

Maite Bernaus Gimeno

**CREACIÓ D'UNA APLICACIÓ WEB PER A LA CERCA DE PÈPTIDS
BIOACTIUS EN PROTEÏNES**

TREBALL DE FI DE GRAU

dirigit per la Dr. Marc Sánchez Artigas

Grau de Enginyeria Informàtica



UNIVERSITAT ROVIRA I VIRGILI

Tarragona

2021

Resum.

Aquest projecte consisteix en el disseny i implementació d'una aplicació web per a la cerca de pèptids bioactius en proteïnes. Aquesta resol la necessitat d'agilitzar aquest procés, ja que pot arribar a ser molt laboriós i lent. Per a realitzar aquest projecte s'ha emprat el framework web2py per a crear una aplicació web amb arquitectura model vista controlador. Addicionalment, s'ha dissenyat i implementant una base de dades amb MySQL, aconseguint una aplicació funcional dissenyada des de zero que resol una necessitat real en el món de la bioinformàtica. Durant el projecte s'han repassat i aprofundit diferents conceptes apresos durant el grau com pot ser l'anàlisi de requisits i el disseny d'una aplicació.

Resumen.

El proyecto consiste en el diseño e implementación de una aplicación web para la búsqueda de péptidos bioactivos en proteínas. Ésta resuelve la necesidad de agilizar este proceso, ya que puede llegar a ser laborioso y lento. Para realizar dicho proyecto se ha utilizado el framework web2py para crear una aplicación web con arquitectura modelo vista controlador. Adicionalmente, se ha diseñado e implementado una base de datos con mySQL, consiguiendo una aplicación funcional diseñada desde cero que resuelve una necesidad real en el mundo de la bioinformática. Durante el proyecto se han repasado y profundizado diferentes conceptos aprendidos durante el grado como el análisis de requisitos y el diseño de una aplicación.

Abstract.

The project consists of the design and implementation of a web application to search for bioactive peptides in proteins. It solves the need to speed up this process, which

can be laborious and time-consuming. To carry out this project, the web2py framework has been used to create a web application with a controller view model architecture. Additionally, a database has been designed and implemented with mySQL, achieving a functional application designed from scratch that solves a real need in the world of bioinformatics. During the project, different concepts learnt during the degree, such as requirements analysis and the design of an application, have been reviewed and deepened.

Índex

0	GLOSSARI	- 1 -
1	INTRODUCCIÓ	- 3 -
2	DESCRIPCIÓ GENERAL DEL PROJECTE	- 5 -
2.1	ENTORN, NECESSITAT I PREVISIONS D'ÚS	- 5 -
2.2	EINES I MATERIAL UTILITZAT	- 6 -
3	REQUISITS	- 8 -
3.1	REQUISITS FUNCIONALS	- 8 -
3.1.1	<i>Guió: Posada en Context</i>	- 8 -
3.1.2	<i>Guió: SearchProteins</i>	- 8 -
3.1.3	<i>Guió: SearchPeptides</i>	- 9 -
3.1.4	<i>Diagrama de casos d'ús</i>	- 11 -
3.1.5	<i>Especificacions textuais dels casos d'ús</i>	- 12 -
3.2	REQUISITS NO FUNCIONALS	- 20 -
3.2.1	<i>Requisits no funcionals del producte</i>	- 20 -
3.2.2	<i>Requisits no funcionals del procés</i>	- 21 -
4	ANÀLISI DELS REQUISITS FUNCIONALS	- 22 -
5	DISSENY	- 38 -
5.1	ARQUITECTURA I ORGANITZACIÓ DE FITXERS DE L'APLICACIÓ	- 38 -
5.2	DISSENY DE LA INTERFÍCIE GRÀFICA	- 42 -
5.3	DISSENY DE LA BASE DE DADES	- 48 -
6	IMPLEMENTACIÓ	- 52 -
6.1	IMPLEMENTACIÓ DE LA BASE DE DADES.....	- 53 -
6.2	IMPLEMENTACIÓ DE LA SOLUCIÓ DELS CASOS D'ÚS.....	- 55 -
7	AVALUACIÓ	- 60 -
8	CONCLUSIONS	- 67 -
9	RECURSOS UTILITZATS	- 68 -

Índex de taules

TAULA 1. PROVES REFERENTS A L'ACCÉS DE L'USUARI ADMINISTRADOR.	- 61 -
TAULA 2. PROVES REFERENTS A L'ADMINISTRACIÓ DE LES BASES DE DADES.	- 61 -
TAULA 3. PROVES REFERENTS A LA CERCA DE PORTEÏNES.	- 63 -
TAULA 4. PROVES REFERENTS A LA CERCA DE PÈPTIDS.	- 65 -

Índex de figuers

FIGURA 1. DIAGRAMA DE CASOS D'ÚS DE L'APLICACIÓ PEPTIDEGO.....	- 11 -
FIGURA 2. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 00.LOGÍN.....	- 22 -
FIGURA 3. DIAGRAMA DE CLASSES DEL CAS D'ÚS 00.LOGÍN.....	- 22 -
FIGURA 4. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 00.LOGÍN.....	- 23 -
FIGURA 5. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 01.LOGOUT.....	- 23 -
FIGURA 6. DIAGRAMA DE CLASSES DEL CAS D'ÚS 01.LOGOUT.....	- 24 -
FIGURA 7. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 01.LOGÍN.....	- 24 -
FIGURA 8. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 02.MANAGEDATA.....	- 24 -
FIGURA 9. DIAGRAMA DE CLASSES DEL CAS D'ÚS 02.MANAGEDATA.....	- 25 -
FIGURA 10. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 02.MANAGEDATA.....	- 25 -
FIGURA 11. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 03.MANAGEPROTEINS_ORGANISM.....	- 26 -
FIGURA 12. DIAGRAMA DE CLASSES DEL CAS D'ÚS 03.MANAGEPROTEINS_ORGANISM.....	- 26 -
FIGURA 13. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 03.MANAGEPROTEINS_ORGANISM.....	- 27 -
FIGURA 14. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 04.MANAGEPEPTIDES_FILES.....	- 28 -
FIGURA 15. DIAGRAMA DE CLASSES DEL CAS D'ÚS 04.MANAGEPEPTIDES_FILES.....	- 28 -
FIGURA 16. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 04.MANAGEPEPTIDES_FILES.....	- 29 -
FIGURA 17. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 05.SEARCHPROTEINS.....	- 29 -
FIGURA 18. DIAGRAMA DE CLASSES DEL CAS D'ÚS 05.SEARCHPROTEINS.....	- 30 -
FIGURA 19. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 05.SEARCHPROTEINS.....	- 30 -
FIGURA 20. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 06.SEARCHBIOPEPTIDES.....	- 31 -
FIGURA 21. DIAGRAMA DE CLASSES DEL CAS D'ÚS 06.SEARCHBIOPEPTIDES.....	- 31 -
FIGURA 22. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 06.SEARCHBIOPEPTIDES.....	- 32 -
FIGURA 23. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 07.SEARCH.....	- 32 -
FIGURA 24. DIAGRAMA DE CLASSES DEL CAS D'ÚS 07.SEARCH.....	- 33 -
FIGURA 25. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 07.SEARCH.....	- 33 -
FIGURA 26. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 08.VIEWMANUAL.....	- 34 -
FIGURA 27. DIAGRAMA DE CLASSES DEL CAS D'ÚS 08.VIEWMANUAL.....	- 34 -
FIGURA 28. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 08.VIEWMANUAL.....	- 34 -
FIGURA 29. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 09.SHOWRESULTS.....	- 35 -
FIGURA 30. DIAGRAMA DE CLASSES DEL CAS D'ÚS 09.SHOWRESULTS.....	- 35 -
FIGURA 31. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 09.SHOWRESULTS.....	- 36 -
FIGURA 32. DIAGRAMA D'ACTIVITATS DEL CAS D'ÚS 10.REMOVEOLDDATA.....	- 37 -
FIGURA 33. DIAGRAMA DE CLASSES DEL CAS D'ÚS 10.REMOVEOLDDATA.....	- 37 -
FIGURA 34. DIAGRAMA DE SEQÜÈNCIES DEL CAS D'ÚS 10.REMOVEOLDDATA.....	- 37 -
FIGURA 35. ESTRUCTURA DELS DIRECTORIS I FITXERS DE L'APLICACIÓ PEPTIDEGO.....	- 38 -
FIGURA 36. PÀGINA PRINCIPAL DE L'APLICACIÓ (MODEL SELECCIÓ DE 2 OPCIONS PRINCIPALS).....	- 43 -
FIGURA 37. PÀGINA PRINCIPAL DE L'ADMINISTRACIÓ DE DADES (MODEL SELECCIÓ DE 2 OPCIONS PRINCIPALS).....	- 43 -
FIGURA 38. PÀGINA DE LOGÍN.....	- 44 -
FIGURA 39. PÀGINA D'ADMINISTRACIÓ DE DADES REFERENTS A ORGANISMES-PROTEÍNES (MODEL ACCIONS BASES DE DADES). - 44 -	
FIGURA 40. PÀGINA D'ADMINISTRACIÓ DE DADES REFERENTS A PÈPTIDS (MODEL ACCIONS BASES DE DADES).	- 44 -
FIGURA 41. FORMULARI PER A INTRODUIR LA INFORMACIÓ REFERENTS ALS PÈPTIDS EN LA CERCA DE PÈPTIDS. (MODEL FORMULARI).	- 45 -
FIGURA 42. FORMULARI PER A INTRODUIR LA INFORMACIÓ REFERENTS ALS PÈPTIDS EN LA CERCA DE PROTEÍNES (MODEL FORMULARI).	- 46 -
FIGURA 43. FORMULARI PER A INTRODUIR LA INFORMACIÓ REFERENTS ALS PARÀMETRES DE PURIFICACIÓ I LA (MODEL FORMULARI).	- 46 -
FIGURA 44. PÀGINA DE TAULA DE RESULTATS I INFORMACIÓ DE LA CERCA.....	- 47 -
FIGURA 45. PÀGINA DE MÉS INFORMACIÓ SOBRE UN RESULTAT.....	- 47 -
FIGURA 46. DIAGRAMA ENTITAT-RELACIÓ EN EL QUE ES REPRESENTEN LES ENTITATS I RELACIONS COMUNS EN TOTES LES CERQUES.	- 48 -
FIGURA 47. DIAGRAMA ENTITAT-RELACIÓ EN EL QUE ES REPRESENTEN LES ENTITATS I RELACIONS PARTICULARS DE LES CERQUES DE PÈPTIDS.	- 49 -

