

GESTCOOK

Grau de Multimèdia

Desenvolupament d'aplicacions interactives

Autor

Francesc Marmol Preciado

Tutor

Kenneth Capseta Nieto

Professor

Carlos Casado Martínez

7 de gener de 2019

Crèdits/Copyright

Memòria:



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

Aplicació Web Progressiva:

Copyright © Francesc Marmol Preciado, 2018. Tots els drets reservats.

L'accés als continguts d'aquesta obra i la seva utilització ha de respectar els drets de la persona autora. Pot ser utilitzada per a consulta o estudi personal, així com per a usos emmarcats en activitats d'investigació i docència. Per a altres utilitzacions es requereix l'autorització prèvia i expressa de la persona titular dels drets. En qualsevol cas, en la utilització dels seus continguts caldrà indicar de forma clara el nom i cognoms de la persona autora i el títol de l'obra. No s'autoritza la seva reproducció o altres formes d'explotació efectuades amb finalitats de lucre ni la seva comunicació pública des d'un lloc aliè a on es troba allotjat.

Serveis de tercers:

Aquest aplicació utilitza serveis de tercers:

- **Angular:** Framework JavaScript/TypeScript (llicència [MIT](https://opensource.org/licenses/MIT))
- **Node.js:** Entorn d'execució per a JavaScript construït amb el motor V8 de *Google* (llicència [MIT](https://opensource.org/licenses/MIT))
- **jQuery:** Llibreria JS (llicència [MIT](https://opensource.org/licenses/MIT))
- **Bootstrap:** Llibreria CSS (llicència [MIT](https://opensource.org/licenses/MIT))
- **Bootstrap:** Llibreria CSS (llicència [MIT](https://opensource.org/licenses/MIT))
- **Firebase:** Plataforma de desenvolupament que ofereix diversos serveis (llicència [Apache 2.0](https://www.apache.org/licenses/LICENSE-2.0)):
 - Firebase Auth: Autenticació d'usuaris amb codi del costal client.
 - Realtime Database: BBDD NoSQL per *back-end*.
 - Firebase Storage: Emmagatzematge al núvol per a la càrrega i descàrrega d'arxius.

FITXA DEL TREBALL FINAL

Títol del treball:	<i>GESTCOOK</i>
Nom de l'autor:	<i>Francesc Marmol Preciado</i>
Nom del col·laborador/a docent:	<i>Kenneth Capseta Nieto</i>
Nom del PRA:	<i>Carlos Casado Martínez</i>
Data de lliurament (mm/aaaa):	<i>01/2019</i>
Titulació o programa:	<i>Grau de Multimèdia</i>
Àrea del Treball Final:	<i>Desenvolupament d'aplicacions interactives</i>
Idioma del treball:	<i>Català</i>
Paraules clau	<i>PWA, ANGULAR, FIREBASE</i>

Resum:

A diari les persones inverteixen gran part del seu temps en llargs desplaçaments, jornades maratonianes i altres compromisos. A aquestes tasques també cal afegir el fet que ens hem d'alimentar, un acte que comporta una gran inversió de temps en planificar, comprar i cuinar.

Però què passa si no tenim aptituds per a la cuina, temps per comprar i/o cuinar? És en aquest punt on encaixa aquest projecte. Actualment existeixen múltiples solucions però molt generalitzades. GESTCOOK vol ser una solució específica per a aquest tipus de públic objectiu.

El projecte GESTCOOK té com a objectiu oferir una aplicació web progressiva (PWA), senzilla, intuïtiva i àgil, que proporcioni als usuaris un punt de trobada on poder compartir coneixements i solucions a un problema comú.

Gràcies a la vessant col·laborativa de GESTCOOK l'usuari esdevé prosumidor. Pot crear, compartir i consultar receptes, a la vegada que li permet gestionar els aspectes relacionats en aquest àmbit, gràcies a un calendari on afegir les receptes, un auto generador de la llista de productes en base a les receptes i un servei de compra en línia amb entrega a domicili.

Aquest projecte serveix per observar cap a on està evolucionant el web i quines noves tècniques s'estan aplicant. Per aquest motiu s'utilitza tecnologia de codi obert que ofereix *Google*. Al *front-end* s'utilitza el *framework Angular* i al *back-end* la plataforma de desenvolupament *Firebase*, que es combinen amb llibreries d'estils com *jQuery* o *Bootstrap* i APIs REST per aconseguir una aplicació responsiva i multi dispositiu.

Abstract:

People every day spend a lot of their time in long journeys, endless day and other commitments. To these tasks we must also add the fact that we have to feed, this act involves a great waste of time in planning, buying and cooking.

But what happens if we don't have skills to kitchen, time to buy and/or to cook? This project fits at this point perfectly. There are currently multiple solutions, but they are widespread. GESTCOOK wants to be a specific solution for this target audience.

GESTCOOK project have the goal to offer a progressive web application (PWA), simple, intuitive and agile, which provides users a meeting point where they can share knowledge and solutions to a common problem.

Thanks to collaborative side of GEESTCOOK user becomes prosumer. He can create, share and check recipes, and at the same time it allows him to manage the related aspects in this topic, thanks to a calendar where add recipes, a product list generator based on the recipes and a purchase online service with home delivery.

This project wants to see where the web is evolving and which new techniques are currently being applied. For this reason, the project is based on open source technology offered by Google. The framework Angular is used at the front end and the Firebase development platform is used at the back end, and they are combined with style libraries such as jQuery or Bootstrap and REST API to achieve a responsive and multi-device application.

Dedicatòria

A la meva família, especialment a la meva dona Ariadna per la seva comprensió, per la seva paciència, pel seu suport, per la seva involucració, per la seva ajuda, pels seus consells i per tots aquells moments que no he pogut passar amb ella. Gràcies per aguantar amb estoïcisme els moments de desesperació i gràcies per compartir els moments d'alegria i satisfacció.

Cita

“Your time is limited, don't waste it living someone else's life. Don't be trapped by dogma, which is living the result of other people's thinking. Don't let the noise of other opinions drown your own inner voice. And most important, have the courage to follow your heart and intuition, they somehow already know what you truly want to become. Everything else is secondary.” - Steve Jobs, (1955 – 2011)

Agraïments

A la meva dona Ariadna per donar-me tot el seu suport per continuar estudiant. A l'Hèctor per la seva ajuda inestimable i la seva enorme paciència durant la meva iniciació a la programació. Als companys de classe amb els qui he compartit penes i glòries. I a totes aquelles persones que han fet possible que arribés fins aquí.

Abstract

People every day spend a lot of their time in long journeys, endless day and other commitments. To these tasks we must also add the fact that we have to feed, this act involves a great waste of time in planning, buying and cooking.

But what happens if we don't have skills to kitchen, time to buy and/or to cook? This project fits at this point perfectly. There are currently multiple solutions, but they are widespread. GESTCOOK wants to be a specific solution for this target audience.

GESTCOOK project have the goal to offer a progressive web application (PWA), simple, intuitive and agile, which provides users a meeting point where they can share knowledge and solutions to a common problem.

Thanks to collaborative side of GEESTCOOK user becomes prosumer. He can create, share and check recipes, and at the same time it allows him to manage the related aspects in this topic, thanks to a calendar where add recipes, a product list generator based on the recipes and a purchase online service with home delivery.

This project wants to see where the web is evolving and which new techniques are currently being applied. For this reason, the project is based on open source technology offered by Google. The framework Angular is used at the front end and the Firebase development platform is used at the back end, and they are combined with style libraries such as jQuery or Bootstrap and REST API to achieve a responsive and multi-device application.

Keywords

Management, Kitchen, Share, Calendar, List, jQuery, Bootstrap, Angular, Firebase, PWA, Progressive Web App, Responsive, Multi-device, Browser web, Front-end, Back-end. Framework, REST API.

Resum

A diari les persones inverteixen gran part del seu temps en llargs desplaçaments, jornades maratonianes i altres compromisos. A aquestes tasques també cal afegir el fet que ens hem d'alimentar, un acte que comporta una gran inversió de temps en planificar, comprar i cuinar.

Però què passa si no tenim aptituds per a la cuina, temps per comprar i/o cuinar? És en aquest punt on encaixa aquest projecte. Actualment existeixen múltiples solucions però molt generalitzades. GESTCOOK vol ser una solució específica per a aquest tipus de públic objectiu.

El projecte GESTCOOK té com a objectiu oferir una aplicació web progressiva (PWA), senzilla, intuïtiva i àgil, que proporcioni als usuaris un punt de trobada on poder compartir coneixements i solucions a un problema comú.

Gràcies a la vessant col·laborativa de GESTCOOK l'usuari esdevé prosumidor. Pot crear, compartir i consultar receptes, a la vegada que li permet gestionar els aspectes relacionats en aquest àmbit, gràcies a un calendari on afegir les receptes, un auto generador de la llista de productes en base a les receptes i un servei de compra en línia amb entrega a domicili.

Aquest projecte serveix per observar cap a on està evolucionant el web i quines noves tècniques s'estan aplicant. Per aquest motiu s'utilitza tecnologia de codi obert que ofereix *Google*. Al *front-end* s'utilitza el *framework Angular* i al *back-end* la plataforma de desenvolupament *Firebase*, que es combinen amb llibreries d'estils com jQuery o Bootstrap i APIs REST per aconseguir una aplicació responsiva i multi dispositiu.

Paraules clau

Gestió, Cuina, Compartir, Calendari, Llistat, jQuery, Bootstrap, Angular, Firebase, PWA, Aplicació Web Progressiva, Responsiu, Multi dispositiu, Navegador web, Front-end, Back-end. Framework, API REST.

Notacions i Convencions

1. Text capítol: Arial, negreta, 20pt, taronja

1.1. Text secció: Arial, negreta, 13pt, taronja

1.1.1. Text sub-secció: Arial, negreta, 13pt, taronja

Text normal: Arial, regular, 10pt, negre.

Text cursiva: Arial, cursiva, 10pt, negre.

Text ressaltat: Arial, negreta, 10pt, taronja.

Text enllaç web: Arial, subratllat, 8pt, taronja.

Text notes al peu de pàgina: Arial, regular, 9pt, taronja.

Text nota al peu de figura: Arial, regular, 8pt, taronja.

- Element llista no ordenada

Text de cita: Arial, cursiva, 12pt, taronja

Text per a notes, avisos o aclariments: Arial, regular, 10pt, taronja

Text per a indicar línies de codi: Arial, regular, 10pt, negre, color ressaltat gris 25%

Índex

1. Introducció	14
1.1. Introducció/Prefaci	14
1.2. Descripció/Definició	15
1.3. Objectius generals	17
1.3.1. Objectius principals	17
1.3.2. Objectius secundaris.....	17
1.4. Metodologia i procés de treball	18
1.5. Planificació	19
1.5.1. Fites	19
1.5.2. Dates clau.....	20
1.5.3. Diagrama Gantt.....	21
1.6. Pressupost	22
1.6.1. Equip humà	22
1.6.2. Equipament tècnic.....	23
1.6.3. Altres recursos	23
1.6.4. Resum pressupost	23
1.7. Estructura de la resta del document	24
2. Anàlisi de mercat	25
2.1. Públic objectiu (i.e. <i>target audience</i>) i perfils d'usuari	26
2.1.1. Públic objectiu	26
2.1.2. Perfils d'usuari.....	26
2.2. Competència	27
2.3. Anàlisi DAFO	28
3. Proposta	29
3.1. Definició d'objectius/especificacions del producte	29
3.2. Model de negoci	30
3.3. Estratègia de màrqueting	30
4. Disseny	31
4.1. Arquitectura general de l'aplicació/sistema/servei	31
4.1.1. Client (<i>front-end</i>):.....	32
4.1.2. Servidor (<i>back-end</i>):.....	33
4.1.3. APIs:	35

4.2. Arquitectura de la informació i diagrames de navegació.....	38
4.3. Disseny gràfic i interfícies.....	42
4.3.1. Estils	42
4.3.2. Usabilitat/UX.....	44
4.4. Llenguatges de programació i APIs utilitzats.....	45
5. Implementació	46
5.1. Requisits d'instal·lació	46
5.2. Instruccions d'instal·lació	46
5.3. Instal·lació i configuració entorn de treball	47
5.4. Desenvolupament de l'aplicació.....	48
5.4.1. <i>Front-end</i>	48
5.4.2. <i>Back-end</i>	51
5.4.3. Control de versions	59
6. Demostració.....	60
6.1. Instruccions d'ús	60
6.2. Prototips	60
6.2.1. Prototips Lo-Fi.....	60
6.2.2. Prototips Hi-Fi	61
6.3. Tests.....	62
7. Conclusions i línies de futur	68
7.1. Conclusions	68
7.2. Línies de futur.....	69
Bibliografia.....	70

Figures i taules

Índex de figures

Figura 1: Aplicacions Web Progressives (PWA).....	16
Figura 2: Metodologia Àgil vs Cascada	18
Figura 3: Diagrama de Gantt.....	21
Figura 4: Esquema funcionament GESTCOOK.....	25
Figura 5: Esquema perfils usuari	26
Figura 6: Esquema objectius GESTCOOK.....	29
Figura 7: Esquema arquitectura GESTCOOK	31
Figura 8: Esquema funcionament SPA.....	32
Figura 9: Esquema funcionament PWA.....	32
Figura 10: Esquema <i>Firebase authentication</i>	33
Figura 11: Exemple model base de dades de <i>Firestore</i>	33
Figura 12: Exemple emmagatzematge <i>Firebase Storage</i>	34
Figura 13: Exemple emmagatzematge <i>Firebase Databse</i>	34
Figura 14: Exemple <i>Firebase Hosting</i>	34
Figura 15: <i>Exemple mètode post amb el programari Postman</i>	35
Figura 16: Exemple cerca a la BBDD de BEDCA.....	35
Figura 17: Exemple cerca a la BBDD de DIETAS.NET	36
Figura 18: Comandament Curl del mètode POST	36
Figura 19: Conversió Curl a PHP.....	37
Figura 20: Estructura de la Bases de dades de GESTCOOK	41
Figura 21: Logotip i icona GESTCOOK	42
Figura 22: Paleta colors	42
Figura 23: Exemple tipografia <i>Ubuntu</i>	42
Figura 24: Camps i elements de formulari.....	43
Figura 25: Botons i icones.....	43
Figura 26: Altres elements gràfics	43
Figura 27: Navegació	44
Figura 28: <i>Sitemap</i>	44
Figura 29: Cercador web.....	46
Figura 30: Accés directe GESTCOOK.....	46
Figura 31: Instal·lació Angular 7	47
Figura 32: Pàgina inici PWA Angular 7	48
Figura 33: Llistat de components del projecte GESTCOOK.....	49
Figura 34: Definició de les rutes del projecte GESTCOOK	50
Figura 35: Utilització del <i>routerLink</i> en la barra de navegació.....	50
Figura 36: Alta del projecte a <i>Firebase</i>	51
Figura 37: Configuració accés aplicació	51
Figura 38: Usuaris creats amb el servei d'autenticació de <i>Firebase</i>	52
Grau de Multimèdia Desenvolupament d'aplicacions interactives	12

Figura 39: Inici de sessió amb un compte <i>Gmail</i> de <i>Google</i>	52
Figura 40: Correu per restablir la contrasenya del compte de l'usuari	53
Figura 41: Apartat d'edició del missatge que apareix al correu de restabliment de la contrasenya	53
Figura 42: Creació de la BBDD	54
Figura 43: Codi per poder treballar amb <i>Firebase</i> des de l'aplicació	54
Figura 44: Codi importacions mòduls <i>Firebase</i> al projecte.....	55
Figura 45: Llistat de serveis del projecte	55
Figura 46: Generació del <i>build</i> de l'aplicació amb èxit	56
Figura 47: Selecció opció <i>Hosting</i>	56
Figura 48: Selecció carpeta que conté el <i>build</i> del projecte	57
Figura 49: Indicació de que es tracta una <i>SPA</i>	57
Figura 50: Indicació de que no es vol sobreescriure l'arxiu <i>index.html</i>	57
Figura 51: Obtenció de l'URL de l'aplicació	58
Figura 52: Pàgina d'inici de <i>Gestcook.es</i>	58
Figura 53: <i>Wireframes</i>	61
Figura 54: Prototip funcional	61
Figura 55: Configuració característiques audició.....	63
Figura 56: Audició en procediment	63
Figura 57: Resultat audició	64
Figura 58: Funcionament dels <i>service worker</i>	65
Figura 59: Resultat audició amb puntuació de 100 en <i>PWA</i>	66
Figura 60: Protecció de rutes	67

Índex de taules

Taula 1: Fites	19
Taula 2: Dates clau	20
Taula 3: Pressupost equip humà	22
Taula 4: Pressupost equipament tècnic.....	23
Taula 5: Pressupost altres recursos	23
Taula 6: Resum pressupost	23
Taula 7: Taula comparativa competència	27
Taula 8: Anàlisi DAFO.....	28

Índex de diagrames

Diagrama 1: Diagrama de casos d'ús	38
Diagrama 2: Diagrama de navegació	39
Diagrama 3: Diagrama de classes	40

1. Introducció

1.1. Introducció/Prefaci

El dia a dia de cada persona comporta l'encadenament d'un seguit de tasques que poden anar des de llevar-se i desplaçar-se al seu lloc de treball, la realització de la jornada laboral, assistir a compromisos de diversa índole, fins a la pràctica d'esports. Tots aquest aspectes varien en funció de cada persona, del dia, etc. Però una de les tasques que no té en compte a l'individu, no varia i es manté com una constant, és l'acte d'alimentar-nos. Ens podem alimentar de diverses maneres: cuinant a casa, anar a casa d'un familiar, utilitzar carmanyoles, anar de restaurant, menjar per emportar, servei a domicili...

Aquest projecte es vol centrar en aquelles persones que mengen a casa i les quals han de cuinar-se el menjar. A aquestes persones, tot i el temps que dediquen a les seves tasques quotidianes, se'ls suma el fet d'alimentar-se, atès que comporta una despesa de temps i recursos molt important. Han de pensar en un menú, escollir quins plats l'han de compondre, quina dificultat d'elaboració tenen, quins productes es necessiten, a on s'ha de realitzar la compra, etc. Com es pot comprovar, encara no ens hem portat ni un tros de menjar a la boca que hem hagut de fer un munt de tasques abans de realitzar un acte tan senzill i quotidià com és el fet d'alimentar-nos.

Però aquest fet s'agreuja en el cas d'aquelles persones que, com és el meu cas, no tenen aptituds culinàries, no disposen de suficient temps per cuinar plats elaborats o tan sols no volen desplaçar-se al supermercat i perdre's entre els lineals atestats de productes. Per a moltes persones se'ls pot fer molt feixuc gestionar totes aquestes tasques i pot arribar a presentar un problema important. És per aquest motiu que s'ha pensat en crear una aplicació que ajudi a solucionar aquest problema amb un intent d'oferir una eina senzilla que faci més fàcil tota aquesta gestió.

Aquest projecte pot resultar interessant i d'ajuda a aquelles persones que comparteixen els problemes presentats als anteriors paràgrafs. A més, una part que pot fer atractiva aquesta aplicació, és aquella que fa que l'usuari s'involucri i formi part d'un conjunt. L'usuari pot consultar les solucions culinàries de la resta de persones col·laboradores, però ell també pot compartir els seus coneixements a partir d'experiències pròpies. Per tant, un dels atractius, a part de les funcionalitats de què disposa l'aplicació, és el fet que converteix a l'usuari en un consumidor i creador de contingut, que es veu motivat per sentir-se identificat amb la resta d'usuaris. Tan la vessant social com la de la gestió i planificació, fan que GESTCOOK sigui atractiva i útil per a l'usuari.

Com a resultat final d'aquest treball es vol obtenir una aplicació web responsiva (PWA), multi dispositiu, que serveixi per cercar, crear i compartir receptes, a la vegada que ofereixi un espai amb calendari per organitzar i planificar el menú, un auto generador de llistat de productes, un apartat de receptes favorites i un servei per realitzar la compra amb servei de lliurament a domicili.

1.2. Descripció/Definició

El projecte que engloba aquest TF pretén desenvolupar d'inici a fi una aplicació web progressiva funcional. Aquesta aplicació té com objectiu principal cobrir la necessitat d'aquelles persones amb poques aptituds per a la cuina, que no disposen de temps per cuinar o que no volen invertir el temps necessari que suposa el fet de desplaçar-se fins al supermercat i realitzar la tria i compra dels productes necessaris per elaborar les diferents receptes.

Amb la solució que aquí es proposa es vol aportar un espai de col·laboració centrat en l'usuari i les seves necessitats. Amb aquest aplicació els usuaris ajuden a altres usuaris aportant solucions pròpies a problemes comuns. Per aquest motiu, aquesta eina ha de resultar una opció atractiva per aquest tipus públic objectiu, suposant una ajuda real i que faciliti les tasques relacionades en aquest àmbit.

