



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



Escola Tècnica  
Superior d'Enginyeria  
Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica  
Universitat Politècnica de València

# Sistema de gestió i anàlisi estratègica de les competències transversals en Enginyeria Informàtica

TREBALL FI DE MÀSTER

Màster en Enginyeria Informàtica

*Autor:* Carles Torró Segura

*Tutors:* Jose Luís Poza Luján  
J.Alberto Conejero Casares

Curs 2017-2018



# Resum

Actualment, és necessari que l'alumnat de les titulacions universitàries estiga preparat per a la seua carrera laboral. Açò implica que hagen adquirit tant competències tècniques (*hard skills*) com competències transversals (*soft skills*). Perquè les organitzacions puguen seleccionar als titulats de forma adequada, han de conèixer les *soft skills* que requereixen els seus treballadors. Aquest treball presenta una aplicació web que permetrà tant a les empreses com a l'entorn universitari conèixer-les i organitzar-les en forma de mapes de competències. A més a més, l'aplicació web permetrà visualitzar l'evolució de les competències al llarg del temps.

**Paraules clau:** competències tècniques, competències transversals, organitzacions, mapes de competències, universitats

---

# Resumen

Actualmente, es necesario que el alumnado de las titulaciones universitarias esté preparado para su carrera laboral. Esto implica que hayan adquirido tanto competencias técnicas (*hard skills*) como competencias transversales (*soft skills*). Para que las organizaciones puedan seleccionar a los titulados de forma adecuada, deben conocer las *soft skills* que requieren sus trabajadores. Este trabajo presenta una aplicación web que permitirá tanto a las empresas como en el entorno universitario conocerlas y organizarlas en forma de mapas de competencias. Además, la aplicación web permitirá visualizar la evolución de las competencias a lo largo del tiempo.

**Palabras clave:** competencias técnicas, competencias transversales, organizaciones, mapas de competencias, universidades

---

# Abstract

Currently, it is necessary that the students of university degrees be prepared for their work career. This implies that they have acquired both technical skills (*hard skills*) and cross-curricular competences (*soft skills*). For organizations to select graduates appropriately, they must know the soft skills required by their employees. This paper presents a web application that will allow both companies and the university environment to know and organize them in the form of competence maps. Additionally, the web application will allow you to visualize the evolution of the competences over time.

**Keywords:** soft skills, hard skills, skill maps, organizations, universities

---



# Índex

Resum	iii
1 Introducció	1
2 Objecte del projecte	2
3 Antecedents	2
4 Descripció de la situació actual	3
4.1 Sistemes Similars	3
4.1.1 Sistema 1: <i>Skills Base</i>	4
4.1.2 Sistema 2: <i>Skills DB Pro</i>	5
4.1.3 Sistema 3: <i>Avilar</i>	6
4.1.4 Sistema 4: <i>Google Education</i>	6
4.2 Resum de característiques	7
4.2.1 Anàlisi quantitativa	7
4.2.2 Anàlisi qualitativa	8
4.3 Resum dels principals inconvenients i oportunitats identificades	10
4.4 Síntesi	10
5 Normes i referències	11
5.1 Disposicions legals i normes aplicades	11
5.2 Mètodes	11
5.3 Ferramentes	12
5.4 Mecanismes de control de qualitat	12
6 Definicions i abreviacions	14
7 Requeriments inicials	15
7.1 Requisits funcionals	16
7.2 Requisits no funcional	17
8 Abast	18

9	Hipòtesis i restriccions . . . . .	19
10	Estudi d'alternatives i viabilitat . . . . .	19
	10.1 Viabilitat econòmica . . . . .	20
	10.2 Viabilitat tècnica . . . . .	20
	10.3 Viabilitat legal . . . . .	20
11	Descripció de la solució proposada . . . . .	20
12	Anàlisi de riscos . . . . .	21
13	Organització i gestió del projecte . . . . .	23
	13.1 Organització . . . . .	23
	13.1.1 Actors i rols del projecte . . . . .	24
	13.1.2 Interfícies externes . . . . .	24
	13.1.3 Estructura interna . . . . .	25
	13.2 Gestió de projectes . . . . .	27
14	Planificació temporal . . . . .	27
15	Resum del pressupost . . . . .	29
	15.1 Costos <i>hardware</i> . . . . .	30
	15.2 Costos <i>software</i> . . . . .	30
	15.3 Costos de ma d'obra . . . . .	30
	15.4 Costos del producte . . . . .	31
	15.5 Resum de costos . . . . .	31
16	Conclusions i treball futur . . . . .	32
	16.1 Dificultats trobades . . . . .	32
	16.2 Aportacions del projecte . . . . .	32
	16.3 Ampliacions futures . . . . .	33
17	Annexos . . . . .	34
	17.1 Annex - Disseny del sistema . . . . .	34
	17.1.1 Capa de presentació . . . . .	34
	17.1.2 Capa de persistència . . . . .	37
	17.1.3 Capa de negoci . . . . .	39
	17.2 Annex - Implementació . . . . .	40
	17.2.1 Pàgina Web . . . . .	40
	17.2.2 Base de Dades . . . . .	44
	17.2.3 API . . . . .	46
	17.3 Annex - Fase de proves . . . . .	50
	17.3.1 Proves amb la <i>Raspberry Pi 3</i> . . . . .	50
	17.3.2 Proves amb bot per Telegram . . . . .	51

17.3.3 Proves amb Postman . . . . .	52
17.3.4 Proves unitàries . . . . .	53
17.4 Annex - Instal·lació . . . . .	58
17.4.1 Software necessari . . . . .	59
17.4.2 Posada en marxa en local . . . . .	59
17.4.3 Implantació en un server . . . . .	60
18 Estudis amb entitat pròpia . . . . .	60





# Índex de figures

1	<i>Skills Base; Skill Matrix</i> . . . . .	4
2	<i>Skill DB Pro; Dashboard</i> . . . . .	5
3	Google <i>Education</i> ; Gràfics amb resultats . . . . .	7
4	Diagrama de <i>Kiviati</i> . . . . .	9
5	Captura de pantalla del projecte a Github . . . . .	13
6	Esquema CoSki . . . . .	18
7	Esquema conceptual . . . . .	21
8	Interfícies externes . . . . .	25
9	Estructura interna . . . . .	26
10	Validació amb el CoSki21 Team . . . . .	27
11	Captura del taulell de Trello . . . . .	28
12	Diagrama de Gantt . . . . .	29
13	<i>Mockup</i> genèric de la pàgina web . . . . .	34
14	<i>Mockup</i> de la capçalera . . . . .	35
15	<i>Mockup</i> d'un formulari . . . . .	36
16	<i>Mockup</i> visualització d'informació . . . . .	36
17	Diagrama Entitat-Relació . . . . .	37
18	Esquema lògic . . . . .	38
19	Diagrama de seqüència . . . . .	39

20	Estructura projecte AngularJS . . . . .	40
21	Implementació del inici de sessió, registre i estil genèric . . . . .	41
22	Implementació del <i>header</i> . . . . .	42
23	Implementació de formularis . . . . .	42
24	Implementació de finestres encarregades de la visualització . . . . .	43
25	Dades de la Taula <i>Skill_Map</i> . . . . .	45
26	Estructura de l'API . . . . .	46
27	Execució de l'API i visualització de missatges . . . . .	47
28	Contrasenyes xifrades . . . . .	48
29	Esquema de les proves realitzades a la Raspberry Pi . . . . .	50
30	Missatges bot de Telegram . . . . .	51
31	Petició amb Postman . . . . .	52

# Índex de taules

1	Anàlisi quantitativa . . . . .	8
2	Escala <i>Likert</i> . . . . .	8
3	Anàlisi qualitativa . . . . .	9
4	Matriu de valoració de riscos . . . . .	22
5	Avaluació de riscos . . . . .	23
6	Personal involucrat en el projecte . . . . .	24
7	Resum costos <i>Hardware</i> . . . . .	30
8	Resum del pressupost . . . . .	31
9	Bateria de proves RF01 . . . . .	54
10	Bateria de proves RF02 . . . . .	55
11	Bateria de proves RF03 . . . . .	55
12	Bateria de proves RF04 . . . . .	56
13	Bateria de proves RF05 . . . . .	56
14	Bateria de proves RF06 . . . . .	57
15	Bateria de proves RF07 . . . . .	57
16	Bateria de proves RF08 . . . . .	58



# Índex de Fragments de codi

1	Exemple d'una cridada des del servei a l'API . . . . .	43
2	Exemple d'una cridada des d'un funcional al servei . . . . .	43
3	Fragment HTML . . . . .	44
4	Creació de la taula Skill_Map a la BD . . . . .	45
5	Exemple de gestió d'una URL a l'API . . . . .	47
6	Exemple de gestió de la petició realitzada a l'API . . . . .	48
7	Estructura d'un token JWT . . . . .	49
8	Gestió del <i>token</i> JWT a l'API . . . . .	49



## 1 Introducció

Actualment, resulta de gran importància tant per a les organitzacions com a per a les persones, disposar de ferramentes que siguin capaces d'alinejar les competències que tenen les persones, amb les competències que sol·liciten les empreses.

Aquest alineament ajuda molt en les tasques de selecció del personal que duen a terme les organitzacions, i facilita en la mesura del possible, que les persones seleccionades siguin les que disposen de les competències demandades i, per tant, siguin capaces de desenvolupar les tasques que es duen a terme en les organitzacions de forma òptima, aconseguint augmentar la productivitat i l'eficiència de les mateixes.

Generalment, les competències que avaluen la gran majoria de les organitzacions a l'hora de seleccionar al personal són les conegudes com a competències dures, o *hard skills*. Aquestes, són les habilitats referents als coneixements adquirits al llarg del període de formació tant acadèmica com laboral. Però aquestes competències, no són les úniques que determinen la professionalitat d'una persona. És per això que en l'actualitat ja hi ha empreses que requereixen el desenvolupament de les habilitats blanetes, conegudes també com a *soft skills*. Aquestes, són habilitats transversals que fan destacar a les persones com a professionals. Com per exemple, la comunicació, el treball en equip i el lideratge entre moltes altres.

Amb aquest treball es proposa desenvolupar una ferramenta que permeti gestionar i analitzar les *soft skills* que tenen les persones i també les que sol·liciten les organitzacions per tal d'aconseguir l'alineament d'ambdues parts.

Per dur a terme la ferramenta de gestió i anàlisi de *soft skills*, s'utilitzarà principalment el que hem anomenat mapes de competències blanetes (*skill map*) que fan referència a les agrupacions de *soft skills* en funció de criteris concrets. Per exemple l'*skill map* d'una organització estarà compost per les *soft skills* que requereixca l'organització en qüestió.

És important esmentar que el sistema que es va a desenvolupar en aquest treball fi de màster es basa en una part d'un projecte europeu de caire educatiu de la plataforma Erasmus+ anomenat CoSki21 (*Core Skills for 21. Century professionals*) en el qual intervenen diferents universitats i col·laboradors europeus entre els que es troba la UPV. Es detallarà en seccions posteriors els propòsits d'aquest treball.

Finalment, és important destacar que aquest document s'ha redactat seguint les pautes de la norma tècnica per a la realització de documentació de projectes en Enginyeria Informàtica del *Consejo de Colegios de Ingenieros en Informática* (CII), actualment un procés d'estandardització de AENOR basat en la norma UNE CCII-N2016-02.

## 2 Objecte del projecte

L'objectiu principal del projecte és proporcionar una ferramenta de gestió i anàlisi de *soft skills* basada en la web i orientada a facilitar el desenvolupament de mapes de *soft skills* a organitzacions. Aquest objectiu principal es pot estructurar en diferents objectius operacionals, llistats a continuació:

- Analitzar els sistemes similars actuals de gestió d'*skills* per a obtenir suficient informació de les funcionalitats que oferisquen i així, aprofitar les deficiències per a poder esbrinar les característiques desitjables del sistema a desenvolupar.
- A partir d'aquestes característiques extretes citades en l'objectiu anterior, determinar els requisits necessaris del sistema a desenvolupar i decidir quins d'aquests es van a desenvolupar en aquest treball.
- Dissenyar un sistema que acomplisca les funcionalitats determinades en l'objectiu anterior, especialment les orientades a l'anàlisi d'*skills* per a facilitar la creació de mapes personalitzats.
- Implementar el sistema dissenyat i avaluar-lo per a comprovar que gestiona les *skills* i les analitza en funció dels objectius anteriors.

## 3 Antecedents

Com ja s'ha esmentat breument en introducció, la majoria d'organitzacions tradicionalment han seleccionat al personal considerant únicament les competencials dures o *hard skills*. Aquest fet és un error, ja que un professional no es determina únicament per les *hard skills* que té desenvolupades.

Aquest fet pot causar problemes a les organitzacions, ja que una persona que tinga desenvolupades les *hard skills* a un alt nivell, però poques *soft skills*, amb molta seguretat es podrà afirmar que no serà capaç de desenvolupar el treball de la millor manera. Es pot aclarir aquest argument amb un exemple. Una persona amb altes capacitats de programació podrà programar a un alt nivell però si el treball que ha de dur a terme implica tindre desenvolupada una *soft skills* com el treball en equip i aquesta persona no és capaç de fer-ho, no serà la més adient per al treball. Per aquestes raons, les *soft skills* són tan rellevants per un professional.

Actualment, hi ha empreses que sol·liciten disposar de *soft skills* per a poder treballar, ja que com s'ha dit, hi ha moltes tasques professionals que no es poden dur a terme sense haver-les desenvolupat. No tot el personal d'una organització té desenvolupades les *soft skills* necessàries per al treball que duen a terme, per això, moltes de les empreses han començat a facilitar la utilització de ferramentes que



ajuden a desenvolupar-les. Amb això apareixen els primers sistemes informàtics que tracten de desenvolupar, potenciar, gestionar i analitzar les *soft skills*.

A més a més, també és important destacar els antecedents i el perquè naix el projecte europeu en el qual es basa aquest treball de fi de màster. Fa uns anys en el consell assessor del Màster Universitari en Enginyeria Informàtica (MUIINF), es decideix que cal introduir-hi al màster una assignatura relacionada amb les *soft skills*, i aquesta s'inicia al curs 2014/2015. Als dos anys, una universitat Finlandesa contacta amb els encarregats d'aquesta assignatura amb la intenció de dur a terme una tesis doctoral que abordarà les diferències de les *softs skills* en enginyeria entre l'empresa i la universitat. Com a conseqüència d'aquest fet sorgeix la possibilitat de dur a terme un projecte europeu Erasmus+, ja que se n'adonen que aquest tema afecta a totes les universitats i és per aquesta raó per la que comença a desenvolupar-se aquest projecte.

## 4 Descripció de la situació actual

### 4.1 Sistemes Similars

En aquest apartat es realitzarà una anàlisi de la situació actual estudiant els sistemes similars existents, on es descriuran les funcionalitats principals dels mateixos i on es valoraran aspectes tant qualitius com quantitius. D'aquesta manera, es podran observar quines característiques posseeixen les ferramentes actuals i això ens ajudarà a prendre decisions sobre els trets que ha de tindre el nostre sistema.

La gran majoria de les ferramentes informàtiques que es poden trobar en l'actualitat relacionades amb les *soft skills*, estan pensades principalment per a desenvolupar tasques que permeten conèixer el nivell que tenen els membres de l'empresa sobre les *soft skills* que necessiten, i en funció d'aquest nivell, realitzar diferents tasques per tractar de millorar-les.

Com bé s'ha especificat en l'apartat anterior (2) Objecte del projecte, el nostre sistema constarà d'una pàgina web que s'encarregarà de gestionar i analitzar *soft skills*. És per això que els sistemes que s'han d'analitzar són pàgines web amb finalitats similars.

### 4.1.1 Sistema 1: Skills Base

*Skills Base* és un software de gestió d'*skills* basat en la web. El seu principal objectiu és augmentar la capacitat de competències de l'organització que l'utilitza fent una avaluació de les mateixes. Posseïx una gran quantitat d'opcions, però el seu funcionament es basa principalment en els següents 4 passos:

- **Creació del compte:** per a l'organització que vol utilitzar el sistema.
- **Afegir *skills*:** sobre les quals es vol dur un seguiment i s'avaluaran per conèixer el seu nivell.
- **Invitació:** dels treballadors per a validar les *skills* introduïdes prèviament.
- **Observar resultats:** en forma de gràfiques i matrius d'*skills*. També realitza *clustering* per agrupar persones amb *skills* semblants i les *skills* més valorades.