FIGURA 48. DIAGRAMA ENTITAT-RELACIÓ EN EL QUE ES REPRESENTEN LES ENTITATS I RELACIONS PARTICULARS DE LES CERQUES DE PROTÉINES.....	- 49 -
FIGURA 49. EXEMPLE DE CONTINGUT FITXER APPLICATION_PATHS.TXT.....	- 52 -
FIGURA 50. FUNCIO OBTAIN_APPLICATION_PATH(DIRE) DEL FITXER PRIVATE/GLOBAL_FUNCTIONS.PY	- 52 -
FIGURA 51. FUNCIO DEL CONTROLADOR REFERENT A LA VISTA USER.HTML.....	- 55 -
FIGURA 52. CAPTURA DEL CODI REFERENT A LA FUNCIO DEL CONTROLADOR DE LA VISTA ADMIN.HTML (PÀGINA PRINCIPAL D'ADMINISTRACIÓ).....	- 55 -
FIGURA 53. FUNCIO ADD_PROTEIN DEL MÒDUL MODULES/DEF_ORGANISM.PY.....	- 57 -
FIGURA 54. FUNCIO VALIDATETEXT() EN EL FITXER /VIEWS/DEFAULT/SEARCHBIOPEPTIDS.HTML.....	- 58 -

0 Glossari

Per fer més entenedor alguns dels conceptes s'ha decidit incorporar un glossari dels temes de caràcter biotecnològic utilitzats.

- **Aminoàcid:** compost orgànic que conté els grups funcionals amino (-NH₂) i carboxil (-COOH), així com una cadena lateral diferent en cada aminoàcid. (s'anomenen residus si formen part d'una seqüència peptídica).
- **Pèptid:** són un tipus de molècules formades per la unió de diversos aminoàcids mitjançant enllaços peptídics.
- **Proteïna:** pèptids que generalment tenen més de 100 residus, una complexitat molt diversa i una funció dins del seu entorn metabòlic original.
- **Seqüència peptídica:** ordre en què els aminoàcids s'encadenen dins dels pèptids i proteïnes.
- **Proteasa / Peptidasa / Enzim de restricció:** enzims (tipus de proteïnes) capaços de trencar enllaços peptídics de pèptids (hidròlisis) produint pèptids més petits o aminoàcids lliures.
- **Etapa:** procés format per una digestió enzimàtica i purificació. És a dir, hidròlisis d'un o diversos pèptids utilitzant proteases i obtenció dels pèptids/aminoàcids d'interès resultants d'aquesta hidròlisis de forma aïllada.
- **In vitro:** s'utilitza per descriure experiments bioquímics que es duen a terme en un ambient artificial (fora d'un organisme viu).
- **In silico:** s'utilitza per descriure simulacions d'experiments biològics dutes a terme exclusivament amb un computador.

- **Format FASTA:** és un format de fitxer en text per a la representació de seqüències tant nucleotídiques com peptídiques, en què els nucleòtids i els aminoàcids es representen usant un codi d'una sola lletra. Una seqüència en format FASTA comença amb una capçalera de descripció d'una sola línia amb el símbol '>' al principi.
- **UniProt (*universal protein*):** és un repositori gratuït de dades sobre pèptids i proteïnes que el qual inclou Swiss-Prot (amb proteïnes confiables associades amb un alt grau d'anotacions) i TrEMB (que proporciona anotacions automatitzades per aquelles proteïnes que només troben en el Swiss-Prot).[1]
- **MEROPS:** base de dades amb informació de proteases i les proteïnes que les inhibeixen. [2]
- **PubMed:** és un motor de cerca gratuït per a accedir al MEDLINE, una base de dades bibliogràfiques de citacions i resums d'articles de recerca en biomedicina i ciències de la vida on cada un dels articles té un codi identificador anomenat PMID associat. [3]
- **Unitats i valors:**
 - **Concentració molar** (M o mol/L): unitat de concentració que indica el nombre de mols per litre de una substància en dissolució.
 - **Massa molecular** (Da o u): dalton o unitat de massa atòmica és una unitat de massa que equival a la dotzena part de la massa del carboni 12 en el seu estat fonamental ($1,660\ 538\ 86 \times 10^{-27}$ kg) utilitzada principalment per expressar masses de molècules.
 - **IC50:** equival a la concentració necessària d'una substància per inhibir la meitat d'un compost determinat. Sovint s'indica com pIC50 calculat com a $-\log(\text{IC}_{50} \text{ en M})$.