Per oferir totes aquestes funcionalitats, aquesta PWA englobarà les següents solucions:

- **Front-end:** el desenvolupament de la part del client es portarà a terme amb el *framework* Angular, en la seva versió 6, que és de codi obert i mantingut per *Google*. Aquest *framework* de *JavaScript* (*TypeScript*)¹ s'utilitza per a crear *single page applications* o *SPA* i que són aplicacions basades en el navegador amb capacitat *MVC*². Amb Angular CLI³ és realment senzill crear una PWA. Per poder treballar amb Angular CLI en la seva versió 6 cal tenir instal·lat:
 - *NodeJS*: entorn en temps d'execució multiplataforma per a la capa del servidor, basat en llenguatge *ECMAScript*⁴ (*JavaScript*), asíncron, amb *inputs* i *outputs* de dades amb una arquitectura orientada a esdeveniments. Basat en el motor V8 de *Google*. Permet construir aplicacions en xarxa escalables i pot gestionar moltes connexions concurrents.
 - *Node Package Manager* (*NPM*): manipulador de paquets per defecte de *NodeJS*, que ofereix un entorn d'execució per a *JavaScript*.
- **Back-end:** el desenvolupament de la part del servidor es portarà a terme amb la plataforma de desenvolupament *Firebase*, en la versió del SDK 5.12.0, mantinguda per *Google*. De tots els serveis que ofereix *Firebase*, s'utilitzaran els següents:
 - *Firebase Auth*: Sistema per autenticar i administrar els usuaris de manera simple i segura.
 - *Realtime Database*: Base de dades NoSQL allotjada al núvol que emmagatzema i sincronitza dades en format JSON⁵ entre usuaris i dispositius en temps real.
 - *Firebase Cloud Storage*: Emmagatzematge d'objectes que permet emmagatzemar i compartir imatges, àudio, vídeo i altres tipus de continguts generat pels usuaris.
- **User interface:** el desenvolupament de la interfície d'usuari es portarà a terme amb el *framework* *Bootstrap*, en la seva versió 4.13, que és de codi obert i mantingut per *Twitter*. Es basa en una llibreria que conté plantilles per ajudar en el disseny del *front-end*. Permet dissenyar aplicacions responsives basades en la filosofia *mobile-first*.

¹ Llenguatge de programació de codi obert mantingut per *Microsoft*. És un superconjunt de *JavaScript* que afegeix tipatge estàtic i objectes basats en classes.

² Model-vista-controlador és un patró d'arquitectura de software que separa el codi en tres capes diferents.

³ Intèrpret de línia de comandaments.

⁴ Especificació d'estandardització de *JavaScript*.

⁵ *Format de text lleuger utilitzar en l'intercanvi de dades*.

Algunes característiques que fan interessants les PWA respecte a les aplicacions natives són:

- Ofereixen un major rendiment i la càrrega de contingut és quasi instantani.
- Ofereix major seguretat atès que operen sota el protocol segur HTTPS.
- Possibilitat de treballar sense connexió i enviament de notifikacions als usuaris.
- Utilització menor de recursos (emmagatzematge, RAM, actualitzacions...).
- Són responsives i es pot accedir directament des de l'enllaç. També es poden instal·lar.
- Requereix menys costos de treball de programació, desenvolupament i manteniment.



Figura 1: Aplicacions Web Progressives (PWA)

<http://halwits.com/progressive-web-app/>

1.3. Objectius generals

A continuació es detallen els objectius que defineixen el propòsit d'aquest treball, ordenats segons la seva rellevància.

1.3.1. Objectius principals

Objectius de l'aplicació/producte/servei:

- Crear un aplicació web progressiva moderna, lleugera i intuïtiva que utilitzi les tècniques més avançades.
- Crear un servei pensat, basat i centrat en l'usuari, amb la finalitat d'oferir un espai de col·laboració del qual pugui formar part, a on pugui compartir els seus coneixements i beneficiar-se de les aportacions de la resta d'usuaris amb els qui comparteix inquietuds i/o afinitats.

Objectius per al client/usuari:

- Disposar d'una aplicació web senzilla, intuïtiva i ràpida que li permeti crear i compartir receptes.
- Facilitar el dia a dia en la gestió de tots els aspectes relacionats amb la cuina, a través de receptes senzilles i ràpides de cuinar, un calendari per organitzar els seus menús i un generador de la llista de la compra.

Objectius personals de l'autor del TF:

- Investigar i conèixer noves tècniques, a la vegada que s'observa cap a on estan evolucionant les aplicacions web.
- Aprendre i aplicar nous mètodes de programació i maquetació per a la realització d'aplicacions web modernes.

1.3.2. Objectius secundaris

Objectius addicionals que enriqueixen el TF:

- Investigar i triar les característiques tècniques més adients per a la realització d'aquest projecte, tot escollint el llenguatge de programació, el motor de BBDD, el servidor, les *APIs*⁶ així com els *frameworks*, entre d'altres.
- Documentar pas a pas tot el procés d'investigació, aprenentatge i posada en pràctica de tots els coneixements apresos durant l'elaboració d'aquest del projecte.

⁶ Interfície de programació d'aplicacions que ofereix biblioteques per a ser utilitzada per altre software como una capa d'abstracció.

1.4. Metodologia i procés de treball

Tot i que qualsevol usuari pot trobar aplicacions molt similar quant a funcions, aquest projecte es basa en el desenvolupament d'una aplicació web progressiva pròpia, que té com a objectiu principal la obtenció d'un producte totalment nou, adaptat a les especificacions i l'abast del projecte. A part del producte en si, la finalitat d'aquest TF és poder realitzar un projecte real a on poder aplicar els coneixements adquirits durant el desenvolupament d'aquest Grau de Multimèdia, però també serveix per anar un pas més enllà, cercant i aprenent noves tècniques.

Per triar la solució tècnica del projecte s'ha investigat sobre les diferents eines que ofereix *Google*, que té una visió molt clara de cap a on està evolucionant el web. Per aquest motiu, ofereix als desenvolupadors entorns de desenvolupaments com *Firebase* que disposa de diverses *APIs* per treballar el *back-end* de les aplicacions. *Angular* és una altra eina que ofereix *Google* i és una basa en *framework* que permet desenvolupar la part del *front-end*. S'ha escollit *Angular* atès que permet crear *PWA* de manera ràpida i fàcil.

Per a la realització d'aquest projecte s'ha optat per una metodologia àgil que es basa en una planificació incremental i iterativa, defugint del mètode tradicional basat en una planificació més seqüencial o també conegut amb el nom de metodologia en cascada. A efectes pràctics i degut a que aquest projecte es desenvolupa de forma individual, el mètode utilitzat es pot entendre com un compendi dels dos mètodes descrits anteriorment, atès que es seguirà un procés lineal però proporcionant un entregues de valor a cada fita, presentant parts funcional del projecte derivades dels *sprints*⁷ seguint el model *SCRUM*⁸.

La metodologia àgil permet abordar molt millor els riscos que es puguin derivar durant el desenvolupament del projecte atès que es treballa per objectius. El projecte es troba en constant retroalimentació, corregint les desviacions, identificant nous problemes i aplicant les millors pràctiques per obtenir els millors resultats. Està clar que la incertesa forma part del procés de desenvolupament i aquest sistema de gestió permet afrontar-la adaptant-se a les diferents situacions.

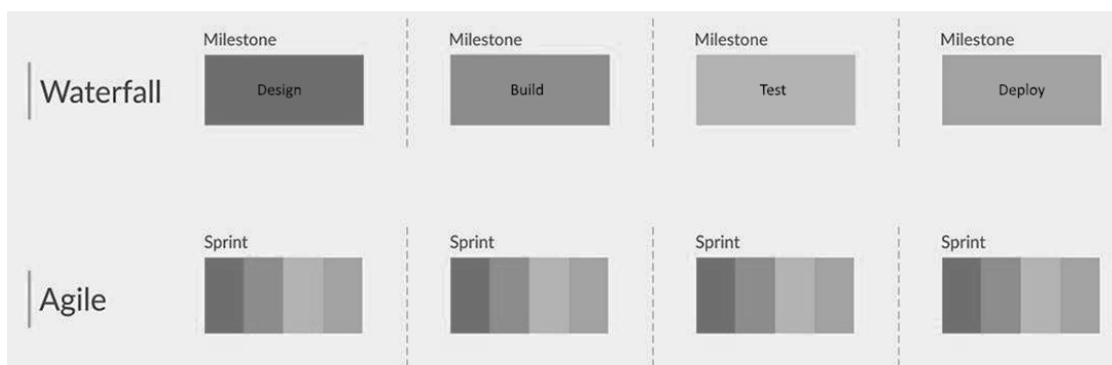


Figura 2: Metodologia Àgil vs Cascada
<https://www.mobifilia.com/devops-future-software-development/>

⁷ Cicle de desenvolupament que compren l'anàlisi de requeriments, desenvolupament, proves i documentació.

⁸ Marc de desenvolupament àgil.

1.5. Planificació

El projecte es desenvoluparà durant el 1º semestre del curs 2018 – 2019, amb data d'inici el dia 19 de setembre de 2018 i data de finalització el dia 07 de gener de 2019.

Atès que l'objectiu d'aquest projecte és el de presentar una aplicació funcional i documentar tot el seu procés d'investigació i desenvolupament, a la planificació es té en compte les diferents fases de documentació a través de les fites (PAC1, PAC 2, PAC 3 i Lliurament final), així com les diferents fases del projecte (disseny, producció i difusió).

Tal i com es pot observar al diagrama de Gantt, tant el procés de documentació com el de la seva revisió es mantindran constants durant el desenvolupament de cada fase. Les correccions de les PACs i el fet d'anar avançant durant els diferents estadis del projecte, faran que aquesta documentació estigui en constant actualització.

1.5.1. Fites

CONCEPTE	DATA	OBJECTIUS
PAC 1	02/10/2018	Proposta del projecte Memòria del projecte (punt 1)
PAC 2	31/10/2018	Memòria del projecte (punts 2, 3 i 4) Esquemes de navegació Diagrames UML Prototips projecte
PAC 3	09/12/2018	Memòria del projecte (punts 5 i 6) Vídeo funcionament versió beta (5 minuts) Versió beta projecte
LLIURAMENT FINAL	07/01/2019	Memòria del projecte (punt 7) Versió final projecte Presentació format lliure Presentació vídeo (15 minuts)

Taula 1: Fites

1.5.2. Dates clau

TASCA	DATA INICI	DATA FI	DURADA
GESTIÓ PROJECTE	19-09-2018	07-01-2019	111
PAC 1 (DEFINICIÓ FORMAL)	19-09-2018	02-10-2018	14
Proposta projecte	19-09-2018	22-09-2018	4
Memòria (1)	23-09-2018	02-10-2018	10
Lliurament PAC 1	02-10-2018	02-10-2018	-
PAC 2 (DESENVOLUPAMENT)	03-10-2018	31-10-2018	29
Revisió memòria	03-10-2018	31-10-2018	29
Memòria (2, 3, 4)	03-10-2018	31-10-2018	29
PWA - Investigació	03-10-2018	07-10-2018	5
PWA - Arquitectura aplicació	08-10-2018	12-10-2018	5
PWA - Estructura aplicació	13-10-2018	18-10-2018	6
PWA – Disseny i prototips	19-10-2018	30-10-2018	12
Instal·lació entorn desenvolupament	31-10-2018	31-10-2018	1
Lliurament PAC 2	31-10-2018	31-10-2018	-
PAC 3 (VERSIÓ BETA)	01-11-2018	09-12-2018	39
Revisió memòria	01-11-2018	09-12-2018	39
Memòria (5, 6)	01-11-2018	09-12-2018	39
PWA – Pàgina inici	01-11-2018	07-11-2018	7
PWA – Autenticació usuaris	08-11-2018	14-11-2018	7
PWA – CRUD receptes	15-11-2018	21-12-2018	7
PWA – Receptes favorites	22-12-2018	28-12-2018	7
PWA – Cercador i filtres	29-12-2018	05-12-2018	7
Realització presentació versió beta	06-12-2018	09-12-2018	4
Lliurament PAC 3	09-12-2018	09-12-2018	-
LLIURAMENT FINAL	10-12-2018	07-01-2019	29
Revisió memòria	10-12-2018	07-01-2019	29
Memòria (7)	10-12-2018	07-01-2019	29
PWA – Planificador menú	10-12-2018	14-12-2018	5
PWA – Llistat productes	15-12-2018	19-12-2018	5
PWA – Integració plataforma compra	20-12-2018	24-12-2018	5
PWA – Integració al servidor i llançament	25-12-2018	25-12-2018	1
Realització presentació format lliure	26-12-2018	30-12-2108	5
Realització vídeo presentació	31-12-2018	07-01-2019	8
Lliurament final	07-01-2109	07-01-2019	-

Taula 2: Dates clau

1.5.3. Diagrama Gantt

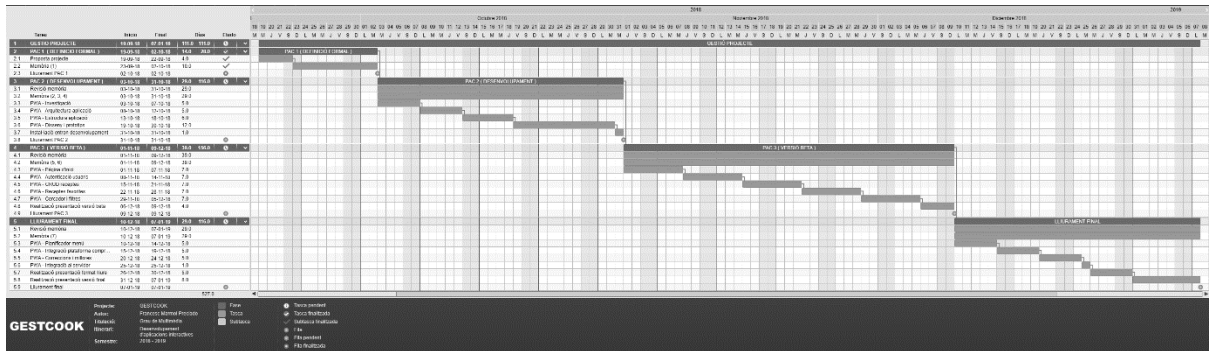


Figura 3: Diagrama de Gantt
<https://www.tomsplanner.es/public/fmarmolp1>

*Nota: fer clic a sobre de la imatge per ampliar.

1.6. Pressupost

Atès que es tracta d'un projecte personal sense cap intervenció externa. A continuació es detalla el cost estimat que ha suposat la realització d'aquest projecte en base de les hores dedicades:

1.6.1. Equip humà

TASCA	HORES		PREU / H	TOTAL
	Parcials	Totals		
GESTIÓ PROJECTE		125	25,00 €	3.125,00 €
PAC 1		14	25,00 €	350,00 €
Proposta projecte	14			
Memòria (1)				
PAC 2		24	25,00 €	600,00 €
Feedback i revisió memòria (1)	24			
Memòria (2, 3, 4)				
PAC 3		32	25,00 €	800,00 €
Feedback i revisió memòria (2, 3, 4)	16			
Memòria (5, 6)				
Realització presentació beta	16			
Lliurament final		55	25,00 €	1.375,00 €
Feedback i revisió memòria (5, 6)	12			
Memòria (7)				
Realització presentació format lliure	18			
Realització vídeo presentació	25			
DESENVOLUPAMENT PROJECTE		320	25,00 €	8.000,00 €
PWA - Investigació		20		
PWA - Arquitectura aplicació		20		
PWA - Estructura aplicació		24		
PWA – Disseny i prototips		48		
Instal·lació entorn desenvolupament		4		
PWA – Pàgina inici		28		
PWA – Autenticació usuaris		28		
PWA – CRUD receptes		28		
PWA – Receptes favorites		28		
PWA – Cercador i filtres		28		
PWA – Planificador menú		20		
PWA – Llistat productes		20		
PWA – Integració plataforma compra		20		
PWA – Integració al servidor i llançament		4		

Taula 3: Pressupost equip humà

1.6.2. Equipament tècnic

DISPOSITIU	UNITATS	PREU / Ut	TOTAL
EQUIP DE TREBALL			230,00 €
<i>Descktop</i>	0,1	1.800,00 €	180,00 €
<i>Tablet</i>	0,1	300,00 €	30,00 €
<i>Smartphone</i>	0,1	200,00 €	20,00 €

Taula 4: Pressupost equipament tècnic

1.6.3. Altres recursos

RECURS	UNITATS	PREU / Ut	TOTAL
DOMINI			0,00 € / any
<i>gestcook.firebaseio.com</i>	1	0,00 € / any	
HOSTING			0,00 € / any
<i>Firebase (Plan Spark)</i>	1	0,00 € / any	
LLICÈNCIES			0,00 € / any
Entorn desenvolupament	1	0,00 € / any	

Taula 5: Pressupost altres recursos

1.6.4. Resum pressupost

CONCEPTE	UNITATS	PREU / Ut	TOTAL
TOTAL PROJECTE			11.355,00 €
Gestió projecte	1	3.125,00€	
Desenvolupament projecte	1	8.000,00 €	
Equip de treball	1	230,00 €	
Domini	1	0,00 €	
<i>Hosting</i>	1	0,00 €	
Llicències	1	0,00 €	

Taula 6: Resum pressupost

1.7. Estructura de la resta del document

Durant els següents capítols de la memòria es defineixen els aspectes que s'han tingut en compte a l'hora de plantejar el projecte:

- A l'apartat dos de la memòria es mostra un resum de l'estat actual del mercat a on està destinada l'aplicació. S'ha procedit a definir el públic objectiu al qual pot anar destinada l'aplicació. S'ha estudiat a la competència per veure com aborden el problema que es planteja i les solucions que aporten. I finalment s'ha realitzat un anàlisi per estudiar la viabilitat del projecte a través de la definició del seus punts febles i punts forts.
- A l'apartat tres de la memòria es defineixen els objectius i les especificacions desitjables que haurà de complir l'aplicació un cop estigui acabada. Altrament, s'ha plantejat un model econòmic dintre del supòsit en què es decidís portar a la realitat el projecte i planejar el seu llançament, tot definint una estratègia de màrqueting.
- A l'apartat quatre de la memòria es tracta tot el que té a veure amb el disseny de l'aplicació. Per un costat es planteja l'arquitectura general de l'aplicació, i per un altre es planteja l'aspecte visual.
- A l'apartat cinc de la memòria es detallen el requisits d'implementació necessaris perquè l'aplicació pugui funcionar i les instruccions d'ús. Altrament, es descriuen els passos necessaris per instal·lar tot l'entorn de producció així com els de desenvolupament de l'aplicació.
- A l'apartat sis de la memòria es presenten els prototips de baixa definició en forma de *wireframes* de les pantalles més rellevants de l'aplicació. Altrament, es pot accedir a un simulador que mostra els prototips en alta definició i a on es pot comprovar la seva interacció a través dels seus elements (icones, botons, navegació...).
- Finalment, a l'apartat setè de la memòria es realitza una reflexió crítica sobre l'assoliment dels objectius plantejats. També es plantegen possibles millores i/o implementacions de cares a futur.

2. Anàlisi de mercat

Actualment a internet es poden trobar múltiples eines relacionades amb el món de la cuina. Aquestes es poden dividir en:

- No instal·lables, com poden ser les aplicacions web.
- Instal·lables, que són creades específicament per a dispositius *Android* i/o *IOS*.

La gran majoria d'aquestes aplicacions es basen en receptaris en línia que ofereixen múltiples funcionalitats. Les més comuns són:

- Cerca de receptes mitjançant múltiples filtres.
- Receptes explicades pas a pas que inclouen imatges i/o vídeos.
- Guardar i organitzar les receptes favorites.
- Planificador per organitzar el menú.
- Generador de llistes de productes arrel de les receptes escollides.
- Recomanacions (segons tipus dieta, més votades, novetats...).

Aquest projecte vol esdevenir una plataforma d'ajuda a usuaris amb unes necessitats específiques:

- Sense aptituds per a la cuina.
- No saben planificar un menú.
- No tenen temps per preparar plats elaborats.
- No volen desplaçar-se al supermercat per comprar productes.

GESTCOOK ha de ser una eina senzilla i intuïtiva que serveixi per simplificar la vida dels seu usuaris.

Per tant, alguns dels avantatges que ha d'oferir són:

- Poder realitzar la compra per a tota la setmana en un sol dia.
- Conèixer amb antelació que hi haurà per menjar.
- Minimitzar el temps dedicat a l'elaboració de plats.
- Facilitar i agilitzar al màxim l'elaboració de plats.
- Aprofitar al màxim tots els ingredients i reduir la despesa.

