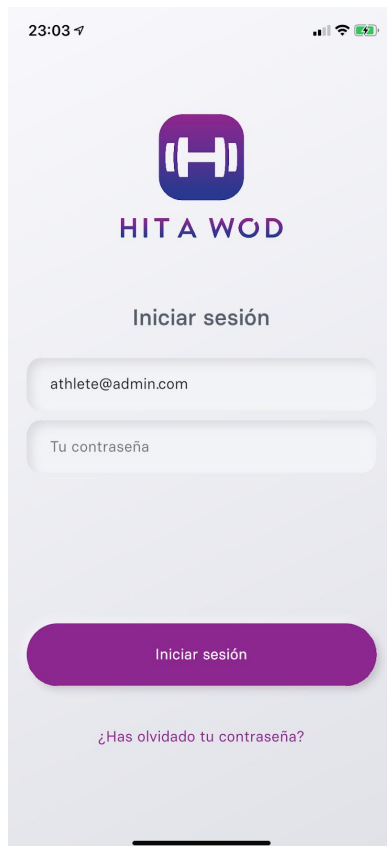


fig.60 - App, Inici de sessió



fig.61 - App, Recuperar contraseña

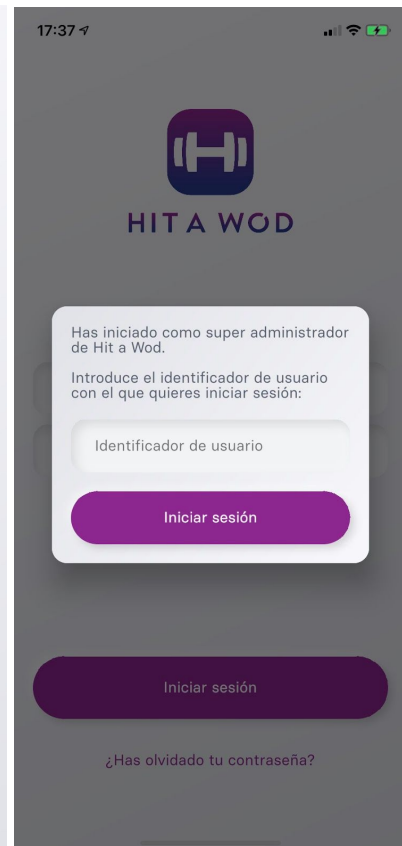


fig.62 - App, Impersonar

Aquesta pantalla permet iniciar sessió a atletes, coaches i administradors. Tot i que l'app està principalment pensada per atletes. No pots registrar-te a través de l'app. També et permet restablir la teva contrasenya. Una altra característica interessant que hem incorporat, és la possibilitat d'iniciar sessió com a super administrador (Només personal de Hit A Wod) i poder impersonar a qualsevol altre usuari.

### 10.1.2.2. Pantalla d'inici

Aquesta pantalla ofereix bastantes dades:

- Subscripció sel·leccionada actualment
- Crèdits setmanals restants
- Última notícia del box
- Informació sobre si algun atleta del box fa anys
- Informació sobre si fa molt que no assisteixes
- Avís sobre la pròxima classe
- Avís sobre entrada del resultat de la última classe.
- Informació sobre el box actiu.

### 10.1.2.3. Selector de subscripcions i historial

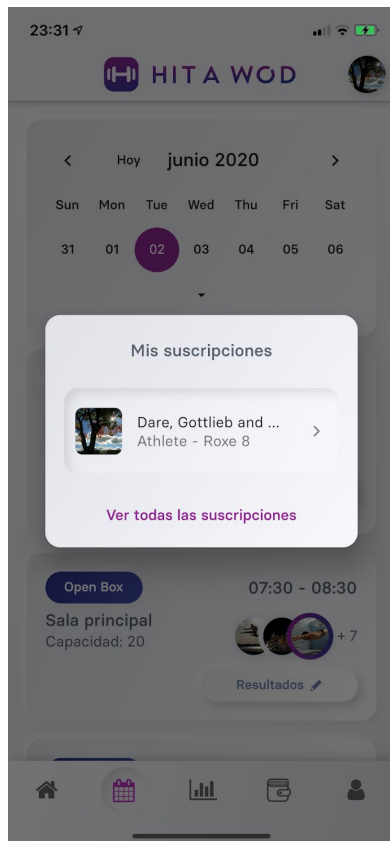


fig.63 - App, selector subscripcions

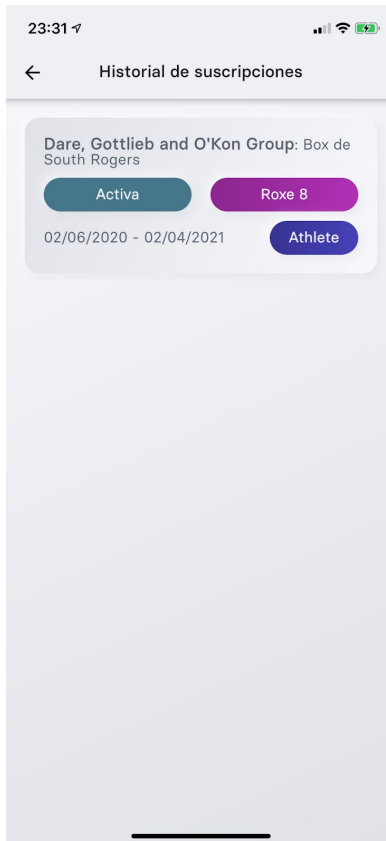


fig.64 - App, historial subscripcions

Es pot accedir al selector de subscripcions de tres formes: Des del widget de la pantalla inicial, des de la icona superior dret que mostra el logo de l'empresa i mantenint premut la icona d'usuari de la barra de navegació. També donem l'opció de poder veure totes les subscripcions ja que la primera pantalla només mostra les actives.