1 Introducció.

El projecte consisteix en la implementació d'una aplicació web per a la cerca de pèptids en proteïnes. Aquest s'ha realitzat durant el curs 2020-2021 en el grup de recerca en Quimioinformàtica i Nutrició en el Departament de bioquímica i biotecnologia de la Universitat Rovira i Virgili (URV).[4]

L'objectiu del projecte és implementar una aplicació web per tal d'agilitzar el procés de recerca d'obtenció de pèptids d'interès biotecnològic i biomèdic. Aquests pèptids es poden arribar a obtenir mitjançant processos de tall i purificació a partir de proteïnes procedents de fonts naturals biològiques, com per exemple, residus d'algunes indústries agroalimentàries. Aquest procés de cerca és rellevant per a la valorització de subproductes proteics d'origen natural com a font d'ingredients actius en cosmètica, veterinària i alimentació funcional i, fins i tot, tractament/prevenió de malalties.

A gran trets, l'aplicació té dos tipus de funcionalitats:

- La cerca de la possible obtenció d'un pèptid proporcionat per l'usuari en proteïnes agrupades per organismes.
- La cerca de la possible obtenció de pèptids amb activitat biològica d'interès en una proteïna o conjunt de proteïnes determinades per l'usuari.

Addicionalment, també s'ha incorporat en la interfície gràfica l'administració de les dades utilitzades (proteïnes organitzades per organisme de procedència i pèptids organitzat per funció, diana i efecte) per a realitzar aquestes cerques. També s'inclourà un manual d'usuari on s'informa de la procedència de les dades usades i com utilitzar l'aplicació.

Per a dur a terme el projecte, s'ha utilitzat l'entorn web2py. Aquest proporciona eines per a realitzar aplicacions web seguint l'arquitectura model-vista-controlador i la utilització de Python (un dels llenguatges de programació més emprats en l'àmbit biotecnològic i en

el qual estaven prèviament implementades al projecte algunes de les funcions necessàries per a realitzar la cerca). També s'ha fet ús dels llenguatges de programació HTML5 i JavaScript per la realització de part de la interfície gràfica web i MySQL per a la gestió de les dades relacionades amb les cerques.

Des de el punt de vista formatiu, s'espera complementar la formació del doble grau en Biotecnologia i Enginyeria Informàtica cursat implementant una aplicació d'àmbit bioinformàtic. També s'espera dissenyar un projecte informàtic des de zero que s'adapti a les necessitats d'un problema real, així com, des d'un punt de vista més personal, aprendre sobre la realització d'aplicacions web i el maneig de dades biotecnològiques d'una forma correcta més enllà dels conceptes apresos en el grau.

2 Descripció General del Projecte

2.1 Entorn, Necessitat i Previsions d'Ús

El projecte es realitza en el grup de recerca de Quimioinformàtica i Nutrició del Departament de Bioquímica i Biotecnologia de la Universitat Rovira i Virgili (URV). Un dels objectius d'aquest grup de recerca és la valorització de subproductes d'origen biològic, entre ells, pèptids i proteïnes. Una de les formes de revaloritzar aquest tipus de molècules és obtenir a partir d'ells altres pèptids amb activitats d'interès biotecnològic. Per fer-ho, primer de tot, es realitzen processos de digestió enzimàtica utilitzant proteases (*in silico* o *in vitro*). Seguidament, s'estudien els pèptids obtinguts per analitzar si alguna d'aquestes seqüències peptídiques té activitat d'interès biològic (ja sigui bibliogràficament o *in vitro*).

En el grup de recerca sorgeix la necessitat d'agilitzar aquest procés mitjançant una aplicació, ja que, és repetitiu i laboriós. A més a més, s'observa que aquest pot ser útil per altres entitats i grups de recerca i es planteja realitzar una aplicació web en la qual realitzar de forma autònoma aquest procés per tal de poder obtenir possibles resultats d'una forma més àgil. Posteriorment al projecte, es preveu utilitzar el servidor del grup de recerca per tal de proporcionar el servi web els membres del grup i a altres entitats a les quals els hi pugui ser d'interès.

2.2 Eines i Material Utilitzat

Per a la realització del projecte es parteix d'un codi base implementat en Python. Aquest consisteix en un mòdul amb algunes de les funcions necessàries per a realitzar la cerca d'un pèptid en una proteïna a partir del seu codi del UniProt. A més, inclou programari per a l'obtenció de les seqüències proteiques d'un organisme a partir del seu nom científic també procedent del UniProt (només Swiss-Prot). Cal tenir en compte que durant la fase d'estudi i compressió d'aquest, s'ha modificat el codi després de trobar alguns errors que afectava el correcte resultat de les cerques; i s'ha adaptat gairebé en la seva totalitat per a la seva utilització en l'aplicació final. També es proporciona dades de peptidases procedents del MEROPS en un document Excel, utilitzades en la digestió *in silico* de les seqüències peptídiques de la cerca, en la qual no s'ha realitzat cap modificació.

Per altra banda, s'han utilitzat un conjunt d'eines per tal d'efectuar el projecte:

- L'eina MagicDraw per a realitzar els esquemes UML del disseny de l'aplicació (diagrames de casos d'ús, seqüències, classes, activitat i seqüències). [5]
- L'eina diagram.net de GoogleDrive per a realitzar els esquemes entitat-relació referents a la base de dades. [5]
- El framework PyCharm per a la implementació i testeig dels diversos mòduls en Python abans d'incorporar-los a l'aplicació web. Aquest és un entorn de desenvolupament integrat (IDE) utilitzat específicament per la programació amb llenguatge Python. [6]
- El paquet de programari lliure XAAMP (Apache, MySQL, PHP i Perl), una distribució d'Apache la qual s'ha utilitzat principalment per al maneig i testeig de la base de dades creada amb del MySQL (MariaDB). [7]

- El framework web2py s'ha utilitzat per a crear l'aplicació web. Aquest és un entorn gratuït de codi obert que permet el desenvolupament d'aplicacions web en Python. L'elecció de la utilització d'aquest ha estat deguda principalment en l'experiència per part d'alguns membres del grup de recerca emprant aquesta eina. Addicionalment, el fet que sigui en Python dóna la facilitat d'incorporar el codi de cerca original d'una forma senzilla.

També s'ha fet servir Git per al control de versions dels mòduls implementats amb el framework PyCharm i GoogleDrive per a tenir copia de seguretat i control de versions de tot el programari.

Dins de les eines i materials utilitzats també cal mencionar l'ús de dades de les bases de dades externes de proteases (MEROPS), proteïnes (UniProtKB) i organismes (NCBI taxonimy). [1,2,8]

3 Requisites

3.1 Requisites Funcionals

Dins dels requisits funcionals de l'aplicació del projecte a continuació es detallen els guions utilitzats en forma d'enunciat i el diagrama de casos d'ús i especificacions d'aquests casos d'ús.

3.1.1 *Guió: Posada en Context*

S'ha de realitzar la cerca de pèptids bioactius d'interès en proteïnes a partir de les seves seqüències i una base de dades de proteases (enzims que tallen proteïnes). Com a resultat es vol obtenir les proteïnes que poden donar el pèptid d'interès juntament amb les etapes de purificació (els enzims necessaris per obtenir-lo i l'ordre en què s'hi ha d'aplicar).

Es vol una aplicació en la qual els usuaris podran consultar ajudes contextuais durant la seva utilització. També ha d'incloure un manual d'usuari per a més informació que contingui una explicació de la procedència de les dades utilitzades i les accions que es poden realitzar.

3.1.2 *Guió: SearchProteins*

S'ha de poder cercar una seqüència peptídica d'interès (en format FASTA) en les seqüències de les proteïnes dels organismes elegits per l'usuari, mitjançant una base de dades amb els punts de tall de les peptidases més freqüents. Addicionalment, es vol que l'usuari tingui la possibilitat de poder elegir diversos paràmetres de purificació: el nombre màxim d'etapes (d'1 a 3) i les diferències mínimes de pes molecular per a poder separar els fragments amb el pèptid d'interès de la resta (amb valors de 3.000, 10.000, 30.000, 50.000 i 100.000 Da). Addicionant s'ha de poder elegir si vol realitzar la cerca utilitzant totes les proteases o només les més habituals (comercials).