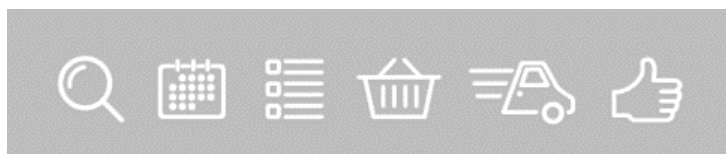


Figura 4: Esquema funcionament GESTCOOK

2.1. Públic objectiu (i.e. *target audience*) i perfils d'usuari

2.1.1. Públic objectiu

L'aplicació GESTCOOK està enfocada a un sector de la població que està habituada a realitzar compres per internet de forma habitual, utilitza dispositius mòbils de forma regular en decurs del seu dia a dia, fa ús de diverses aplicacions, són persones actives i valoren el temps lliure. Alguns perfils que poden englobar aquest tipus de *target audience* poden ser persones de divers perfil i amb edats compreses entre els 18 i els 50 anys:

- **Estudiants:** que pel fet d'assistir a les classes i continuar amb els estudis o els deures a casa, no disposen de temps per cuinar o desplaçar-se al supermercat.
- **Esportistes/as:** que entrenen llargues jornades i no disposen de temps per cuinar o desplaçar-se al supermercat o que necessiten planificar la seva dieta de forma senzilla.
- **Treballadors/es:** per compte propi o aliè, que després de realitzar la jornada han de continuar complint amb obligacions de diversa índole, fet que provoca que no disposin de temps suficient per cuinar, planificar un menú o desplaçar-se al supermercat per comprar productes.
- **Pares i mares:** que es troben en actiu o no i que pel fet d'atendre les obligacions i tot el que comporta gestionar una família fa que no disposin de temps per cuinar, desplaçar-se al supermercat i que necessiten que els portin els productes a casa.
- **Altres:** persones que no tenen aptituds culinàries, no tenen suficient temps per cuinar plats elaborats, no volen pensar en planificar un menú o no volen desplaçar-se fins al supermercat per a realitzar la compra dels productes.

2.1.2. Perfils d'usuari

Els perfils d'usuaris que poden interactuar amb l'aplicació GESTCOOK són bàsicament de tres tipus:

- **Usuari no registrar:** no pot navegar per l'aplicació. Per accedir al contingut s'ha de donar d'alta i/o registrar-se.
- **Usuari registrat:** pot navegar per tota l'aplicació sense restriccions. Pot veure, cercar, crear i valorar receptes. Pot afegir receptes a la llista de favorits. Pot planificar el menú a través del calendari. Pot generar una llista de productes i realitzar la seva compra en línia.
- **Administrador:** disposa d'accés total i sense restriccions sobre el contingut de l'aplicació. Pot crear, eliminar i editar les receptes.



Figura 5: Esquema perfils usuari

2.2. Competència

Alguns dels casos de competència més rellevant són:

- **Planificador de Nestlé:** planificador molt elaborat que genera automàticament una dieta setmanal en funció de les característiques de l'usuari. Informa del valor nutricional de les receptes. Permet descarregar el menú en *pdf* i comprar els productes d'una recepta cada cop.
- **Planificador de Consum:** planificador senzill i intuïtiu que genera automàticament una dieta setmanal en funció dels filtres escollits. Permet descarregar en format *pdf* la llista de productes però es pot realitzar la compra en línia.
- **Planificador de Gallina Blanca:** planificador poc intuïtiu que permet crear un plat en funció dels ingredients que tenim. Bàsicament és un receptari a l'ús. No permet descarregar llistats i només es poden comprar productes de la pròpia marca en línia.
- **Planificador de Eroski:** planificador poc intuïtiu que mostra diferents planificacions per a una mateixa setmana en funció del tipus de menú triat. El menú es tancat i no es poden modificar els plats. No es genera llistat de productes ni la seva compra en línia.
- **Planificador tudespensa.com:** planificador poc elaborat en format PWA. No disposa de planificador. Permet cercar receptes i afegir els seus productes directament a la cistella de compra en línia.

TAULA COMPARATIVA						
CASOS D'ESTUDI	CARACTERÍSTIQUES					
	Varietat de receptes	Múltiples filtres de cerca	Planificador setmana vista	Generador automàtic de llistat	Plataforma de compra en línia	Aplicació responsiva
Nestlé	X	X	O	O	O	O
Consum	O	O	O	O	X	O
Gallina Blanca	O	X	X	X	O	O
Eroski	O	O	O	X	X	O
Tudespensa	X	O	X	X	O	O
GESTCOOK	O	O	O	O	O	O

Taula 7: Taula comparativa competència

En conclusió, es poden trobar diversitat d'ofertes i variants. Algunes d'elles no deixen de ser receptaris en línia amb filtres de cerca i d'altres ofereixen eines molt complertes com és el cas del planificador de Nestlé. Altrament, s'ha pogut comprovar que algunes aplicacions que són fortes en alguns aspectes no ofereixen totes les funcionalitats desitjables i és en aquest punt on vol tenir cabuda el projecte GESTCOOK.

Tal i com demostren les aplicacions web analitzades, aquestes són receptaris tancats a on la finalitat de la majoria és el fet d'oferir receptes basades en els productes de la seva marca. Per aquest motiu, GEESTCOOK es vol desmarcar mitjançant la col·laboració social i l'interès comú, a la vegada que ofereix les eines necessàries per fer la vida més fàcil als seus usuaris.

2.3. Anàlisi DAFO

	ANÀLISIS INTERNA	ANÀLISIS EXTERNA
FACTORS NEGATIUS	<p>Manca de participació</p> <p>Dependència d'un dispositiu</p> <p>Desconeixement de les PWA</p> <p>Dependència dels serveis de tercers</p> <p>DEBILITATS</p>	<p>Seguretat i pirateria</p> <p>Fallida de serveis de tercers</p> <p>Ús inadequat de la plataforma</p> <p>Poca acceptació per part dels usuaris</p> <p>AMENACES</p>
FACTORS POSITIUS	<p>Requeriments de maquinari mínims</p> <p>Multi dispositiu i multi plataforma</p> <p>Planificació àgil i senzilla</p> <p>Contingut específic</p> <p>FORTALESES</p>	<p>Ús de les noves tecnologies</p> <p>Fomentar la participació</p> <p>Compartir coneixement</p> <p>Generació d'ingressos</p> <p>OPORTUNITATS</p>

Taula 8: Anàlisi DAFO

3. Proposta

Analitzat el mercat a través de diversos exemples de possibles casos de competència, s'han pogut determinar aspectes comuns i debilitats que presenten cadascun d'ells. GESTCOOK es diferencia de la competència gràcies al fet de resultar una eina col·laborativa, que no només permet a l'usuari consumir contingut, sinó que també li permet la possibilitat de crear-lo a través de la seva pròpia experiència i compartir-lo amb la resta de la comunitat que comparteix les mateixes inquietuds.

Tal i com s'ha pogut observar, els casos d'estudi de competència són webs que pretenen cobrir certes necessitats dels usuaris que les visiten. Poden anar des de cercar receptes, planificar un menú o fins i tot la compra dels productes d'una recepta. Però la realitat és que cap d'elles cobreixen tots els aspectes. És en aquest buit on es vol encabir GESTCOOK a través d'una xarxa social per compartir receptes fàcils i senzilles entre usuaris amb poques capacitats per a la cuina. Però que els ajudi en totes aquelles tasques relacionada amb la planificació del menú i la compra dels productes.



Figura 6: Esquema objectius GESTCOOK

3.1. Definició d'objectius/especificacions del producte

L'aplicació web progressiva GESTCOOK té com a objectius oferir una xarxa social per a crear i compartir receptes senzilles i ràpides a través de les experiències dels propis usuaris. També pretén facilitar totes aquelles tasques que corresponen amb la gestió dels plats com poden ser: l'elaboració d'un menú, el llistat de productes o la seva compra.

Algunes de les opcions més rellevants que ofereix GESTCOOKE són:

- És un servei multi plataforma.
- És una aplicació multi dispositiu.
- Permet registre d'usuaris.
- Permet crear, editar i compartir receptes.
- Permet pujar fotografies i vídeos.
- Permet afegir text i *tags*⁹.
- Permet realitzar cerques per filtres i/o *tags*.
- Permet generar llista de receptes favorites.
- Permet planificar un menú setmanal.
- Permet generar llista de productes de forma automàtica segons el menú.
- Permet realitzar la compra online dels productes de la llista.

⁹ Anglisme utilitzat per a definir una etiqueta que actua com a paraula clau assignada a una dada emmagatzemada en un repositori i que facilita la seva recuperació.

3.2. Model de negoci

Al tractar-se d'una xarxa social, el model de negoci és centra en l'usuari i en oferir-li una eina atractiva que li resulti d'utilitat. Per tant, GESTCOOK ha de cobrir les necessitats del seu públic objectiu. Per altre costat, com a possible font d'ingressos s'ha pensat en la possibilitat d'incorporar un sistema d'afiliació. Aquests sistemes es basen en oferir una comissió per les compres de productes de tercers que els usuaris realitzen a través de l'aplicació web.

3.3. Estratègia de màrqueting

Com a estratègia de màrqueting per a reforçar el *branding* de GESTCOOK es considera cabdal que la comunitat d'usuaris estigui interessada en continuar participant, atès que la seves aportacions enriqueixen el contingut que ofereix l'aplicació als seus usuaris. Per aquest motiu, una part important de l'estratègia ha d'estar enfocada en el propi usuari a través de diverses accions que promoguin i premiïn la seva participació. Un exemple podria ser la incorporació d'un espai per a ofertes de productes de tercers amb la finalitat de que l'usuari es pugui beneficiar de descomptes i altres ofertes.

Atès que el *branding* fa referència al procés de fer i construir una marca amb la finalitat d'oferir una experiència determinada, no es pot deixar de banda l'essència de GESTCOOK que no és més que el contingut que ofereix als seus usuaris. Per aquest motiu, una part important recaurà en la feina interna i que es passa per oferir un contingut acurat, clar i de qualitat. Per aconseguir aquest objectius caldria reforçar la plantilla d'administradors/editors de contingut per vetllar que les receptes creades pels usuaris s'adeqüen a un format concret i amb uns mínims de qualitat.

Per finalitzar, una part molt important és el posicionament i la conversió de l'aplicació web, això significa que les accions hauran d'estar dirigides a treballar l'avantatge competitiu que fa a GESTCOOK diferent dels seus competidors. Algunes accions digitals més habituals poden ser el SEO¹⁰ o les xarxes socials. La publicació d'una *landing page*¹¹ a diferents xarxes socials que mostri les característiques principals de l'aplicació pot ser un clar exemple de captació de *leads*¹², que pot derivar en un augment en la conversió d'usuaris.

Una altra opció o possible estratègia seria la de treballar conjuntament amb alguns prescriptors del ram de la cuina, com per exemple: cuiners famosos, blocaires, youtubers, instagramers, etc.

¹⁰ Posicionament en buscadors optimització de motors de cerca per millorar la visibilitat d'un lloc web.

¹¹ Pàgina web d'aterratge a una pàgina web a la qual s'arriba a través d'un enllaç, botó, *banner*...

¹² Usuaris que entreguen les seves dades i passen a ser un registre de la base de dades del sol·licitant.

4. Disseny

4.1. Arquitectura general de l'aplicació/sistema/servei

L'aplicació GESTCOOK consisteix en una aplicació web progressiva (*PWA*). Aquesta aplicació web es basa en una *single page application* (*SPA*) a la qual se li han afegit diverses funcionalitats a la part del client (*front-end*) per millorar l'experiència d'usuari. La part del servidor (*back-end*) es basa en una plataforma d'emmagatzematge i sincronització de dades al núvol per aplicacions mòbils. Aquesta plataforma de desenvolupament ofereix diversitat de serveis sota un mateix entorn.

La part del client ha estat desenvolupat amb el *framework* Angular de *Google*, mentre que a la part del servidor s'ha utilitzat la plataforma *Firebase*. La unió d'aquestes dues tecnologies proporciona un conjunt de serveis *RESTFUL*¹³ que es comunica mitjançant el format de text lleuger *JSON* per a l'intercanvi de dades.

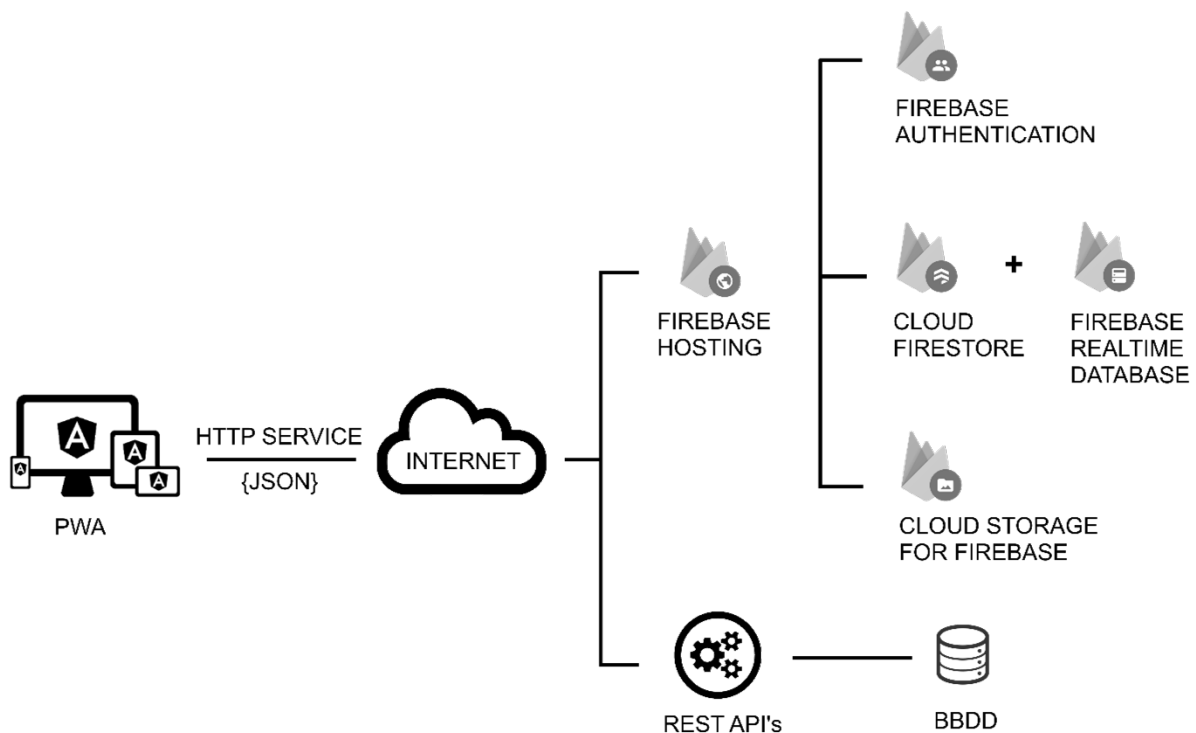


Figura 7: Esquema arquitectura GESTCOOK

¹³ Servei web que implementa la arquitectura REST.

4.1.1. Client (*front-end*):

La part visible per a l'usuari ha estat desenvolupada amb el *framework* Angular, de codi obert i propietat de Google. Aquest *framework* per desenvolupar aplicacions web es basa en HTML, CSS i en *TypeScript*, que és un superconjunt del llenguatge de programació *JavaScript*. S'utilitza per a crear i mantenir aplicacions web d'una sola pàgina o *simple page applications* (SPA), basades en navegador amb capacitat de Model Vista Controlador (MCV).

Una SPA és una aplicació web que carrega tot el contingut d'aquesta a la mateixa pàgina. Això significa que la pàgina no refresca cada cop que cliquem un enllaça dintre de la pròpia aplicació. L'aplicació allibera al servidor d'una part del treball, fet que redueix la quantitat de crides i millora la percepció de velocitat de l'usuari.

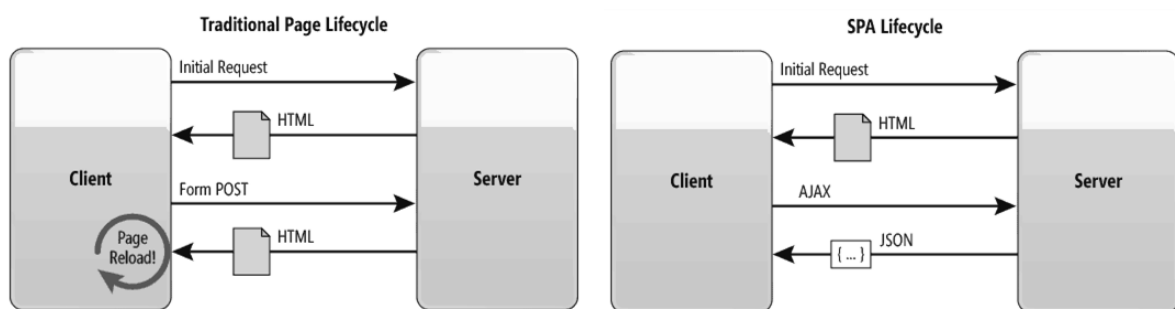


Figura 8: Esquema funcionament SPA
<https://moz.com/blog/optimizing-angularjs-single-page-applications-googlebot-crawlers>

Angular permet crear una *PWA* de manera ràpida i senzilla gràcies als *Service Workers* que afegeixen funcionalitats clau perquè l'aplicació web es comporti com una aplicació nativa: servei *offline*, actualització en segon pla, emmagatzematge de *caché*, Web Push...

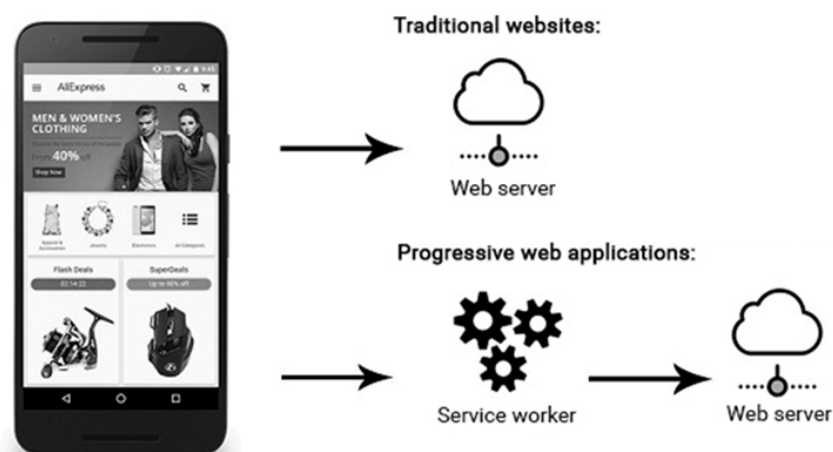


Figura 9: Esquema funcionament PWA
<http://www.torresburriel.com/weblog/2017/01/25/disenando-aplicaciones-web-progresivas-pwa/>

4.1.2. Servidor (*back-end*):

La part del servidor ha estat desenvolupada amb la plataforma de desenvolupament mòbil *Firebase* propietat de *Google*. Aquesta plataforma proveeix d'un *back-end* al núvol que disposa de diverses funcionalitats que es poden combinar segons les necessitats de l'aplicació. Pel desenvolupament de l'aplicació web GESTCOOK s'han utilitzat els següents serveis:

- **Firebase authentication:** s'utilitza per facilitar la creació del sistema d'autenticació segura a l'aplicació. Proporciona una solució d'identitat de extrem a extrem compatible amb comptes de correu electrònic i contrasenya, autenticació telefònica, *Google*, *Twitter*, *Facebook*...

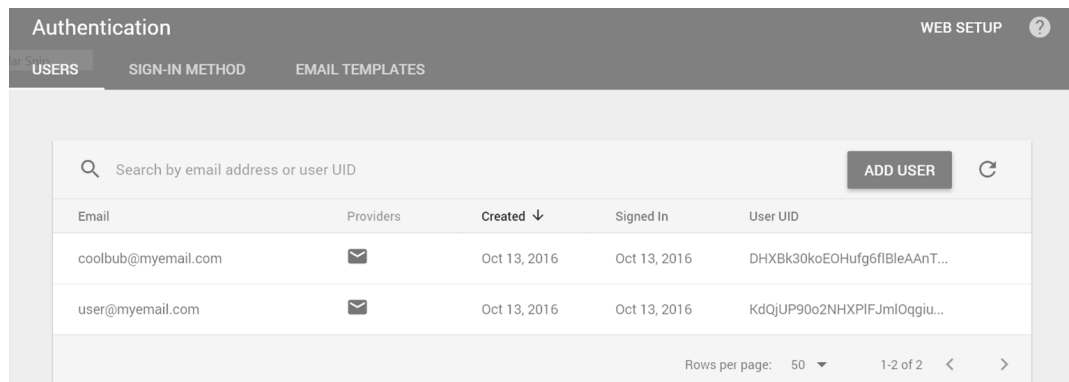


Figura 10: Esquema *Firebase authentication*

<https://stackoverflow.com/questions/40012816/firebase-auth-with-email-and-password-check-user-already-registered>

- **Cloud Firestore:** és una base de dades de documents *NoSQL* que permet emmagatzemar, sincronitzar i consultar fàcilment dades a escala global. És flexible, escalable i permet mantenir les dades sincronitzades entre els usuaris oferint assistència sense connexió. A partir del model de dades *NoSQL* emmagatzema les dades en documents i aquests documents s'emmagatzemen en col·leccions, que són contenidors dels documents de l'aplicació i que es poden utilitzar per organitzar les dades i compilar consultes.