### 10.1.2.4. Calendari i sessions

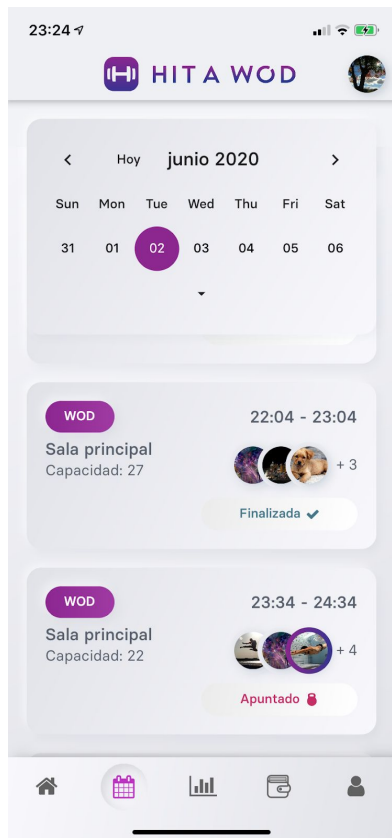


fig.65 - App, Calendari

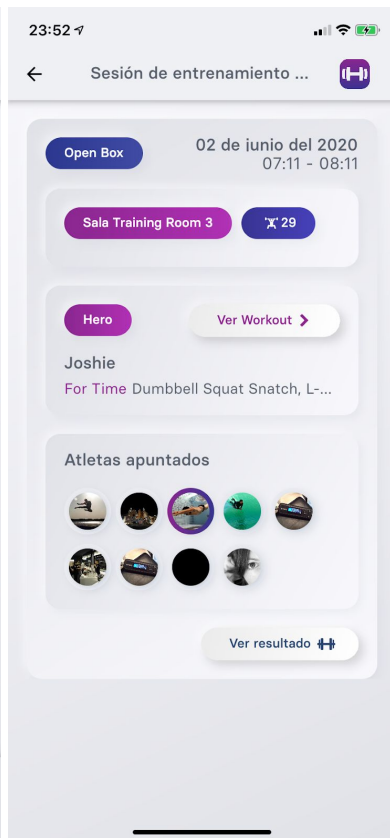


fig.66 - App, Detall sessió

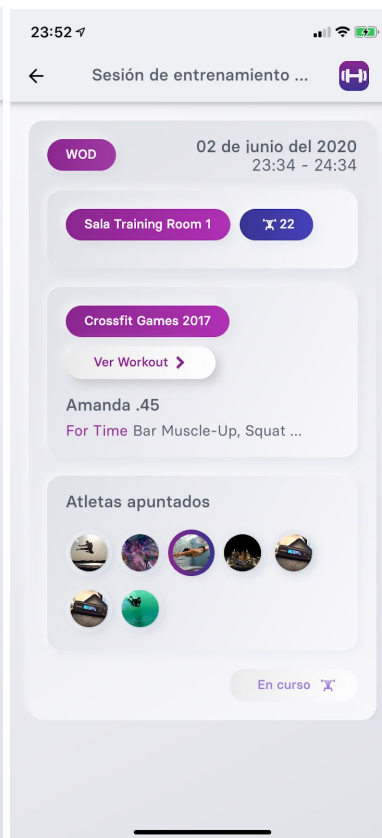


fig.67 - App, Detall sessió



## 10.1.2.5. Usuari

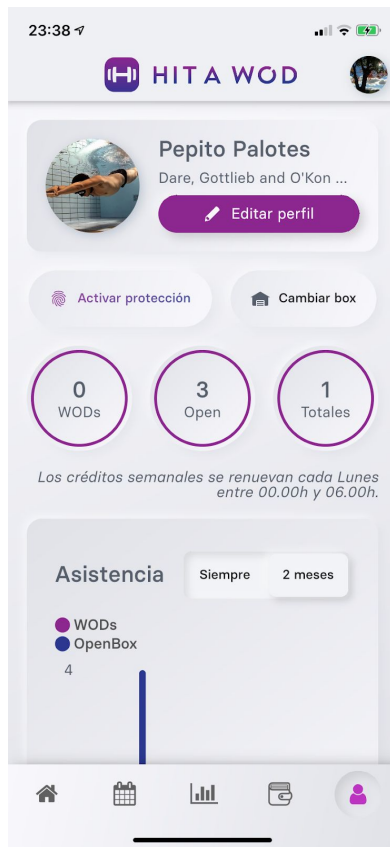


fig.68 - App, Usuari, part 1

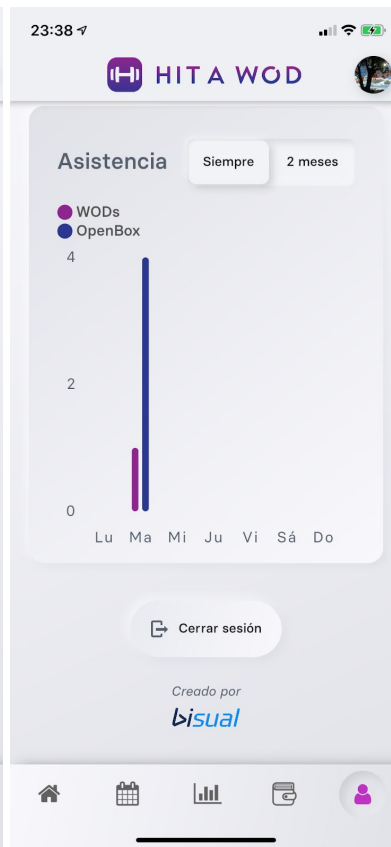


fig.69 - App, Usuari, part 2

fig.70 - App, Edita usuari

## 10.1.2.6. Facturació, subscripcions i box orders

Pantalla molt modular en funció si l'empresa té Stripe activat, si tens configurat un mètode de pagament, si estàs en una subscripció automàtica o no, etc. Pots processar pagaments únics, iniciar subscripcions, canviar el teu mètode de pagament.