El resultat d'aquesta cerca ha de ser una taula de resultats amb: el codi del UniProt de la proteïna, el nom de la proteïna, l'organisme d'on procedeix, el nombre d'etapes usades,

els pèptids finals que s'obtenen, els enzims utilitzats en cada etapa i les fraccions de pes molecular dels pèptids obtinguts. Aquests resultats han de tenir l'opció de ser filtrats i ordenats pels diferents paràmetres o ser descarregats en formats csv. Addicionalment, per a cada un dels resultats s'ha de poder consultar amb detall, mitjançant un botó accessible en la mateixa taula, informació completa sobre el resultat incloent la traçabilitat de les etapes realitzades. Per cada etapa es demana mostrar la peptidasa usada, la posició de tall de la peptidasa, els fragments obtinguts, la fracció de pes molecular filtrada i els pèptids retinguts en aquesta.

Per altra banda, es busca tenir les dades amb les seqüències de les proteïnes presents en l'UniProt agrupades per organismes, així com el nom comú i un enllaç amb més informació sobre l'organisme per tal de poder identificar-lo fàcilment. Es demana tenir una interfície gràfica per administrar aquestes dades. En aquesta s'ha de poder afegir o eliminar les proteïnes d'un organisme determinat a partir del seu nom científic. Es vol també tenir la possibilitat d'actualitzar tota la base de dades en qualsevol moment creant una nova versió d'aquesta (actualitzant totes les seqüències del UniProt de tots els organismes que hi ha en la versió anterior). En aquest cas, per a l'administració es requereix tenir un usuari administrador amb una contrasenya per tal que només ell pugui accedir a la gestió de les dades.

3.1.3 *Guió: SearchPeptides*

S'ha de poder fer una cerca de pèptids en una o varies proteïnes (introduïdes en format FASTA o amb el codi del UniProt), mitjançant una base de dades amb els punts de tall de les peptidases més freqüents. Els pèptids, elegits per l'usuari segons la seva activitat, diana i efecte determinat, s'han d'incloure en una base de dades local. També es demana que l'usuari tingui la capacitat d'elegir que els pèptids a cercar tinguin un mínim valor de bioactivitat. Addicionalment, com en la primera cerca, l'usuari també ha de poder elegir

el nombre màxim de etapes i les diferències mínimes de pes molecular per a poder separar els fragments amb el pèptid d'interès de la resta. A més a més, s'ha de poder elegir si es vol realitzar la cerca utilitzant totes les proteases o només les més habituals (comercials).

La sortida d'aquesta cerca ha de ser una taula de resultats amb: la seqüència, l'activitat, diana i efecte del pèptid, nombre d'etapes usades i els pèptids finals que s'obtenen, així com els enzims utilitzat en cada etapa i la diferència de pes molecular entre el pèptid buscat i la resta de pèptids dels quals s'ha de separar en cada etapa. Aquest resultats han de tenir l'opció de ser filtrats i ordenats pels diferents paràmetres o ser descarregats en formats *csv*. Addicionalment, per a cada un dels resultats s'ha de poder consultar amb detall la seva informació (incloent informació sobre la procedència dels valors d'activitat del pèptid obtingut) i traçabilitat de les etapes realitzades igual que en la primera cerca.

Per altra banda, es vol tenir una base de dades amb els pèptids més usats. Es tenen fitxers *csv* amb el nom en format *bioactivitat-diana-efecte.csv* els quals es volen emmagatzemar (afegir al sistema), actualitzar o esborrar. Aquest fitxers contenen com a mínim la seqüència peptídica en una sola lletra, el PMID o DOI de la referència on s'ha trobat, la localització d'aquesta informació en l'article (pàgina, taula...), el valor IC_{50} amb les unitats originals i el pX (valor de bioactivitat equivalent a $-\log(IC_{50})$ on IC_{50} té unitats de M). Els pèptids també poden contenir informació complementaria dels quals interessa guardar el tipus de inhibició. Es vol tenir una interfície gràfica per administrar les dades d'aquests fitxers de pèptids. En aquest cas per a l'administració es requereix també tenir un usuari administrador amb una contrasenya per tal de que només ell pugui accedir a la gestió de les dades.

3.1.4 Diagrama de casos d'ús

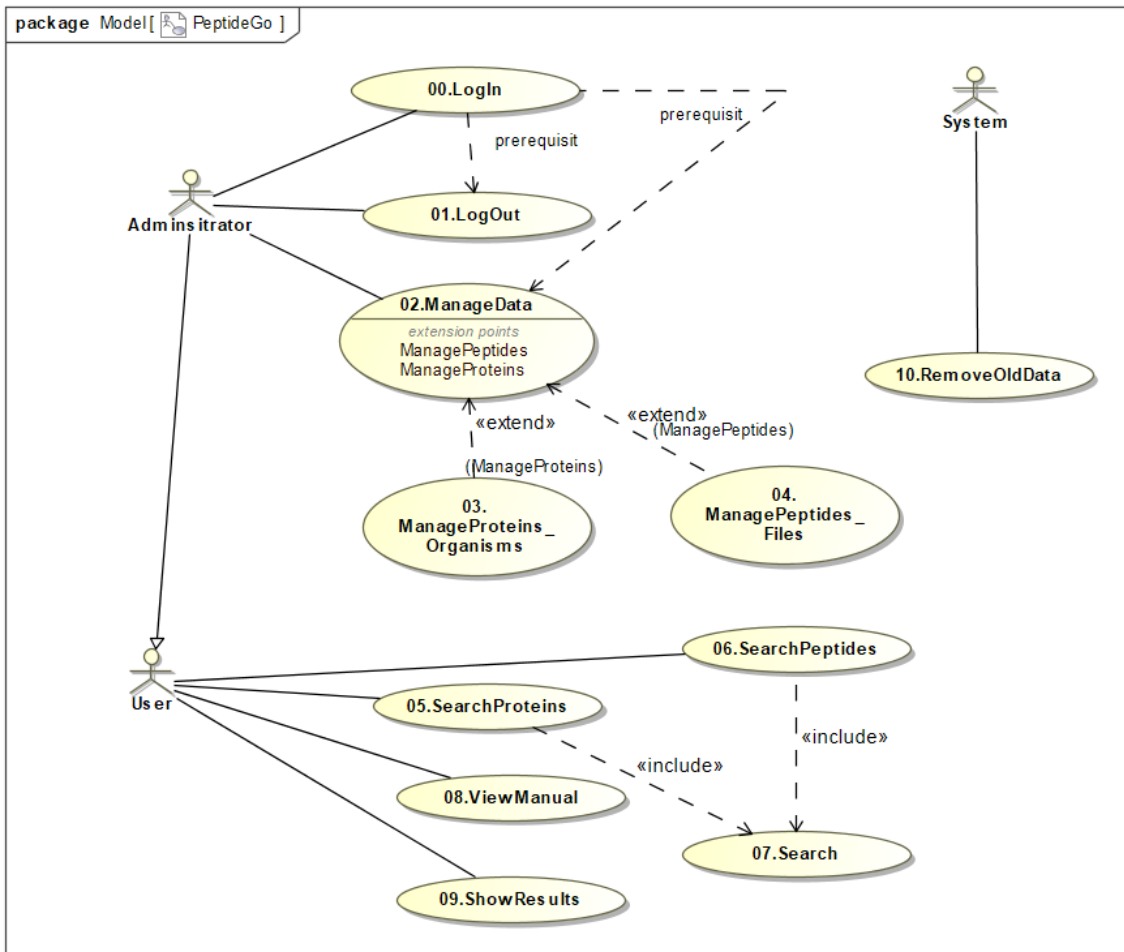


Figura 1. Diagrama de casos d'ús de l'aplicació PeptideGo.