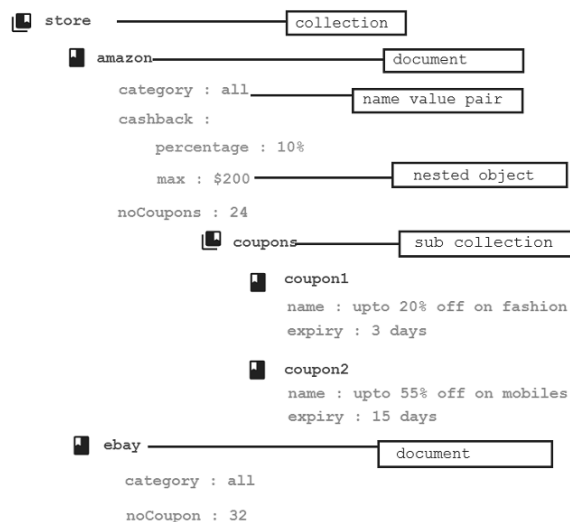


Figura 11: Exemple model base de dades de *Firestore*

<http://www.torresburriel.com/weblog/2017/01/25/disenando-aplicaciones-web-progresivas-pwa/>

- **Cloud Storage for Firebase:** és un servei d'emmagatzematge d'objectes i està destinat a cobrir les necessitat d'emmagatzematge i publicació de contingut generat pels usuaris: fotos, vídeos i/o àudio.

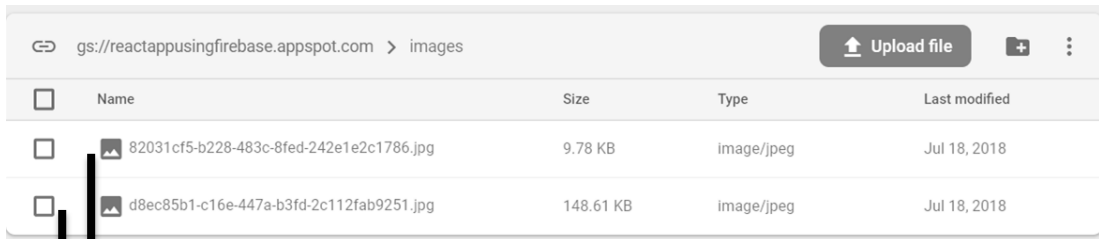


Figura 12: Exemple emmagatzematge *Firebase Storage*

<https://c1ctech.com/reactnative-working-with-firebase-using-friendllychatapp-example/>

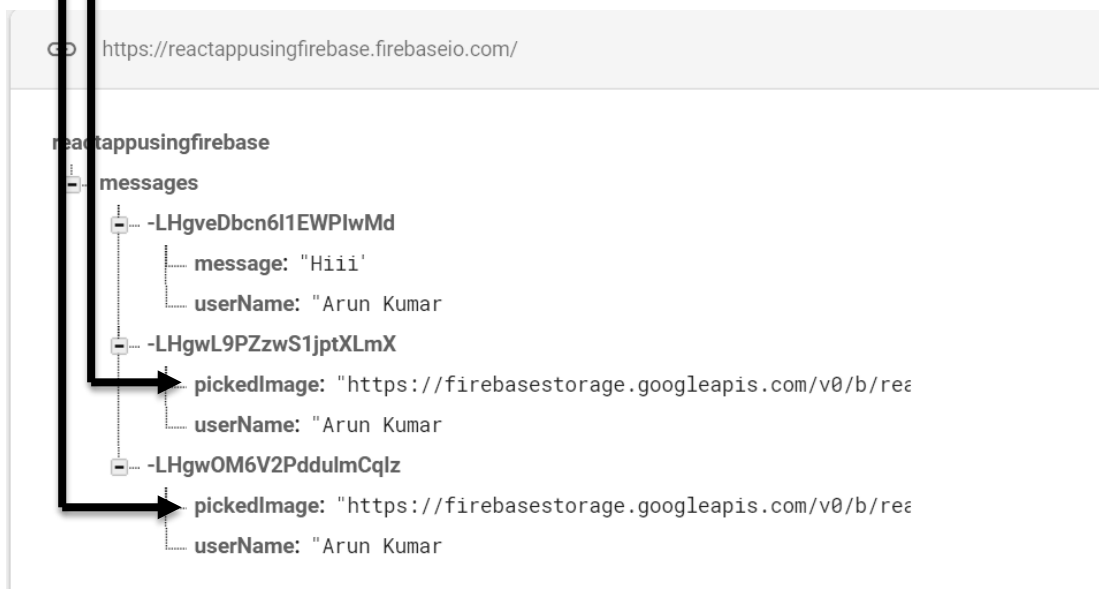
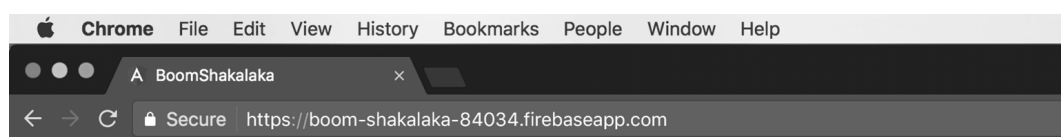


Figura 13: Exemple emmagatzematge *Firebase Database*

<https://c1ctech.com/reactnative-working-with-firebase-using-friendllychatapp-example/>

- **Firebase Hosting:** és un servei dissenyat especialment per a implementar aplicacions web d'una pàgina (*SPA*), una pàgina de destí (*landing page*) o una app web progressiva (*PWA*). Proporciona un *hosting* estàtic, ràpid i segur sota una xarxa de distribució de contingut (*CDN*) global que emmagatzema els arxius en *cache* SSD. Inclou certificat SSL per oferir el web de forma segura, de forma automàtica i sense necessitat de configuració.



app works!

Figura 14: Exemple *Firebase Hosting*

<https://houssein.me/continuous-integration-angular-firebase-travisci>

4.1.3. APIs:

- **BEDCA:** base de dades espanyola de composició d'aliments que permet cercar per aliment o categoria d'aliments. Ofereix informació sobre nutrients, calories... Aquesta BBDD no disposa de cap *API REST*. Per tant, s'ha procedit a realitzar tasques d'enginyeria inversa per observar com es realitzen les crides al servidor. Gràcies al programari en línia *Postman* s'ha pogut comprovar el codi realitzant consultes i comparant-les amb el web.

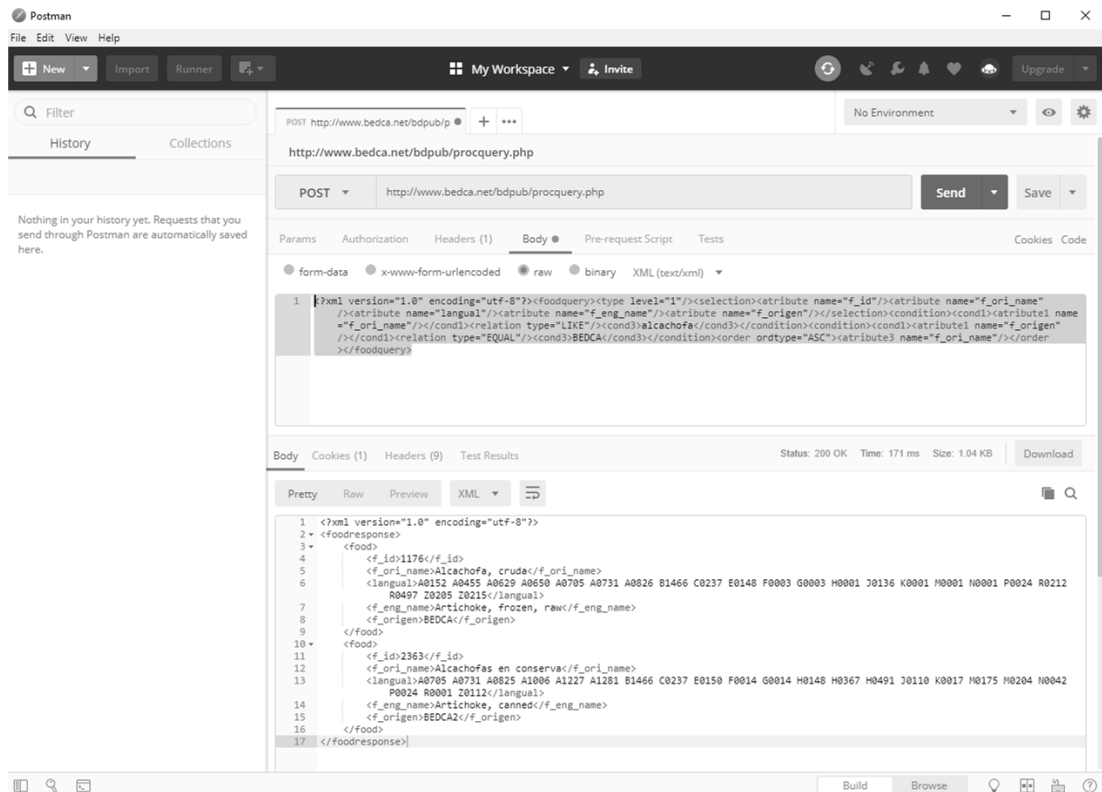


Figura 15: Exemple mètode post amb el programari Postman



Figura 16: Exemple cerca a la BBDD de BEDCA

<http://bedca.net/bdpub/>

- **DIETAS.NET**: pàgina web de planificació de dietes que ofereix un cercador que ofereix una BBDD extensa d'aliments. A l'igual que BEDCA, també permet recopilar informació sobre nutrients, calories... Aquesta BBDD tampoc disposa de cap *API REST*. Per tant, s'ha procedit a realitzar tasques d'enginyeria inversa per observar com es realitzen les crides al servidor. Amb el programari en línia *Codepunker* s'ha pogut trobar els paràmetres que utilitza per a realitzar els mètodes *GET* i *POST*.

The screenshot shows the Codepunker web interface. At the top, there's a navigation bar with 'CODE' and various menu items. A privacy notice banner is present. The main content area is titled 'Simple HTTP POST And GET Requests'. Below this, there's a form for sending requests. The URL is 'http://www.dietas.net/inc/ajax/alim/buscar.php'. The method is set to 'POST'. A parameter 'search' is set to 'maiz'. There's a CAPTCHA and a 'SEND REQUEST' button. The response is shown in a raw format, listing various food items like 'Aceite de maiz', 'Cereales de desayuno con base de maiz', etc.

Figura 17: Exemple cerca a la BBDD de DIETAS.NET

<https://www.codepunker.com/tools/http-requests>

Codepunker permet generar un document amb informació sobre el mètode *POST*, mostrant la comanda *Curl* per mostrar els resultats.

```
Request Response: Raw / Parsed / Generate Docs

## Request method: `POST`
### Request url: `http://www.dietas.net/inc/ajax/alim/buscar.php`
### Post params:
- search: maiz
### Curl command:
curl 'http://www.dietas.net/inc/ajax/alim/buscar.php' -X POST -d 'search=maiz'
```

Figura 18: Comandament Curl del mètode POST

<https://www.codepunker.com/tools/http-requests>

Amb el programari *Curl-to-PHP* s'ha pogut convertir la crida *Curl* en *PHP*.

Curl-to-PHP

Instantly convert [curl](#) commands to [PHP](#) code

This tool turns a Curl command into PHP code. Currently, it knows the following options: `-d/--data/--data-binary`, `-H/--header`, `-I/--head`, `-u/--user`, `compressed`, `--url` and `-X/--request`. There's probably bugs; please [contribute on GitHub!](#)

This script derives from Matt Holt's excellent [curl-to-Go](#).

[Example 1](#) · [Example 2](#) · [Example 3](#) · [Example 4](#) · [Example 5](#)

```
curl 'http://www.dietas.net/inc/ajax/alim/buscar.php' -X POST -d 'search=maiz'
```

```
// Generated by curl-to-PHP: http://incarnate.github.io/curl-to-php/  
$ch = curl_init();  
  
curl_setopt($ch, CURLOPT_URL, "http://www.dietas.net/inc/ajax/alim/buscar.php");  
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);  
curl_setopt($ch, CURLOPT_POSTFIELDS, "search=maiz");  
curl_setopt($ch, CURLOPT_POST, 1);  
  
$headers = array();  
$headers[] = "Content-Type: application/x-www-form-urlencoded";  
curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);  
  
$result = curl_exec($ch);  
if (curl_errno($ch)) {  
    echo 'Error:' . curl_error($ch);  
}  
curl_close ($ch);
```

Figura 19: Conversió Curl a PHP

<https://incarnate.github.io/curl-to-php/>

4.2. Arquitectura de la informació i diagrames de navegació

- Diagrama ULM de casos d'ús:

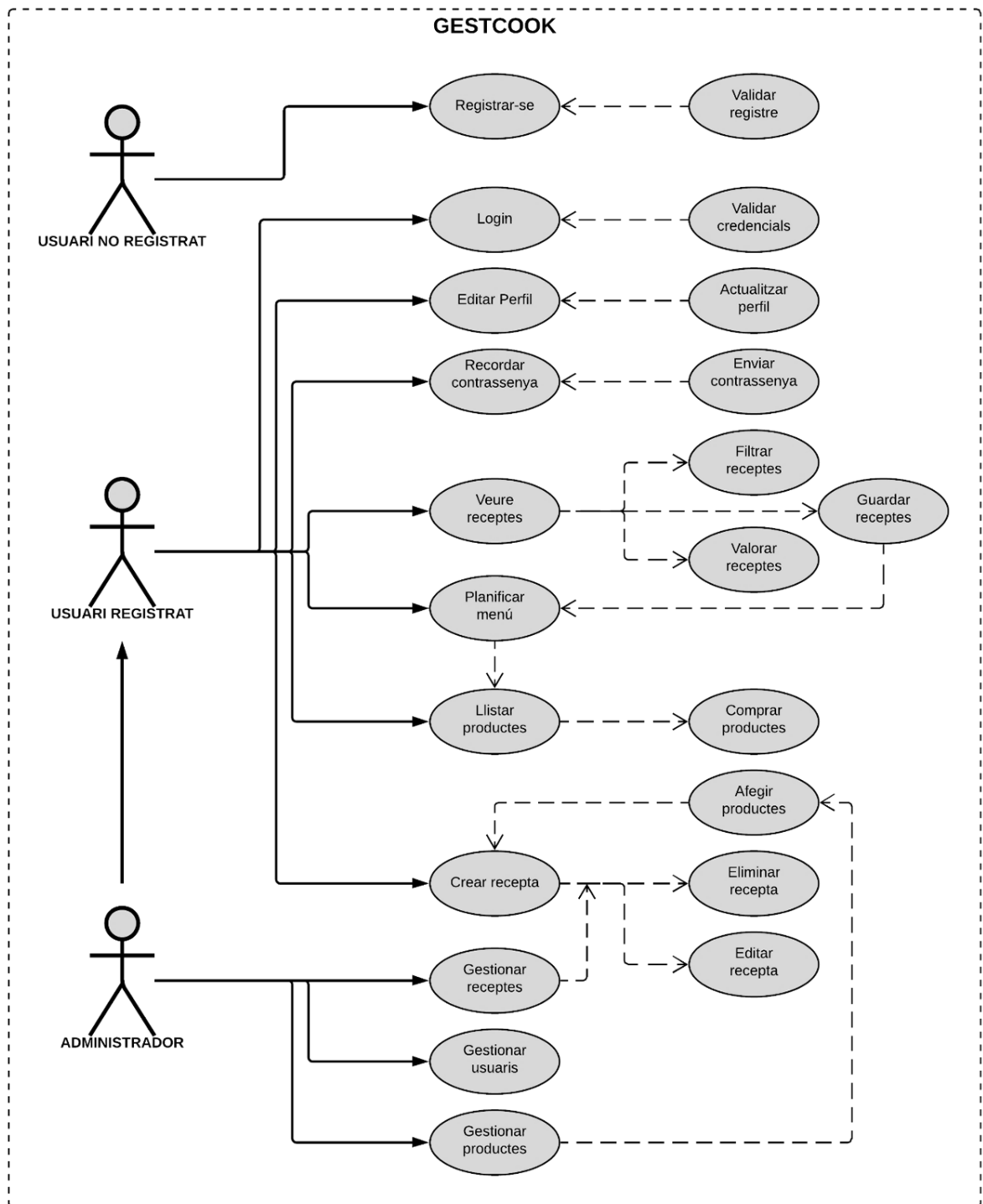


Diagrama 1: Diagrama de casos d'ús

- Diagrama de navegació:

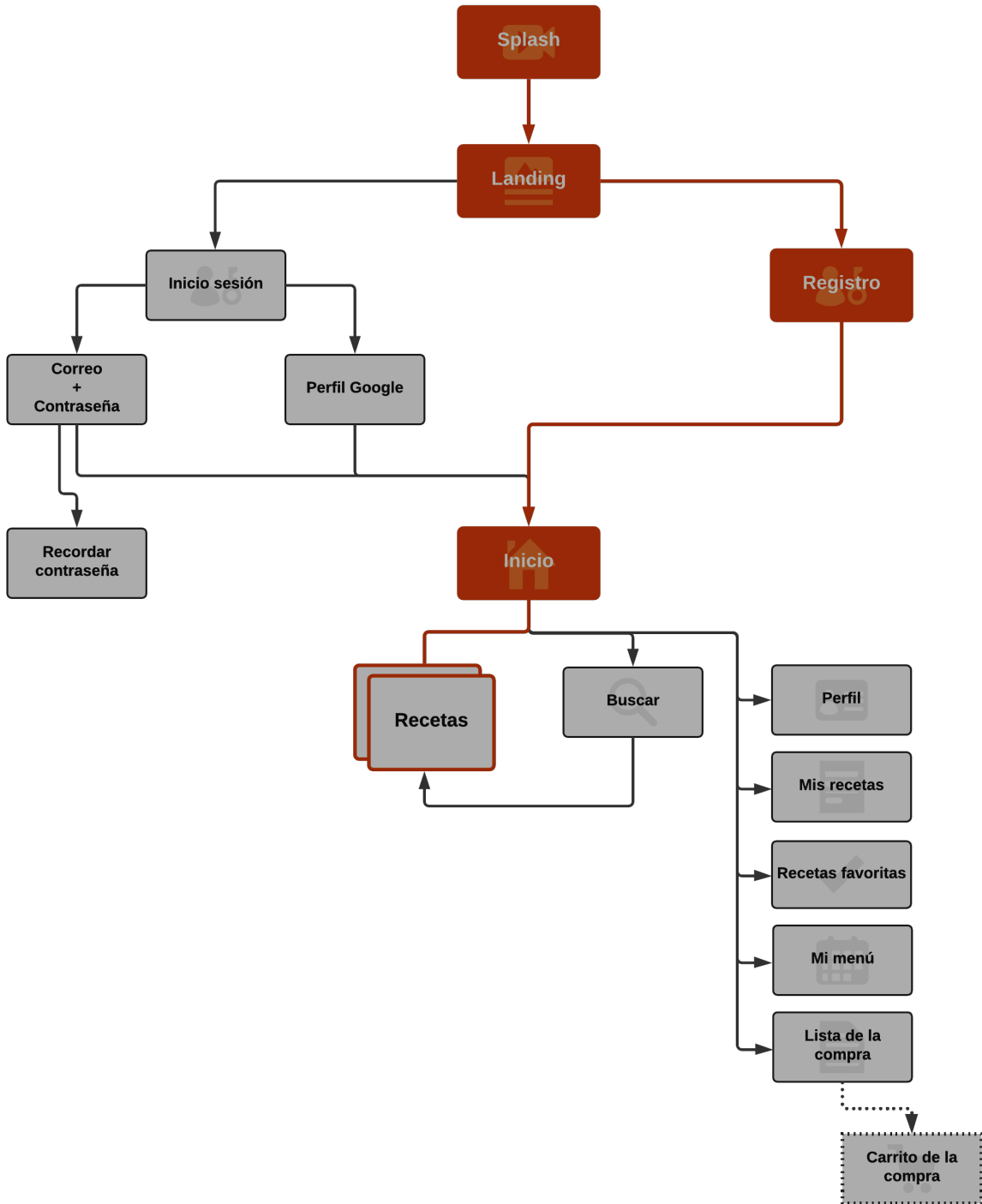


Diagrama 2: Diagrama de navegació

- Diagrama de classes:

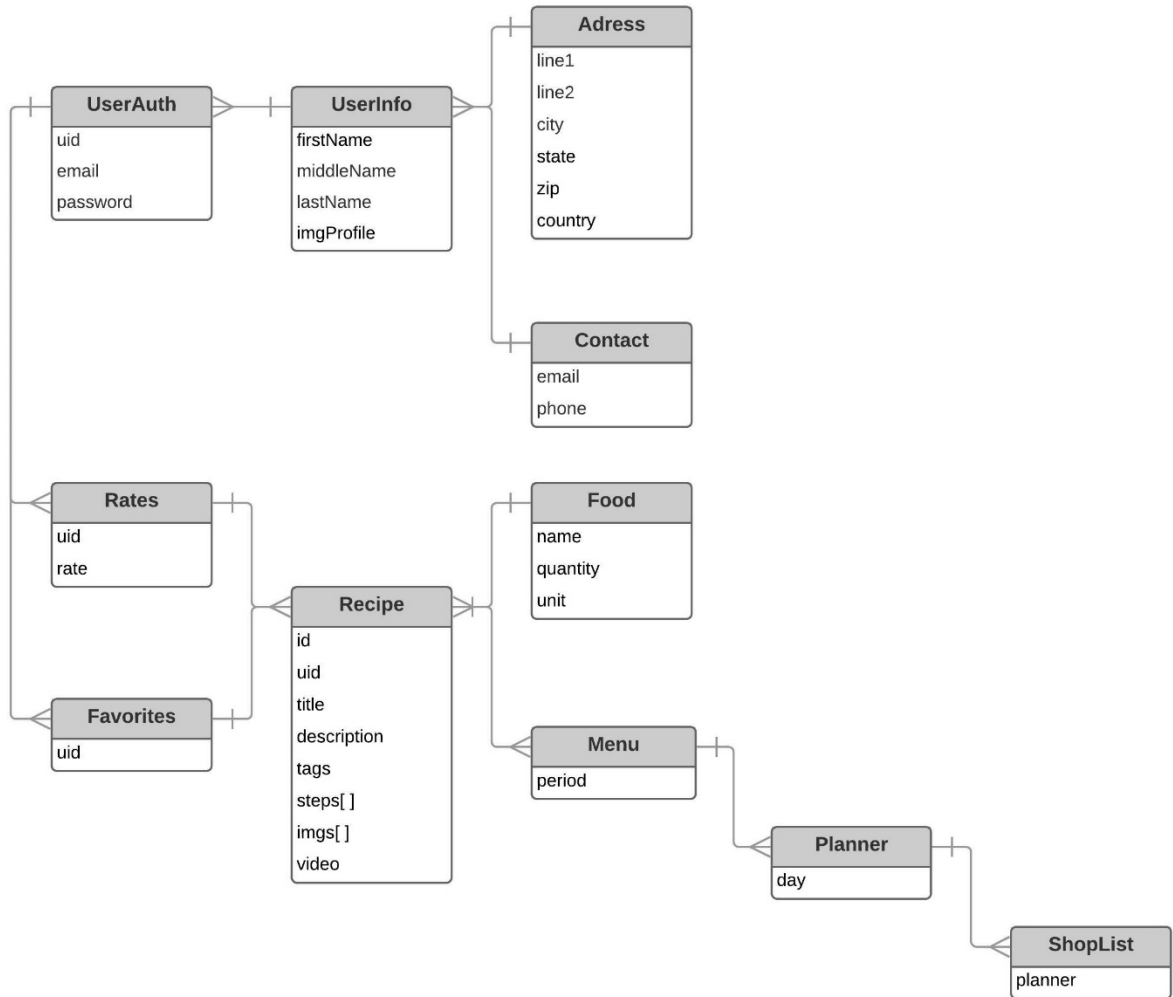


Diagrama 3: Diagrama de classes

- Bases de dades:

```

1 {
2   "UserAuth":
3   {
4     "uid": 1,
5     "email": "fmarmolp@uoc.edu",
6     "password": "passUOC"
7   },
8
9   "UserInfo":
10  {
11    "firsName": "Francesc",
12    "middleName": "Marmol",
13    "lastName": "Preciado",
14    "imgProfile": "url",
15    "adress": [
16      {
17        "line1": "Passeig Migdia",
18        "line2": 60,
19        "city": "Palafrugell",
20        "state": "SP",
21        "zip": 17200,
22        "country": "Girona"
23      }
24    ],
25    "contact": [
26      {
27        "email": "fmarmolp@uoc.edu",
28        "phone": 600600600
29      }
30    ]
31  },
32
33  "Recipes":
34  {
35    "id": 1,
36    "uid": 1,
37    "title": "Ensalada mixta",
38    "description": "Ensalada con tomates y atun",
39    "tags": ["lechuga", "fresco"],
40    "steps": [
41      {"stepDescription": "Limpiar lechuga"},
42      {"stepDescription": "Cortar tomates"}
43    ],
44    "imgs": [
45      {"url": "https://firebasestore.googleapis.com/v8/b/image1.jpg"},
46      {"url": "https://firebasestore.googleapis.com/v8/b/image2.jpg"}
47    ],
48    "video": "https://firebasestorage.googleapis.com/v8/c/video.avi",
49    "food": [
50      {"name": "lechuga", "quantity": 1, "unit": "pieza"},
51      {"name": "atun", "quantity": 100, "unit": "gramos"}
52    ]
53  },
54
55  "ShopList":
56  {
57    "planner": 1,
58    "day": [
59      {
60        "id": "31-10-2018",
61        "period": [
62          {"id": "desayuno", "reciepe_id": 1},
63          {"id": "comida", "reciepe_id": 35}
64        ]
65      }
66    ]
67  }
68 }

```

object ▶ UserInfo ▶

- object {4}
 - ▶ UserAuth {3}
 - ▶ UserInfo {6}
 - ▶ Recipes {9}
 - ▶ ShopList {2}

Ln: 68 Col: 1

Figura 20: Estructura de la Bases de dades de GESTCOOK

4.3. Disseny gràfic i interfícies

4.3.1. Estils

- Logotips i anagrames:



Figura 21: Logotip i icona GESTCOOK

- Paleta de colors:



Figura 22: Paleta colors

- Paleta tipogràfica, grandària i estil de fonts: la tipografia escollida ha estat “Ubuntu” de l’empresa Dalton Maag patrocinada per Canonical Ltd. Aquesta tipografia ve predeterminada al tema de la plantilla “United” de Bootswatch. Pertany a la família OpenType de les tipografies sans-serif humanístic. Totes les tipografies incloses a aquest *framework* són responsives i sota llicència [MIT](#).

Typography

Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

Heading with muted text

Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor.

Example body text

Nullam quis risus eget urna mollis ornare vel eu leo. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Nullam id dolor id nibh ultricies vehicula.

This line of text is meant to be treated as fine print.

The following is **rendered as bold text**.

The following is *rendered as italicized text*.

An abbreviation of the word attribute is `attr`.

Figura 23: Exemple tipografia Ubuntu

- Fons, icones, botons i altres elements gràfics:
 - Camps i elements de formularis:



Figura 24: Camps i elements de formulari

- Botons:

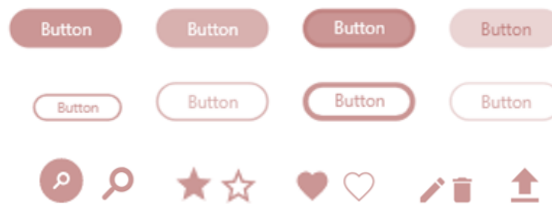


Figura 25: Botons i icones

- Elements emergents:

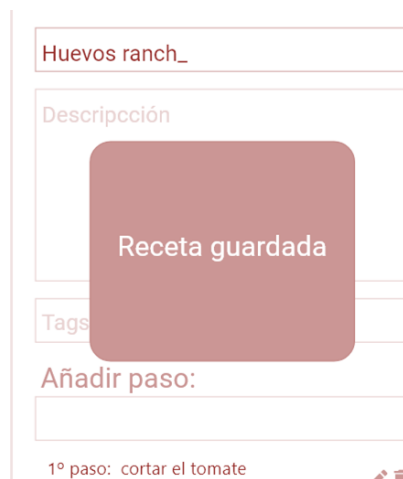


Figura 26: Altres elements gràfics

4.3.2. Usabilitat/UX

- **Formes d'interacció:** un cop registrat i iniciada la sessió, l'usuari pot:
 - Navegar per l'aplicació a través del menú lateral.
 - Realitzar cerques i llistar informació.
 - Crear, editar i eliminar informació o pujar arxius.
- **Navegació:** barra principal de navegació a la part superior de la pantalla i que es manté constant a tots els apartats de l'aplicació. El logotip actua a mode de "home" que permet accedir a la pantalla d'inici en tot moment. El botó del menú desplega un menú lateral que mostra els diferents apartats de l'aplicació a on es pot navegar.

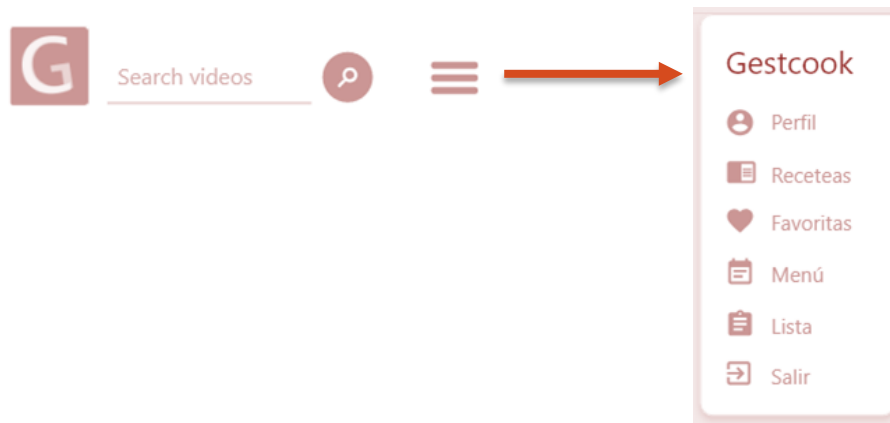


Figura 27: Navegació

- **Sitemap:** estructura jeràrquica poc profunda amb no més de tres nivells.

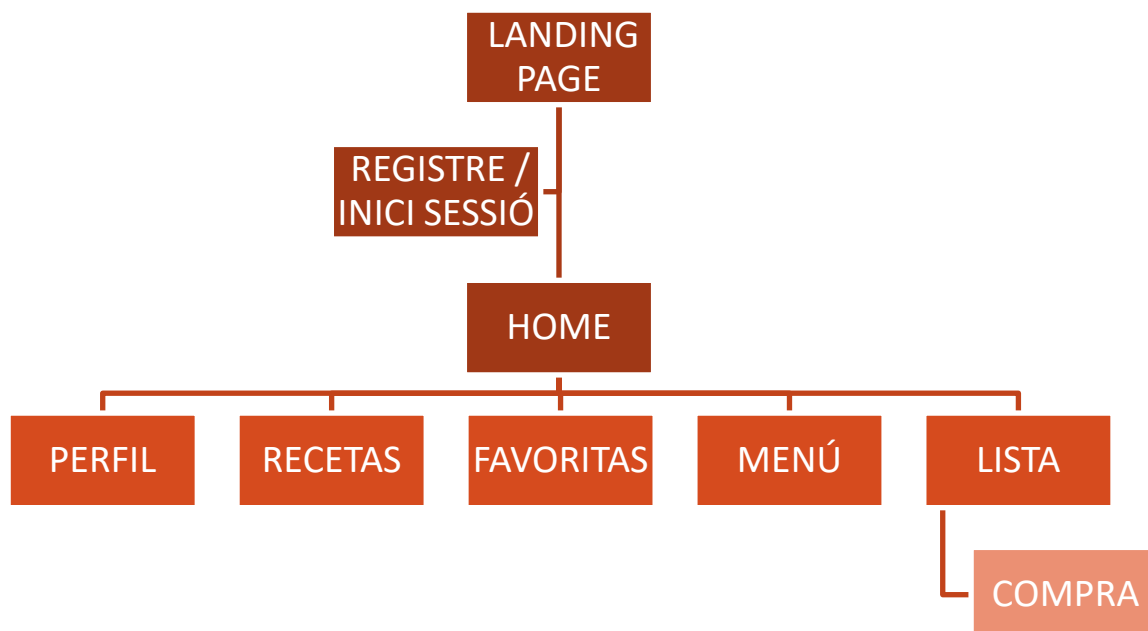


Figura 28: Sitemap

4.4. Llenguatges de programació i APIs utilitzats

Descripció del procés realitzat per triar la plataforma de desenvolupament (p.ex. CMS, sistema operatiu, llenguatge de programació, eines a utilitzar, etc.). Enumerar les principals plataformes i eines existents per realitzar el producte/servei i justificar l'elecció de la solució escollida.

- **Llenguatges:**
 - HTML5 i CSS: llenguatge de marques per configurar i presentar el contingut.
 - JavaScript i TypeScript: llenguatge de programació per codificar el front-end
 - JSON: format de text per intercanvi de dades amb la BBDD de Firebase.
- **Software:**
 - Sistema operatiu: Windows 10 Professional 64bits (v. 1803).
 - Desenvolupament:
 - Visual Studio Code (v. 1.27.2): Editor de codi font.
 - Angular (v. 7.0.1): *framework front-end*.
 - Angular CLI (v. 7.0.3): Interfície de línia de comandaments d'*Angular*.
 - Node.js (v. 8.12.0): entorn d'execució del costat del servidor JavaScript.
 - Npm (v. 6.4.1): gestor de paquets per a Node.js.
 - Firebase CLI (versió 5.1.1): Interfície de línia de comandaments de *Firebase*.
 - Bootstrap (v. 4.1.3): *framework* d'estils CSS .
 - Google Chrome (v. 61.0.3163.100): Navegador web.
 - Disseny:
 - Adobe Illustrator (v. CC 2018): Programari per a la realització de gràfics vectorials.
 - Adobe XD (v. CC 2018): Programari per a la realització de prototips.
 - Microsoft Office Professional Plus 2016 (v. 1809): Programari editor de textos.
- **APIs de tercers, complements, plug-ins:**
 - BEDCA i DIETAS.NET: Base de dades d'aliments.
 - Firebase: Plataforma de desenvolupament al núvol per emmagatzemar dades i arxius.
 - Authentication.
 - Cloud Firestore (versió Beta).
 - Cloud Storage.
 - Hosting.
- **Hardware:**
 - Desktop: Intel Pentium I7-4771 3.50GHz, 16 GB RAM.
- **Altres:**
 - tomsplanner.es: programari en línia per crear diagrames de *Gantt*.
 - material.io: programari en línia per crear paletes de color per UI.
 - materialpalette.com: programari en línia per crear paletes de color per UI.
 - lucidchart.com: programari en línia per realitzar diagrames UML.
 - mockflow.com: programari en línia per realitzar *wireframes*.

5. Implementació

5.1. Requisits d'instal·lació

Atès que es tracta d'una aplicació web progressiva, aquesta s'executa al navegador del dispositiu. Per tant, l'usuari només necessita d'un dispositiu de visualització amb accés a Internet i un navegador instal·lat per poder introduir la direcció web de l'aplicació.



Figura 29: Cercador web

5.2. Instruccions d'instal·lació

La utilització de l'aplicació GESTCOOK no requereix cap mena d'instal·lació per part de l'usuari. Al resultar una aplicació web progressiva només cal accedir al web mitjançant la seva URL. A l'accedir al web, apareixerà un avís a peu de pantalla per indicar si es vol crear un accés directe a l'aplicació, simulant la icona d'una aplicació nativa



Figura 30: Accés directe GESTCOOK

5.3. Instal·lació i configuració entorn de treball

L'entorn de desenvolupament està format per les següents eines:

- **Visual Studio Code:** per treballar amb el *framework* Angular 7 s'ha triat aquest IDE ja que és gratuït, *open source* i multi plataforma, entre d'altres. Està preparat per a *TypeScript* i és compatible amb Angular *IntelliSense*. Aquest sistema *IntelliSense* permet detectar llibreries importades i permet auto completar text, mostrar informació de la variable/mètode/classe seleccionada... També incorpora un gestor de *Git* que permet realitzar *commit*, *pull* i *push*, i un terminal per defecte que permet treballar amb els comandaments de Angular CLI.

Per instal·lar *Visual Studio Code* només cal anar a la pàgina [web](#) principal, triar el paquet en funció del nostre S.O i executar l'instal·lable descarregat.

- **Angular 7:** per treballar amb Angular 7 cal realitzar la instal·lació d'alguns paquets:
 - Node.js: per instal·lar Node.js només cal accedir a la pàgina [web](#) principal i descarregar l'última versió estable que està identificada amb la terminació LTS. En aquest cas s'utilitzarà la versió 10.13.0 LTS per a Windows (x64). Per comprovar que s'ha instal·lat correctament, a través del terminal s'ha d'introduir el comandament:

```
node -v
```

- npm: el paquet npm ve incorporat dintre de Node.js. Per tant, un cop instal·lat Node.js també tindrem instal·lat npm en la seva versió estable 6.4.1. També es pot descarregar des de la pàgina [web](#) principal. Per comprovar que s'ha instal·lat correctament, a través del terminal s'ha d'introduir el comandament:

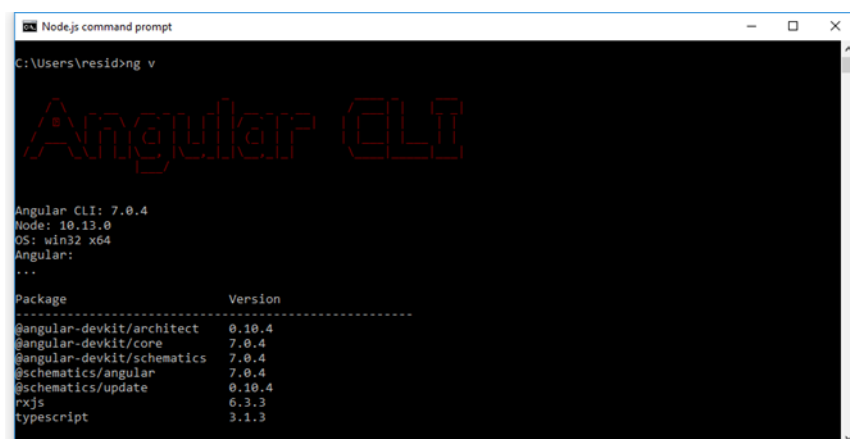
```
npm -v
```

- Angular CLI: aquesta eina d'Angular és una interfície de línia de comandament que s'utilitza per a la creació de SPA. Per instal·lar Angular CLI cal fer-ho des del terminal del nostre PC:

```
npm install -g @angular/cli
```

- Un cop instal·lat el CLI d'Angular podem comprovar que tot l'entorn de producció està instal·lat correctament amb el següent comandament:

```
ng v
```



```

Node.js command prompt
C:\Users\resid>ng v

Angular CLI
Angular CLI: 7.0.4
Node: 10.13.0
OS: win32 x64
Angular:
...
Package          Version
-----
@angular-devkit/architect    0.10.4
@angular-devkit/core        7.0.4
@angular-devkit/schematics   7.0.4
@schematics/angular         7.0.4
@schematics/update          0.10.4
rxjs                       6.3.3
typescript               3.1.3

```

Figura 31: Instal·lació Angular 7

5.4. Desenvolupament de l'aplicació

5.4.1. Front-end

- **Creació del projecte:** per poder crear i executar un nou projecte amb Angular 7 es pot fer a través del terminal del propi *Visual Studio Code* amb els següents comandaments:

```
ng new gestcook
```

```
cd gestcook
```

```
ng serve --open
```

En aquest cas es crea un projecte amb el nom "gestcook". A continuació ens desplaçem a la carpeta a on s'ha creat el projecte i s'executa. Des del navegador, a través de la següent direcció del servidor intern es pot accedir a la SPA que s'ha creat: <http://localhost:4200/>

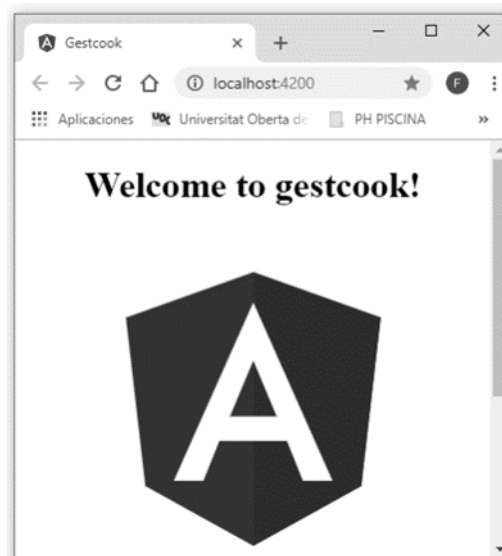


Figura 32: Pàgina inici PWA Angular 7

Per a treballar els estils de l'aplicació s'instal·la la biblioteca Bootstrap 4 per a Angular. Aquesta biblioteca permet aplicar components de UI moderns que funcionen de forma responsiva a la web, mòbil i escriptori. També s'instal·la un *theme* de *Bootswach* que deriva del propi *Bootstrap*. La instal·lació es fa a través del terminal:

```
npm install bootstrap
```

```
npm install bootswach
```

Per convertir la SPA en PWA només cal introduir els següent comandament que instal·lar la llibreria a on es disposa de tot el necessari per oferir els serveis d'instal·lació a l'escriptori, rebre notificaciones push del servidor... Totes aquestes directrius venen especificades al manifest de l'aplicació en un arxiu *JSON*. També fa que la *PWA* treballi amb *Service Workers*.

```
ng add @angular/pwa
```


- **Components:** la pàgina principal d'Angular és l'aplicació *Shell*. El Shell està controlat per un component d'Angular anomenat *AppComponent*. Els components són els blocs de construcció fonamentals en les aplicacions d'Angular. Tenen una plantilla *HTML* i només un sol component pot ser instanciat per un element a una plantilla. Aquests components mostren dades per pantalla, escolten els inputs de l'usuari i prenen accions en funció d'aquests inputs.