fig.71 - App, Facturació, part 1

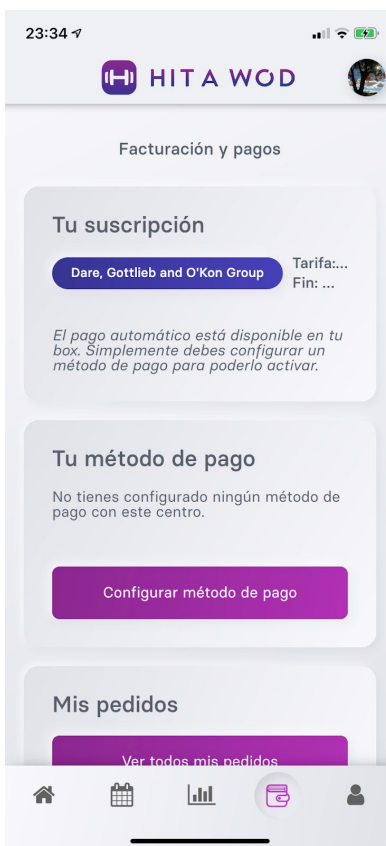


fig.72 - App, Facturació, part 2

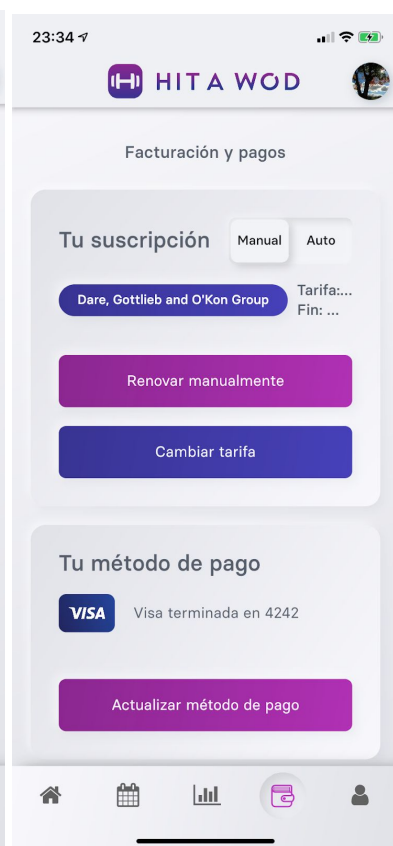


fig.73 - App, Facturació, part 3



fig.74 - App, Facturació, part 4



fig.75 - App, Facturació, part 5



fig.76 - App, Facturació, part 6

## 10.1.2.7. Workouts

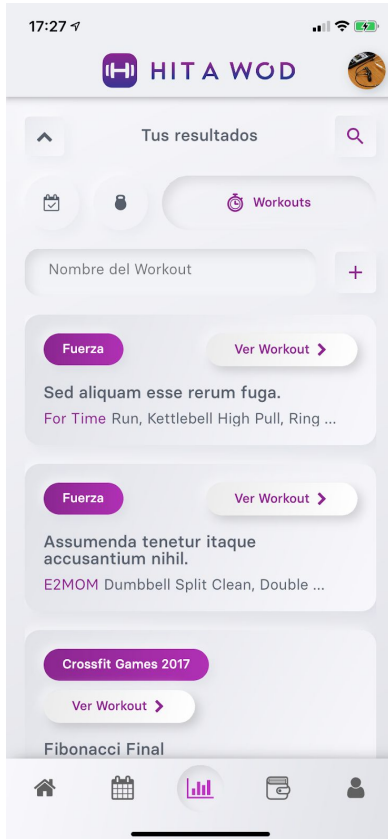


fig.77 - App, Workouts

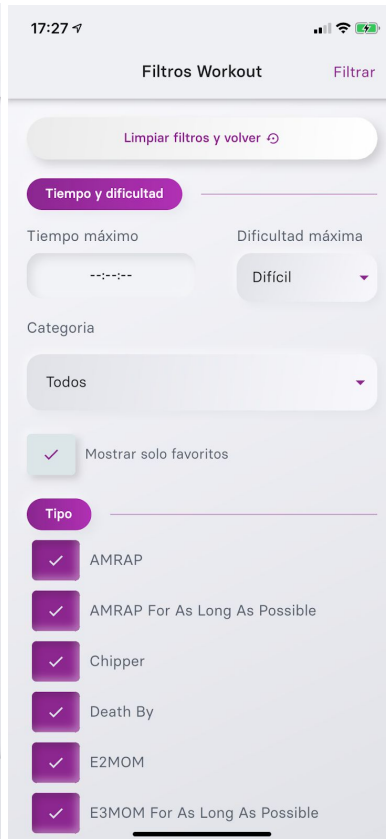


fig.78 - App, Workouts, Filtre

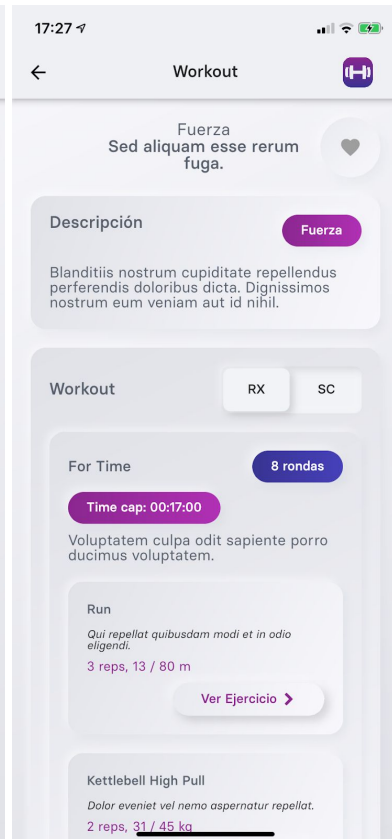


fig.79 - App, Workouts, Detall

Aquí els atletes podran accedir a les pantalles de Workouts, filtrar de forma avançada en funció de la tipologia de WOD que busquin i els exercicis que prefereixin fer, així com una descripció detallada de les modalitats RX o Scaled, tots els detalls del mateix i veure els últims resultats d'altres cops que l'hagin realitzat.