3.1.5 Especificacions textuais dels casos d'ús

A continuació s'indiquen les especificacions textuais dels 11 casos d'ús definits en l'apartat 3.1.4.

00. LogIn

Resum de la funcionalitat: Identificar l'usuari com administrador i donar permís d'accés a la interfície d'administració de les dades.

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Administrador.**

Precondició: L'usuari actual no té permisos d'administrador.

Postcondició: L'usuari actual té permisos d'administrador.

Procés normal principal:

1. L' **Administrador** indica al sistema que vol fer logIn.
2. El sistema demana nom d'usuari i contrasenya del administrador de les bases de dades.
3. L' **Administrador** introdueix les dades.
4. El sistema comprova que les dades són correctes.
5. El sistema mostra la pàgina principal de administració de les bases de dades.

Alternatives de procés i excepcions:

4a. L' **Administrador** posa nom d'usuari o contrasenya incorrectes:

- 4a1. El sistema detecta que les dades són incorrectes.
- 4a2. El sistema mostra al usuari un missatge d'error.
- 4a3. El sistema torna al pas 1.

01. LogOut

Resum de a funcionalitat: Deixar d'identificar l'usuari com administrador.

Paràmetres d'entrada: cap

Paràmetres de sortida: cap

Usuaris: **Administrador**

Precondició: L'usuari actual té permisos de administrador.

Postcondició: L'usuari actual passa a no tenir permisos de administrador.

Procés normal principal:

1. L' **Administrador** indica al sistema que vol fer logOut.
2. El sistema retira els permisos d'administració de l'usuari.
3. El sistema mostra la pàgina principal de l'aplicació.

02.ManageData

Resum de la funcionalitat: Addició, esborrat i/o actualització de les dades referents a proteïnes i pèptids utilitzades durant la realització de les cerques.

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Administrador, Usuari.**

Precondició: L'usuari actual ha de tenir permisos de administrador.

Postcondició: Modificacions realitzades per l'usuari en les dades referents a proteïnes i pèptids de la base de dades.

Procés normal principal:

1. El sistema comprova que l'usuari té permís d'administració.
2. El sistema mostra les opcions a administrar (proteïnes o pèptids).
3. L'**Administrador** elegeix la opció d'administrar les dades de proteïnes/organismes.
4. El sistema executa el cas d'ús 03.ManageProteins Organisms.

Alternatives de procés i excepcions:

2a. L'**Usuari** no té permís d'administració:

2a1. El sistema executa el cas d'ús 00. LogIn.

2a2. El sistema torna al pas 2.

3a. L'**Administrador** vol administrar les dades dels fitxers que contenen els pèptids:

3a1. L'**Administrador** elegeix l'opció d'administrar les dades dels fitxers que contenen els pèptids.

3a2. El sistema executa el cas d'ús 04.ManagePeptides Files.

03.ManageProteins Organisms

Resum de a funcionalitat: Administrar les dades referents les proteïnes agrupades per organismes utilitzades en al cerca de proteïnes (afegir o esborrar proteïnes d'un organisme o actualitzar tota la base de dades referent a proteïnes d'organismes).

Paràmetres d'entrada: cap

Paràmetres de sortida: cap

Usuaris: **Administrador**

Precondició: L'usuari actual ha de tenir permisos de administrador.

Postcondició: Modificacions realitzades per l'usuari en les dades referents a proteïnes de la base de dades.

Procés normal principal:

1. El sistema mostra la finestra amb diàleg per afegir/actualitzar/esborrar proteïnes d'organismes.
2. L'**Administrador** proporciona el nom científic del organisme.
3. L'**Administrador** elegeix la opció d'afegir organisme.
4. El sistema cerca el nom/s comú/ns del organisme.
5. El sistema cerca l'enllaç a la taxonomia del organisme.
6. El sistema afegeix/actualitza l'organisme (nom científic, nom comú i enllaç taxonomia) en la base de dades.
7. El sistema obté els codis de les proteïnes del organisme del UniProt
8. El sistema es descarrega els fitxers FASTA de les proteïnes de l'organisme del UniProt.
9. El sistema obté les seqüències d'aquestes proteïnes.
10. El sistema guarda/actualitza les seqüències d'aquestes proteïnes a la base de dades.
11. El sistema crea un relació per cada una de les proteïnes amb l'organisme en la base de dades.
12. El sistema mostra a l'usuari que s'ha realitzat correctament la operació demanada.

Alternatives de procés i excepcions:

2a. L'**Administrador** vol actualitzar les proteïnes de tots dels organismes del sistema:

2a1. L'**Administrador** elegeix l'opció d'actualitzar la base de dades.

2a3. El sistema recopila el nom científics dels organismes de la versió actual de la base de dades.

2a4. El sistema cerca les proteïnes dels organismes en el UniProt.

2a5. El sistema actualitza la informació a la base de dades.

2a6. El sistema torna al pas 21.

3a. L'**Administrador** vol eliminar les proteïnes d'un organisme:

3a1. L'**Administrador** elegeix l'opció d'eliminar l'organisme.

3a2. El sistema esborra les dades de l'organisme de la base de dades. (nom científic, nom comú, enllaç i relació entre proteïna-organisme).

3a3. El sistema va al pas 12.

12a. S'ha produït algun error durant el procés d'afegir, esborra o actualitzar:

12a1. El sistema mostra un missatge d'error.

04.ManagePeptides_Files

Resum de a funcionalitat: : Administrar les dades referents als pèptids agrupats en fitxer segons activitat, diana i efecte utilitzades en al cerca de pèptids (afegir, actualitzar o esborrar la informació referent a cada un d'aquest grups).

Paràmetres d'entrada: cap

Paràmetres de sortida: cap

Usuaris: **Administrador**

Precondició: L'usuari actual ha de tenir permisos de administrador.

Postcondició: Modificacions realitzades per l'usuari en les dades referents a proteïnes de la base de dades.

Procés normal principal:

1. El sistema mostra finestra amb diàleg afegir/actualitzar/esborrar fitxer de pèptids.
2. L'**Administrador** selecciona un fitxer.
3. L'**Administrador** elegeix l'opció d'afegir fitxer.
4. El sistema comprova que no existeix el fitxer.
5. El sistema guarda el fitxer en les dades locals.
6. El sistema guarda la informació referent a les seqüències peptídiques, referencia, localització d'informació, IC50 original, pX i tipus d'inhibició en la base de dades.
7. El sistema mostra a l'usuari que s'ha realitzat correctament la operació.

Alternatives de procés i excepcions:

2a. L'**Administrador** vol eliminar un fitxer:

2a1. L'**Administrador** escriu el nom del fitxer a eliminar.

2a2. L'**Administrador** elegeix l'opció eliminar el fitxer.

2a3. El sistema comprova que existeix el fitxer.

2a4. El sistema esborra el fitxer localment.

2a5. El sistema esborra la seva informació referent de la base de dades.

2a6. El sistema torna al pas 5.

3a. L'**Administrador** vol actualitzar un fitxer:

3a1. L'**Administrador** elegeix l'opció d'actualitzar fitxer.

3a2. El sistema comprova que existeix el fitxer.

3a3. El sistema sobreescrui el fitxer local

3a4. El sistema actualitza les dades a la base de dades.

3a4. El sistema torna al pas 5.

5a. S'ha produït algun error durant el procés (existeix un fitxer amb el mateix nom al afegir el fitxer o no existeix un fitxer amb el mateix nom en cas de actualitzar o esborrar el fitxer):

5a1. El sistema mostra un missatge d'error.