**Accounting Skills**

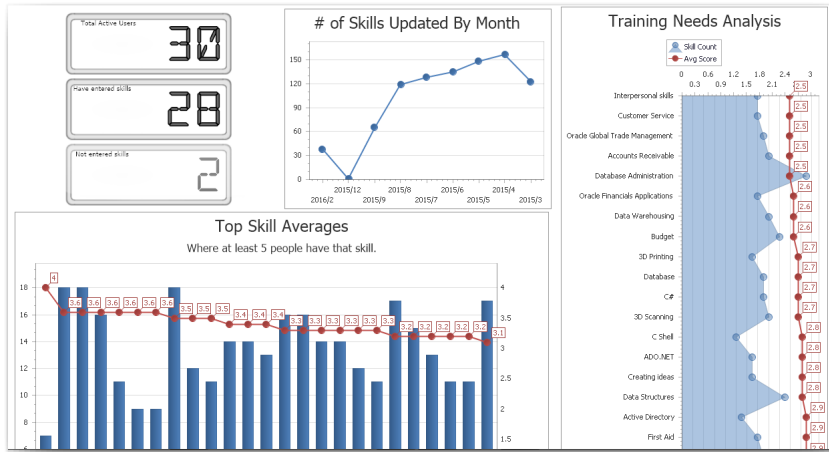
	Average cost method	Bank reconciliation	Bookkeeping	Creative accounting	Equity method	General ledger	Imprest system	Lease accounting	Purchase ledger	Purchasing	Stock and flow	Tax accounting	Unlimited ledger
Christina Rosenthal	2.00	1.50	2.00	3.50	2.00	3.00	1.00	1.00	2.00	1.00	2.00	3.00	3.00
Donald Woods	2.50	2.00	2.00	2.50	3.00	3.00	3.00	2.00	1.50	1.00	3.50	3.00	3.00
Dwight Hoyle	2.50	1.50	3.50	2.50	3.00	4.00	3.00	3.50	2.00	1.00	2.50	4.00	3.50
Erica Bowden	2.50	2.00	4.00	4.00	3.50	5.00	4.00	3.50	3.00	2.00	3.00	3.50	3.50
Gretchen Joseph	2.50	1.50	3.00	3.50	3.50	4.00	3.50	3.00	2.00	2.00	3.50	3.00	3.00
Hazel Raynor	2.00	1.50	1.00	2.50	2.50	3.50	3.00	3.00	1.00	1.00	2.50	2.50	2.50
Jean Griffin	4.00	1.50	3.00	3.50	3.50	3.50	1.00	3.50	4.50	1.50	4.50	5.00	4.00
John Supervisor	2.00	1.00	3.00	3.00	3.00	5.00	4.00	4.00	3.00	1.00	4.00	5.00	4.00
Karen Hamilton	2.50	1.50	4.00	3.00	3.00	4.50	2.50	4.50	4.00	1.50	2.50	4.00	3.50
Louis Mangum	3.00	1.50	2.50	2.00	2.50	4.50	1.00	2.00	1.50	2.50	3.50	3.00	3.00
Neal Lawrence	2.50	1.50	3.00	3.00	4.00	4.50	3.00	3.50	2.50	1.50	3.00	2.00	3.00
Stephanie Baker	3.50	1.50	3.00	2.50	3.00	3.50	4.50	4.00	4.00	2.50	3.50	3.50	2.00
Wesley Grant	3.00	1.50	3.00	3.00	3.00	4.50	3.00	3.50	1.50	3.50	3.00	3.00	3.00

**Figura 1:** *Skills Base; Skill Matrix*  
**Font:** <https://www.skills-base.com/>

El seu preu varia en funció dels membres de l'empresa que vulguen fer servir la ferramenta, sent gratuït fins a de 10 treballadors. Es fa una diferenciació de rols i, a més a més, presenta una versió de prova per a mostrar la ferramenta.

#### 4.1.2 Sistema 2: Skills DB Pro

L'objectiu principal de *Skills DB Pro* és tractar d'alinejar les *skills* dels treballadors d'una empresa amb els objectius de la mateixa. Amb la utilització d'aquesta ferramenta, es podran conèixer tant les *skills* necessàries per a treballar en la companyia com les *skills* que els treballadors necessiten entrenar. Tot açò es podrà visualitzar amb gràfiques i taules com les que mostra la següent captura.



**Figura 2:** *Skill DB Pro*; Dashboard  
Font: <https://skillsdbpro.com/>

Es pot visualitzar informació referent al número d'usuaris que estan fent ús de la ferramenta. També s'observa un registre de les *skills* més desenvolupades i, com s'ha dit anteriorment, gràfics amb informació sobre les *skills* que fa falta millorar. Respecte a aquestes últimes, el sistema permet realitzar tasques per tractar de millorar-les. Tot queda registrat en el sistema visualitzant d'aquesta manera el progrés de cada treballador.

Finalment, és important destacar que el sistema és de pagament però presenta una versió de prova i, a més a més, fa una diferenciació de rols en funció del càrrec dintre de l'empresa.

### 4.1.3 Sistema 3: Avilar

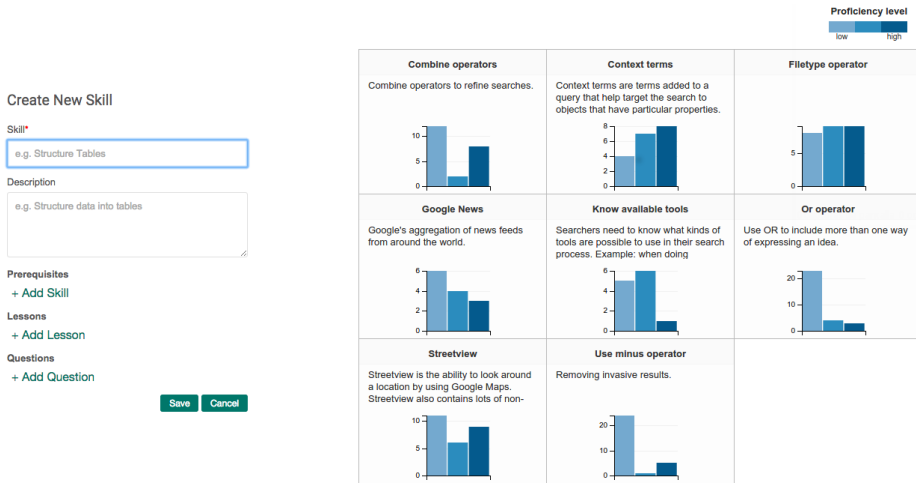
Avilar es presenta com a líder en gestió de competències basades en web com a conseqüència de la gran quantitat de clients que posseïxen. La utilització d'aquesta plataforma permet iniciar un pla d'anàlisi de rendiment avaluant les competències de la plantilla de l'empresa que el fa servir. Aquest pla està format pels següents passos:

- Identificar les llacunes en les competències.
- Optimitzar la força laboral.
- Complir amb els requisits reglamentaris.
- Alinear la formació amb objectius corporatius.
- Realitzar un inventari de capacitats.
- Avaluació d'empleats i avaluació del rendiment.

De la mateixa manera que els sistemes anteriors, aquest també presenta una diferenciació de rols i permet provar el sistema abans de pagar per la seua utilització.

### 4.1.4 Sistema 4: Google Education

Aquesta ferramenta de Google, inclosa dintre de Google *Education*, està pensada per a la creació de cursos amb l'objectiu d'ensenyar una sèrie d'*skills* als participants. La ferramenta permet crear, modificar i esborrar *skills* i relacionar-les amb el seu contingut. Es poden mostrar les *skills* als participants del curs per donar-los una perspectiva dels continguts i les competències que adquiriran.



**Figura 3:** Google *Education*; Gràfics amb resultats  
**Font:** <https://edu.google.com/>

Es poden observar grafs d'*skills* que mostren relacions i dependències entre les mateixes. A més a més, per a la visualització de resultats els participants dels cursos poden observar els seus progressos i estadístiques.

## 4.2 Resum de característiques

Tant per a l'anàlisi qualitativa com quantitativa s'utilitzaran taules que valoraran les seues característiques. L'anàlisi qualitativa es farà de forma objectiva mentre que l'anàlisi quantitativa es farà valorant de forma subjectiva els sistemes estudiats. Per a les dues anàlisis es descriurà el significat de les columnes que s'utilitzaran a les taules. En ambdós casos la columna més a l'esquerra farà referència al sistema a valorar.

### 4.2.1 Anàlisi quantitativa

Les columnes de la taula per a l'anàlisi quantitativa seran les següents:

- **Registre:** Autenticar-se en l'aplicació per a poder utilitzar-la.
- **Rols:** Diferenciació de diferents perfils dintre del sistema.
- **Preu:** Saber si el sistema és gratuït o no.

- **€/y**: Cost econòmic anual mitjà per la utilització del sistema.
- **Skill Map**: Saber si la ferramenta gestiona les *skills* utilitzant *skills maps*.
- **Clustering**: Saber si la ferramenta fa una agrupació d'*skills* seguint algun criteri concret.

Anàlisi quantitativa						
	Registre	Rols	Preu	€/y	Skill Map	Clustering
<i>Skills Base</i>	Si	Si	Pagament	600€	No	Si
<i>Skills DB Pro</i>	Si	Si	Pagament	240€	No	No
<i>Avilar</i>	Si	Si	Pagament	No Info	No	No
<i>Google education</i>	Google	No	Gratuit	0€	Si	No

Taula 1: Anàlisi quantitativa

#### 4.2.2 Anàlisi qualitativa

Al tractar-se d'aplicacions web, aquestes s'han pogut provar en la mesura del possible, i per tant, es poden valorar subjectivament. Per a valorar les característiques numèricament s'utilitzarà una escala de *Likert* amb 5 nivells. Açò facilitarà poder realitzar comparatives entre els diferents sistemes analitzats utilitzant diagrames de *Kiviat*.

Gens adequat	Poc adequat	Normal	Adequat	Molt adequat
1	2	3	4	5

Taula 2: Escala *Likert*

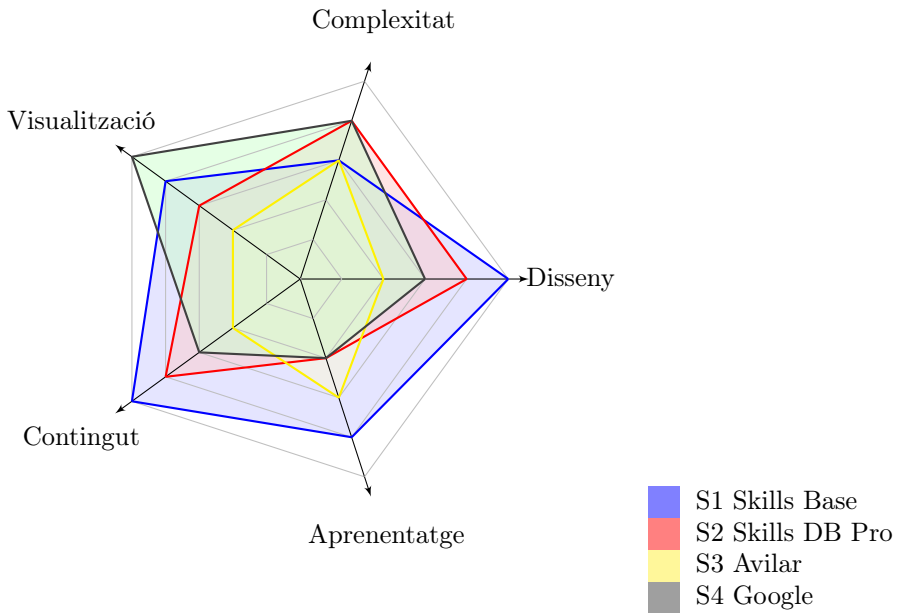
Les columnes de la taula per a l'anàlisi qualitativa seran les següents:

- **Disseny**: Aspecte i estètica de la ferramenta.
- **Complexitat**: Esforç requerit per a utilitzar el sistema correctament.
- **Visualització**: Adequació de la visualització de la informació.
- **Contingut**: Adequació del contingut en comparació amb els propòsits d'aquest treball.
- **Aprenentatge**: Corba d'aprenentatge de la ferramenta.

	Anàlisi qualitativa				
	Disseny	Complexitat	Visualització	Contingut	Aprenentatge
<i>Skills Base</i>	5	3	4	5	4
<i>Skills DB Pro</i>	4	4	3	4	2
<i>Avilar</i>	2	3	2	2	3
<i>Google Education</i>	3	4	5	3	2

Taula 3: Anàlisi qualitativa

Amb el següent diagrama de *Kiviat* es pot observar la valoració dels sistemes de manera sobreposada. Les diferents arestes corresponen a les característiques de la taula anterior. L'altura on cada àrea talla a les diferents arestes correspon als valors de la taula. Per tant, el sistema que tinga una major àrea serà el més complet.

Figura 4: Diagrama de *Kiviat*

Analizant el diagrama anterior podem concloure que dels diferents sistemes analitzats el més complet és *Skills Base* tot i que existeixen sistemes amb una complexitat menor i amb visualitzacions lleugerament millors, la resta de característiques

són les més completes. A més a més, és de totes les ferramentes trobades la que té uns propòsits similars als quals es pretén abordar en aquest treball.

### 4.3 Resum dels principals inconvenients i oportunitats identificades

Amb l'anàlisi de diversos sistemes s'ha arribat a la conclusió que la principal deficiència que tenen és que la gran majoria disposen d'una gran varietat d'opcions i, per tant, presenten una corba d'aprenentatge poc empinada.

D'altra banda, molts dels sistemes són de pagament i orientats a empresa, per tant la versió de prova que ofereixen és personalitzada, és a dir, no deixen provar el sistema sense previ registre i contacte amb l'empresa.

Finalment, la gran majoria d'aquest sistemes no fa un gestió de les *soft skills* utilitzant *skill maps*.

### 4.4 Síntesi

Una vegada analitzats els sistemes i tenint clars els objectius del projecte, podem establir les característiques ideals que haurà de tindre el nostre sistema.

Característiques:

- **CA01:** El sistema a desenvolupar serà una aplicació Web amb interfície de pàgina web adaptable.
- **CA02:** Es requerirà un registre i autenticació per a utilitzar la ferramenta que complisca els requisits legals.
- **CA03:** Hi haurà una diferenciació de rols per a accedir al sistema, però la funcionalitat del sistema serà idèntica per a tots els rols.
- **CA04:** S'utilitzaran formularis per a la gestió d'*skills* i de mapes d'*skills*.
- **CA05:** L'anàlisi d'*skills* i de mapes d'*skills* es podrà realitzar en funció del temps i sols es podran fer modificacions de l'any actual.
- **CA06:** El sistema podrà incorporar mapes d'*skills* propis però també mapes creats per altres entitats, i només es podran realitzar modificacions sobre els mapes propis.
- **CA07:** Les dades s'emmagatzemaran de forma persistent a una BD utilitzant una API comú per poder accedir-hi.



## 5 Normes i referències

### 5.1 Disposicions legals i normes aplicades

En aquest apartat, es mostraran les normatives aplicades al llarg del desenvolupament d'aquest projecte. Al tractar-se d'un projecte d'enginyeria del software, on es realitza un desenvolupament *full stack* d'una pàgina web, les normatives més destacades que s'han d'acomplir són les següents:

- ***Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal.***: Com bé s'ha esmentat en la característica CA02 de l'apartat anterior (4.4) Síntesi, hi haurà un registre i autenticació a la web, és per això que aquesta norma s'ha de complir.
- ***Real Decreto Legislativo 1/1996, de 12 de abril, de la Ley de Propiedad Intelectual.***: La pàgina web es desenvoluparà respectant la propietat intel·lectual i evitant plagis, fent ús de la documentació corresponent a la tecnologia que es farà servir.

Més informació: <http://www.boe.es>

### 5.2 Mètodes

El procediment que s'ha dut a cap per al desenvolupament d'aquest projecte consta de:

1. Estimació del temps i dels recursos necessari per al desenvolupament del projecte.
2. Recerca i estudi del diferents sistemes similars existents per realitzar posteriors anàlisis dels mateixos que ens ajudaran a determinar millor el nostre sistema.
3. Disseny i implementació de la solució proposada. Utilitzant les fonts necessàries per documentar-se i poder realitzar la implementació de la forma més correcta possible.
4. Implantació i *testing* i manteniment del sistema.
5. Explotació i comercialització.