## 10.1.2.8. Exercicis

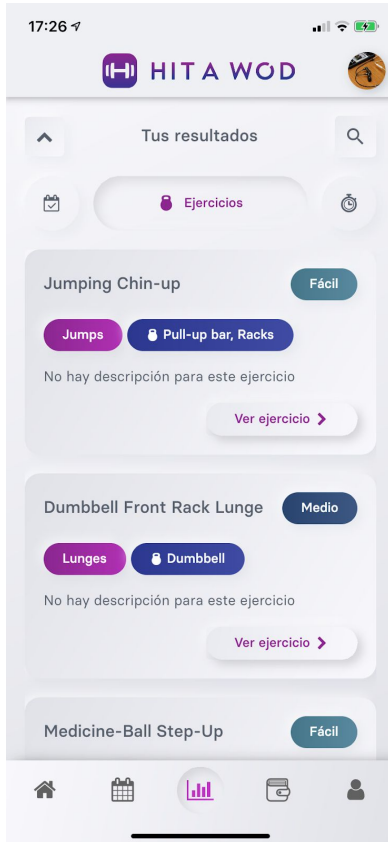


fig.77 - App, Exercicis

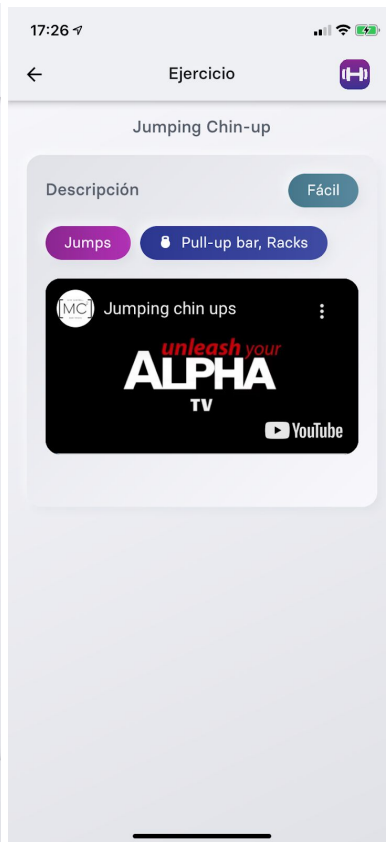


fig.78 - App, Exercicis, Detail

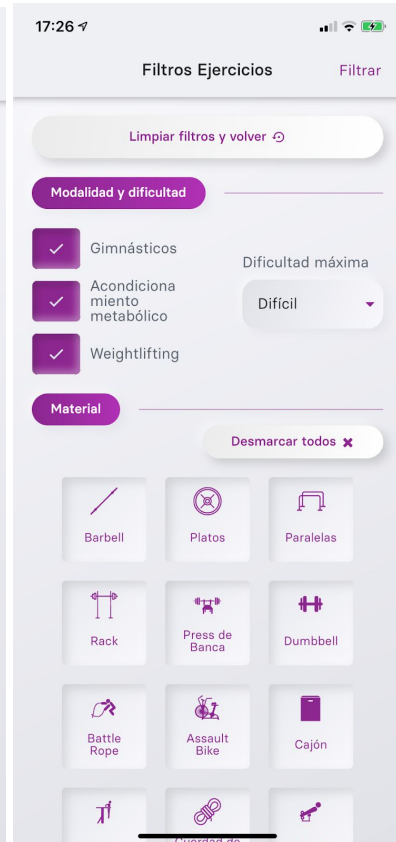


fig.79 - App, Exercicis, Filtre

Aquestes són les pantalles referents a la llibreria d'exercicis que disposa Hit a Wod. Permet tenir filtres molt avançats, i si ja hem realitzat l'exercici molts cops, podem veure el progrés de l'atleta en quant a l'exercici (per exemple, veure l'augment de pes).

### 10.1.2.9. Benchmarks d'un workout

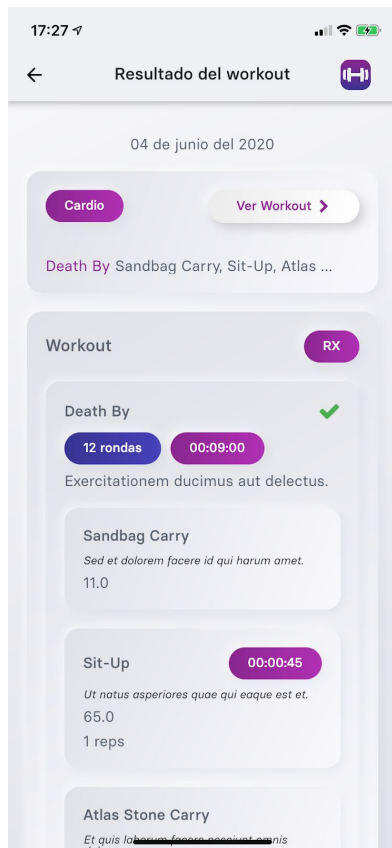


fig.80 - App, Benchmark

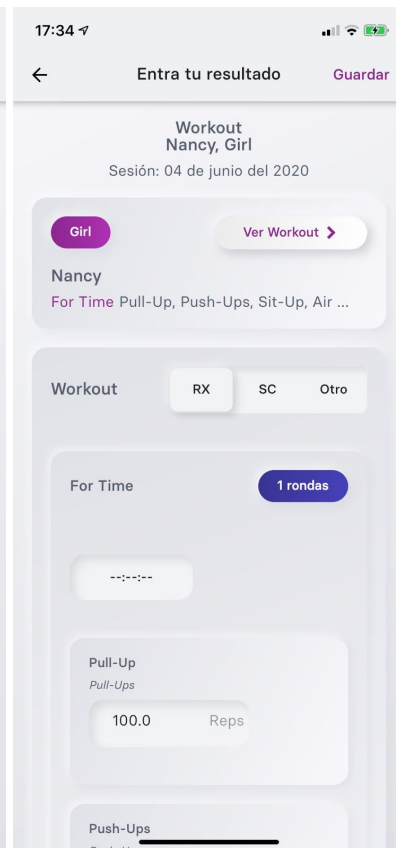


fig.81 - App, Benchmark

Un cop realitzada una sessió, si té un workout establert pels coaches, podrem entrar el nostre resultat per mantenir una evolució dels nostres resultats. Podràs introduir rondes, pes, modalitats, temps dels teus resultats, perfectament adaptat a cadascun dels workouts.