05. SearchProteins

Resum de la funcionalitat: Efectuar la cerca de proteïnes que contengun un pèptid proporcionat per l'usuari tenint en compte els paràmetres de purificació.

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Usuari**.

Precondició: cap.

Postcondició: resultat de la cerca guardat en el sistema de forma persistent

Procés normal principal:

Procés normal principal:

1. El sistema demana les dades de input la seqüència del pèptid en format FASTA.
2. **L'Usuari** introdueix la seqüència del pèptid.
3. El sistema demana la elecció dels organismes en els quals es vol buscar el pèptid.
4. **L'Usuari** selecciona els organismes en els quals vol buscar el pèptid.
5. El sistema executa el cas d'ús 09.Search.

06. SearchPeptides

Resum de la funcionalitat: Efectuar la cerca de pèptids en una o varies proteïnes proporcionada per l'usuari tenint en compte els paràmetres de purificació.

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Usuari**.

Precondició: cap.

Postcondició: el resultat de la cerca guardat en el sistema de forma persistent.

Procés normal principal:

1. El sistema demana en quin format vol introduir la/les proteïnes on es vol cercar (codi UniProt o seqüència FASTA)
2. **L'Usuari** introdueix la/les proteïnes en el format escollit.
3. El sistema demana els tipus de pèptids que es volen buscar (activitats, dianes i efectes disponibles dels pèptids i mínima activitat) .
4. **L'Usuari** indica els tipus de pèptids a buscar.
5. El sistema demana la mínima bioactivitat dels pèptids a cercar.
6. **L'Usuari** indica el valor de mínima activitat dels pèptids a cercar segons el seu interès.
7. El sistema executa el cas d'ús 09.Search.

07.Search

Resum de la funcionalitat: Efectuar la cerca de pèptids en una proteïna proporcionada per l'usuari tenir en compte els paràmetres de purificació (nombre màxim d'etapes i mínim diferència de pes molecular).

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Usuari.**

Precondició: cap.

Postcondició: el resultat de la cerca guardat en el sistema de forma persistent.

Procés normal principal:

1. El sistema demana les dades de purificació (nombre màxim d'etapes, mínim diferència de pes molecular i proteases utilitzades), el nom de la cerca i correu electrònic.
2. **L'Usuari** introdueix les dades demanades.
3. **L'Usuari** indica que vol realitzar la cerca.
4. El sistema comprova que les dades introduïdes tenen el format correcte.
5. El sistema realitza la cerca de pèptids.
6. El sistema genera i envia la URL aleatòria creada per aquesta cerca.
7. El sistema guarda la informació de la cerca al a base de dades
8. El sistema envia un correu al usuari amb la informació de la cerca.

Alternatives de procés i excepcions:

5a. El sistema ha trobar errors en el format de les dades:

5a1. El sistema indica al usuari les dades amb errors de format.

5a2. El sistema torna al pas 2.

5a. Si la cerca es de proteïnes:

5a1. El sistema realitza la cerca de proteïnes.

5a2. El sistema segueix al pas 6.

08. ViewManual

Resum de la funcionalitat: Mostrar a l'usuari la informació necessària per a realitzar la cerca.

Paràmetres d'entrada: cap.

Paràmetres de sortida: cap.

Usuaris: **Usuari.**

Precondició: Cap.

Postcondició: Cap.

Procés normal principal:

1. L'**Usuari** indica que vol consultar el manual.
2. El sistema mostra la informació del manual.
3. L'**Usuari** llegeix en el manual la informació d'interès.

09. ShowResults

Resum de la funcionalitat: Mostrar al usuari els resultats d'una cerca a partir del url intruït.

Paràmetres d'entrada: url.

Paràmetres de sortida: cap.

Usuaris: **Usuari**.

Precondició: Cap.

Postcondició: Cap.

Procés normal principal:

1. El sistema busca les dades associades al resultat de la url.
2. El sistema mostra les dades generals de la cerca i els resultats en una taula.
3. El **Usuari** demana filtrar els resultats de la taula o ordenar els resultats segons un paràmetre.
4. El sistema realitza l'operació de filtratge o ordenació indicada pel usuari i mostra els resultats.

Alternatives de procés i excepcions:

2a. La URL no té cap cerca associada:

2a1. El sistema mostra un missatge d'error

2b. La cerca corresponent a la URL no ha acabat

2b1. El sistema mostra les dades generals de la cerca i informa a l'usuari que la cerca encara no ha acabat.

2b. La cerca corresponent a la URL ha acabat però no té cap resultat

2b1. El sistema mostra les dades generals de la cerca i informa a l'usuari que la cerca ha acabat però no té cap resultat.

4a. El **Usuari** vol consultar la informació de traçabilitat d'algun de resultats.

4a1. El **Usuari** indica que vol consultar la informació de traçabilitat d'un dels resultats.

4a.2. El sistema mostra consultat la informació de traçabilitat del resultat indicat per l'**Usuari** (peptidasa, posició de tall i fragments obtinguts en cada etapa).

4b. El usuari vol descarregar-se els resultats en un fitxer .csv.

4b1. El **Usuari** indica que vol descarregar-se els resultats en un fitxer .csv.

4b.2. El sistema envia a l'usuari el fitxer .csv.

4b3. El sistema torna al pas 2.

10.RemoveOldData

Resum de a funcionalitat: Un cop al mes eliminar les dades de les cerques amb una antiguitat superior a un més.

Paràmetres d'entrada: cap

Paràmetres de sortida: cap

Usuaris: Sistema

Precondició: cap

Postcondició: Eliminació d'informació antiga.

1. El sistema busca els directoris amb els resultats de les cerques realitzades mesos anteriors.
2. El sistema elimina aquestes cerques.
3. El sistema realitza una actualització de la base de dades d'organismes.

3.2 Requisits no funcionals

Referent als requisits no funcionals s'ha decidit dividir aquests en dues categories: requisits no funcionals del producte i del procés. En la primera s'indiquen algunes puntualitzacions dels principals trets (usabilitat, fiabilitat, portabilitat, seguretat i rendiment i eficiència) i en la segona categoria una breu descripció dels requisits addicionals que no tenen a veure amb el producte final.

3.2.1 *Requisits no funcionals del producte*

- **Usabilitat:** L'aplicació serà utilitzada principalment per persones familiaritzades en altres aplicacions de caràcter biotecnològic com per exemple UniProt o NCBI. Es demana un disseny similar per a facilitar la seva usabilitat i ajudes contextuais per tal de guiar a l'usuari en aspectes com per exemple el format de les dades a introduir. A més a més d'accés a un manual i a contacte amb el grup de recerca.
- **Fiabilitat:** L'aplicació ha de proporcionar a l'usuari la informació vàlida (millor no donar resultats que donar resultats erronis o no vàlids).
- **Portabilitat:** Inicialment s'espera que sigui exclusivament un aplicació web accessible i funcional des dels diferents navegadors més freqüents. També s'ha de destacar que l'aplicació s'ha de poder executar des d'un servidor amb sistema operatiu Linux.
- **Seguretat:** La part de l'administrador ha d'estar protegida amb, com a mínim, una contrasenya per a que no es pugui accedir a les dades així com a la manipulació d'aquestes. En el cas dels resultats s'espera que no es pugui realitzar una traçabilitat mitjançant l'URL per tal de reduir el risc de que un altre usuari pugui accedir als resultat de cerques no realitzades per ell (es proposa realitzar URL amb caràcters aleatoris associats a una cerca mitjançant codificació). A més a més, com es lògic, totes les dades sensibles han de tenir protecció front a un intent d'accés no permès.