L'aplicació Shell està formada per tres arxius, a l'igual que la resta de components:

- `app.component.ts`: és el codi de la classe del component, escrit en *TypeScript*.
- `app.component.html`: és la plantilla del component, escrita en *HTML*.
- `app.component.css`: és l'estil CSS privat del component.

Per a crear nous components es pot fer amb el següent comandament des del terminal:

```
ng g c nomDelComponent
```

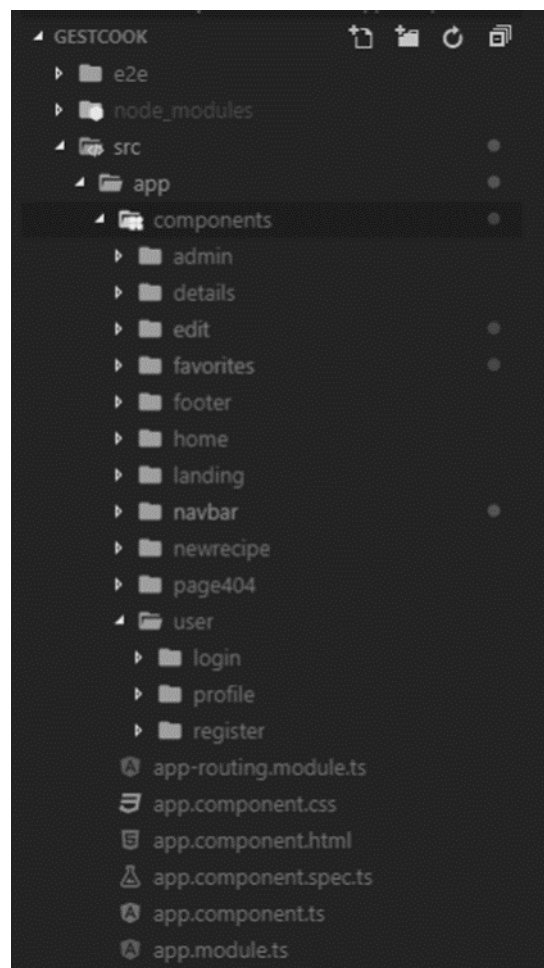


Figura 33: Llistat de components del projecte GESTCOOK

- **Rutes:** les diferents pàgines de l'aplicació estan enrutades al propi navegador. Per tant, s'aconsegueix alliberar al servidor d'una part del treball, reduint la quantitat de trucades a la vegada que es millora la percepció de velocitat per part de l'usuari.

Les rutes de navegació s'estableixen a l'arxiu "app-routing.module.ts" que s'autogenera en crear el projecte. A aquest arxiu s'importen els diferents components creats indicant les seves rutes. Un cop importats els components, es pot definir el comportament de la navegació.

En aquest cas s'ha indicat amb el comandament `redirectTo`: la ruta per defecte principal, que serà la *Landing Page*. Per a rutes no contemplades s'ha definit una ruta a una pàgina d'error 404 a través del `path: '**'`

```

4 import { LandingComponent } from './components/landing/landing.component';
5 import { HomeComponent } from './components/home/home.component';
6 import { FavoritesComponent } from './components/favorites/favorites.component';
7 import { NewrecipeComponent } from './components/newrecipe/newrecipe.component';
8 import { DetailsComponent } from './components/details/details.component';
9 import { EditComponent } from './components/edit/edit.component';
10 import { Page404Component } from './components/page404/page404.component';
11 import { ProfileComponent } from './components/user/profile/profile.component';
12 import { RegisterComponent } from './components/user/register/register.component';
13 import { LoginComponent } from './components/user/login/login.component';
14
15
16 const routes: Routes = [
17   {path: '', redirectTo: '/landing', pathMatch: 'full'},
18   {path: 'landing', component: LandingComponent},
19   {path: 'home', component: HomeComponent},
20   {path: 'favorites', component: FavoritesComponent},
21   {path: 'newrecipe', component: NewrecipeComponent},
22   {path: 'details/:id', component: DetailsComponent},
23   {path: 'edit/:id', component: EditComponent},
24   {path: 'profile', component: ProfileComponent},
25   {path: 'register', component: RegisterComponent},
26   {path: 'login', component: LoginComponent},
27   {path: '**', component: Page404Component},
28 ];

```

Figura 34: Definició de les rutes del projecte GESTCOOK

A Angular, els enllaços web tradicionals a on s'utilitzen els elements `` es substitueixen pels atributs `routerLink`.

```

<div class="collapse navbar-collapse" id="navbarColor01">
  <ul class="navbar-nav mr-auto">
    <li class="nav-item active">
      <a class="nav-link" routerLink="/recipes">Recetas <span class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
      <a class="nav-link" routerLink="/favorites">Favoritas</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" routerLink="/menu">Menú</a>
    </li>
  </ul>

```

Figura 35: Utilització del `routerLink` en la barra de navegació

5.4.2. Back-end

- **Inici a Firebase:** per treballar amb la base de dades de *Firebase* cal accedir al [web](#) i entrar a la consola amb un compte de *Google*. Un cop s'ha accedit podem crear el projecte amb el què es treballarà a l'aplicació *GESTCOOK*.



Figura 36: Alta del projecte a *Firebase*

- ***Firebase Authentication:*** per a la part del registre a l'aplicació s'utilitza *Firebase Authentication*. En aquest cas s'han habilitat les opcions de correu electrònic/contrasenya i *Google* per poder iniciar la sessió.

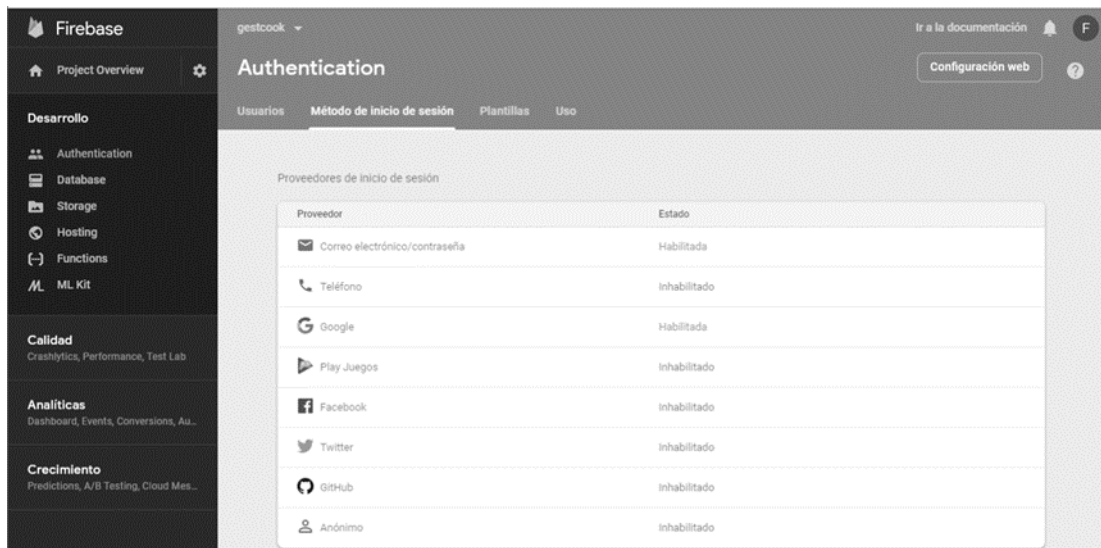


Figura 37: Configuració accés aplicació

Un cop creat el servei d'autenticació de *Firebase* a l'arxiu "services/auth.services.ts". En fer clic al botó de *Google* s'obre la finestra per triar l'usuari amb el qual es vol iniciar la sessió a l'aplicació:

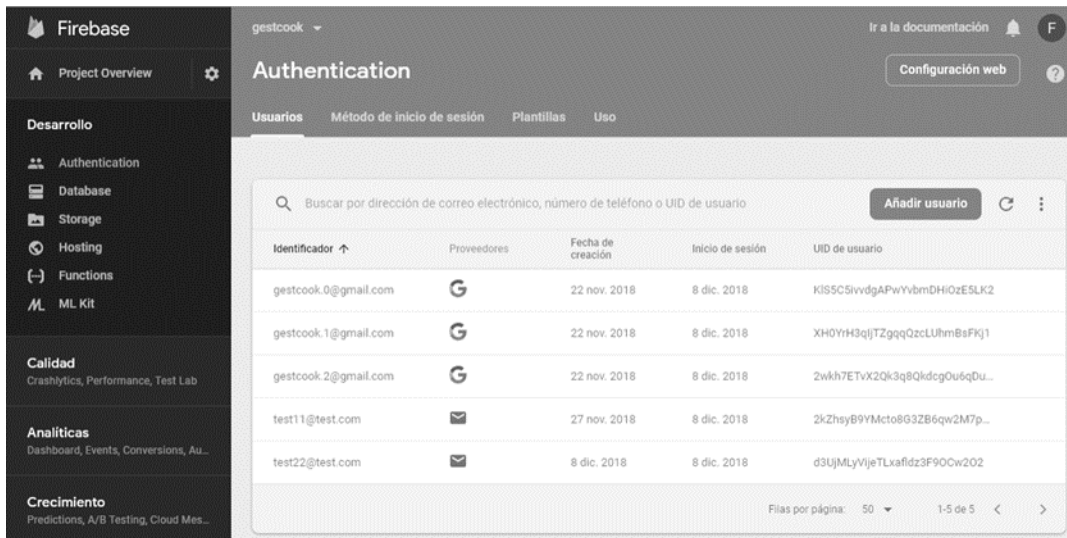


Figura 38: Usuaris creats amb el servei d'autenticació de *Firebase*

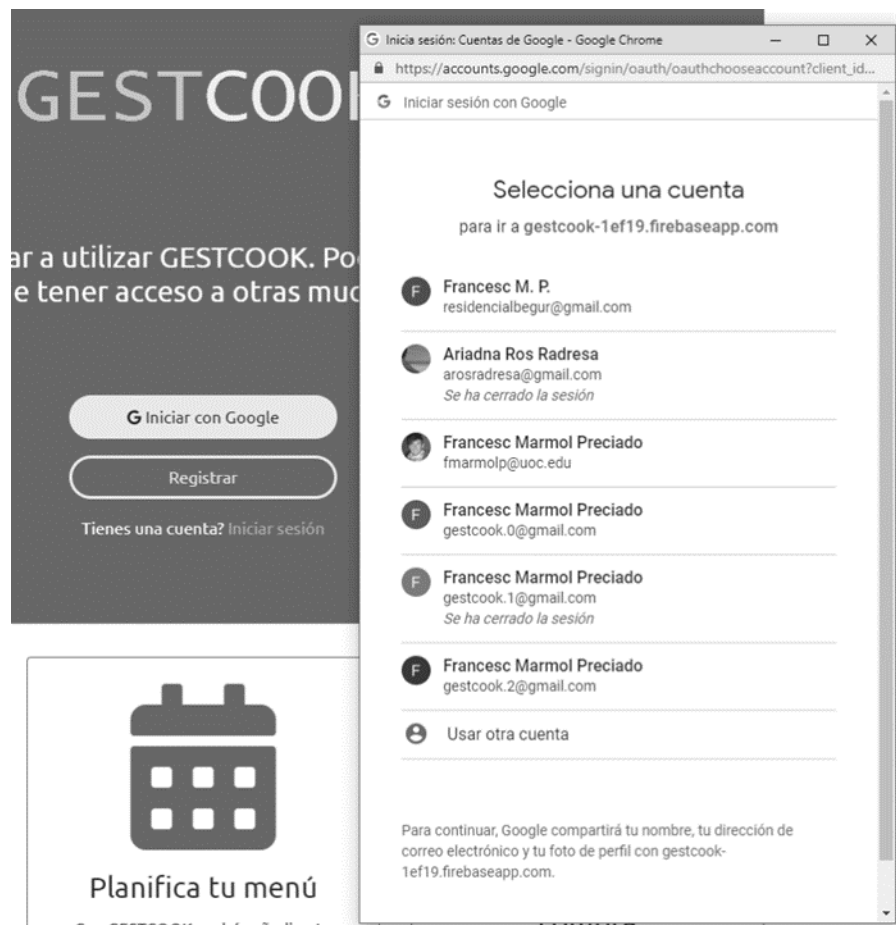


Figura 39: Inici de sessió amb un compte *Gmail* de *Google*

Aquest servei també proporciona un procediment per poder restablir la contrasenya en cas que l'usuari s'hagi registrat pel mètode de correu i contrasenya i la vulgui substituir o necessiti generar una de nova degut a que no s'enrecorda d'ella.

En aquest cas només cal prémer l'enllaç corresponent, introduir el correu amb el qual es va realitzar l'enregistrament a la plataforma i seguir el passos que venen indicats al correu enviat.



Figura 40: Correu per restablir la contrasenya del compte de l'usuari

El cos del missatge que rebrà l'usuari es pot personalitzar des de la pròpia plataforma de *Firebase*.

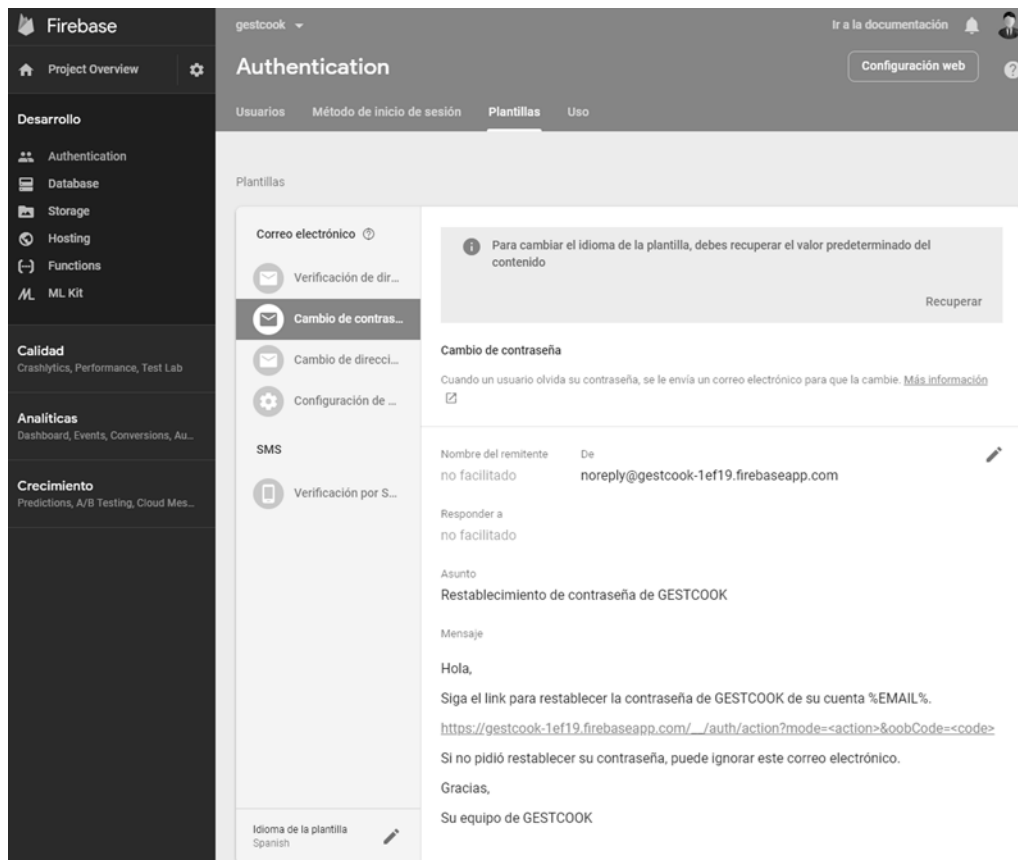


Figura 41: Apartat d'edició del missatge que apareix al correu de restabliment de la contrasenya

- **Cloud Firestore i Firebase:** es crea la base de dades amb *Cloud Firestore*, que és la nova generació de la base de dades que utilitza *Firebase*. Fins al moment es feia servir *Realtime Database*, que encara es pot utilitzar.

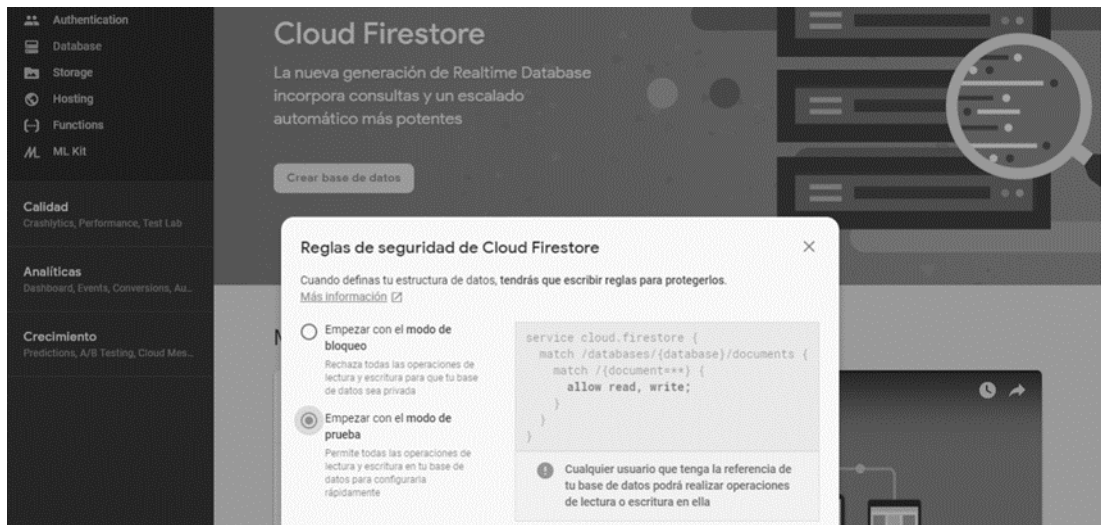


Figura 42: Creació de la BBDD

- **Afegir Firebase al projecte:** un cop configurada tota la part del *back-end* es procedeix a afegir el codi que proporciona *Firebase*. *Angular* permet treballar en dos entorns, un per a desenvolupament, que es pot trobar *environments/environments.ts* i un altre per a producció, que es pot trobar a *environments/environments.prod.ts*.

Per tant, a *Firebase* podem crear un projecte per a desenvolupament (GESTCOOK-DEV) i que es compilarà amb el comandament `ng build`, i un altre per a producció (GESTCOOK) que es compilarà amb el comandament `ng build - -prod`. Aquest últim és el que realment implementa les funcions pròpies d'una PWA.

Finalment, només caldrà incorporar a cada arxiu de la carpeta *environments* les claus que proporciona *Firebase* de cada projecte.