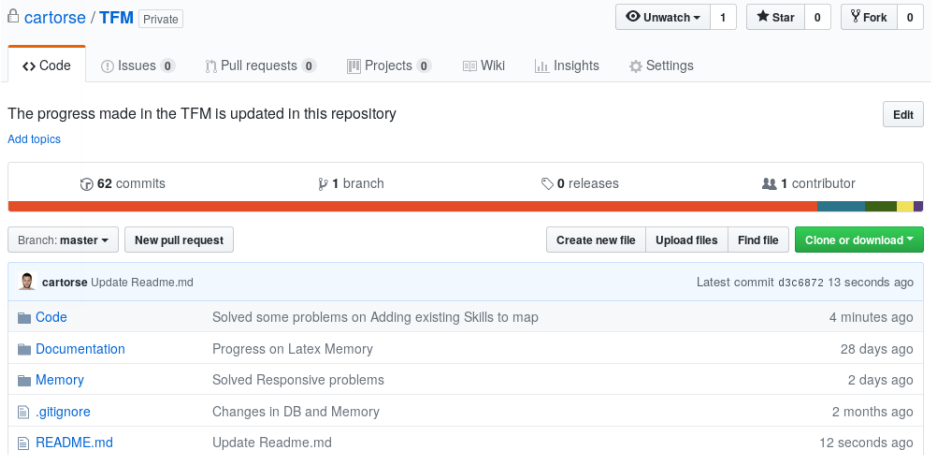
### 5.3 Ferramentes

En base a les característiques determinades anteriorment a l'apartat (4.4) Síntesi, i tenint com a objectiu principal desenvolupar un sistema de gestió i anàlisi de *soft skills* utilitzant tecnologies web, es poden determinar les ferramentes que s'utilitzaran per a abordar el projecte.

- **Angular4 JS:** És un *framework frontend* desenvolupat amb *TypeScript* per a la creació d'aplicacions web, mantingut per Google, i que s'utilitza principalment per a crear aplicacions web dinàmiques i d'una sola pàgina. Angular inclou HTML I CSS per a dissenyar les vistes, i per a aquest sistema s'ha utilitzat també *Bootstrap* amb l'objectiu de poder emprar elements de disseny predefinitos.
- **ExpressJS:** És un *framework* desenvolupat amb *NodeJS* per a la creació aplicacions WEB i APIs. En el nostre sistema s'utilitza *ExpressJS* per a crear una API Rest amb capacitat per a comunicar l'aplicació web amb la BD mitjançant peticions *https*.
- **MySQL:** És un sistema de gestió de bases de dades (SGBD) relacional i que utilitza el llenguatge SQL. En el nostre sistema s'encarregara de la persistència de les dades de la pàgina web.
- **Apache:** Apache es un servidor HTTP de codi obert i multiplataforma. Al nostre projecte s'utilitzarà per a servir la pàgina web i estarà instal·lat a una *Raspberry Pi 3* la qual allotjarà la pàgina, permetent d'aquesta manera poder accedir a la web des de l'exterior i aconseguir un entorn més real.
- **Postman:** Postman és una ferramenta que permet construir i gestionar de forma còmoda les peticions als serveis REST. La ferramenta és capaç de realitzar peticions de diferents tipus, capturar les respostes i mostrar-les de forma clara ordenada i en diferents formats. Al nostre sistema el farem servir per a comprovar l'estat dels *endpoints* i validar un correcte funcionament de l'API.

### 5.4 Mecanismes de control de qualitat

Al tractar-se d'un projecte d'enginyeria del software, s'ha utilitzat el sistema de control de versions Git amb la plataforma GitHub, d'aquesta manera s'ha tingut un control del desenvolupament del projecte facilitant la recuperació del codi front a fallades.



**Figura 5:** Captura de pantalla del projecte a Github

També amb Git s'han pogut etiquetar 3 versions sobre el sistema a desenvolupar, cada una d'aquestes versions, ha sigut provada i, per tant, lliure d'errors i consegüentment estaria llesta per a mostrar al client, aconseguint d'aquesta manera un *feedback* per la seua part. Les versions en qüestió són les següents.

- **Versió 1.0:** Gestió de *soft skills* i *skills maps* bàsica.
- **Versió 2.0:** Gestió de *soft skills* i *skills maps* mitjana, incloent *merging* (més informació sobre *merging* apartat (6) Definicions i abreviacions).
- **Versió 3.0:** Gestió de *soft skills* i *skills maps* avançat, incloent generació automàtica amb *clustering* (més informació sobre *clustering* apartat (6) Definicions i abreviacions).

D'altra banda, el codi s'ha escrit tractant de respectar al màxim els estàndards i les bones pràctiques de programació corresponents a les tecnologies especificades en l'apartat (5.3) Ferramentes.

Finalment, s'ha utilitzat Trello per a gestionar les tasques plantejades a les reunions amb el tutor i portar un millor control sobre el que s'ha de desenvolupar i els terminis que s'han d'acomplir. Per a més informació sobre la gestió realitzada amb Trello, consultar l'apartat (13.2) Gestió de projectes.

## 6 Definicions i abreviacions

En aquest apartat es definiran els conceptes i acrònims relacionats amb aquest treball per tal de aclarir el seu significat. Es deixaran de banda les explicacions sobre conceptes tecnològics que s'assumeixen com a sabuts o en cas que no siga així, es poden consultar en fonts externes.

- Definicions:
  - **Sistema:** Tot el desenvolupament d'aquest projecte. També fa referència a les ferramentes tecnològiques que s'han utilitzat i/o consultat.
  - **Hard Skill:** Fa referència a les habilitats referents als coneixements adquirits al llarg del període de formació tant acadèmica com laboral (Exemple: saber programar en el llenguatge de programació Java).
  - **Soft Skill:** Fa referència a les habilitats transversals que fan destacar a les persones com a professionals. Al llarg del treball també es trobaran esmentades com a *skills* o competències (Exemple: Treball en equip, comunicació, lideratge ...).
  - **Skill Map:** Conjunt de *soft skills* agrupades per a finalitats concretes (Exemple: conjunt de *soft skills* d'una organització).
  - **Skill Intensity:** Percentatge assignat a una *soft skill* al llarg d'un any, reflectant la importància de la *soft skill* en eixe període.
  - **Bloqued Skill Map:** Un *skill map* que no podrà ser modificat per cap usuari al nostre sistema.
  - **Merged Soft Skill:** Un *soft skill* que s'ha creat en base a dos o més *soft skills* diferents. Aquest procediment també es referència al llarg del treball com *Merging*.
  - **Skill Cluster:** Agrupació de *soft skills* per percentatge de similitud. Aquest procediment també es referència al llarg del treball com *Clustering*.
  - **Missió d'una empresa** Defineix la activitat de mercat d'una empresa.
  - **Visió d'una empresa:** Són les metes que pretén assolir una empresa.
  - **Valors d'una empresa** Són els valors ètics que segueix una empresa.
  - **Recruiters:** Persona encarregada de la selecció de personal en una organització o empresa.

- **Stakeholders:** Fa referència a les parts interessades en el projecte, és a dir les persones o organitzacions que es veuen afectades per les decisions que es prenen en el projecte.
  - **API Endpoint:** És tracta d'una URL que pot rebre paràmetres i indica on es trobaran les dades corresponents a una petició.
  - **Targeta:** Fa referència a una tasca dins de Trello, té propietari, data de venciment i descripció.
  - **Sprint:** Fa referència a l'execució d'una iteració en projecte que utilitza metodologies àgils.
  - **Organització:** Entitat, empresa, companyia, indústria tant se val si és gran, mitjana, xicoteta i pública o privada. S'utilitza indistintament el terme organització o empresa.
  - **Competències:** Esmentat en la definició anterior de *Soft Skills*.
  - **Ferramenta:** Esmentat en la definició anterior de Sistema.
- Acrònims:
    - **BD:** Fa referència a la base de dades (BD) del nostre sistema i es desenvoluparà en MySQL.
    - **API:** Fa referència a l'API REST del nostre sistema i es desenvoluparà en Express.
    - **CEO:** Fa referència al director executiu d'una organització.
    - **CRUD:** Fa referència a les operacions permeses en una base de dades. És l'acrònim de *Create, Read, Update, Delete*.

## 7 Requeriments inicials

Després d'haver realitzat l'anàlisi de la situació actual (4.1) Sistemes Similars i una vegada determinades les característiques que hauria de disposar el nostre sistema especificades en l'apartat (4.4) Síntesi, és el moment d'establir els requisits que s'han d'acomplir en aquest projecte. Açò ens ajudarà a poder fer un disseny més complet. S'han dividit els requisits en dos tipus, es poden observar a continuació:

- **Funcionals:** Defineix una funció o component del sistema. S'expressaran amb el codi RFxx sent xx el número del requisit en qüestió. Al tractar-se

d'un sistema que gestiona i analitza *soft skills*, es diferenciaran els requisits funcionals en requisits de gestió, requisits d'anàlisi i altres requisits.

- **No funcionals:** Característiques de funcionament i atributs de qualitat. S'expressaran amb el codi RNFxx sent xx el número del requisit en qüestió.

## 7.1 Requisits funcionals

### Requisits funcionals de gestió

- **RF01 Gestió d'usuaris:** Un usuari estarà format per: nom, cognoms, correu electrònic, nom d'usuari, contrasenya, tipus d'usuari, gènere i podrà tindre una foto de perfil. La gestió d'usuaris permetrà crear-los, modificar els camps desitjats i esborrar-los.
- **RF02 Gestió de *skill maps*:** Els *skill maps*, tindran un nom, una descripció, una url, una organització, un any i una font indicant d'on provenen. La gestió de *skill maps* permetrà crear-los, modificar els camps desitjats, llistar-los, esborrar-los i crear mapes a partir d'altres. Aquesta gestió únicament es durà a terme si l'usuari és l'autor de la *skill map* i no es tracta d'un *skill map* bloquejat.
- **RF03 Gestió de *soft skills*:** Les *soft skills* tindran un nom, una descripció i aniran associades a un *skill map*. La gestió de les *soft skills* permetrà crear-les, modificar els camps desitjats, llistar-les i esborrar-les. Aquesta gestió únicament es durà a terme si l'usuari és l'autor del *skill map* al que pertany la *soft skill* i no pertanyen a un *skill map* bloquejat.

### Requisits funcionals d'anàlisi

- **RF04 Anàlisi d'intensitat:** es podran realitzar anàlisis de l'evolució de la intensitat de skills en funció del temps sempre i quan no pertanyen a mapes bloquejats. (definició de *skill intensity* a l'apartat anterior (6) Definicions i abreviacions).
- **RF05 *Merging* de *soft skills*:** es podrà fer un *merge* de dos o més *soft skills* que siguin semblants i s'afegiran a un *skill map* que no estiga bloquejat (definició de *merging* a l'apartat anterior (6) Definicions i abreviacions).
- **RF06 *Clustering* de *soft skills*:** es podran fer un *clustering* (agrupació) de les *soft skills* per percentatge de similitud (definició de *clustering* a l'apartat anterior (6) Definicions i abreviacions).
- **RF07 *Skill Map* amb PDF:** Es podrà generar i descarregar un *Skill Map* seleccionat amb format PDF sempre que aquest siga propi. El PDF mostrarà

informació de l'usuari del *skill map* seleccionat i de les *soft skills* associades al mapa seleccionat.

### Altres Requisits funcionals

- **RF08 Sessió:** Una vegada iniciada la sessió en la web, aquesta serà vàlida durant 1 hora, a més a més es visualitzarà un comptador amb el temps restant.
- **RF09 Informació:** Hi haurà una finestra amb informació referent al projecte.
- **RF10 Formularis:** Els requisits funcionals de gestió (RF01, RF02, RF03) s'administraran amb formularis que tindran camps obligatoris i camps opcionals, només es podran enviar els formularis si estan tots els camps obligatoris complets correctament.
- **RF11 Alertes:** Els formularis mostraran missatges satisfactoris i/o d'error en funció del resultat de l'enviament del formulari. També es mostraran alertes quan no s'ompliguin els camps obligatoris i, també quan es causen fallades als serveis.
- **RF12 Notificacions:** S'enviaran notificacions informatives per correu electrònic tant si un usuari es dona d'alta com si es dona de baixa de la ferramenta.

### 7.2 Requisits no funcional

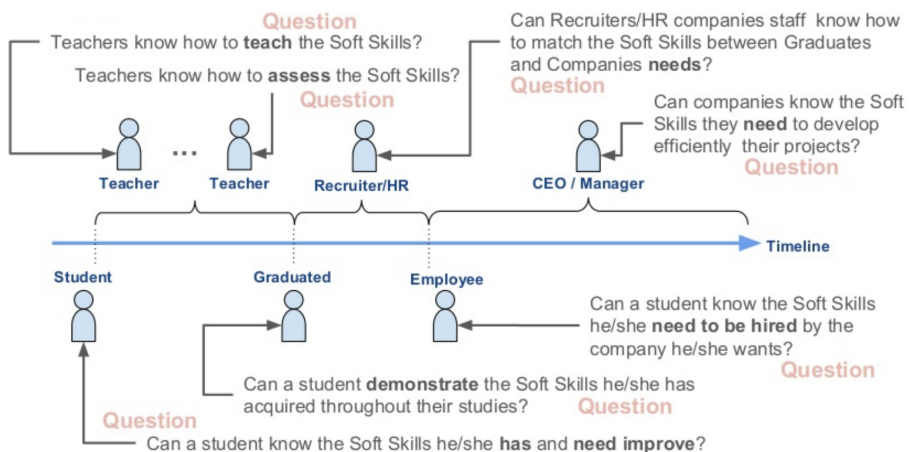
- **RNF01 Seguretat:** Les comunicacions entre *frontend* i *backend* es realitzaran amb *https* per garantir la seguretat. A més a més, només els usuaris autoritzats poden fer ús de la web i les contrasenyes dels usuaris que es registren s'emmagatzemen xifrades.
- **RNF02 Usabilitat:** La web tractarà de ser el més fàcil d'utilitzar possible amb una interfície senzilla i amigable. És per això que la interacció de l'usuari amb la web no serà complicada.
- **RNF03 Robustesa:** Es garantirà la robustesa del sistema gestionant els errors que puguen ocórrer amb missatges informatius i a més a més es controlaran els formularis amb la intenció de validar l'enviament de dades.
- **RNF03 Adaptabilitat:** La pàgina web s'adaptarà a diferents grandàries de pantalla per a ser el més *responsive* possible. Conseqüentment es podrà visualitzar correctament als dispositius mòbils.

- **RNF04 Persistència:** Les dades utilitzades a la web estaran sempre en relació amb la BD associada i seran consistents en tot moment.

## 8 Abast

El present treball de fi de màster, com ja s'ha dit a la (1) Introducció, i s'ha contextualitzat a l'apartat (3) Antecedents, s'inspira en un projecte europeu anomenat CoSki21, és a dir, s'agafa la idea d'una part d'aquest projecte per a desenvolupar el treball de fi de màster.

L'objectiu del projecte CoSki21, a grans trets, és tractar d'aconseguir que les universitats produïsquen graduats i professionals equipats en les competències i *soft skills* necessàries per accedir al món laboral el més preparats possible. Per tractar d'entendre com s'enfoca l'objectiu d'aquest projecte a continuació es pot observar un esquema temporal amb les preguntes que el projecte pretén abordar.



**Figura 6:** Esquema CoSki

Per trobar més informació sobre el projecte CoSki21 es pot consultar el següent enllaç:

<https://www.slideshare.net/jopolu/coski21-project-overview>.



## 9 Hipòtesis i restriccions

A pesar que el projecte no ha variat en gran mesura des de la proposta del mateix fins al final, si que es cert que les hipòtesis i punts de partida inicials han variat lleugerament a mesura que s'ha desenvolupat el projecte, donant lloc al sistema complet que s'especifica en l'apartat (11) Descripció de la solució proposada. Els factors més importants que han anat canviant han sigut sobretot aspectes funcionals.

Al tractar-se d'un projecte inspirat en un projecte europeu en el qual intervenen diferents col·laboradors, el desenvolupament d'algunes de les funcionalitats del sistema, s'han hagut de consultar per tractar de respectar-les o abordar-les de la forma més similar possible.

Com a conseqüència d'aquest fet, alguns requisits funcionals s'han anat modificant a mesura que s'ha anat implementant el projecte i en alguns casos s'han hagut de re-implementar xicotetes funcionalitats, com per exemple l'aparició de mapes bloquejats (*blocked*) especificada a l'apartat (7) Requeriments inicials. Inicialment la gestió d'*skill maps* no contemplava aquesta opció, és a dir, no existien els mapes bloquejats però finalment es va decidir afegir aquesta funcionalitat per a aconseguir que hi hagen mapes que no puguin ser modificats.

D'altra banda, la principal restricció d'aquest projecte és la gestió que es fa sobre la diferenciació de rols. Com bé s'ha esmentat al llarg del projecte i ocorre en alguns dels sistemes similars estudiats anteriorment, es fa una diferenciació de rols, i cadascun d'ells pot desenvolupar una funció diferent dintre del sistema. Amb aquesta ferramenta, es diferenciarà el rol però la funcionalitat que desenvoluparà per a cada rol serà idèntica, com ja s'ha determinat a les característiques de l'apartat (4.4) Síntesi.