- **Rendiment i eficiència:** No s'espera un rendiment ni eficiència elevat sinó una eina amb fiabilitat, tot i que, es valora qualsevol aportació per a millorar el rendiment i només en cas necessari si s'observa que pot afectar la usabilitat o fer inviable l'ús del sistema. La implementació ha de permetre paralelitzar el codi ja que les cerques realitzades tenen un temps d'execució elevat.

3.2.2 Requisits no funcionals del procés

El llenguatge de programació principal ha de ser Python ja que el codi inicial està en aquest llenguatge de programació, a més a més, és un llenguatge molt utilitzat en l'àmbit biotecnològic amb moltes eines lliures adaptades per les diferents taques que es volen realitzar dins del tractament de les dades.

Les dades s'han d'emmagatzemar i/o poder descarregar en fitxer *excel* i *csv* per una posterior fàcil gestió i consulta per part de persones no familiaritzades amb el sistema. El codi ha d'estar comentat i acompanyat d'un manual d'usuari amb el contingut que s'acordi en les reunions.

4 Anàlisi dels Requisits Funcionals

En aquest apartat del treball es mostren els diagrames realitzats durant el disseny de l'aplicació, concretament, els diagrames d'activitats, classes i seqüències dels 11 casos d'ús descrits en el diagrama de casos d'ús i especificats anteriorment.

00.Login

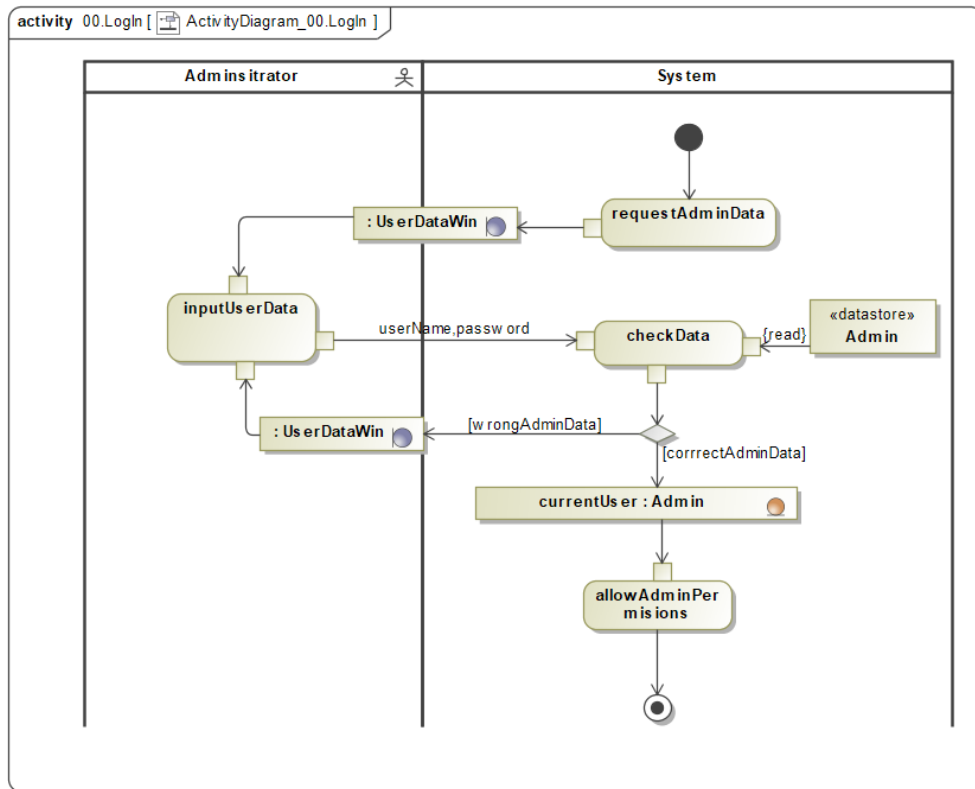


Figura 2. Diagrama d'activitats del cas d'ús 00.Login.

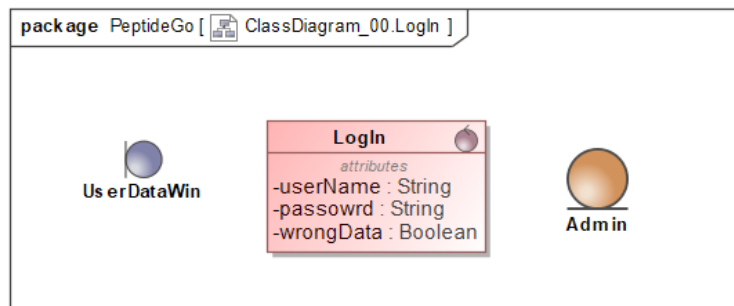


Figura 3. Diagrama de classes del cas d'ús 00.Login.

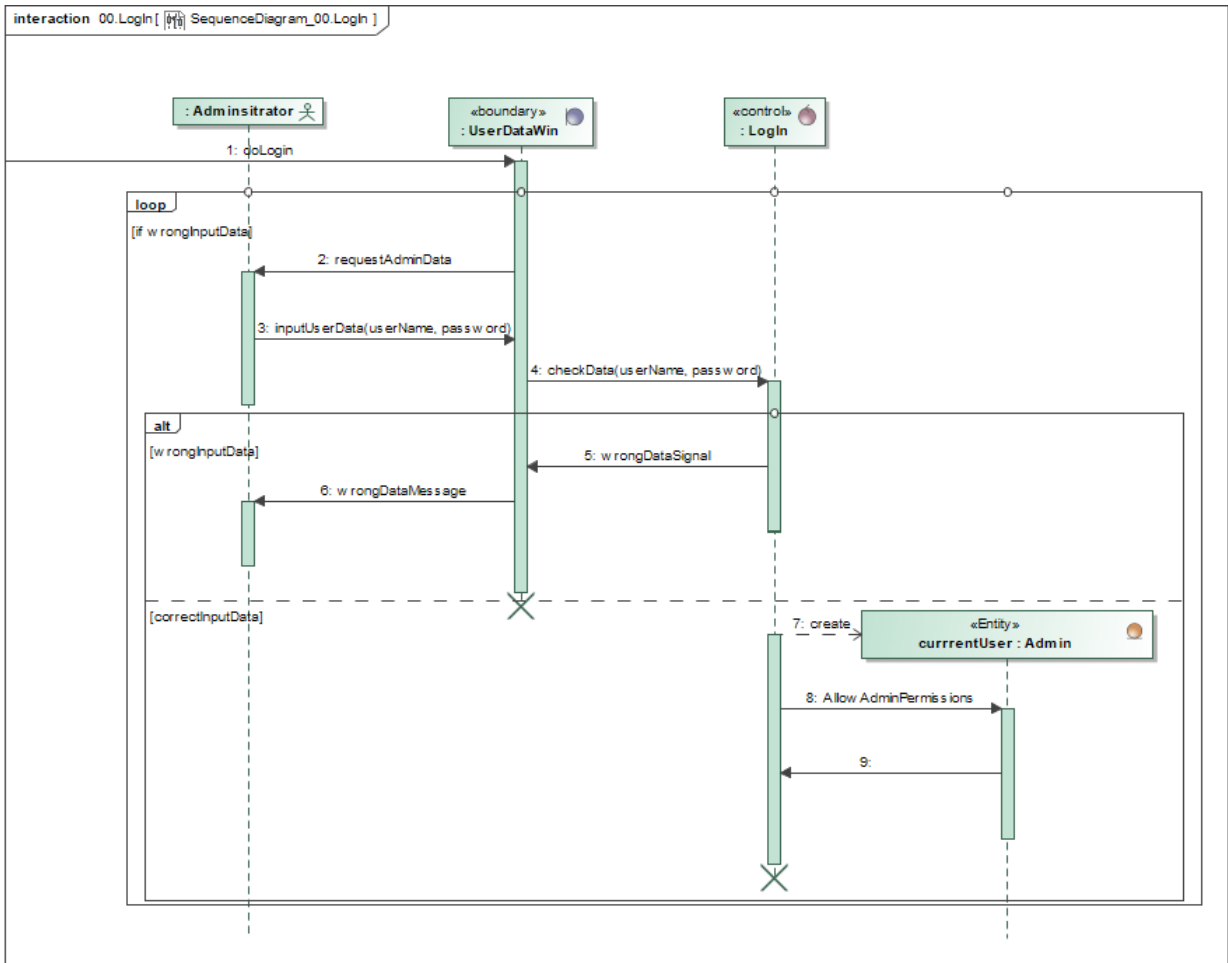


Figura 4. Diagrama de seqüències del cas d'ús 00.Login.