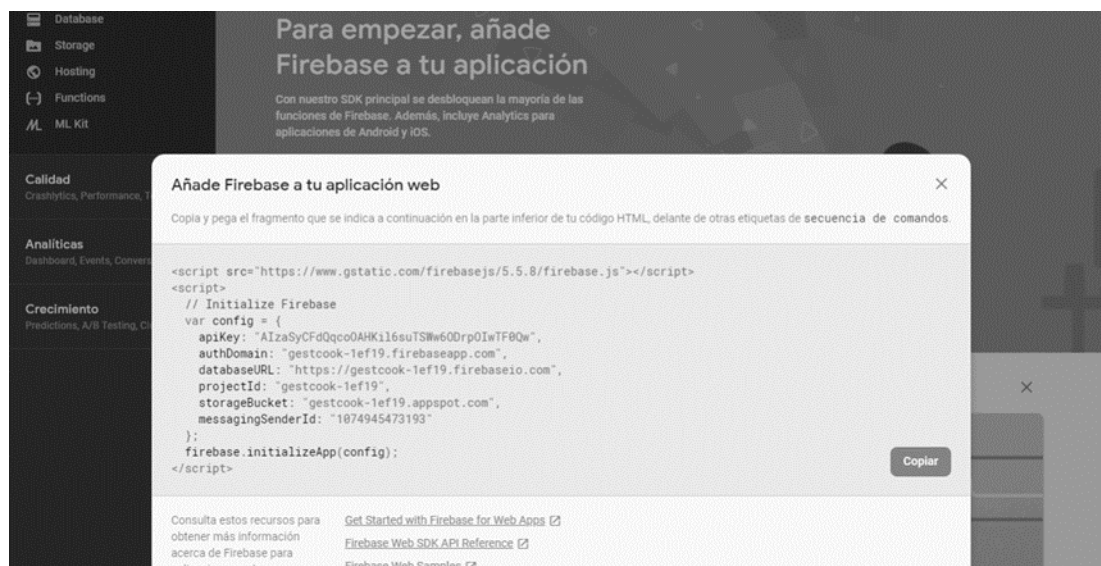


Figura 43: Codi per poder treballar amb *Firebase* des de l'aplicació

Des de la terminal d'Angular cal instal·lar les següents dependències de *Firebase*:

```
npm install -g @angular/cli@latest
```

```
npm install -g firebase-tools
```

```
npm install angularfire2 firebase - - save
```

Per poder treballar amb els diferents serveis que ofereix *Firebase*, cal importar el mòdul corresponents a l'arxiu "app.module.ts". També cal importar l'arxiu "environments/environment.ts" que guarda les credencial per accedir a *Firebase*:

```
// Firebase
import { AngularFireModule } from 'angularfire2';
import { AngularFireAuthModule } from 'angularfire2/auth';
import { AngularFireStoreModule } from 'angularfire2/firestore';
import { AngularFireStorageModule } from 'angularfire2/storage';
import { AuthService } from './services/auth.service';
export const firebaseConfig = environment.firebaseConfig;
// Environments
import { environment } from '../environments/environment';

56 ],
57 imports: [
58   BrowserModule,
59   FormsModule,
60   AppRoutingModule,
61   ServiceWorkerModule.register('ngsw-worker.js', { enabled: environment.production }),
62   BrowserAnimationsModule,
63   AngularFontAwesomeModule,
64   AngularFireModule.initializeApp(firebaseConfig),
65   AngularFireAuthModule,
66   AngularFireStoreModule,
67   AngularFireStorageModule
```

Figura 44: Codi importacions mòduls *Firebase* al projecte

A continuació cal crear els arxius corresponents a cada servei. Aquests es troben aglutinats a la nova carpeta "services":

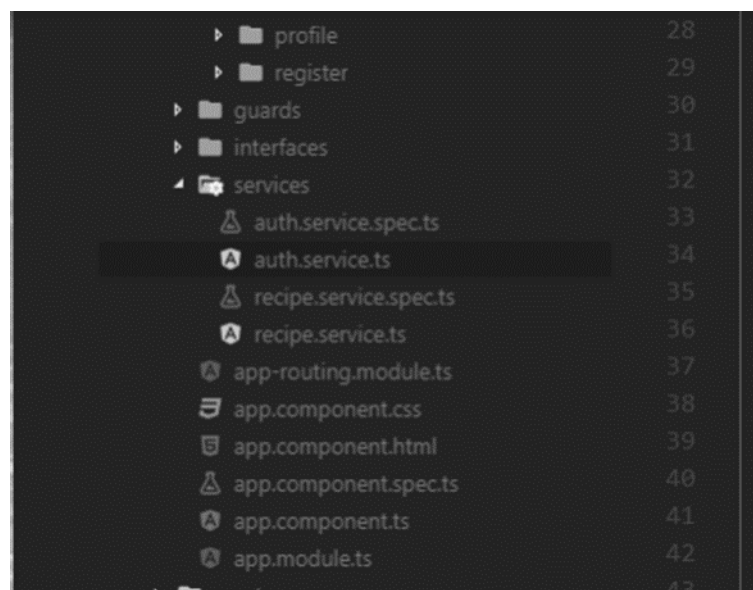


Figura 45: Llistat de serveis del projecte

Pujar el projecte a *Firebase*: gràcies al servei de *Hosting* al núvol de què disposa *Firebase* es fa molt fàcil implementar una aplicació web. Aquest hosting proporciona i configura automàticament un certificat SSL per a cada lloc web que s'implementa. També permet enllaçar un domini extern a través d'unes senzilles verificacions.

Per implementar una *app* des del directori local només cal seguir els següents passos:

```
npm install -g firebase-tools
```

```
firebase login
```

```
ng build
```

```
PS C:\Users\resid\Desktop\gestcook> firebase login
Already logged in as residencialbegur@gmail.com
PS C:\Users\resid\Desktop\gestcook> ng build

Date: 2019-01-04T18:55:03.626Z
Hash: ea05d8b24e357faf5523
Time: 11929ms
chunk {main} main.js, main.js.map (main) 288 kB [initial] [rendered]
chunk {polyfills} polyfills.js, polyfills.js.map (polyfills) 223 kB [initial] [rendered]
chunk {runtime} runtime.js, runtime.js.map (runtime) 6.08 kB [entry] [rendered]
chunk {scripts} scripts.js, scripts.js.map (scripts) 293 kB [rendered]
chunk {styles} styles.js, styles.js.map (styles) 3.19 MB [initial] [rendered]
chunk {vendor} vendor.js, vendor.js.map (vendor) 7.74 MB [initial] [rendered]
PS C:\Users\resid\Desktop\gestcook> |
```

Figura 46: Generació del *build* de l'aplicació amb èxit

```
firebase init
```

```
PROBLEMAS 6 SALIDA CONSOLA DE DEPURACIÓN TERMINAL
#####
##
#####
##
#####
##
#####
##
#####
##
#####
#####

You're about to initialize a Firebase project in this directory:

C:\Users\resid\Desktop\gestcook

Before we get started, keep in mind:

* You are initializing in an existing Firebase project directory

? Are you ready to proceed? Yes
? Which Firebase CLI features do you want to setup for this folder? Press Space to select features, then Enter to confirm your choices.
  ( ) Database: Deploy Firebase Realtime Database Rules
  ( ) Firestore: Deploy rules and create indexes for Firestore
  ( ) Functions: Configure and deploy Cloud Functions
>(*) Hosting: Configure and deploy Firebase Hosting sites
  ( ) Storage: Deploy Cloud Storage security rules
```

Figura 47: Selecció opció *Hosting*


```

PROBLEMAS 6 SALIDA CONSOLA DE DEPURACIÓN TERMINAL

* You are initializing in an existing Firebase project directory

? Are you ready to proceed? Yes
? Which Firebase CLI features do you want to setup for this folder? Press Space to select features, then Enter to confirm your choices.

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

i .firebaserc already has a default project, skipping

=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? (public) dist/gestcook

```

Figura 48: Selecció carpeta que conté el *build* del projecte

```

PROBLEMAS 6 SALIDA CONSOLA DE DEPURACIÓN TERMINAL

? Are you ready to proceed? Yes
? Which Firebase CLI features do you want to setup for this folder? Press Space to select features, then Enter to confirm your choices.

=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

i .firebaserc already has a default project, skipping

=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? dist/gestcook
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? File dist/gestcook/index.html already exists. Overwrite? (y/N) n

```

Figura 49: Indicació de que es tracta una SPA

```

PROBLEMAS 6 SALIDA CONSOLA DE DEPURACIÓN TERMINAL

You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

i .firebaserc already has a default project, skipping

=== Hosting Setup

Your public directory is the folder (relative to your project directory) that
will contain Hosting assets to be uploaded with firebase deploy. If you
have a build process for your assets, use your build's output directory.

? What do you want to use as your public directory? dist/gestcook
? Configure as a single-page app (rewrite all urls to /index.html)? Yes
? File dist/gestcook/index.html already exists. Overwrite? No
i Skipping write of dist/gestcook/index.html

i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...

+ Firebase initialization complete!
PS C:\Users\resid\Desktop\gestcook>

```

Figura 50: Indicació de que no es vol sobre escriure l'arxiu *index.html*

firebase deploy

```

PROBLEMAS 5 SALIDA CONSOLA DE DEPURACIÓN TERMINAL
i Writing project information to .firebaserc...
+ Firebase initialization complete!
PS C:\Users\resid\Desktop\gestcook> firebase deploy

=== Deploying to 'gestcook-1ef19'...

i deploying hosting
i hosting[gestcook-1ef19]: beginning deploy...
i hosting[gestcook-1ef19]: found 36 files in dist/gestcook
+ hosting[gestcook-1ef19]: file upload complete
i hosting[gestcook-1ef19]: finalizing version...
+ hosting[gestcook-1ef19]: version finalized
i hosting[gestcook-1ef19]: releasing new version...
+ hosting[gestcook-1ef19]: release complete

+ Deploy complete!

Project Console: https://console.firebase.google.com/project/gestcook-1ef19/overview
Hosting URL: https://gestcook-1ef19.firebaseio.com
PS C:\Users\resid\Desktop\gestcook>

```

Figura 51: Obtenció de l'URL de l'aplicació

Després de completar tot el procés, finalment es genera una URL de l'aplicació:

<https://gestcook-1ef19.firebaseio.com>



Figura 52: Pàgina d'inici de Gestcook.es

5.4.3. Control de versions

Per treballar amb més seguretat en un projecte a on s'estan desenvolupant i incorporant noves implementacions constantment, cal portar un control de les versions del projecte. Per aquest motiu s'ha creat un repositori Git remot a GitHub: <https://github.com/fmarmolpUOC/gestcook.git>

Els passos a seguir per generar el repositori Git de forma són:

- Descarregar i instal·lar el programari GitHub
- Seleccionar la carpeta del projecte i amb el boto dret seleccionar Git Bash Here
- Iniciar Git amb el següent comandament:
`git init`
- Guardar tots els arxius al repositori per fer-ne el seguiment:
`git add .`
- Crear un punt de restauració i nomenar-lo:
`git commit`
- Un cop s'ha acabat un sprint del projecte es guarda:
`git branch nom_del_sprint`
- Es crea un nou sprint i es continua treballant a partir d'aquest:
`git branch nom_del_sprint`
`git checkout nom_del_sprint`
- Per pujar al repositori de GitHub:
`git remote add origin https://github.com/fmarmolpUOC/gestcook.git`
`git push -u origin master`
- Per afegir les diferents versions a la versió principal (situats a la versió master):
`git merge nom_del_branch_a_afegit_a_master`
- Per veure el nom i el correu de l'usuari que treballa en el repositori:
`git config user.name`
`git config user.email`

Detall de les versions generades:

- master: versió inicial.
- landing: versió a on es crea la *landing page*.
- page404: versió a on es crea la *page not found 404*.
- routerlink: versió a on es creen les rutes de navegació.
- firebase: versió a on es creen els serveis de *Firebase*.
- recipe: versió a on s'implementa el CRUD entre *Angular* i *Firebase*.
- favorites: versió a on s'implementa l'apartat de favorits.
- menu: versió a on s'implementa el calendari per a la creació dels menús.
- list: versió a on s'implementa l'apartat de la llista d'ingredients del menú seleccionat.
- serviceWorker: on s'implementen els serveis per comportar-se com una PWA.
- master: versió final a on s'ha fet un git merge de totes les versions.

6. Demostració

6.1. Instruccions d'ús

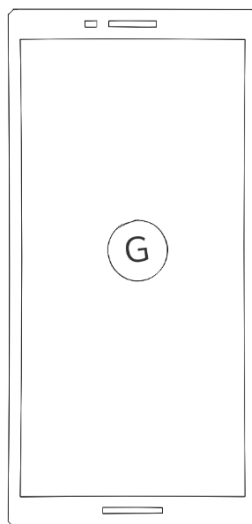
Per a la utilització de l'aplicació GESTCOOK només cal disposar d'un dispositiu amb navegador i accés a Internet. Un cop implementada es podrà fer a través de la següent URL:

- <https://www.gestcook.es>

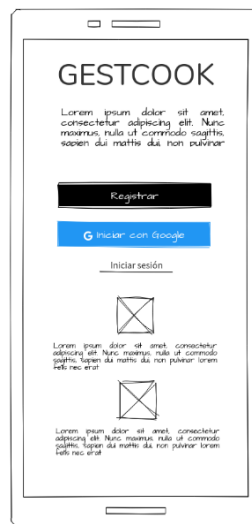
6.2. Prototips

6.2.1. Prototips Lo-Fi

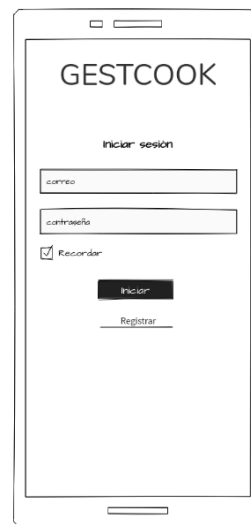
- *Wireframes*



SPLASH



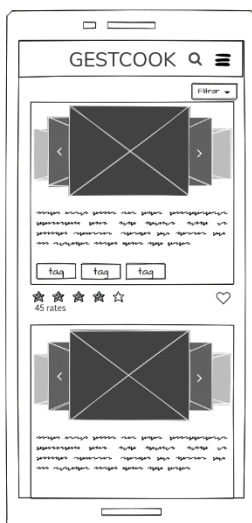
LANDING PAGE



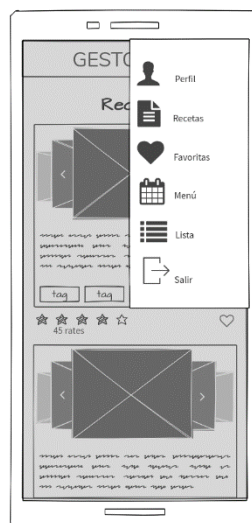
INICI SESSIÓ



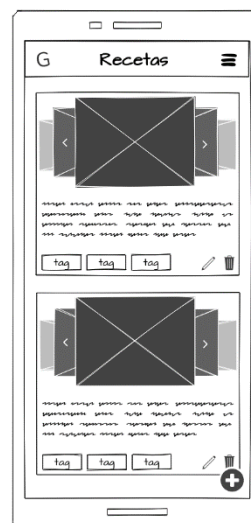
REGISTRE



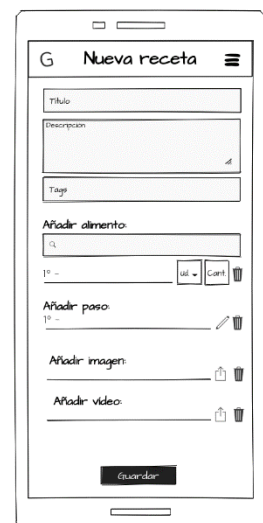
INICI



MENU LATERAL



RECEPTES
GUARDADES



NOVA / EDICIÓ
RECEPTA



Figura 53: Wireframes

6.2.2. Prototips Hi-Fi

Maquetes funcionals: s'ha dissenyat un prototip interactiu per poder comprovar la navegació i usabilitat de l'aplicació abans de procedir al seu desenvolupament. Per accedir al simulador del prototip funcional de l'aplicació GESTCOOK es pot fer a través del següent enllaç:

- <https://xd.adobe.com/view/34abfb7e-a3be-42e9-7977-4b39b3fbf8bd-429a/?hints=off>

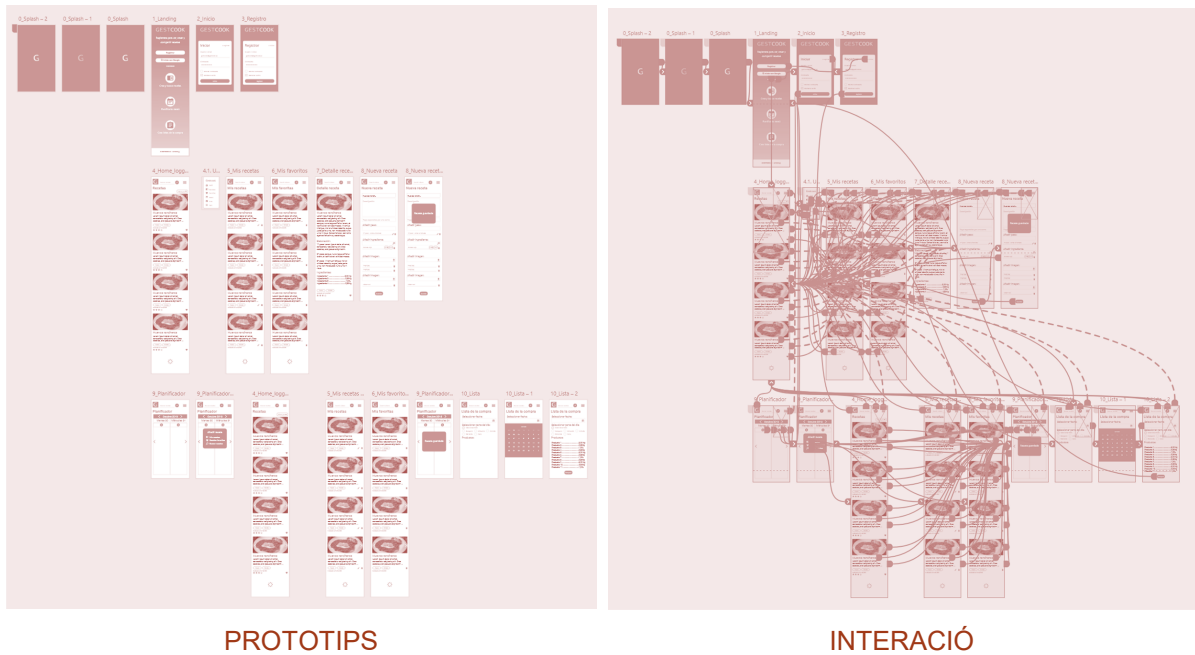


Figura 54: Prototip funcional

6.3. Tests

Per a testejar l'aplicació web s'ha creat un perfil d'administrador que pot accedir a tots els registres per editar-los o eliminar-los. Aquest administrador s'autentica a la plataforma a través del seu compte de *Gmail*. Altrament, s'han creat 4 perfils d'usuaris, dos d'ells s'autentiquen a través dels seus comptes de *Gmail* i els altres dos a través dels comptes creats amb correu i contrasenya.

Cadascun d'aquests perfils d'usuaris poden accedir a la pàgina principal a on es publiquen les receptes de la resta d'usuaris. També poden accedir a l'apartat de receptes personals a on crear, editar i eliminar les seves receptes. Altrament, poden accedir a l'espai de favorits a on es llisten les receptes creades per altres usuaris i que han marcat com a favorites.

Per poder comprovar el comportament de la plataforma s'han creat 5 receptes per a cada usuari. Altrament, l'usuari que estigui testejant l'aplicació és lliure d'accedir amb un compte de *Gmail* propi o crear un compte nou per poder consultar les receptes o crear-ne de noves.

- **Usuari:** per a la realització dels test d'usuari s'han creat els següents perfils:
 - Comptes d'usuari amb *Gmail* de *Google*:
 - Administrador:
 - correu: gestcook.0@gmail.com
 - contrasenya: fmarmolpTFG
 - Usuari 1:
 - correu: gestcook.1@gmail.com
 - contrasenya: fmarmolpTFG
 - Usuari 2:
 - correu: gestcook.2@gmail.com
 - contrasenya: fmarmolpTFG
 - Comptes d'usuari amb correu electrònic i contrasenya:
 - Usuari 1:
 - correu: test11@test.com
 - contrasenya: test11
 - Usuari 2:
 - correu: test22@test.com
 - contrasenya: test22

- **Usabilitat:** per auditar *apps* web Google disposa de Lighthouse. Aquesta eina es troba disponible a les eines per desenvolupadors a l'apartat "Audits". Aquest auditor executa una sèrie de proves contra la pàgina web, que en finalitzar genera un informe sobre el rendiment de la pàgina. Aquest informe mostra un llistat amb indicadors per a millorar l'app. Lighthouse actualment està enfocada en les funcions de les PWA, como *Add to homescreen* i suport sense connexió.