### 10.1.2.10. Notícies

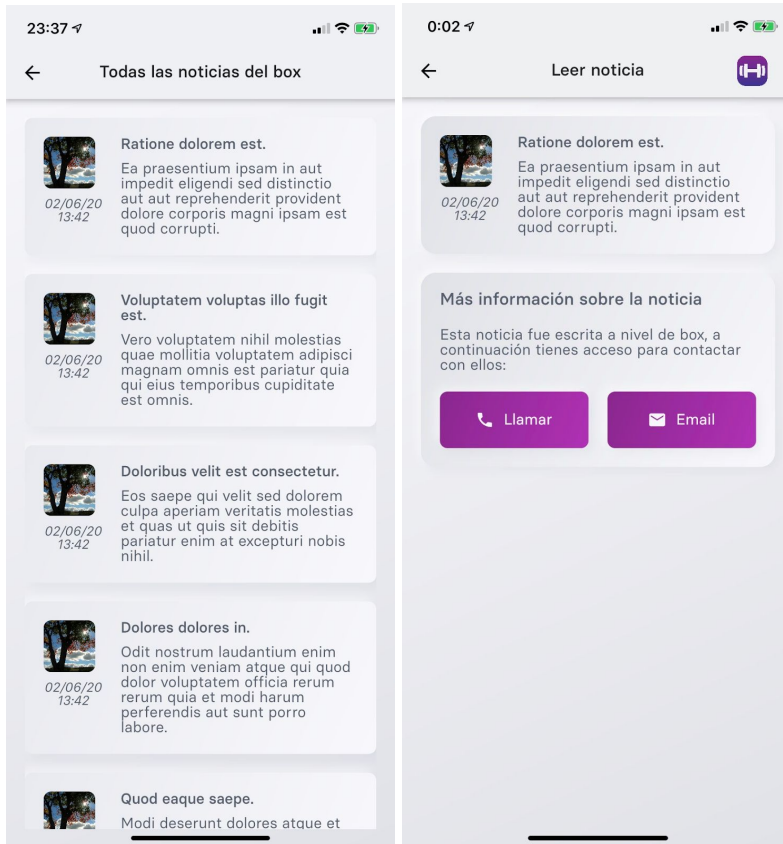


fig.82 - App, Notícies

fig.83 - App, Notícies, Detall

## 10.2. Producció i deploy

El nostre TFG no ha inclòs l'apartat de posta en producció i deploy de l'aplicació, ja que el projecte ha estat molt gran i no hi ha hagut prou temps. Tot i així, a data de Maig de 2020 podem confirmar que ja estem treballant i provant diferents estratègies de deployment com per exemple Dockers, tenint en compte escalabilitat i possibles opcions com balanceig de càrrega. També estem considerant diversos entorns de producció, com ara incloure preproducció per minimitzar errors i tenir un entorn entremig entre desenvolupament i producció.

Com es pot suposar, les aplicacions no han estat penjades als seus respectius stores (App Store i Google Play Store), però s'hi pujaran en els pròxims mesos quan el producte es llanci i es comenci la seva acció comercial.

## 10.3. Legalitat

Respecte a l'aspecte legal, el nostre sistema tracta dades de caràcter personal, de caràcter identificatives, característiques personals i econòmiques/financeres i d'assegurances. Ens hem hagut d'assegurar l'aplicació de la legislació actual quant a la "*Ley Orgánica de Protección de Datos y la Ley de Servicios de la Sociedad de la Información y Comercio Electrónico*".

El nostre sistema recupera l'avís legal, l'avís de cookies, els termes i condicions i la política de privacitat a través d'una plantilla que generem en el backend i substitueix les dades d'aquesta per les dades de l'empresa, que es troben guardades a una taula de la base de dades del backend; atributs com el nom, el número telèfon del responsable, etc. Ja que cada atleta haurà d'acceptar:

1. Quan es registri, saber i acceptar que les seves dades es guarden en la base de dades de Hit a Wod.
2. Quan iniciï una subscripció, saber i acceptar que les seves dades seran cedides a l'empresa per poder facturar-lo.

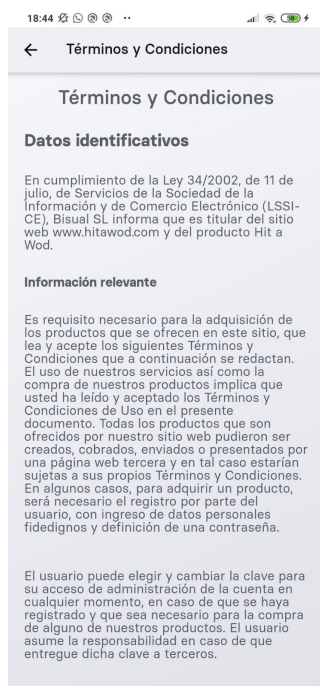


fig.84: Legalitat al client mòbil

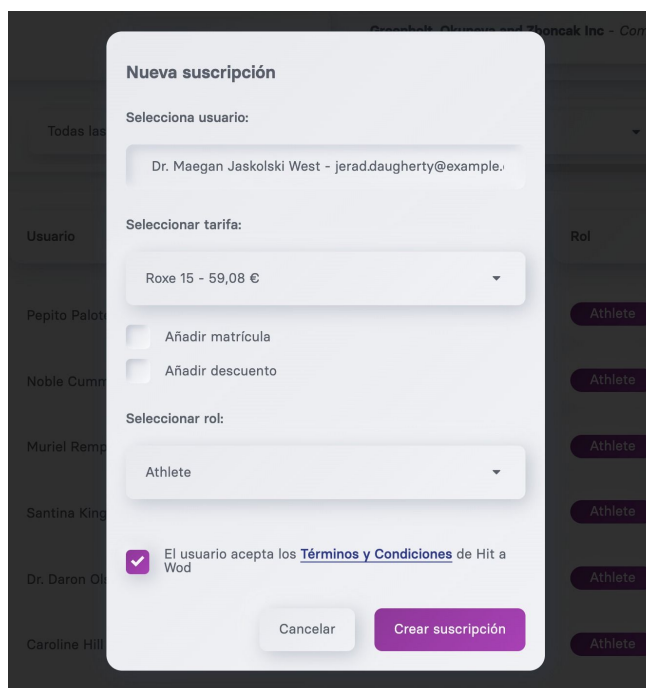


fig.85: Legalitat al client web, crear subscripció

## Política de protección de datos

Tus datos están seguros con Hit a Wod. Aquí tienes un resumen de nuestra política de protección de datos.

Información básica sobre protección de datos	
<b>Responsable</b>	Bisual SL <a href="#">+Info</a>
<b>Finalidad</b>	Gestión de las peticiones realizadas a través del Sitio Web o App y, en su caso, envío de comunicaciones comerciales electrónicas. <a href="#">+Info</a>
<b>Legitimación</b>	Consentimiento del interesado <a href="#">+Info</a>
<b>Destinatarios</b>	Se podrán ceder datos a Stripe y OneSignal. <a href="#">+Info</a>
<b>Derechos</b>	Acceder, rectificar, suprimir y oponerse, así como otros derechos, tal y como se explica en la "información adicional". <a href="#">+Info</a>
<b>Información adicional</b>	Puede consultar la información adicional y detallada sobre la protección de datos en la <a href="#">Política de Privacidad</a> <a href="#">+Info</a>

Acepto el tratamiento de mis datos personales y corporativos de conformidad con la [Política de Privacidad](#).

Atrás

Registrar y pagar

fig.86: Legalitat al registre d'una nova companyia



# 11. Conclusions

Vam iniciar el projecte el Setembre de 2019, conscients que seria un treball amb molta feina. Hem aconseguit presentar un projecte que assoleix l'aplicació que vam plantejar-nos fer des del principi, millorar les aplicacions que hi havien al mercat i oferir noves característiques que simplifiquessin i milloressin l'experiència tant des de la vessant de l'administrador del negoci com de l'atleta.