## 10 Estudi d'alternatives i viabilitat

Pel que respecta a les alternatives, en l'apartat (4.1) Sistemes Similars s'han descrit i analitzat diferents sistemes similars. Aquests sistemes comparteixen el fet que gestionen *soft skills*, en alguns casos fins i tot comparteixen funcionalitats, però baix ningun concepte es poden assolir els objectius proposats per a aquest projecte utilitzant aquests sistemes. És per això que es descarten les alternatives estudiades i sorgeix la necessitat de desenvolupar aquest treball.

Una vegada descartades les alternatives estudiades i tenint la necessitat de desenvolupar un projecte per tal d'assolir els objectius proposats, s'estudiarà la viabilitat del mateix on es consideraran aspectes econòmics, tècnics i també legals.

### 10.1 Viabilitat econòmica

A l'apartat (15) Resum del pressupost, s'ha especificat el pressupost econòmic detallat tant per al *software*, com per al *hardware* i també els costos de mà d'obra. Al tractar-se d'un projecte emmarcat dins d'un treball de fi de màster, és viable econòmicament, ja que els costos de mà d'obra no suposen un cost real a pesar d'haver-se calculat. És per això que aquest projecte és viable des del punt de vista econòmic.

### 10.2 Viabilitat tècnica

A l'apartat (7) Requeriments inicials, s'han especificat els requisits funcionals del sistema. Aquest requisits es poden desenvolupar utilitzant diferents ferramentes tecnològiques dintre del món de la informàtica i, a més a més, aquest projecte s'ha desenvolupat per un estudiant de màster en enginyeria informàtica. És per això que al disposar tant de la tecnologia com del personal, aquest projecte és viable a nivell tècnic.

### 10.3 Viabilitat legal

A l'apartat (5) Normes i referències, s'han especificat les normatives legals que s'han de complir en aquest projecte. A més a més, el projecte realitza connexions segures i encripta les dades confidencials de la base de dades. Per aquesta raó, el projecte és viable des del punt de vista de la legalitat.

## 11 Descripció de la solució proposada

Amb la intenció de facilitar la comprensió de la solució proposada, s'ha dissenyat un esquema conceptual. Aquest, fa una separació entre els components que formen el *Frontend* i el *Backend* que integraran la web. A més a més, mostra les tecnologies que s'han utilitzat per a la implementació, especificades anteriorment a l'apartat (5.3) Ferramentes.

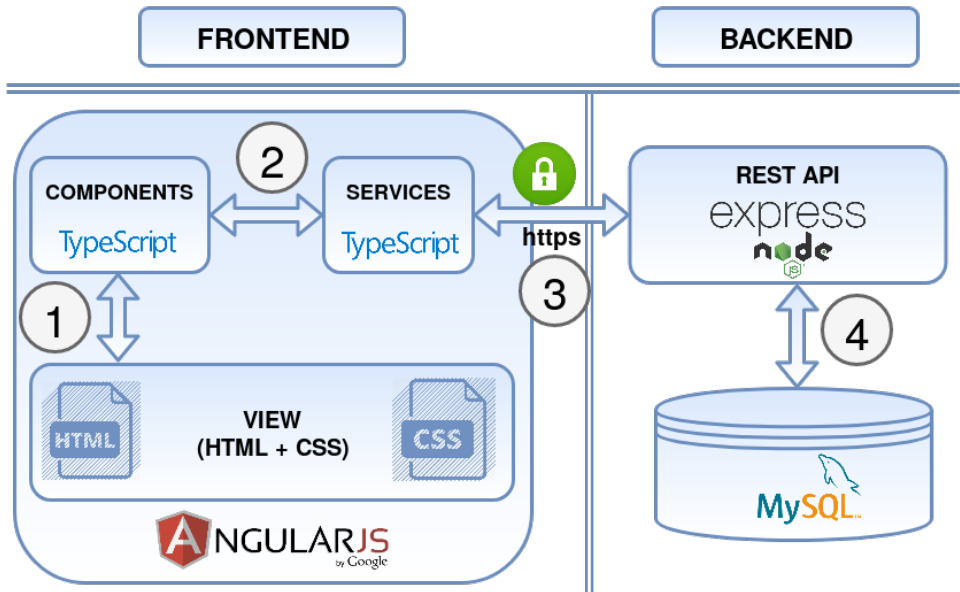


Figura 7: Esquema conceptual

S'han numerat els passos a l'esquema anterior 7 Esquema conceptual que seguirà la ferramenta amb la interacció d'un usuari per a una major claredat i comprensió. En primer lloc, l'usuari interactua amb el sistema utilitzant qualsevol navegador, i aquest carrega la vista corresponent. Cada vista té associada un component que s'encarrega d'executar la lògica implementada (punt 1) i efectuar crides al servei (punt 2). El servei crida a l'API utilitzant peticions *https* aconseguint així transferir les dades de forma segura (punt 3). Posteriorment l'API interacciona amb la BD realitzant les operacions CRUD necessàries en funció de la crida efectuada (punt 4). Finalment, es realitza el mateix procediment de forma inversa fins a arribar a la vista, és per això que es poden observar a la imatge fletxes amb doble sentit.

## 12 Anàlisi de riscos

Com a conseqüència de la gran quantitat d'amenaques que es poden identificar en els projectes d'enginyeria del *software*, s'ha decidit donar un enfocament pràctic, distingint únicament les més rellevants i que poden afectar directament al projecte, evitant d'aquesta manera allargar la secció de forma innecessària.

Les amenaces identificades són les següents:

- **Fallada del servidor:** Una fallada de servidor deixaria la web inaccessible fins que aquest es restablira.
- **Problemes de visualització a dispositius mòbils:** El fet que actualment existisquen gran quantitat de dispositius mòbils fa que hi haja diversitat en el tamany de les pantalles. Açò pot suposar un problema amb la visualització d'algunes vistes sobretot als dispositius més vells amb pantalles més xicotetes.
- **Problemes de visualització a algun navegador:** Al desenvolupament web no tots els navegadors mostren les vistes de la mateixa manera, poden haver components que es visualitzen de forma diferent. Açò pot ocasionar problemes a l'hora visualitzar alguns components ja que no van a mostrar-se igual en tots els navegadors.
- **Lentitud en la resposta dels serveis:** Un flux lent en la navegació de la pàgina web, seria un problema ja que no permetria utilitzar-la de forma òptima i còmoda per als usuaris que la fan servir.
- **Inconsistència a la BD:** La inconsistència entre les dades de la web i de la BD del *Backend* no permetria utilitzar el sistema correctament i trencaria amb el compliment d'alguns requisits funcionals especificats per a aquest projecte.
- **Mala gestió dels usuaris:** Una mala gestió dels usuaris produiria comportaments no desitjables dintre del nostre sistema i també trencaria en el compliment d'alguns requisits funcionals.

Per a cada risc identificat, s'avaluarà tant la probabilitat de que es materialitze així com l'impacte que tindria sobre el projecte. Aquestes qualificacions es faran en funció de la següent matriu de probabilitat per impacte.

		IMPACTE		
		Baix	Mitjà	Alt
PROBABILITAT	Baixa	Molt Baix	Baix	Mitjà
	Mitja	Baix	Mitjà	Alt
	Alta	Mitjà	Alt	Molt Alt

**Taula 4:** Matriu de valoració de riscos

Els colors indiquen la tolerança del risc en qüestió, de manera que les caselles amb tonalitats verdoses són considerades riscos tolerables, i les caselles grogues i roges indiquen que s'han de prendre mesures. Concretament les roges es consideren intolerables i s'han de buscar alternatives per a tractar de disminuir el risc urgentment.

Amenaça	Risc	Probabilitat	Impacte
Fallada de Servidor	Mitjà	Baixa	Alt
Mala visualització a dispositius mòbils	Baix	Baixa	Baix
Mala visualització a alguns navegadors	Baix	Baixa	Baix
Lentitud	Mitjà	Mitja	Mitjà
Inconsistència a la BD	Mitjà	Baixa	Alta
Mala gestió d'usuaris	Mitjà	Baixa	Alta

**Taula 5:** Avaluació de riscos

Es pot observar a la taula, que actualment no hi ha riscos alts (de color roig). Els riscos de visualització s'han considerat baixos ja que són aïllats i que apliquen a molts pocs casos i/o dispositius.

D'altra banda, la resta de riscos han sigut considerats mitjans, com a conseqüència de que l'impacte que tenen és gran, però realment la probabilitat de que ocorreguen és baixa. S'ha posat especial atenció al desenvolupament i *testing* d'aquests components per tal de minimitzar el risc al màxim.

## 13 Organització i gestió del projecte

### 13.1 Organització

Als següents subapartats s'aniran descrivint els diferents aspectes referents a l'organització del projecte, considerant els actors que duren a terme el projecte, l'estructura interna, interfícies, rols i responsabilitats.

### 13.1.1 Actors i rols del projecte

La informació referent a les persones involucrades en el projecte, es podrà visualitzar a les següents taules, que mostren nom i cognoms, informació de contacte, i responsabilitats al càrrec dintre del projecte.

<b>Nom i cognoms:</b>	Carles Torró Segura
<b>Contacte:</b>	cartorse@inf.upv.es
<b>Rol:</b>	Programador

<b>Nom i cognoms:</b>	Jose Luis Poza Luján
<b>Contacte:</b>	jopolu@disca.upv.es
<b>Rol:</b>	Coordinador

<b>Nom i cognoms:</b>	Jose Alberto Conejero Casares
<b>Contacte:</b>	aconejero@upv.es
<b>Rol:</b>	Coordinador

<b>Nom i cognoms:</b>	Víctor Ramírez Flores
<b>Contacte:</b>	vcramflo@inf.upv.es
<b>Rol:</b>	Programador (extern)

<b>Nom i cognoms:</b>	Coski21 Team
<b>Contacte:</b>	<a href="http://www.skills-4u.eu">http://www.skills-4u.eu</a>
<b>Rol:</b>	<i>Stakeholders</i>

**Taula 6:** Personal involucrat en el projecte

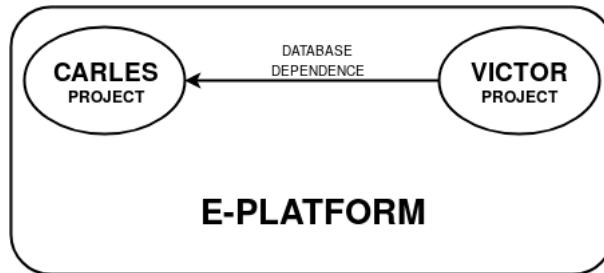
### 13.1.2 Interfícies externes

Com bé s'ha mencionat a l'apartat (8) Abast, aquest projecte s'inspira en un projecte europeu més gran anomenat CoSki21. L'alumne Víctor Ramírez Flores, citat en l'apartat (13.1.1) Actors i rols del projecte, s'encarrega de desenvolupar una altra part inspirada en el mateix projecte i, conseqüentment relacionat amb aquest, però s'emmarcarà dintre del seu treball de fi de grau.

El projecte d'aquest alumne, a grans trets, s'encarrega de gestionar les soft skills relacionades amb la missió, visió i valors d'una organització (per recordar consultar a l'apartat (8) Abast). Com que el seu treball necessita gestionar *soft skills*, la BD que utilitzarà dependrà de la BD dissenyada per a aquest treball, amb la qual es comunicarà agafant la informació referent a les *Soft Skills* emmagatzemades.

La dependència en qüestió s'especifica posteriorment a l'apartat (17.1) Annex - Disseny del sistema.

D'altra banda, altra interfície externa, és l'E-platform, una ferramenta per a administrar els serveis d'un servidor de forma gràfica. Aquesta plataforma s'encarrega de gestionar-la dintre del projecte europeu CoSki21 el personal de la universitat de Turku a Finlàndia. L'E-Platform inclouria el resultat de desenvolupar aquest treball i el treball de l'alumne Víctor Ramírez Flores per tal de facilitar la seua explotació, el manteniment o fins i tot la seua comercialització.

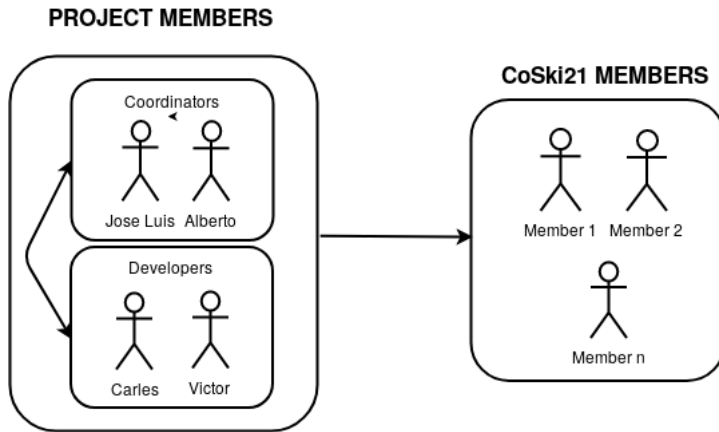


**Figura 8:** Interfícies externes

Es pot observar la relació que té aquest projecte amb el projecte de l'alumne Víctor Ramírez Flores i la E-Platform a la imatge anterior (8) Interfícies externes.

### *13.1.3 Estructura interna*

L'estructura i la relació que tenen els diferents membres que formen part del projecte, especificats amb detall a l'apartat anterior, es pot observar amb la següent imatge.

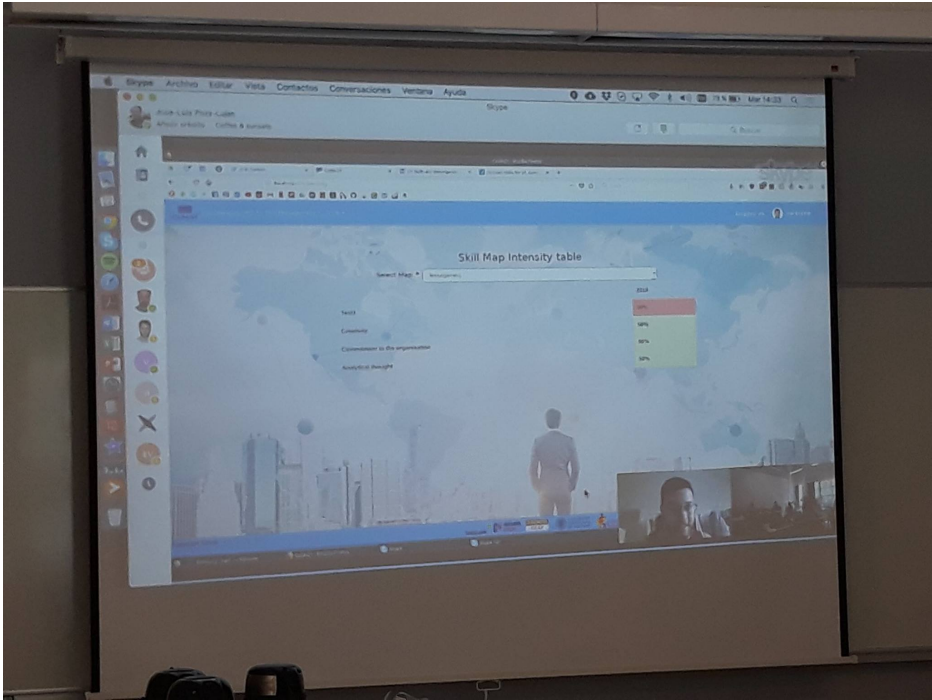


**Figura 9:** Estructura interna

S'observa a la part esquerra els membres encarregats de desenvolupar el projecte, ja siga a nivell de coordinació, supervisió o de desenvolupament, aquests dos grups dintre del projecte estan directament relacionats per tal desenvolupar-lo seguint els objectius marcats. A la part dreta s'observen els membres del *CoSki21 team* que com ja s'ha dit a l'apartat anterior, són els encarregats de valorar o validar el sistema.

Per tal de validar el projecte, s'han aprofitat les reunions realitzades a Finlàndia pels membres del projecte *CoSki21*, mostrant la ferramenta per Skype. S'ha explicat el disseny fet, el funcional implementat i s'ha navegat a través de la ferramenta com ho faria un usuari habitual per tal de mostrar el correcte funcionament validant d'aquesta manera la ferramenta. A continuació, es mostrarà una captura d'una de les reunions realitzades.