En realitzar una primera auditoria el resultat no es gaire bo atès que l'apartat relacionat amb les funcionalitats com a PWA són molt baixos amb una puntuació total de 38 punts:

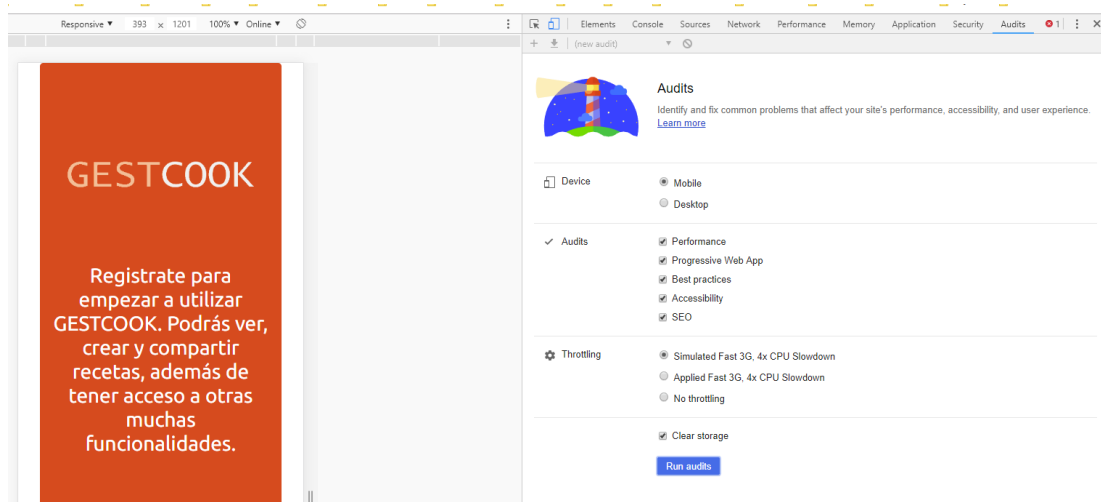


Figura 55: Configuració característiques audició

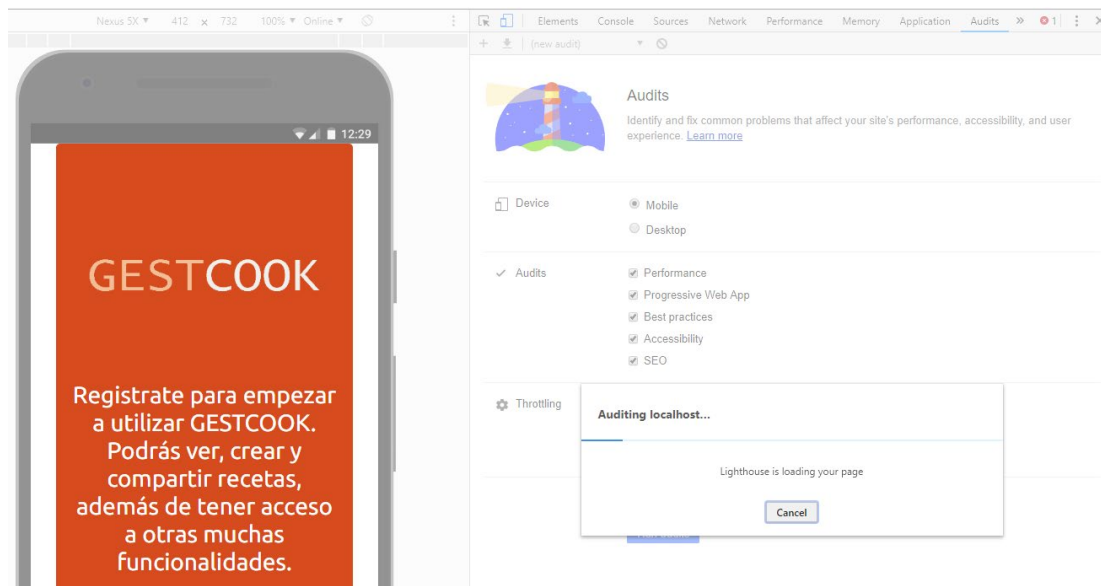


Figura 56: Audició en procediment

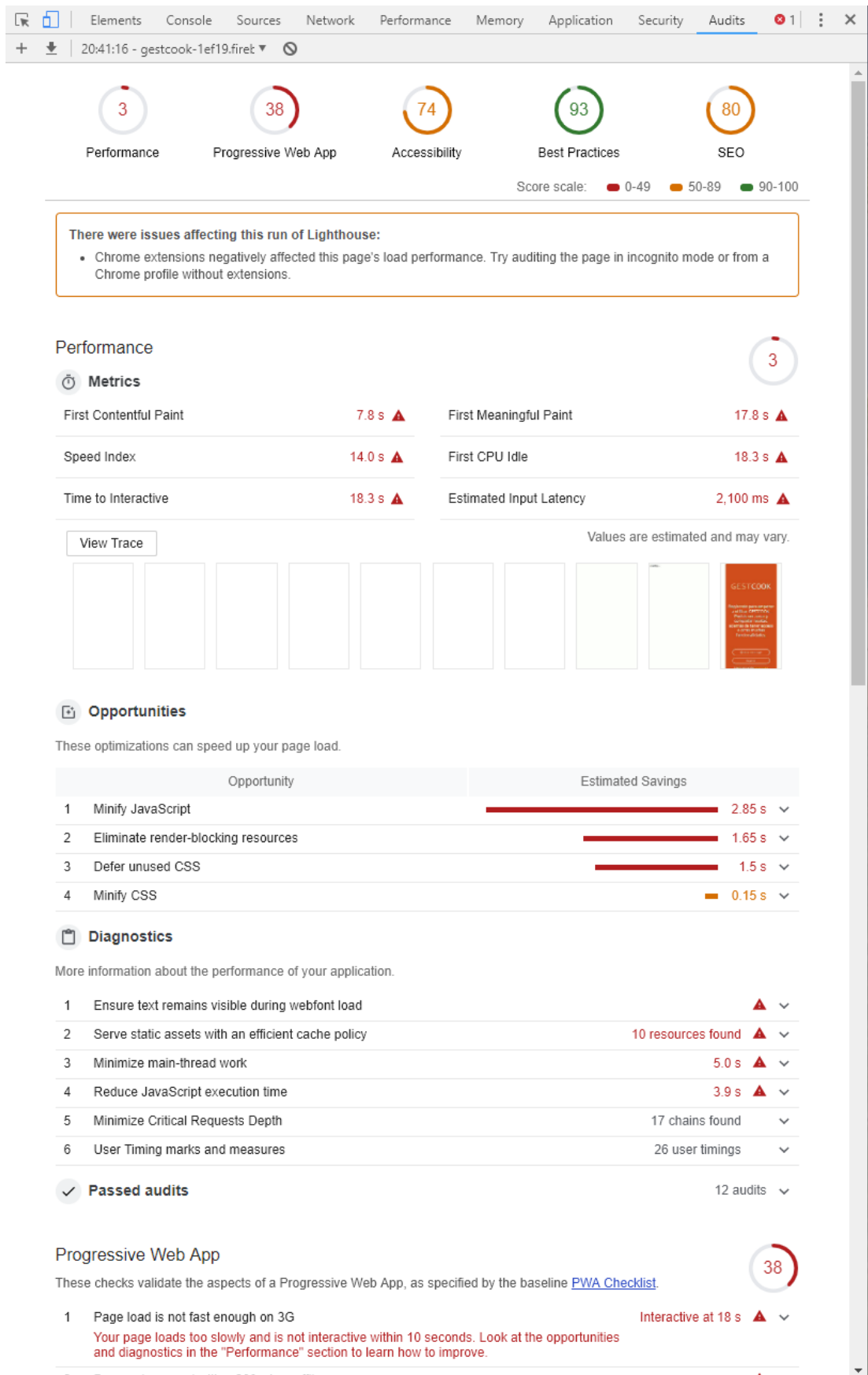


Figura 57: Resultat audició

Per solucionar les mancances detectades només cal posar en funcionament el *service worker* que ha de generar l'arxiu `ngsw-worker.js`:

```
ng add @angular/pwa
npm install @angular/service-worker --save
```

Cal afegir un script a l'arxiu `index.html` de l'app que serà l'encarregat de detectar si el navegador admet *service workers*, en cas afirmatiu registrarà l'arxiu `ngsw-workers.js` com a *service worker*:

```
<script>
  if ('serviceWorker' in navigator) {
    navigator.serviceWorker
      .register('/ngsw-worker.js')
      .then(function() {
        console.log("Service Worker Registered");
      });
  }
</script>
```

Es torna a generar un *build* de l'aplicació, per en aquest cas ha de ser per a producció. I finalment es torna a pujar a *Firebase*:

```
ng build --prod
firebase deploy
```

Ara el *service worker* ja es troba operatiu. En realitzar una altra audició amb *Lighthouse* la puntuació relacionada amb les funcionalitats de la PWA és de 100:

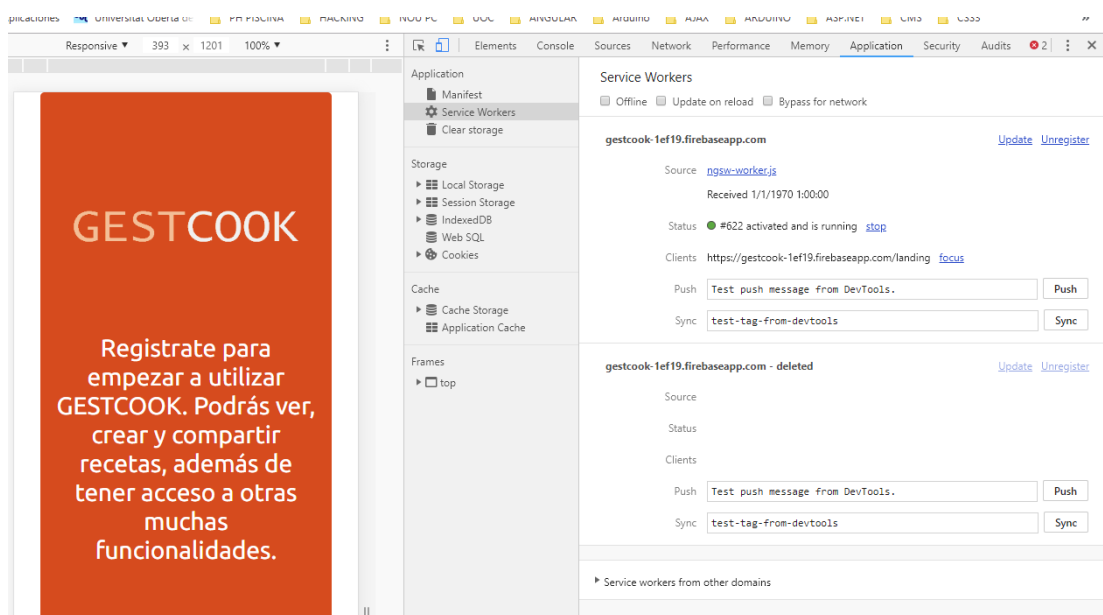


Figura 58: Funcionament dels *service worker*

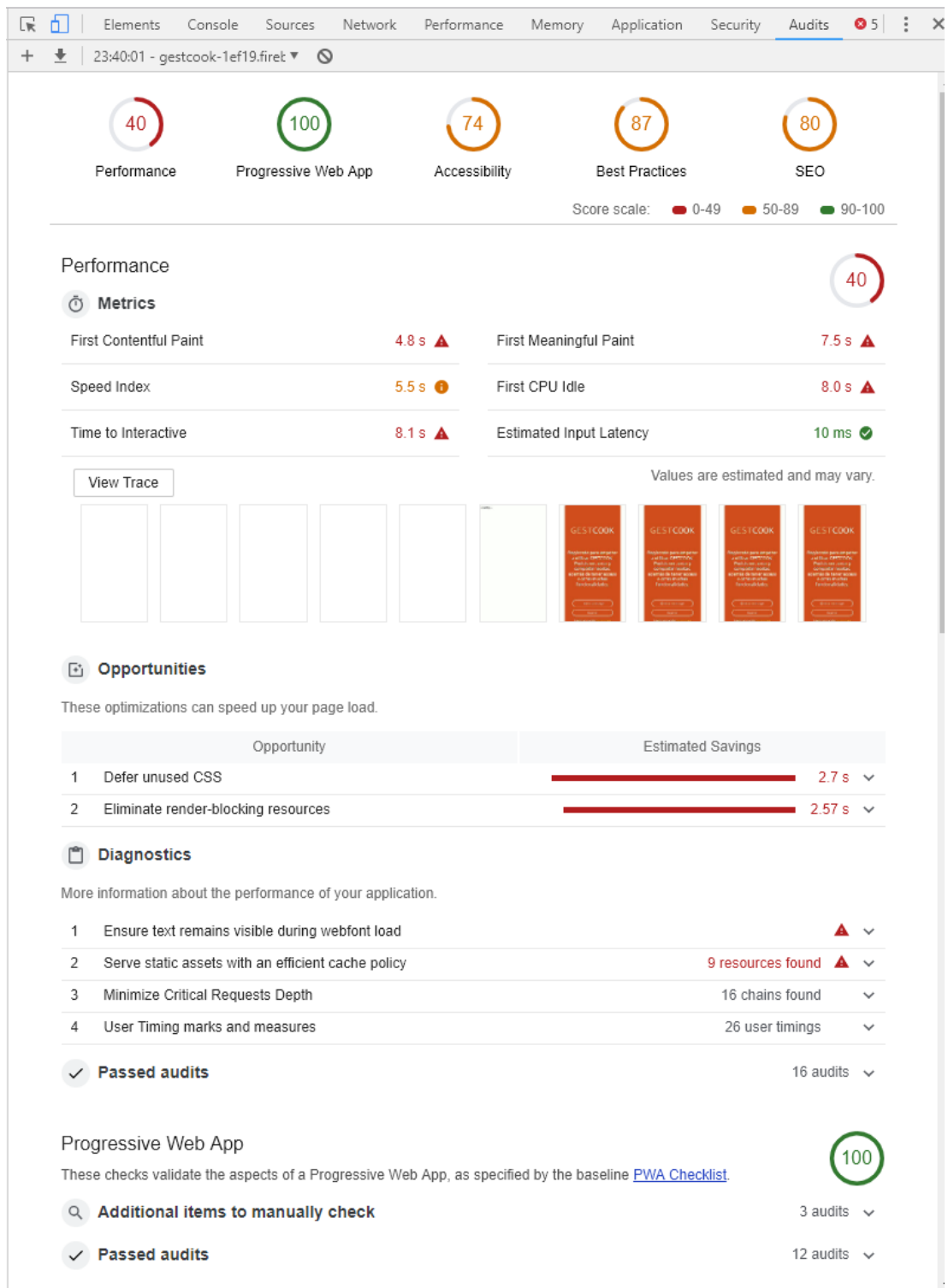


Figura 59: Resultat audició amb puntuació de 100 en PWA

- **Seguretat:** com a mesura principal de seguretat, tots els esforços s'han centrat en que un usuari que no ha iniciat sessió no pugui accedir al contingut a través de les URL. Altrament, una mesura igualment important és que l'usuari registrat pugui editar i eliminar les receptes que ha creat ell mateix.
 - Protecció de rutes: per a la protecció de rutes a Angular es fan servir "Guards". Es crea un arxiu a "guards/auth.ts" a on es protegiran totes les rutes en cas de que l'usuari no hagi iniciat la sessió. En cas contrari l'usuari sempre serà redirigit a la *Landing Page* perquè inici la sessió.

```
ng g g guards/auth
```

```

app-routing.module.ts  app.module.ts  environment.ts  auth.service.ts  auth.guard.ts
1  import { Injectable } from '@angular/core';
2  import { CanActivate, ActivatedRouteSnapshot, RouterStateSnapshot, Router } from '@angular/router';
3  import { Observable } from 'rxjs';
4  import 'rxjs/add/operator/do';
5  import 'rxjs/add/operator/map';
6  import 'rxjs/add/operator/take';
7  import { AuthService } from '../services/auth.service';
8
9
10 @Injectable({
11   providedIn: 'root'
12 })
13 export class AuthGuard implements CanActivate {
14   constructor(
15     private router: Router,
16     private authService: AuthService
17   ) {}
18
19   canActivate(
20     next: ActivatedRouteSnapshot,
21     state: RouterStateSnapshot): Observable<boolean> | Promise<boolean> | boolean {
22     return this.authService.afAuth.authState
23       .take(1)
24       .map(authState => !! authState)
25       .do(logado => {
26         if (!logado) {
27           this.router.navigate(['/landing']);
28         }
29       });
30   }
31 }

```

Figura 60: Protecció de rutes

7. Conclusions i línies de futur

7.1. Conclusions

Conclusions personals sobre el projecte realitzat, el procés de treball i els resultats obtinguts.

- Conclusions del treball:
 - Durant l'estudi i desenvolupament d'aquest TFG s'han après diverses qüestions relacionades amb el que podria ser el desenvolupament d'un projecte real al món laboral i que van des de l'estudi previ, passant per l'estudi econòmic i la planificació, fins al seu desenvolupament i llançament.
 - També s'han après la utilització de noves eines (*Angular* i *Firebase*) i nous llenguatges de programació (*Typescript* i *NoSQL*). Així com la implementació d'una aplicació basada en una plataforma al núvol que proporciona gran diversitat de funcionalitats.
 - El més important ha estat el fet de poder mostrar i fer valer el *know how* que la UOC impregna a cada alumne gràcies a la seva metodologia d'estudi. Gràcies a l'avaluació continuada i a l'esforç per part de l'alumne a l'hora d'haver d'investigar i documentar-se per poder afrontar els diversos reptes que se li plantegen en forma de PAC, aquest bagatge fa que qualsevol estudiant de la UOC pugui afrontar reptes als que no s'havia afrontat amb anterioritat i amb la convicció de que té disposta d'habilitats suficients per a superar-los.
- Reflexió crítica sobre l'assoliment dels objectius plantejats inicialment:
 - Respecte el plantejament inicial i un cop finalitzat el projecte s'ha pogut comprovar que s'ha assolit aproximadament un 80% dels objectius.
 - Alguns dels motius que han causat la impossibilitat d'assolir el 100% dels objectius plantejats han estat:
 - El termini imposat per a la realització del projecte és insuficient si es té en compte el fet de que s'ha de realitzar una proposta de projecte viable, invertir temps en investigació i documentació, aprendre la utilització de noves eines, plataformes i llenguatges de programació, així com el desenvolupament d'una aplicació funcional que compleixi amb els requisits especificats a la proposta.
 - La corba d'aprenentatge del *framework* Angular així com el llenguatge de programació *Typescript*.
 - La utilització de la base de dades *Firestore* en versió beta que proporciona *Firebase*. Aquesta base de dades és molt recent i aquest fet fa que es trobin pocs exemples d'aplicacions d'una certa complexitat.
 - També s'ha de tenir en compte que fins a l'actualitat no es tenien coneixements de *NoSQL*. A aquest fet s'ha de sumar la poca documentació que hi ha entorn a *Firestore* pel que fa a l'hora de treballar amb *arrays* com a dades dels documents.
- Anàlisi crítica del seguiment de la planificació i metodologia al llarg del projecte:
 - Un cop finalitzat el projecte es pot dir que si s'ha seguit la planificació.

- La metodologia prevista basada en *sprints* per aconseguir parts funcionals de l'aplicació ha estat l'adequada.
- No s'han introduït grans canvis, només han estat a nivell d'estructura de bases de dades atesa la complexitat de treballar amb *NoSQL*. Altrament, s'han hagut d'obviar o deixar d'implementar algunes funcionalitats per poder garantir l'èxit del treball. Això és degut a que s'han perdut moltes hores que s'han invertit en aprenentatge i documentació que no s'han pogut destinar a la implementació d'una base de dades d'ingredients ni al servei de compra online dels ingredients.

7.2. Línies de futur

Possibles ampliacions a futur del treball, i/o millores a realitzar en hipotètiques futures versions del producte/servei.

- Atès que no s'ha pogut integrar durant el desenvolupament del projecte, l'objectiu principal seria oferir als usuaris una cerca dinàmica en una base de dades d'aliments. D'aquesta manera tots els usuaris disposarien d'un estàndard a l'hora de confeccionar les seves receptes. Gràcies a aquesta estandardització, la posterior integració amb un servei de compra en línia dels ingredients la faria molt més viable.
- Integració d'un servei de compra en línia dels ingredients en funció de les receptes afegides al menú creat per l'usuari. Gràcies als programes d'afiliació de d'alguns comerços i supermercats, posen a disposició dels desenvolupadors les seves APIs amb la finalitat de poder integrar a un web moltes de les funcionalitats que disposen els seus propis portals.

Cal a dir que aquesta seria una de les propostes més interessants de cares a un futur proper degut a que les ventes realitzades a través de l'aplicació suposarien una font d'ingressos de diners gràcies al percentatge que ofereixen els comerços per cada venda indirecta.

- Altres millores a realitzar venen donades per la varietat d'opcions que ofereix la plataforma *Firebase* amb el servei *Cloud Functions*:
 - Seguiment d'usuaris: un usuari pot seguir a un altre, gràcies a les *Cloud Functions* de *Firebase*. L'usuari pot rebre una notificació quan, per exemple, els usuaris que segueix pugen noves receptes.
 - Enviar un correu electrònic quan els usuaris es subscriuen, es donen de baixa o de benvinguda.
 - Supervisar vocabulari inadequat i imatges de contingut adult o violent. A l'igual que moltes xarxes socials, cada cop s'estan invertint més esforços en la censura de continguts inadequats, com pot ser la pornografia o els insults. En aquest aspecte *Cloud Functions* ofereix eines per eliminar aquest continguts.
 - Redimensionar les imatges que es pugen amb la finalitat d'oferir imatges amb el mateix format i amb un pes reduït.
 - Eliminar comptes inactius de manera periòdica.

Bibliografia

angular.io. One framework. Mobile & desktop.

En línia (setembre 2018)

<https://angular.io>

firebase.google.com. Cómo agregar Firebase a tu proyecto de JavaScript.

En línia (setembre 2018)

<https://angular.io/guide/quickstart>

ahorradoras.com. Planificadores semanales de menús.

En línia (octubre2018)

<https://www.ahorradoras.com/2013/02/planificadores-semanales-de-menús/>

digitaltrends.com. Ahorra tiempo y cuida tu salud con las mejores apps para preparar comida.

En línia (octubre2018)

<https://es.digitaltrends.com/guias/apps-para-planificar-menús/>

code.visualstudio.com. Using Angular in Visual Studio Code.

En línia (novembre 2018)

<https://code.visualstudio.com/docs/nodejs/angular-tutorial>

blog.enriqueoriol.com. Configurar Visual Studio Code para Angular 2.

En línia (novembre 2018)

<http://blog.enriqueoriol.com/2016/06/vscode-angular-2.html>

acadèmia-binaria.com. Páginas y rutas Angular SPA.

En línia (novembre 2018)

<https://academia-binaria.com/paginas-y-rutas-angular-spa/>

codigofacilito.com. Deploy de una aplicación de Angular en Firebase.

En línia (deseembre 2018)

<https://codigofacilito.com/articulos/deploy-angular-firebase>

developers.google.com. Agregado de un Servicio Worker y Trabajo sin conexión a tu app web.

En línia (deseembre 2018)

<https://developers.google.com/web/fundamentals/codelabs/offline/?hl=es>