Ara bé, al llarg de tot aquest temps se'ns ha i ens hem proposat nous objectius i noves característiques per l'aplicació que la portarien a un altre nivell, com ara una botiga integrada, una televisió del box, entre altres, que no hem pogut presentar en aquesta entrega però que ja tenim plantejat posar-nos a desenvolupar en les pròximes setmanes. També ens hagués agradat tenir ja un sistema d'integració contínua implementat i funcionant i poder-lo presentar també en aquesta entrega, però hem prioritzat entregar un producte solvent, fiable i que assolís els objectius de l'inici.

Creiem que ho hem aconseguit, i tot i no haver arribat a la versió que tenim planejada amb totes les característiques, podem dir que ens hem situat tant a nivell d'usabilitat com de característiques pel davant de la nostra competència.

## 11.1. Resultat i comparativa amb la competència

Hit a Wod és un projecte que es va pensar des d'un inici per arribar a ser un producte comercial i ser ofert com una solució real a un problema real, amb un estudi de mercat i de viabilitat i una posterior execució i posada a producció.

No només com a solució, sinó com a la millor solució. Per això, als inicis, vam analitzar la competència amb les característiques que ofereixen i així saber els punts on podíem destacar i els que podíem millorar.

Vam preparar una taula comparativa amb el que és i serà Hit a Wod respecte la competència:

	<b>Hit a Wod</b>	<b>SugarWod</b>	<b>Wodbuster</b>	<b>Virtuagym</b>
<b>Llançament</b>	-	2013-07-03	2017-01-16	2014-03-21
<b>Descàrregues (Google Play Store)</b>	-	+100.000	+50.000	+5.000.000
<b>Estat</b>	-	Estable	A la baixa	Estancat
<b>Per a Crossfit?</b>	Sí	Sí	Sí	Per gimnàs

<b>Multi-box</b>	Sí	No	No	No
<b>Idiomes</b>	1 (més idiomes en pròximes setmanes)	1	1	5
<b>TV Box</b>	En desenvolupament	Sí	Sí	No
<b>Reserva de classes</b>	Sí	Sí	Sí	Sí
<b>Historial d'estadístiques</b>	Sí, molt complet i detallat	Sí, però sense relacions i poc detallat	Sí, però sense relacions i poc detallat	No
<b>Notificacions</b>	Sí	No	No	No
<b>Botiga integrada</b>	En pròximes setmanes	No	No	No
<b>Comunicació i xat amb el box</b>	Sí. Xat en pròximes setmanes	No	Sí	No
<b>Gestió de pagaments i subscripcions automàtiques</b>	Sí	No	Sí	Sí
<b>Convidats i bonos</b>	Sí	No	Sí	No
<b>Facturació completa</b>	Sí	No	Sí	Sí
<b>Informes BI</b>	Sí	Sí	Sí	No
<b>Migració</b>	En pròximes setmanes	No	Sí	No
<b>Avís lloc lliure a la sessió</b>	Sí	No	Sí	No
<b>Sincronització calendari</b>	En pròximes setmanes	No	Sí	No
<b>Box Branding</b>	Sí	Sí	Sí	Sí
<b>Suport</b>	Sí	Sí	Sí	Sí
<b>Llibreria de</b>	Sí, estructurat i	Text pla	Text pla	No

<b>WODs</b>	detallat			
<b>Llibreria d'exercicis</b>	Sí, estructurat i detallat	Text pla	No	Sí, però no de Crossfit

Els principals punts que faran que Hit a Wod sigui un producte especial i destaquí per sobre els altres són:

- **Multi box i multi empresa:** Actualment no hi ha cap aplicació de Crossfit al mercat que permeti, amb el mateix usuari, estar subscrit a varis boxes a la vegada i per tant no perdre registres d'estadístiques. Manté separat l'usuari i registres amb les subscripcions. Per exemple: Si un dia te'n vas de vacances a una ciutat on hi ha un box que usa Hit a Wod, els registres que facis allà contribuiran a les teves estadístiques, amb independència del box en el que ho hakis registrat.
- **Llibreria d'exercicis i WODs:** Algunes de les aplicacions actuals de Crossfit pots escriure'ls el WOD o exercicis que es farà en una sessió. Però no permeten ni reutilitzar WODs, ni tenir centralitzats els WODs coneguts per tot el món, i a més, són a text pla, amb el que informàticament es fa impossible de mantenir un registre estadístic com el que ofereix Hit a Wod. A més, permetem filtrar per competició, tipus de Wod, seccions, si és un AMRAP, temps, rondes, pes, modalitat RX, Scaled, etc. I en el cas dels exercicis, permetem veure quants cops s'ha realitzat, l'equipament necessari, la dificultat i podem montar filtres.
- **Les estadístiques més completes d'evolució d'atleta:** Molt lligat amb el punt anterior. Com que tenim una estructura de WODs i exercicis relacionats entre ells, podem mostrar estadístiques molt detallades de l'evolució de l'atleta amb un exercici o un wod en concret. En canvi les altres aplicacions, al ser de text pla, no poden oferir aquest nivell d'estadístiques. És un punt que serà molt útil i atraparà a l'atleta que usi l'aplicació.
- **Botiga integrada:** Tot i que encara no és una característica que estigui disponible al moment d'entregar aquest treball, sí que serà una característica estrella que al moment de llegir aquest document probablement ja s'estarà implementant. Permetrà integrar la botiga del box amb Hit a Wod i poder generar més ingressos als boxes. I els atletes podran pagar a un clijk amb el mètode de pagament configurat.
- **Pagaments i facturació:** Hit a Wod manté un resum d'ingressos i despeses per tal que els administradors puguin veure-ho fàcilment des del dashboard. Genera factures amb tota la informació fiscal necessària i també genera els PDFs de les factures. Gestiona també les subscripcions recurrents i els cobra cada període configurat de forma automàtica.

Tot i que encara no és una aplicació acabada al 100%, sabem que tenim un producte molt potent i que si aconseguim vendre'l bé, pot ser un èxit en tots els sentits.

## 12. Repartiment de feina

Des de l'inici fins al final, ens hem involucrat tots dos en totes les fases del projecte, anàlisi, planificació i desenvolupament i a nivells iguals. No s'han calculat línies de codi ni els temps emprats per cadascun de nosaltres, però sí que s'ha col·laborat molt activament (i quan diem molt, ens referim a que potser hem estat fent quasi una reunió diària, com a mínim per solucionar problemes). Tot i que és cert que cadascú, en la fase de desenvolupament, s'ha dedicat més a uns temes, quan un tenia un problema o dubte, sempre recorria a l'altre, és a dir, els dos al final hem estat involucrats en tots els àmbits del producte final.