**Figura 10:** Validació amb el CoSki21 Team

## 13.2 Gestió de projectes

El projecte s'ha desenvolupat utilitzant metodologies àgils gestionant les tasques a dur a terme en cada iteració amb a la ferramenta Trello fent servir un taulell compartit amb els supervisors del projecte. A més a més, s'han anat celebrant reunions presencials setmanalment i, puntualment s'han dut a terme reunions per *Skype*. Durant aquestes sessions, s'ha debatut quines tasques s'han de traslladar a Trello i conseqüentment, dur a terme en la iteració corresponent.

## 14 Planificació temporal

El desenvolupament del projecte s'ha dut a terme entre els mesos de Febrer de 2018 i Juliol de 2018 i s'ha desenvolupat de forma iterativa seguint metodologies àgils amb Trello, com ja s'ha esmentat en l'apartat anterior. A la següent imatge es pot observar una captura de pantalla del taulell de Trello.

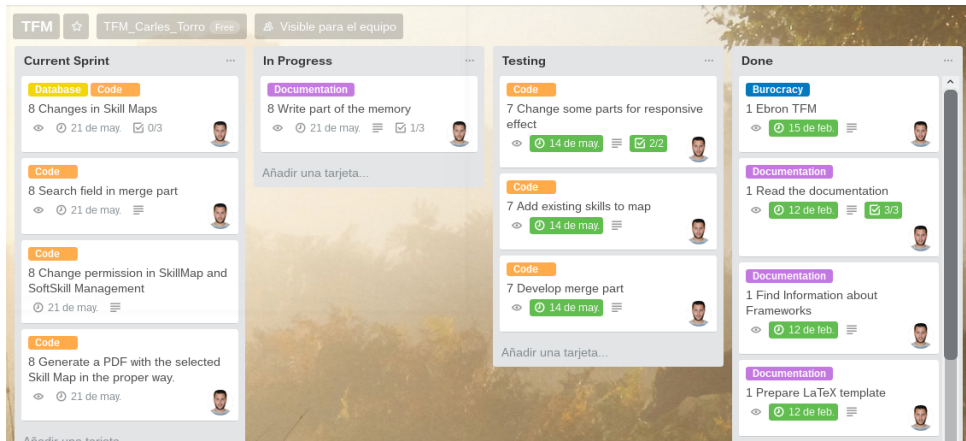


Figura 11: Captura del taulell de Trello

Aquest taulell està organitzat en quatre llistes. Les targetes (tasques) tenen un número abans del títol indicant la iteració i van movent-se d'una llista a altra en funció de l'estat en el que es troben. Les llistes en qüestió són les següents:

- **Current Sprint:** Conté les targetes que s'han de dur a terme en el *sprint* actual.
- **In Progress:** Conté totes les targetes que s'estan desenvolupant actualment i prèviament es trobaven a la llista *Current Sprint*
- **Testing:** Conté les targetes que ja han sigut desenvolupades però estan en fase de proves. Les targetes que es troben en aquesta llista han estat prèviament a *In Progress*.
- **Done:** Conté totes les targetes que ja han sigut desenvolupades i també provades. Les targetes que estiguen en aquesta llista prèviament estaven a *Testing*. Al final del projecte totes les targetes han d'estar en aquesta llista.

A les reunions setmanals s'ha avaluat l'estat de les targetes corresponent a la iteració en la que es troba el projecte per tal de veure com evoluciona i també se'n proposen de noves, aquestes es desenvoluparan amb la data límit de la següent iteració. Per tal d'observar com s'han anat acomplint els objectius marcats en les iteracions, es mostrarà un diagrama de Gantt



### 15.1 Costos *hardware*

Per a la implementació del projecte, s'ha utilitzat un ordinador portàtil MSI i, com s'ha dit anteriorment a l'apartat (5.3) Ferramentes, per a realitzar proves del sistema, s'ha utilitzat una *Raspberry Pi 3*. A continuació es pot observar una taula amb les prestacions del *hardware* utilitzat per al desenvolupament del projecte

Components	Portàtil	<i>Raspberry Pi</i>
Marca i model	MSI GE60 2PC	<i>Raspberry Pi 3</i>
Disc Dur	1TB	32GB
Memoria RAM	8GB	1GB
Tarjeta gràfica	GTX850m Nvidia	Broadcom VideoCore IV
Procesador	i5-420H	ARMv8 quad-core
Sistema Operatiu	Debian 9	Raspbian
Preu	700€	30€

**Taula 7:** Resum costos *Hardware*

### 15.2 Costos *software*

El *software* utilitzat ha sigut en la seua totalitat gratuït, totes les tecnologies especificades anteriorment (5.3) Ferramentes són gratuïtes i a més a més, tant l'ordinador portàtil amb el que s'ha implementat el projecte com la *Raspberry Pi 3* fan servir una distribució Linux, de manera que el seu cost econòmic és 0.

### 15.3 Costos de ma d'obra

Com bé s'ha especificat a l'apartat (14) Planificació temporal, aquest projecte s'ha desenvolupat al llarg de 5 mesos, començant acordant la idea del treball amb el tutor al Febrer de 2018 fins al lliurament del mateix al Juliol de 2018. Tant la implementació com la redacció del projecte s'han fet de forma alternada i uniforme al llarg d'aquest temps.

El cost estimat d'aquest treball es pot comptabilitzar amb 400 hores i conseqüentment es pot emmarcar dins d'un treball de fi de màster de 15 crèdits.

## 15.4 Costos del producte

També és important a l'hora d'abordar un projecte calcular els pressupostos del producte desenvolupat. Els principals aspectes econòmics que s'haurien de considerar per abordar un projecte d'aquestes dimensions són els següents:

- Llicències de les ferramentes que s'utilitzen.
- Servidors web.
- Dominis.
- Pla d'explotació.

## 15.5 Resum de costos

Per tal de quantificar els costos del treball que implica el desenvolupament del projecte i la generació d'un producte emmarcats dintre d'aquest treball de fi de màster, es va a considerar el cost de la vida útil dels components *hardware* utilitzats, els costos de ma d'obra i finalment, els costos de generar un producte.

Pel que respecta als costos de llicències comercials del *software* les ferramentes que s'han utilitzat no requereixen d'aquest tipus de pagament.

### HARDWARE

	Nº mesos	Preu/mes	Total
Raspberry PI	5	3 €	15€
Ordinador MSI	5	140 €	700 €

### PRODUCTE

	Nº mesos	Preu/mes	Total
Lloguer de servidor (Amazon Linux 0.5 GB)	12	4 €	48 €
Comprar domini (coski21.com)	12	10 €	120 €

### MA D'OBRA

	Nº hores	Preu/mes	Total
Cosotos d'implementació	400h	40 €	16.000 €
<b>SubTotal</b>			<b>16.883 €</b>

**Taula 8:** Resum del pressupost

## 16 Conclusions i treball futur

Aquest apartat recollirà les conclusions del projecte esmentant les dificultats que s'han trobat a l'hora de del desenvolupament, aportacions que fa el projecte i ampliacions que es podrien dur a terme en un futur.

### 16.1 Dificultats trobades

A continuació es detallaran les dificultats trobades. L'ordre en el que es descriuen les dificultats està relacionat amb l'ordre en què s'ha desenvolupat el projecte.

Una de les principals dificultats a l'hora d'escollir els sistemes similars a analitzar, ha sigut tractar de sintetitzar quins de tots els sistemes trobats són els que més concordança tenien amb els objectius del projecte, havent de descartar alguns dels que no compartien els objectius principals.

Pel que respecta al disseny, la principal dificultat trobada ha sigut determinar quin seria el disseny més òptim de la web per als usuaris. S'han hagut de fer modificacions del mateix per tal d'assolir el que s'ha considerat el disseny més òptim aconseguint facilitar d'aquesta manera l'ús de la ferramenta per als usuaris que la faran servir.

Les dificultats més grans a l'hora d'abordar el projecte han sigut les vinculades a la implementació, ja que s'han utilitzat tecnologies sobre les que no es tenien pràcticament coneixements. Desenvolupar l'API rest o aprendre a utilitzar de 0 el *framework* AngularJS no ha sigut fàcil però si satisfactori, ja que gràcies al projecte s'han conegut noves ferramentes que amplien el coneixement sobre les noves tecnologies.

Tot i que aquests aspectes s'han considerat dificultats per a desenvolupar el projecte, no s'han de considerar problemes, ja que el món de les noves tecnologies està contínuament en desenvolupament. És per això que adaptar-se i aprendre a utilitzar noves ferramentes i tecnologies no ha de suposar un problema, més bé al contrari, s'ha d'afrontar amb positivisme.

### 16.2 Aportacions del projecte

La principal aportació realitzada és el sistema compost per l'aplicació web, l'API rest i la BD que aconsegueixen l'objectiu principal del projecte. Addicionalment també s'aporta l'estudi dels diferents sistemes similars descrits en l'apartat (4.1) Sistemes Similars.

A nivell personal el desenvolupament del projecte ha sigut gratificant, ja que com s'ha indicat en l'apartat anterior, no es posseïen pràcticament coneixements de la

majoria de les tecnologies utilitzades i aquests han sigut assolits gràcies a haver realitzat aquest projecte.

Ja que es tracta d'un projecte relacionat amb les *soft skills* és important destacar les competències transversals de la UPV que es considera que s'han treballat. Aquestes són:

- **Disseny i projecte** com a competència principal ja que és pràcticament en el que s'ha basat el TFM.
- **Anàlisi i resolució de problemes**, ja que totes les dificultats trobades s'han hagut d'analitzar i solucionar per tal de garantir el desenvolupament del projecte.
- **Planificació i gestió del temps**, s'ha adquirit aquesta competencial per tal d'assolir els objectius del projecte en el termini acordat.

### 16.3 Ampliacions futures

Amb la intenció de millorar el projecte desenvolupat, es podrien seguir implementat funcionalitats. Una de les ampliacions que es podrien dur a terme, seria la ja descrita anteriorment a l'apartat (9) Hipòtesis i restriccions, que consisteix en fer una diferenciació de rols, de manera que es podrien desenvolupar funcionalitats diferents en funció del rol amb el que es registra l'usuari a la plataforma. Açò implicaria desenvolupar noves funcionalitats i restringir l'ús de les mateixes en funció del rol del perfil de l'usuari.

També es podria ampliar la funcionalitat corresponent al requisit encarregat del *Clustering*, afegint una opció que pugui eliminar les paraules que no són considerades *stop words* però que a l'hora de comparar les descripcions de les *skills* tampoc siguin importants, com per exemple *ability*, *skill*.. entre altres, aconseguint d'aquesta manera una comparació més precisa.

Finalment, també es podrien fer enquestes per saber la satisfacció en la utilitat de la ferramenta i algun apartat per a contactar i rebre informació sobre problemes i millores que pugui tindre la plataforma.

## 17 Annexos

### 17.1 Annex - Disseny del sistema

En aquest annex s'especificaran tots els detalls relacionats amb el disseny del sistema, realitzant l'especificació formal fent una descripció per capes.

Tant el disseny com la implementació s'ha basat en el patró MVC (*Model, View, Controller*) amb l'objectiu de separar les capes de presentació, negoci i de dades, ja que es tracta d'una arquitectura *Client-Server* típica al desenvolupament Web. A continuació, s'especifiquen les diferents capes.

#### 17.1.1 Capa de presentació

Coneguda també com capa de l'usuari (interfície gràfica). S'encarrega de mostrar la informació i arregar dades de l'usuari i, a més a més, ha de ser el més amigable possible. En aquest apartat es mostraran una serie de *Mockups* juntament amb la seua explicació. Corresponen al disseny de les interfícies gràfiques de les principals finestres de la web.

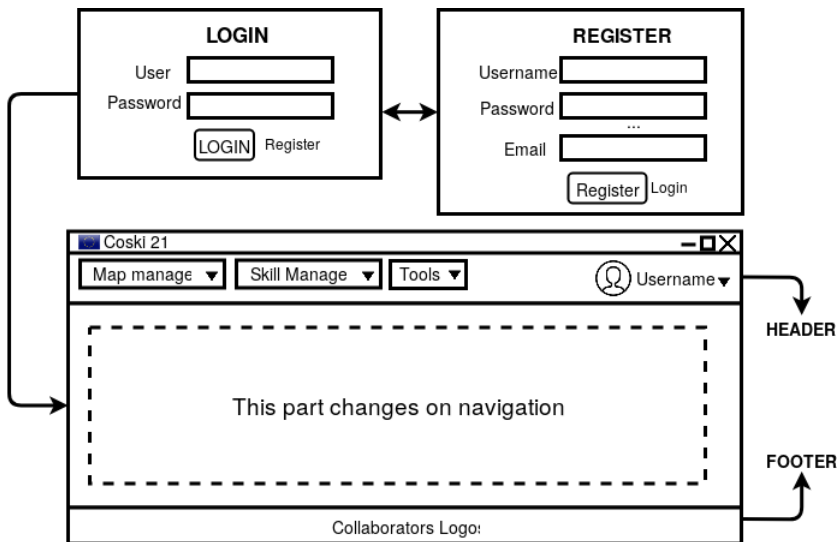


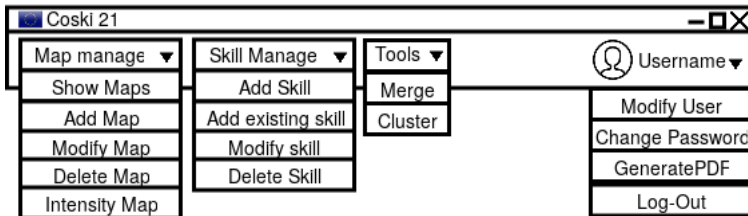
Figura 13: *Mockup* genèric de la pàgina web



A la imatge anterior s'observa un *mock-up* que inclou el formulari de registre, el d'inici de sessió i l'estructura general que tindran les finestres de la pàgina web una vegada s'ha accedit amb usuari i contrasenya.

A l'hora de dissenyar les interfícies gràfiques, s'ha decidit que totes les finestres de la pàgina web tinguin en comú la utilització d'una capçalera (*header*) i un peu de pàgina (*footer*) exceptuant l'encarregada de l'inici de sessió i l'encarregada del registre. Aleshores, a mesura que es navegue per la web l'única part que canviarà serà la part central. Aquest disseny es pot observar en la imatge anterior.

La capçalera (*header*), serà la que ens permetrà navegar al llarg de les diferents finestres de la pàgina web, mostrant el contingut de la navegació en qüestió a la part central de la pàgina Web. Es pot observar la capçalera al següent *mock-up*:



**Figura 14:** *Mockup* de la capçalera

Per tant, la capçalera permetrà fer una gestió de mapes d'*skills* (*Map Manage* a la imatge), una gestió de soft skills (*Skill Manage* a la imatge), utilitzar les ferramentes avançades (*Tools* a la imatge) que corresponen al *Merge* i *Clustering* i també gestionar l'usuari (*Username* a la imatge).

La gran majoria d'opcions de la gestió d'*skills*, la gestió de mapes d'*skills* i fins i tot la gestió de l'usuari, s'encarreguen d'accions com mostrar, afegir, esborrar i modificar. Aquestes accions, estan gestionades amb formularis, com s'ha especificat als requisits funcionals apartat (7) Requeriments inicials. Els formularis en qüestió seran de la següent manera i com s'ha dit anteriorment es mostraran el part central de la pàgina web:

**Figura 15:** *Mockup* d'un formulari

Òbviament no tots els formularis seran idèntics, els seus camps dependran de la gestió que es duga a terme (afegir, esborrar, modificar ...) i també si gestionen mapes, *skills*, o l'usuari però generalment seguiran l'estil de la imatge anterior.

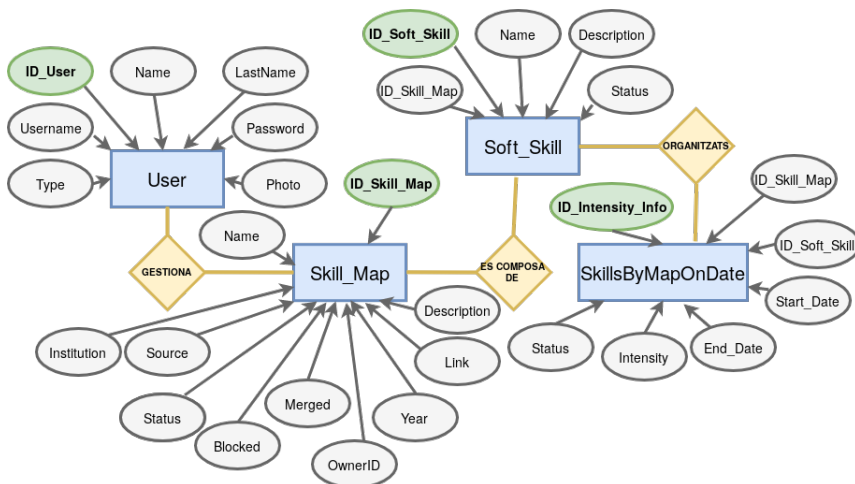
D'altra banda, i per acabar aquesta secció, cal dir que les finestres encarregades de la visualització de contingut a la web tindran un estil similar entre elles. A continuació es mostrarà el disseny de la visualització de mapes ja que la part corresponent a les ferramentes avançades (*Merge i Clustering*), la generació de PDFs i la visualització d'Intensitat, són molt similars a la visualització de mapes.

**Figura 16:** *Mockup* visualització d'informació

Bàsicament el comportament es basa en seleccionar les dades sobre les que es vol mostrar la informació i es visualitzarà a la part central de la web. Dependent del funcional que s'execute i del que se seleccione, la informació que es mostrarà anirà canviant.

### 17.1.2 Capa de persistència

És on resideixen les dades i l'encarregada d'accedir a les mateixes, rebent peticions de lectura i/o escriptura. Com bé s'ha especificat en l'apartat (5.3) Ferramentes, la tecnologia utilitzada per la BD serà MySQL. Per a clarificar el disseny realitzat per a la BD, a continuació es mostrarà un diagrama entitat-relació.



**Figura 17:** Diagrama Entitat-Relació

Amb el diagrama anterior (17) Diagrama Entitat-Relació, es poden observar les entitats més importants que es gestionaran, pintades de color blau així com les seues relacions pintades de color groc i les propietats pintades de color gris exceptuant els que seran identificadors a les taules de la Base de dades que estan pintats de verd.

D'aquesta manera, la informació que emmagatzemarà la base de dades correspondrà als usuaris que gestionaran *skills maps* composant-se aquests per diferents *soft skills* que s'organitzen temporalment. A partir d'aquest diagrama es poden definir les taules i els atributs que tindrà la base de dades i realitzar d'aquesta manera l'esquema lògic relacional.

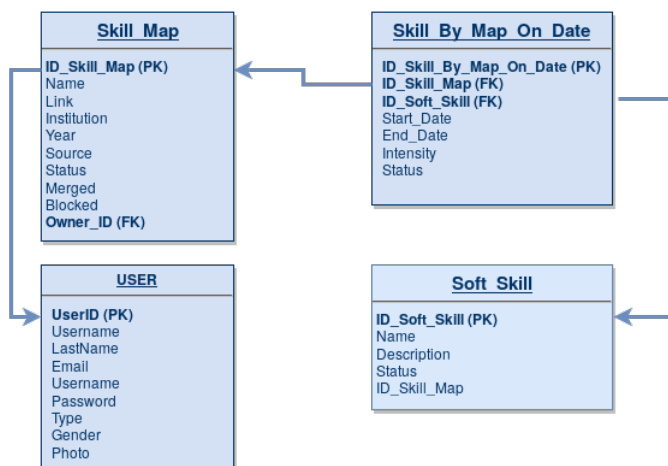


Figura 18: Esquema lògic

A continuació s'explicaran exclusivament el propòsit de les taules i els atributs més importants, deixant de banda la resta d'atributs, ja que són molts i el significat es pot deduir observant el diagrama anterior (18 Esquema lògic). Les taules seran les següents:

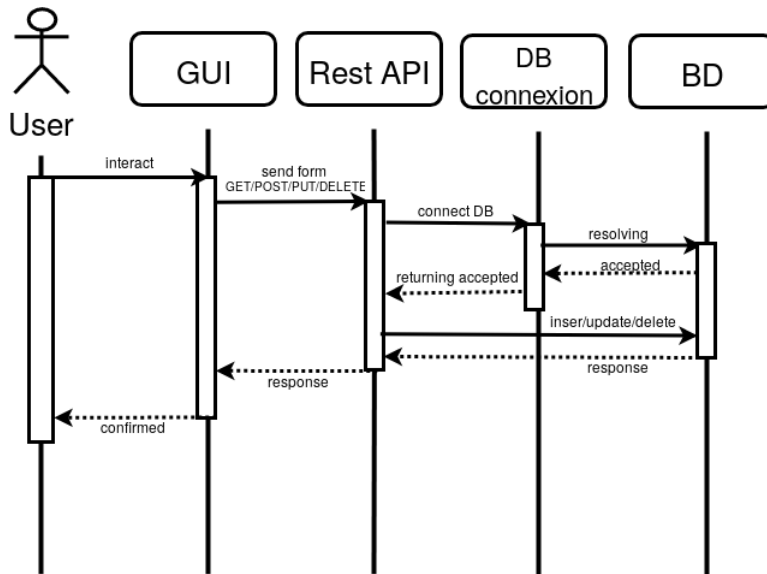
- **User:** Emmagatzema la informació dels usuaris registrats en la Web.
- **Soft\_Skill:** Conté la informació referent a les *soft skill*. Cadascuna d'elles pertany a un *skill map*.
- **Skill\_Map:** Conté la informació referent al *skill map*. D'aquesta taula cal destacar l'atribut *blocked*, que indicarà si és un mapa bloquejat o no. La definició d'aquest atribut es pot revisar a l'apartat (6) Definicions i abreviacions.
- **SkillsByMapOnDate:** Emmagatzema informació de les *soft skills* al llarg del temps, així com el *skill map* al que pertanyen. Per a aquesta taula és important destacar l'atribut *Intensity*, que inclou la intensitat en forma de percentatge que té cada *soft skill* amb una data d'inici i una data de fi.

És important destacar en aquest punt que l'alumne Víctor Ramírez Flores, esmentat a l'apartat (13.1.1) Actors i rols del projecte utilitzarà la taula *Soft\_Skill* de la base de dades especificada en aquesta secció.

### 17.1.3 Capa de negoci

La capa de negoci, també coneguda com a capa lògica, és l'encarregada d'executar el codi que implementa les funcionalitats. La majoria de funcionalitats que s'han de desenvolupar en aquest treball repeteixen un patró que consisteix en què l'usuari interacciona amb la interfície emplenant un formulari, aquest s'envia amb una petició https a l'API per a posteriorment comunicar-se amb la BD i realitzar la modificació pertinent en funció de la funcionalitat. Aquest comportament ja s'ha descrit anteriorment a l'apartat (11) Descripció de la solució proposada .

A continuació es mostrarà un diagrama de seqüència que resumeix el patró descrit en el paràgraf anterior. Aquest diagrama es pot aplicar a la gran majoria dels requisits funcionals proposats per aquest treball.



**Figura 19:** Diagrama de seqüència

S'ha decidit no realitzar un diagrama per a cada funcionalitat com a conseqüència de la similitud que tindrien els mateixos i també per tractar de no allargar la secció amb informació redundant.

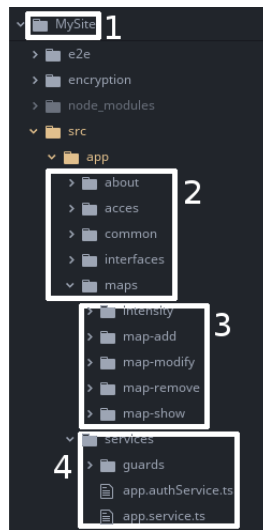
## 17.2 Annex - Implementació

En aquest annex, es detallaran tots els aspectes referents a la implementació del sistema corresponent al disseny realitzat a l'annex anterior. Es parlarà de la tecnologia utilitzada, es mostraran captures reals del sistema i s'inclouran les línies de codi amb especial rellevància així com les corresponents explicacions.

L'annex es dividirà en 3 seccions, cadascuna d'elles detallarà la implementació de la pàgina web, la BD i l'API, respectivament.

### 17.2.1 Pàgina Web

Com ja s'ha esmentat anteriorment a l'apartat (5.3) Ferramentes, l'aplicació web s'ha desenvolupat en el *framework* AngularJS utilitzant la versió 4, amb l'editor de text Atom. A grans trets, un projecte amb AngularJS està format principalment per diferents components. Cada component generalment s'encarrega d'implementar la funcionalitat (amb TypeScript en aquest cas) associada a una vista (corresponent als fitxers HTML i CSS). A continuació es mostra una captura amb l'estructura de carpetes i fitxers del projecte.



**Figura 20:** Estructura projecte AngularJS

Amb aquesta imatge, 20 Estructura projecte AngularJS, es pot observar l'estructura del projecte AngularJS, que s'ha anomenat *MySite* (punt 1) Estructura projecte AngularJS). S'observen varies carpetes dintre de *MySite* però la més destacable

es *src/app/* que al seu interior mostra diverses carpetes que agrupen components amb finalitats semblants (punt 2). Entre aquestes carpetes s'observa la carpeta *maps* que al seu interior inclou els components que gestionen *skill maps* (punt 3). Per cadascun d'aquests components trobarem el fitxer HTML, CSS i TypeScript corresponent. També s'observa una carpeta anomenada *services* (punt 4) que és la que conté els serveis implementats i s'encarregarà d'efectuar les crides a l'API amb peticions *https*.

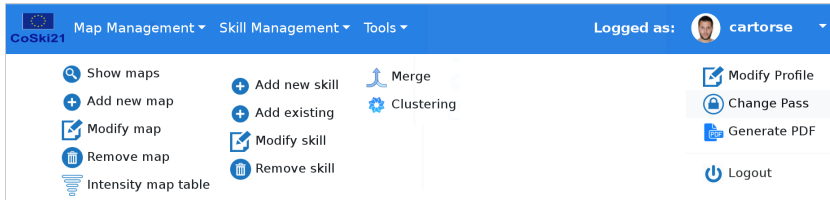
El resultat d'implementar el disseny de l'annex anterior (5.3) Annex - Disseny del sistema, el podem observar en les següents imatges:

- Imatge que correspon al inici de sessió, registre i estil general de la pàgina, mostrant a la part central informació sobre el projecte.



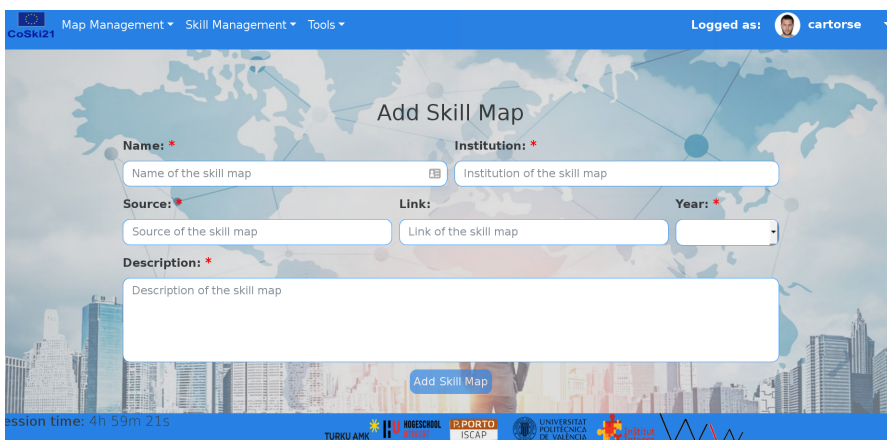
**Figura 21:** Implementació del inici de sessió, registre i estil genèric

- Imatge que correspon a la capçalera (*header*), amb tots els seus camps desplegats:



**Figura 22:** Implementació del *header*

- Imatge que correspon a la implementació dels formularis, concretament l'encarregada d'afegir un *Skill Map*:



**Figura 23:** Implementació de formularis

- Imatge que correspon a la implementació de finestres encarregades de visualització de contingut, concretament l'encarregada de mostrar informació dels mapes



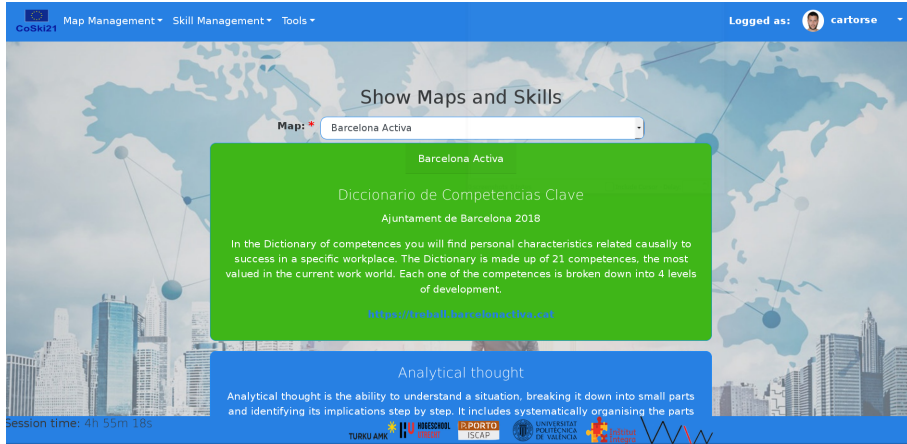


Figura 24: Implementació de finestres encarregades de la visualització

Es pot realitzar una comparativa entre les imatges anteriors i les imatges corresponents al disseny mostrades a l'annex anterior (5.3) Annex - Disseny del sistema amb la intenció de validar el disseny proposat.

Pel que respecta a les línies de codi rellevants, a continuació es mostrarà una funció allotjada dins d'un servei que fa una petició a l'API encarregada d'obtenir les *Soft Skills* associades a un *Skill Map* del que es coneix la seua ID.

```

1 //Return an array of Skills
2 getAllSkillsByMap(id): Observable<Skills []> {
3   return this.https.get(this.basicURL+"/api/skillsByMap/"+id,this.
4     options)
5     .map((response: Response) => {
6       return <Skills []>response.json();
7     })//map
8     .catch(this.handleError);//catch the error
9 }//getAllSkillsByMap

```

Codi 1: Exemple d'una cridada des del servei a l'API

L'anterior fragment de codi, realitza una petició GET a la URL emmagatzemada en la variable `this.basicURL +/api/skills` (línia 3) rebent la ID del *Skill Map* per paràmetre i si no es produeix cap error retorna una resposta mapejada a una *array* de tipus **Skills** (línia 5).

El següent fragment de codi amb TypeScript, efectua una crida a la funció anterior allotjada dins del servei *apiservice* (línia 2) i si no es produeix cap error emmagatzema el resultat de la crida en la variable `this.SkillsArray` (línia 8)

```

1  getAllSkillsByMap(id): void {
2      this.apiService.getAllSkillsByMap(id)
3          .catch((err: Response) => {
4              this.services=true
5              return Observable.throw("Error getAllSkillsByMap")
6          })//catch the error
7          .subscribe(resultArray => {
8              this.SkillsArray = resultArray
9          })//subscribe
10 }//getAllSkillsByMap

```

**Codi 2:** Exemple d'una cridada des d'un funcional al servei

La vista HTML s'encarrega de cridar a la funció anterior i a partir d'eixe moment ja pot utilitzar la variable **this.SkillsArray** per a mostrar els valors per la interfície gràfica. Ho podem observar en el següent fragment de codi HTML, on a la línia 3 es fa una iteració sobre la variable **this.SkillsArray**:

```

1  <select class="form-control" [(ngModel)]="id.ID_Soft_Skill" (change)
2      ="getSkillAttributes($event.target.value)" formControlName="
3      ID_Soft_Skill" required id="ID_Soft_Skill">
4      <option disabled value=0>--Select Map--</option>
5      <option *ngFor="let skill of SkillsArray" value={{skill.
6          ID_Soft_Skill}}>{{skill.Name}}</option>
7  </select>

```

**Codi 3:** Fragment HTML

Els tres fragments de codi anteriors són un exemple del funcionament genèric de la web, que es basa en realitzar crides a l'API per a fer operacions a la BD obtindre resultats i mostrar-los per la interfície gràfica.

### 17.2.2 Base de Dades

Com ja s'ha indicat anteriorment a l'apartat (5.3) Ferramentes, la tecnologia que s'utilitzarà per a la BD serà MySQL. S'ha implementat amb aquest sistema de gestió de base de dades la BD dissenyada en l'annex anterior (17.1) Annex - Disseny del sistema. Aquesta implantació s'ha fet utilitzant un *script* que crea les taules i les relacions a la BD.

S'ha desenvolupat l'*script* amb la intenció de facilitar la recuperació de la BD mentre es desenvolupava el sistema i es feien proves, ja que en aquesta fase la BD moltes vegades arriba a estats no desitjats i és una forma d'automatitzar el procés de recuperació. A continuació es mostrarà un fragment de l'*script* corresponent a la creació de la taula *Skill\_Map*, la resta de taules es creen de la mateixa forma però amb els camps corresponents .

```

1 --CREATE TABLE Skill_Map
2 CREATE TABLE IF NOT EXISTS Skill_Map (
3   ID_Skill_Map INT AUTO_INCREMENT,
4   Name VARCHAR(250) NOT NULL,
5   Description TEXT,
6   Link VARCHAR(250),
7   Institution VARCHAR(250),
8   Years INT,
9   Source VARCHAR(250),
10  Blocked INT,
11  Status INT,
12  ID_Owner INT,
13  PRIMARY KEY(ID_Skill_Map)
14 );

```

**Codi 4:** Creació de la taula *Skill\_Map* a la BD

A més a més, l'*script* inserta 3 mapes (i les seues corresponents *skills*) amb els quals es va a treballar. L'explicació de que són i perquè s'utilitzen aquests mapes s'especifica en detall a la secció (18) Estudis amb entitat pròpia.