Creiem que seria injust posar un repartiment de feina que no fos del 50% per cadascú, així que aquesta és la nostra valoració. A més, som conscients que hi hem dedicat moltes més hores de les que es farien normalment en un projecte de fi de grau, ja que portem des de Setembre 2019 treballant-hi i ho continuarem fent després d'aquesta entrega.

Adjuntem en les següents línies curiositats sobre com ens hem organitzat:

Tauler de Trello (veure fig.87):

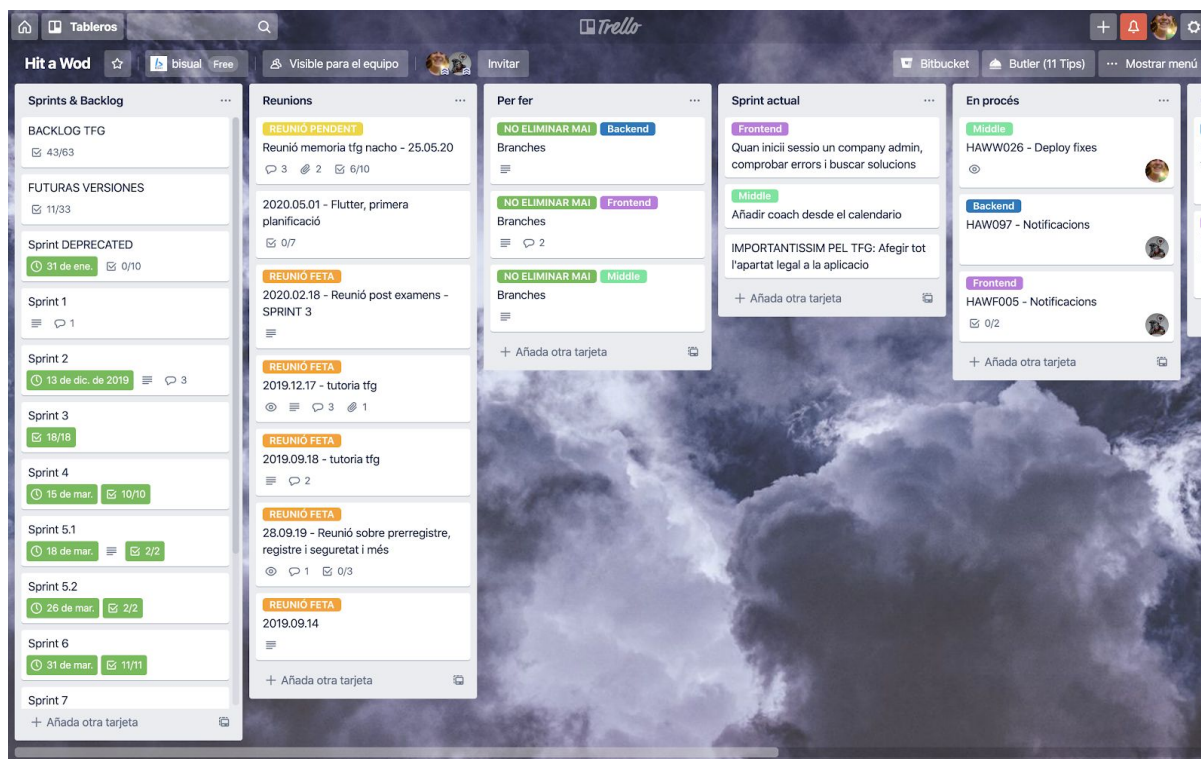
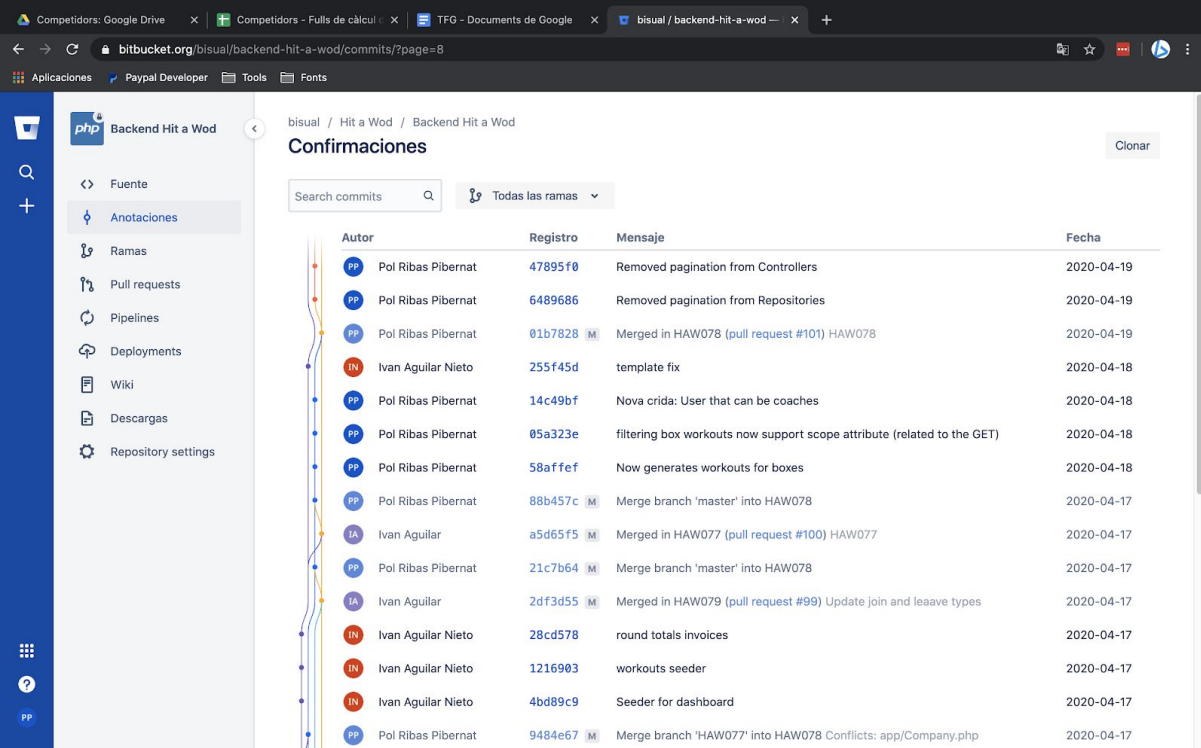


fig. 87: Trello

## Repositori Laravel de Bitbucket (veure fig.87):



The screenshot shows the Bitbucket web interface for the repository 'bisual / Hit a Wod / Backend Hit a Wod'. The page title is 'Confirmaciones' and it displays a list of 14 commits. The interface includes a search bar for commits, a dropdown for branches (set to 'Todas las ramas'), and a 'Clonar' button. The commit list is organized into columns: 'Autor', 'Registro', 'Mensaje', and 'Fecha'. The commits are ordered chronologically from newest at the top to oldest at the bottom. The commit history is visualized with a vertical timeline on the left side of the table, showing the progression of branches and merges.