A continuació es mostrarà una captura de pantalla on es podrà observar el contingut de la taula *Skill\_Map* on es podran observar els 3 mapes citats anteriorment, així com un mapa creat per a realitzar diferents proves i validar el funcionament del sistema, aquest últim és el mapa sobre el qual es realitza la majoria de bateries de proves de l'apartat (17.3) Annex - Fase de proves.

```

mysql> SELECT ID_Skill_Map, Name, Institution, Year, Blocked, Status, ID_Owner FROM Skill_Map;

```

ID_Skill_Map	Name	Institution	Year	Blocked	Status	ID_Owner
1	Barcelona Activa	Ajuntament de Barcelona	2018	1	1	1
2	Competencias transversales UPV	Universitat Politècnica de Valencia	2018	1	1	1
3	MOSSA	Fondazione casa di carita	2018	1	1	1
4	Carles Map Test	Carles Torró Segura Organization	2018	0	1	2

```

4 rows in set (0.00 sec)

mysql> SELECT ID_Skill_Map, Source, Link FROM Skill_Map;

```

ID_Skill_Map	Source	Link
1	Diccionario de Competencias Clave	<a href="https://treball.barcelonactiva.cat">https://treball.barcelonactiva.cat</a>
2	UPV	<a href="http://www.upv.es/contenidos/COMPTRAN/">http://www.upv.es/contenidos/COMPTRAN/</a>
3	The Model Of Soft Skills Assessment	<a href="http://valorize.odl.org/outputs/I02%20-%20MOSSA%20VHSM.pdf">http://valorize.odl.org/outputs/I02%20-%20MOSSA%20VHSM.pdf</a>
4	Carles Torró Segura Source	<a href="https://www.linkedin.com/in/carles-torr%C3%B3-segura-46a762119/">https://www.linkedin.com/in/carles-torr%C3%B3-segura-46a762119/</a>

```

4 rows in set (0.00 sec)

```

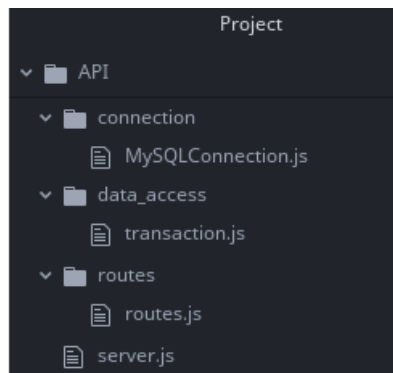
**Figura 25:** Dades de la Taula *Skill\_Map*

A la captura anterior, s'ha mostrat el contingut de la taula *skill map* amb dues consultes per a una millor visualització. No s'ha mostrat el camp descripció perquè al ser més llarg no es visualitza correctament a una captura.

### 17.2.3 API

Com ja s'ha detallat anteriorment a l'apartat (5.3) Ferramentes, l'API ha sigut desenvolupada amb ExpressJS i ha sigut estructurada en 4 fitxers diferents:

- **server.js**: Fitxer encarregat de posar el servidor en funcionament a la *URL* i port especificats utilitzant el certificat auto-firmat per garantir la seguretat.
- **routes.js**: Fixa els *endpoints* que es faran servir i s'encarrega de cridar a la funció que connecta amb la BD.
- **transaction.js**: Conté les funcions implementades que connecten amb la BD.
- **MySQLConnection.js**: S'encarrega d'establir la connexió amb la BD MySQL amb usuari i contrasenya.



**Figura 26:** Estructura de l'API

Per posar en marxa l'API simplement caldrà executar el fitxer `server.js`

```
1 node server.js
```

A mesura que l'API va rebent peticions, imprimirà per pantalla quina funció a executat. A continuació es mostra una captura amb l'execució del servidor i els missatges d'algunes funcions que ha rebut l'API.

```

x-INT ~/Work/TFM/Code/API [master|+ 3...2]
18:49 $ node server.js
Server Listening on port 55555
User: cartorse logged

getUser executed

getAllMaps executed

getAllSkillsByMap executed

getUser executed

getMapsById executed

```

**Figura 27:** Execució de l'API i visualització de missatges

Amb la intenció d'exemplificar com es gestionen les peticions que rep l'API, a continuació es mostraran les línies de codi juntament amb les corresponents explicacions del processament d'una petició. Concretament es mostrarà la gestió de la petició explicada anteriorment al subapartat (17.2.1) Pàgina Web encarrega de mostrar totes les *skills* associades a un *skill map*.

En primer lloc, quan es rep la petició, aquesta va dirigida a un *endpoint* determinat. A la nostra API, com ja s'ha dit, el fitxer `routes.js` és el que s'encarrega d'aquesta gestió. A continuació es visualitza el codi associat:

```

1  app.get('/api/skillsByMap/:id', function (req, res) {
2    isAuthenticated(req, res, function(){
3      transactions.getAllSkillsByMap(req.params.id, res);
4      console.log("getAllSkillsByMap executed\n")
5    }) // isAuthenticated
6  }) // get

```

**Codi 5:** Exemple de gestió d'una URL a l'API

En aquest cas, com ja se sabia es tracta d'una petició GET (línia 1). El primer que fa es comprovar que la petició prové d'un usuari autenticat i registrat a la BD, es per això que es crida a la funció *isAuthenticated* (línia 2). Tot seguit, s'efectua la crida a la funció que s'encarrega de contactar amb la BD i realitzar la operació en qüestió (línia 3), la implementació d'aquesta, es pot observar a continuació:

```

1 //Get All the Skills by map
2 this.getAllSkillsByMap = function (id,res) {
3   connection.init();
4   connection.acquire(function (err, con) {
5     var query='SELECT SS.ID_Soft_Skill, SS.Name,
6               SS.Description, SS.ID_Skill_Map
7               FROM Skill_Map SM, Soft_Skill SS
8               WHERE SS.ID_Skill_Map = SM.ID_Skill_Map
9               AND SS.Status=1 AND SM.ID_Skill_Map=?';
10    con.query(query, id, function (err, result) {
11      con.release();
12      res.send(result);
13    });
14  });
15 };//getAllSkillsByMap

```

**Codi 6:** Exemple de gestió de la petició realitzada a l'API

Aquesta funció estableix una connexió amb al BD (línia 3) i realitza una operació, en aquest cas una consulta, emmagatzemada a la variable *query* (línia 10) i que com ja s'ha dit retorna les *skills* associades a un *skill map*. Finalment, s'encarrega d'enviar el resultat retornat per la consulta (línia 12).

Les crides a funcions que es realitzen els fragments de codi anteriors i que repeteixen per a totes les crides que fan els funcionals a l'API, segueixen una seqüència de *callbacks* per garantir l'execució ordenada i lliure d'errors.

### Xifrat de la contrasenya

L'API com s'ha mencionat als requisits funcionals, xifra les contrasenyes. Per al xifrat, s'utilitza l'algoritme *bcrypt* amb una llibreria per NodeJS. *Bcrypt* es un algoritme per a realitzar funcions *hash* de contrasenyes. No s'explicaran els aspectes tècnics del funcionament d'aquest algoritme ja que escapen dels propòsits d'aquest treball, però si és important destacar l'ús que es fa de la llibreria.

```

mysql> SELECT ID_User,Password FROM User;
+-----+-----+
| ID_User | Password |
+-----+-----+
| 1 | $2a$10$sFuDppKmkrrjwFv56pPJie8y9frz/clxpTaERNhDuNAIMEYrqMgmC |
| 2 | $2a$10$bXv7hAv7l25kWlj4amCnj0c3IDWyD5Y.mLcB6Q4X4s.fognyGmVXS |
+-----+-----+
2 rows in set (0.00 sec)

```

**Figura 28:** Contrasenyes xifrades

El xifrat de la contrasenya, com és lògic permet ocultar a l'administrador i a qualsevol persona que poguera accedisca a la BD, les contrasenyes dels usuaris registrats en el sistema.

Tant a l'**inici de sessió** com al **registre** a la web, la contrasenya arriba al servidor per *https*, és a dir, amb una connexió segura però visualitzant-se en text pla. Conseqüentment, al fer el **registre** d'un usuari, es xifra la contrasenya abans d'emmagatzemar-se a la BD. A l'**inici de sessió**, es compara la contrasenya en text pla provinent del *login* amb la contrasenya xifrada emmagatzemada a la BD. Si la comparació es correcta l'autenticació serà correcta, en cas contrari no es podrà accedir al sistema.

### Establiment de sessió

Per a establir la sessió d'un usuari registrat, s'utilitza l'estàndard **JWT** (*Json Web Tokens*). El funcionament fonamental d'aquest estàndard es basa en la creació de *tokens* d'accés. Quan un usuari inicia sessió i el servidor respon amb *OK*, aquest li envia un *token*, de manera que a partir d'aquest moment totes les peticions que realitzi el client inclouran aquest *token* per a poder accedir als recursos del servei. Aquests *tokens* tenen un temps de validesa que correspon al temps que durarà la sessió a la web. A continuació es pot observar l'estructura d'un *token* **JWT**:

```
1 eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
2 eyJ1c2VybmFtZSI6ImNhcncvcnNlIiwiaWF0IjoxNTI3MTcxMDY3LCJleHAiOjE1M.
3 PDGluV6nr7RUm7AxajceokXaGcMo23f_Km688YLH-B0
```

**Codi 7:** Estructura d'un token JWT

Es pot observar que són tres *strings* separades per punts, on cadascuna d'elles té un significat. No es detallarà el significat d'aquestes *strings*, però si que es mostraran les línies de codi corresponents a la creació del *token*

```
1  bcrypt.compare(body.Password, pass, function(err, bool) {
2    if(bool==true){
3      var token = jwt.sign({ username: body.Username }, config.secret
4        , {
5          expiresIn: 1*60*60
6        });
7      res.send({ auth: true, username: body.Username, user: body, token:
8        token });
9    }//if
10   else{
11     res.send({ auth: false, username: body.Username, user: body, });
12   }//else
13 });//bcrypt
```

**Codi 8:** Gestió del *token* JWT a l'API

Aquest fragment de codi, correspon a una part de la funció d'inici de sessió (*login*). En primer lloc compara les contrasenyes per veure si coincideixen (línia 1), en cas afirmatiu, es genera un token vàlid amb un temps de validesa (línies 3-4) i s'envia al client (línia 6)

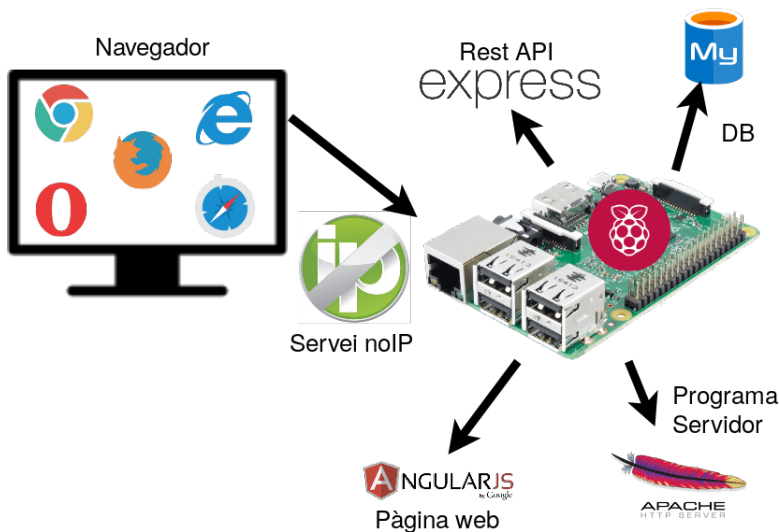
### 17.3 Annex - Fase de proves

En aquest annex es detallaran les proves realitzades per a validar el correcte funcionament del sistema. Per tal de donar un enfocament pràctic a la secció es detallaran les proves tècniques que s'han fet, estructurades de la següent manera :

1. Proves amb la *Raspberry Pi 3*
2. Proves amb bot per Telegram
3. Proves amb Postman
4. Proves unitàries

#### 17.3.1 Proves amb la *Raspberry Pi 3*

A la *Raspberry Pi 3*, s'ha instal·lat i configurat un servidor Apache i un servei noIP. S'ha generat el projecte AngularJS i s'ha allotjat en una carpeta configurada per al servidor Apache. Paral·lelament, s'ha migrat l'API i la BD a la *Raspberry Pi* i s'ha configurat el servei noIP per tal de que els serveis de l'API i de la Web apunten a la nova direcció noIP. D'aquesta manera s'ha aconseguit generar un entorn més real i accessible des de qualsevol dispositiu.



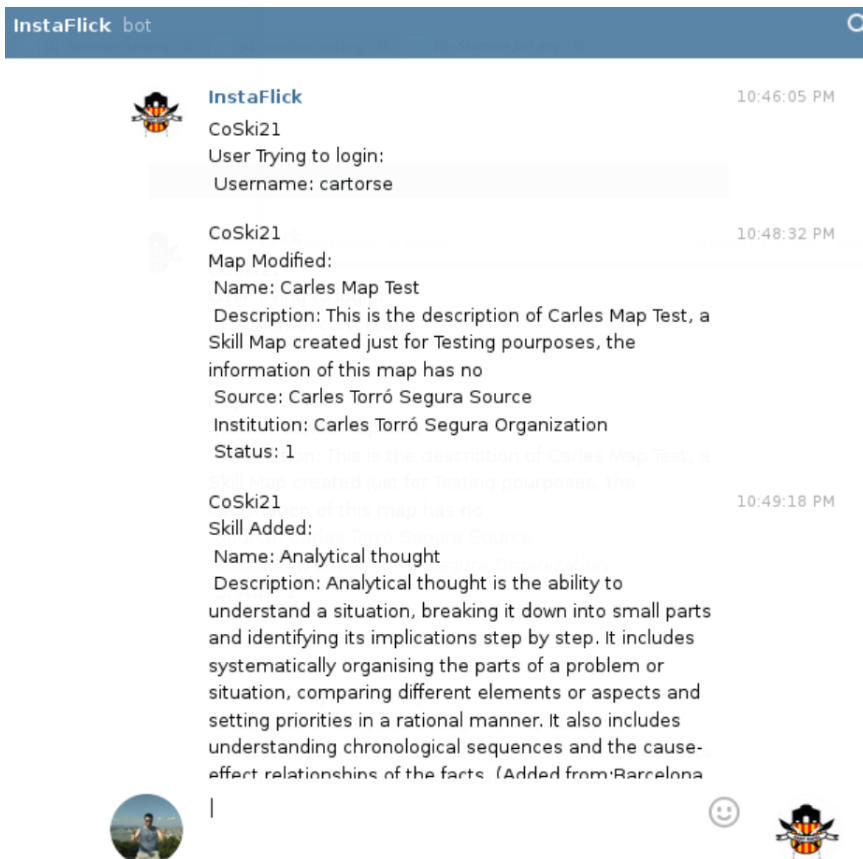
**Figura 29:** Esquema de les proves realitzades a la Raspberry Pi



A la imatge anterior es pot observar un esquema en l'estructura descrita anteriorment. Aquesta configuració ha permès provar la pàgina web per distints usuaris i navegadors i observar el comportament en un entorn més real observant d'aquesta manera els errors que podrien trobar-se els usuaris.

### 17.3.2 Proves amb bot per Telegram

S'ha desenvolupat un bot per Telegram amb una llibreria per NodeJS, que envia missatges des de l'API sobre les accions que es fan a la web. Aquests missatges s'envien a una única conversa de manera que qualsevol persona que intente utilitzar el bot no rebrà cap missatge. A continuació es pot visualitzar una captura de Telegram amb missatges generats pel bot.



**Figura 30:** Missatges bot de Telegram

És una forma de dur un seguiment sobre les accions que es fan a la web amb propòsits únicament de *testing*. En el moment que el sistema passara a un entorn real de producció aquesta funcionalitat s'eliminarà.

### 17.3.3 Proves amb Postman

Postman s'ha utilitzat cada vegada que s'ha afegit un *endpoint* nou a l'API. D'aquesta manera es pot validar el funcionament correcte de l'*endpoint* observant les dades que torna el programa. A continuació, es mostra una captura d'un petició realitzada amb Postman.