Autor	Registro	Mensaje	Fecha
Pol Ribas Pibernat	47895f0	Removed pagination from Controllers	2020-04-19
Pol Ribas Pibernat	6489686	Removed pagination from Repositories	2020-04-19
Pol Ribas Pibernat	01b7828 M	Merged in HAW078 (pull request #101) HAW078	2020-04-19
Ivan Aguilar Nieto	255f45d	template fix	2020-04-18
Pol Ribas Pibernat	14c49bf	Nova crida: User that can be coaches	2020-04-18
Pol Ribas Pibernat	05a323e	filtering box workouts now support scope attribute (related to the GET)	2020-04-18
Pol Ribas Pibernat	58affeef	Now generates workouts for boxes	2020-04-18
Pol Ribas Pibernat	88b457c M	Merge branch 'master' into HAW078	2020-04-17
Ivan Aguilar	a5d65f5 M	Merged in HAW077 (pull request #100) HAW077	2020-04-17
Pol Ribas Pibernat	21c7b64 M	Merge branch 'master' into HAW078	2020-04-17
Ivan Aguilar	2df3d55 M	Merged in HAW079 (pull request #99) Update join and leave types	2020-04-17
Ivan Aguilar Nieto	28cd578	round totals invoices	2020-04-17
Ivan Aguilar Nieto	1216903	workouts seeder	2020-04-17
Ivan Aguilar Nieto	4bd89c9	Seeder for dashboard	2020-04-17
Pol Ribas Pibernat	9484e67 M	Merge branch 'HAW077' into HAW078 Conflicts: app/Company.php	2020-04-17

fig. 88: Bitbucket

## 13. Bibliografía

1. Laravel Installation  
<https://laravel.com/docs/7.x/installation>
2. Laravel Eloquent  
<https://laravel.com/docs/7.x/eloquent>
3. Laravel Database: Migrations  
<https://laravel.com/docs/7.x/migrations>
4. Laravel Database: Seeding  
<https://laravel.com/docs/7.x/seeding>
5. Laravel Database: Transactions  
<https://laravel.com/docs/7.x/database#database-transactions>
6. Carbon PHP  
<https://carbon.nesbot.com/docs/>
7. Fernando Herrera. Flutter, Tu guía completa de desarrollo para IOS y Android  
<https://www.udemy.com/course/flutter-ios-android-fernando-herrera/>
8. Flutter Installation  
<https://flutter-es.io/docs/get-started/install>
9. Flutter Widgets  
<https://flutter-es.io/docs/development/ui/widgets>
10. Flutter Dio  
<https://pub.dev/packages/dio>
11. Flutter Calendar  
[https://pub.dev/packages/flutter\\_calendar](https://pub.dev/packages/flutter_calendar)
12. Flutter Youtube Plater  
[https://pub.dev/packages/youtube\\_player\\_flutter](https://pub.dev/packages/youtube_player_flutter)
13. Flutter OneSignal  
<https://documentation.onesignal.com/docs/flutter-sdk-setup>
14. OneSignal Push Notification  
<https://documentation.onesignal.com/docs/sending-notifications>

15. Berkay Kaya, 26 d'Abril del 2016, Laravel OneSignal  
<https://github.com/berkayk/laravel-onesignal>
16. Flutter Neumorphic  
[https://pub.dev/packages/flutter\\_neumorphic](https://pub.dev/packages/flutter_neumorphic)
17. Legislació AEPD  
<https://www.aepd.es/es/guias-y-herramientas/guias>
18. Fernando Herrera, Angular, De cero a experto creando aplicaciones (Angular 9+)  
<https://www.udemy.com/course/angular-2-fernando-herrera/>
19. Angular Charts, ngx-charts  
<https://swimlane.gitbook.io/ngx-charts/>
20. Angular Skeleton Loader, ngx-skeleton-loader  
<https://stackblitz.com/edit/ngx-skeleton-loader-sample?file=app%2Fapp.component.html>
21. Stripe API Reference  
<https://stripe.com/docs/api>

## 14. Treball futur

Com hem explicat a l'apartat de conclusions, aquest treball no està acabat. Volem incorporar més característiques noves al projecte abans de llançar-lo al mercat, tot i que actualment ja tenim implementades les més significatives.

### 14.1. Continuous Integration


Hem començat a preparar estratègies de deployment i d'integració contínua per tal de fer el desenvolupament en producció fluïd.

L'opció que té més números que integrem és la de fer un deploy basat amb dockers.

### 14.2. Afegir noves característiques

Com es pot apreciar en la captura de Trello que hem adjuntat anteriorment, es pot veure que tenim un backlog i característiques per a futures versions. Les característiques més rellevants que volem implementar en les pròximes setmanes són:

- Migració automàtica des d'altres softwares
- Sincronització amb el calendari preferit
- Oferir suport de forma fàcil per les empreses

- 
- Crear una pàgina web pública per vendre el producte (la típica de màrqueting)
  - TV Box, per veure els resultats del wod del dia en una pantalla dins el box.
  - Botiga integrada dins l'aplicació, perquè les empreses puguin vendre productes als atletes.
  - Internacionalitzar l'aplicació: afegir suport per més idiomes.
  - Sincronitzar els ingressos de Hit a Wod amb el programa de facturació de Bisual

### **14.3. Llançament al mercat i venda**

Tenim previst preparar una campanya de màrqueting potent per vendre l'aplicació un cop haguem incorporat les últimes característiques que considerem fonamentals prèvies al llançament.

La nostra idea és preparar una campanya agressiva, amb promocions, vídeos de demo, FAQs i incorporar gent a l'equip com ara comercials, per suport tècnic, etc.