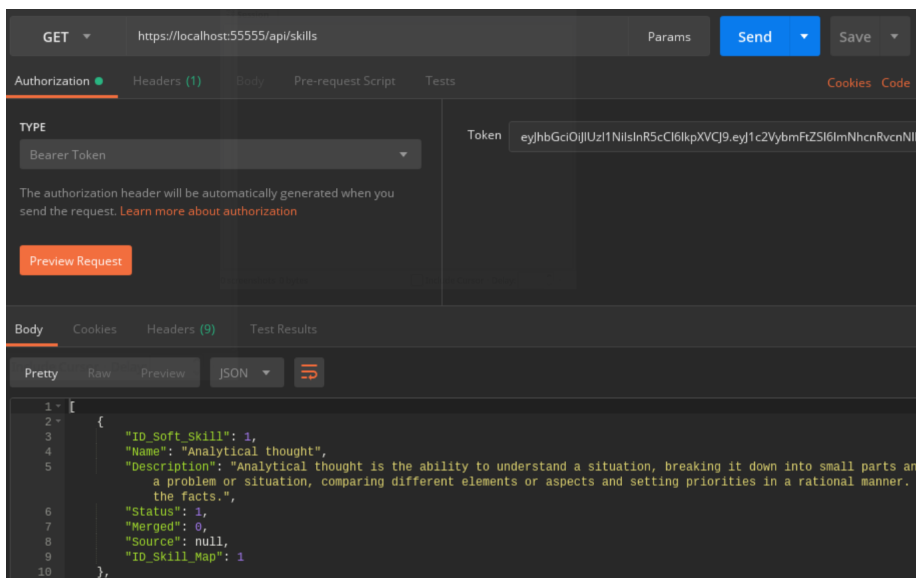


Figura 31: Petició amb Postman

Es pot observar a la imatge com es realitza una petició GET dirigida a l'*endpoint* `https://localhost:55555/api/skills`. També es pot observar com es fa ús d'un token vàlid per poder realitzar la petició. Finalment, a la part inferior, es pot visualitzar un fragment de la resposta retornada.

#### 17.3.4 Proves unitàries

Les proves unitàries s'han anat fent a mesura que s'ha anat realitzant la implementació del treball. Amb la implementació de cada funcionalitat nova s'han realitzat unes bateries de proves per tal de validar el correcte funcionament de cada requisit funcional.

Com s'ha dit en el paràgraf anterior, cada bateria de proves correspon a la validació d'un requisit funcional, i cada prova de cada bateria, estarà codificada amb la següent nomenclatura: TXXRFYY, sent XX el número de *test* i sent YY el número del requisit funcional que s'està provant.

Cada bateria de proves, s'expressarà en una taula, que tindrà les següents columnes:

- **Codi de la prova:** Expressat amb TXXRFYY
- **Descripció:** es descriurà la prova realitzada
- **Resultat esperat:** és el resultat que s'espera a l'efectuar la prova.
- **Resultat obtingut:** el resultat que s'obté a l'efectuar la prova

Per tal que el *test*, siga considerat vàlid, el resultat esperat i el resultat obtingut han de ser idèntics, en cas contrari el *test* no es considerarà vàlid. En els casos que es considere necessari s'inclouran observacions. A continuació es mostraran les bateries de proves que s'han realitzat sobre cada funcional, validant d'aquesta manera el correcte funcionament del treball.

**Bateria de proves RF01**

<b>T01RF01</b>	
<b>Descripció:</b>	Crear un usuari amb tots els camps existents
<b>Resultat esperat</b>	Creació correcta de l'usuari a la BD
<b>Resultat obtingut</b>	Es crea correctament de l'usuari a la BD

<b>T02RF01</b>	
<b>Descripció:</b>	Crear un usuari que ja existeix
<b>Resultat esperat</b>	Que no es cree l'usuari
<b>Resultat obtingut</b>	No es crea l'usuari

<b>T03RF01</b>	
<b>Descripció:</b>	Crear un usuari al que li falte algun camp obligatori
<b>Resultat esperat</b>	Que no s'habilite el botó per crear l'usuari
<b>Resultat obtingut</b>	No s'habilita el botó per crear l'usuari

<b>T04RF01</b>	
<b>Descripció:</b>	Modificar la contrasenya d'un usuari
<b>Resultat esperat</b>	Que s'actualitze la contrasenya a la BD
<b>Resultat obtingut</b>	S'actualitza la contrasenya a la BD

<b>T05RF01</b>	
<b>Descripció:</b>	Modificar la contrasenya d'un usuari de forma erronea
<b>Resultat esperat</b>	Que no actualitzi la contrasenya de l'usuari
<b>Resultat obtingut</b>	No actualitza la contrasenya de l'usuari

<b>T06RF01</b>	
<b>Descripció:</b>	Afegir a l'usuari una foto de perfil molt pesada
<b>Resultat esperat</b>	Que no s'afegisca la fotografia
<b>Resultat obtingut</b>	No s'afegix tota la fotografia

<b>T07RF01</b>	
<b>Descripció:</b>	Esborrar un usuari
<b>Resultat esperat</b>	Ha de desaparèixer de la BD
<b>Resultat obtingut</b>	S'esborra correctament

**Taula 9:** Bateria de proves RF01

**Bateria de proves RF02**

<b>T01RF02</b>	
<b>Descripció:</b>	Crear modificar esborrar o llistar un Skill Map
<b>Resultat esperat</b>	Observar les modificacions a la BD
<b>Resultat obtingut</b>	S'observen els resultats

<b>T02RF02</b>	
<b>Descripció:</b>	Repetir el test T01RF02 sense algun camp obligatori
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

<b>T03RF02</b>	
<b>Descripció:</b>	Repetir el test T01RF02 per a un mapa d'un altre usuari
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

**Taula 10:** Bateria de proves RF02**Bateria de proves RF03**

<b>T01RF03</b>	
<b>Descripció:</b>	Crear modificar esborrar o llistar una Soft Skill
<b>Resultat esperat</b>	Observar les modificacions a la BD
<b>Resultat obtingut</b>	S'observen els resultats

<b>T02RF03</b>	
<b>Descripció:</b>	Repetir el test T01RF03 sense algun camp obligatori
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

<b>T03RF03</b>	
<b>Descripció:</b>	Repetir el test T01RF03 per a un soft skill d'altre usuari
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

**Taula 11:** Bateria de proves RF03

**Bateria de proves RF04**

<b>T01RF04</b>	
<b>Descripció:</b>	Visualitzar l'anàlisi d'intensitat d'un mapa propi
<b>Resultat esperat</b>	Observar l'anàlisi amb la taula pertinent'
<b>Resultat obtingut</b>	S'observa l'anàlisi

<b>T02RF04</b>	
<b>Descripció:</b>	Repetir el test T01RF04 per a un mapa no propi
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

**Taula 12:** Bateria de proves RF04**Bateria de proves RF05**

<b>T01RF05</b>	
<b>Descripció:</b>	Tractar de fer el merging amb 1 sola soft skill
<b>Resultat esperat</b>	El botó per fer el merging no apareix
<b>Resultat obtingut</b>	Es crea correctament de l'usuari

<b>T02RF05</b>	
<b>Descripció:</b>	Fer el merging de 2 soft skills
<b>Resultat esperat</b>	S'ha de crear la nova merged soft skill
<b>Resultat obtingut</b>	S'ha creat la nova merged soft skill

<b>T03RF05</b>	
<b>Descripció:</b>	Fer el merging de més de 2 skills
<b>Resultat esperat</b>	S'ha de crear la nova merged soft skill
<b>Resultat obtingut</b>	S'ha creat la nova merged soft skill

<b>Observacions:</b>	Les recerques buides mostren totes les coincidències
----------------------	--

**Taula 13:** Bateria de proves RF05

**Bateria de proves RF06**

<b>T01RF06</b>	
<b>Descripció:</b>	Fer cluster al 90% en Carles Soft Skill 1
<b>Resultat esperat</b>	Observar 9 Soft Skills
<b>Resultat obtingut</b>	S'observen les 9 Soft Skills

<b>T02RF06</b>	
<b>Descripció:</b>	Fer cluster al 50% en Carles Map Test
<b>Resultat esperat</b>	Observar 9 Soft Skills
<b>Resultat obtingut</b>	S'observen les 9 Soft Skills

<b>T03RF06</b>	
<b>Descripció:</b>	Fer cluster al 0% en Carles Map Test
<b>Resultat esperat</b>	Apareixen totes les soft skills
<b>Resultat obtingut</b>	S'ha creat la nova merged soft skill

<b>T04RF06</b>	
<b>Descripció:</b>	Repetir el procediment aplicant el remove stop words
<b>Resultat esperat</b>	Observar com generalment apareixen menys skills
<b>Resultat obtingut</b>	Apareixen menys skills

<b>Observacions:</b>	Amb les stop words el clustering serà més real
----------------------	--

**Taula 14:** Bateria de proves RF06**Bateria de proves RF07**

<b>T01RF07</b>	
<b>Descripció:</b>	Generar el PDF per a mapes propis
<b>Resultat esperat</b>	Visualitzar el PDF generat
<b>Resultat obtingut</b>	S'observa el PDF

<b>T02RF07</b>	
<b>Descripció:</b>	Repetir el test T01RF07 per a un mapa no propi
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

**Taula 15:** Bateria de proves RF07

**Bateria de proves RF08**

<b>T01RF08</b>	
<b>Descripció:</b>	Iniciar sessió amb un usuari existent contrasenya correcta
<b>Resultat esperat</b>	Visualitzar el PDF generat
<b>Resultat obtingut</b>	S'observa el PDF

<b>T02RF08</b>	
<b>Descripció:</b>	Iniciar sessió un usuari existent contrasenya correcta
<b>Resultat esperat</b>	No s'activarà el botó encarregat de permetre l'operació
<b>Resultat obtingut</b>	No s'habilita el botó

<b>T03RF08</b>	
<b>Descripció:</b>	Iniciar sessió amb un usuari no existent
<b>Resultat esperat</b>	Visualitzar el PDF generat
<b>Resultat obtingut</b>	S'observa el PDF

**Taula 16:** Bateria de proves RF08

La resta de requisits funcionals queden validats en les bateries anteriors, ja que són requisits referents als formularis, alertes i notificacions. Per a més informació dels requisits tornar a consultar a l'apartat (7) Requeriments inicials

**17.4 Annex - Instal·lació**

En aquest annex es detallaran els passos que s'han de dur a terme per a instal·lar la ferramentada desenvolupada. S'especificarà el software a instal·lar, posteriorment s'explicarà com iniciar la ferramentada en local i finalment com s'hauria de configurar per a utilitzar-la en un servidor.

Com ja s'ha dit anteriorment a l'apartat (5.3) Ferramentes, l'eina proporcionada s'ha desenvolupat utilitzant el sistema operatiu Linux, però realment també es podria instal·lar tant a Windows com a MacOS. L'explicació que es detallarà en els següents subapartats serà per màquines Linux.



### 17.4.1 *Software necessari*

Les tecnologies a utilitzar ja han sigut especificades a l'apartat (5.3) Ferramentes. Seran necessàries les següents versions del *software* que s'ha utilitzat.

- NodeJS - Versió 5.6.0
- Angular - Versió 4
- MySQL - Versió 5.7.22
- Express - Versió 5.6.0

Per descarregar el software anterior cal accedir a les pàgines web de cada ferramenta *software* i seguir les instruccions d'instal·lació.

### 17.4.2 *Posada en marxa en local*

El treball consta de tres components principals que són: la pàgina web, l'API i la Base de dades. Cada component esta allotjat en la seua corresponent carpeta, de manera que dintre de la carpeta principal del projecte, es podran observar tres subcarpetes

- **Web:** Aquesta carpeta contindrà la web. Configurada per a ser executada a la url `https://localhost:4200`.
- **API:** Contindrà l'API. Configurada per a ser executada a la `https://localhost:55555`.
- **Database:** Contindrà un script per configurar i crear automàticament la BD. Amb els mapes especificats a l'apartat (18) Estudis amb entitat pròpia.

La posada en marxa constaria dels següents passos:

1. **Pàgina Web:** simplement s'ha d'accedir la carpeta "Web" amb un intèrpret d'ordres i escriure:

```
1 ng serve
```

Aquesta ordre s'encarregarà d'iniciar la pàgina web.

2. **API:** Per poder utilitzar l'api, prèviament s'hauran d'instal·lar totes les dependències, per a fer açò, s'ha d'executar amb un intèrpret d'ordres dintre de la carpeta "API" la següent ordre.

```
1 sudo npm install
```

L'ordre anterior, només cal executar-la la primera vegada, ja que les dependències només és necessari instal·lar-les una vegada. Quan acabe l'execució, de l'anterior ordre, simplement caldrà teclejar el següent comandament per posar l'API en marxa:

```
1 node server.js
```

3. **Base de dades:** per a restaurar la BD, caldrà accedir al software de MySQL, típicament amb:

```
1 mysql -u username -p password;
```

A continuació executar l'*script* que restaura la BD amb:

```
1 source real_data.sql;
```

Una vegada s'han seguit tots els passos simplement caldrà accedir a la url `https://localhost:4200`

### 17.4.3 Implantació en un server

Per poder instal·lar el sistema a un servidor Linux, s'hauria de seguir exactament el mateix procediment descrit a l'apartat (17.3.1) Proves amb la *Raspberry Pi 3*, però utilitzant un servidor real en lloc d'una *Raspberry Pi 3*. Addicionalment ens faria falta un domini per a la pàgina web i tindre instal·lat tot el software necessari.

## 18 Estudis amb entitat pròpia

En aquesta secció es fa referència als *skill maps* que s'ha decidit introduir per defecte al nostre sistema. Cal destacar, que aquests seran mapes bloquejats sobre els que no es podran fer anàlisis d'intensitat ni modificacions. Sols es podran consultar les seues *soft skills*. Aquestes es podran afegir a mapes propis, i es podran utilitzar per a fer les tasques d'anàlisi corresponents al *merging* i *clustering*.

1. **Barcelona Activa:** Aquest mapa conte les *soft skills* que seria recomanable tindre desenvolupades per a treballar de la forma més òptima a la ciutat de Barcelona.
2. **UPV:** Aquest mapa format per 13 competències, determina quines son les competències transversals que s'han d'haver adquirit al finalitzar un grau a la Universitat Politècnica de València.
3. **Mossa:** Són les sigles de *Model Of Soft Skills Assessment*. Aquest mapa presenta 12 *soft skills* que la fundació italiana *Casa di Carita* considera necessàries per a la inserció en el món laboral.

S'ha decidit incloure aquests mapes perquè es consideren importants i complets. Per trobar més informació d'aquests mapes es poden consultar els següents enllaços:

- **Barcelona Activa:** <https://treball.barcelonactiva.cat>
- **UPV:** <http://www.upv.es/contenidos/COMPTRAN/>
- **Mossa:** <http://valorize.odl.org/outputs/I02%20-%20MOSSA%20VHSM.pdf>



# Bibliografia

- [1] Micah GodBoy. *Frontend Architecture for Design Systems*. 1a ed. Vol. 1. 1. The address: O'Reilly Media Inc, febr. de 2016. ISBN: 9781491926789.
- [2] Anton Moiseev Yakov Fain. *Angular Development with TypeScript*. 2a ed. Vol. 1. 1. The address: Manning Publications, juny de 2018. ISBN: 1617295348.
- [3] Robert C. Martin. *UML Para desarrolladores Java*. 1a ed. Vol. 1. 1. The address: Pearson Educacion, març de 2004. ISBN: 9788420541099.
- [4] Jayesh Agrawal. *MySQL Data Access API Development Using Express.JS, Node.JS*. 2017. URL: [www.c-sharpcorner.com/article/](http://www.c-sharpcorner.com/article/).
- [5] Walther Smulders. *Using Observables in Angular 4 to get data from an API service*. 2017. URL: [www.symbiotics.co.za](http://www.symbiotics.co.za).
- [6] Angular University. *Angular Security - Authentication With JSON Web Tokens (JWT)*. 2018. URL: <https://blog.angular-university.io/>.
- [7] Cory LaViska. *Hashing Passwords with Node.js and Bcrypt*. 2017. URL: <https://www.abeautifulsite.net/>.
- [8] Sebastian Eschweiler. *Using Bootstrap with Angular*. 2017. URL: <https://codingthesmartway.com>.
- [9] *W3Schools Online Web Tutorials*. <https://www.w3schools.com/>. Accessed: Abril 2018.
- [10] *Stack Overflow Where Developers Learn, Share and Build Careers*. <https://stackoverflow.com/>. Accessed: Març 2018.