01.LogOut

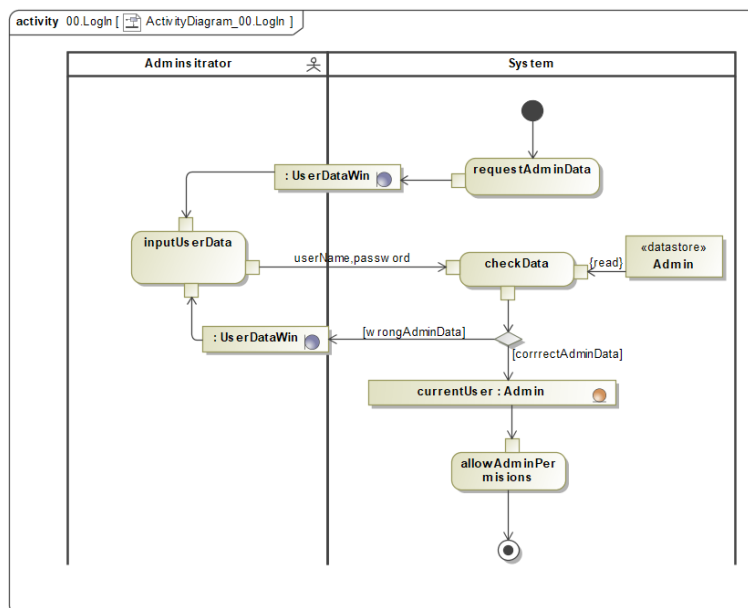


Figura 5. Diagrama d'activitats del cas d'ús 01.LogOut.

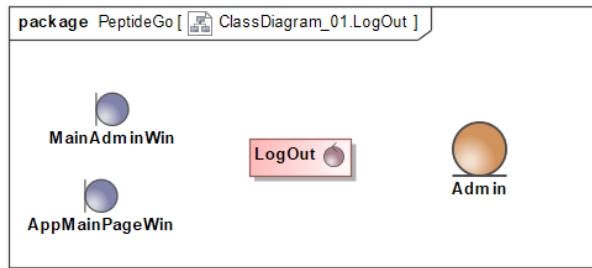


Figura 6. Diagrama de classes del cas d'ús 01.LogOut.

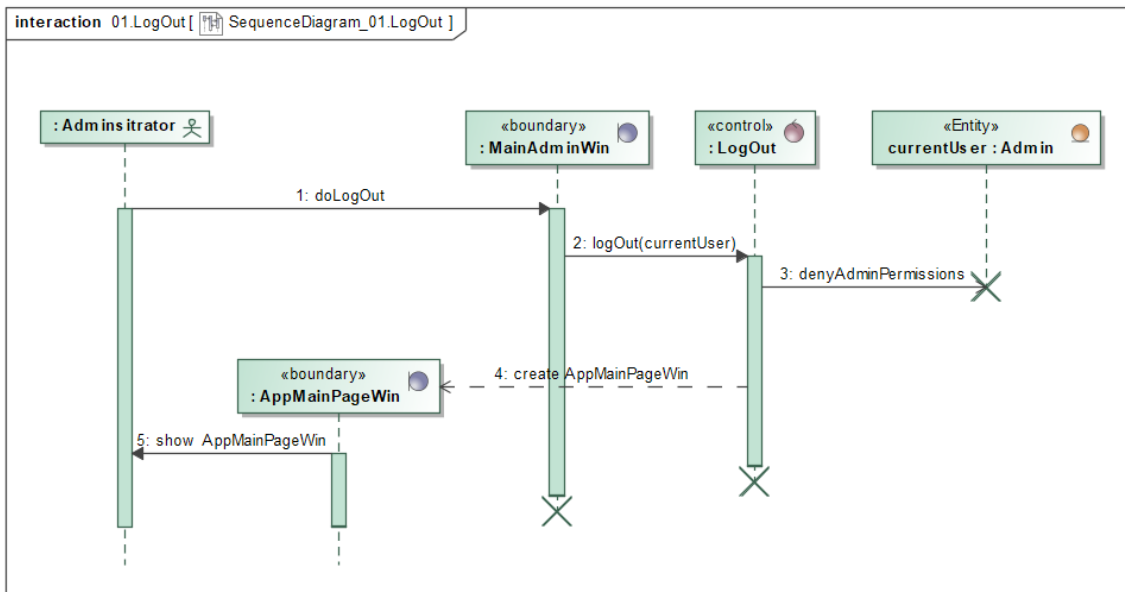


Figura 7. Diagrama de seqüències del cas d'ús 01.LogIn.

02.ManageData

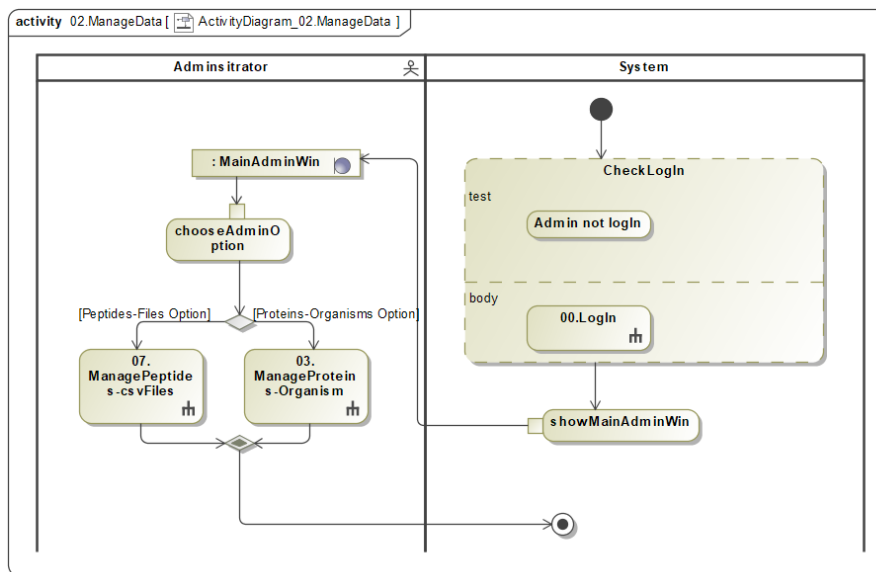


Figura 8. Diagrama d'activitats del cas d'ús 02.ManageData.

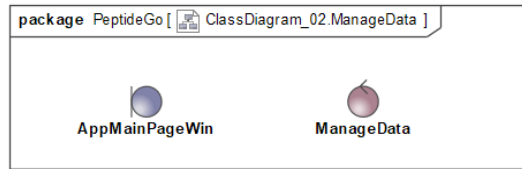


Figura 9. Diagrama de classes del cas d'ús 02.ManageData.

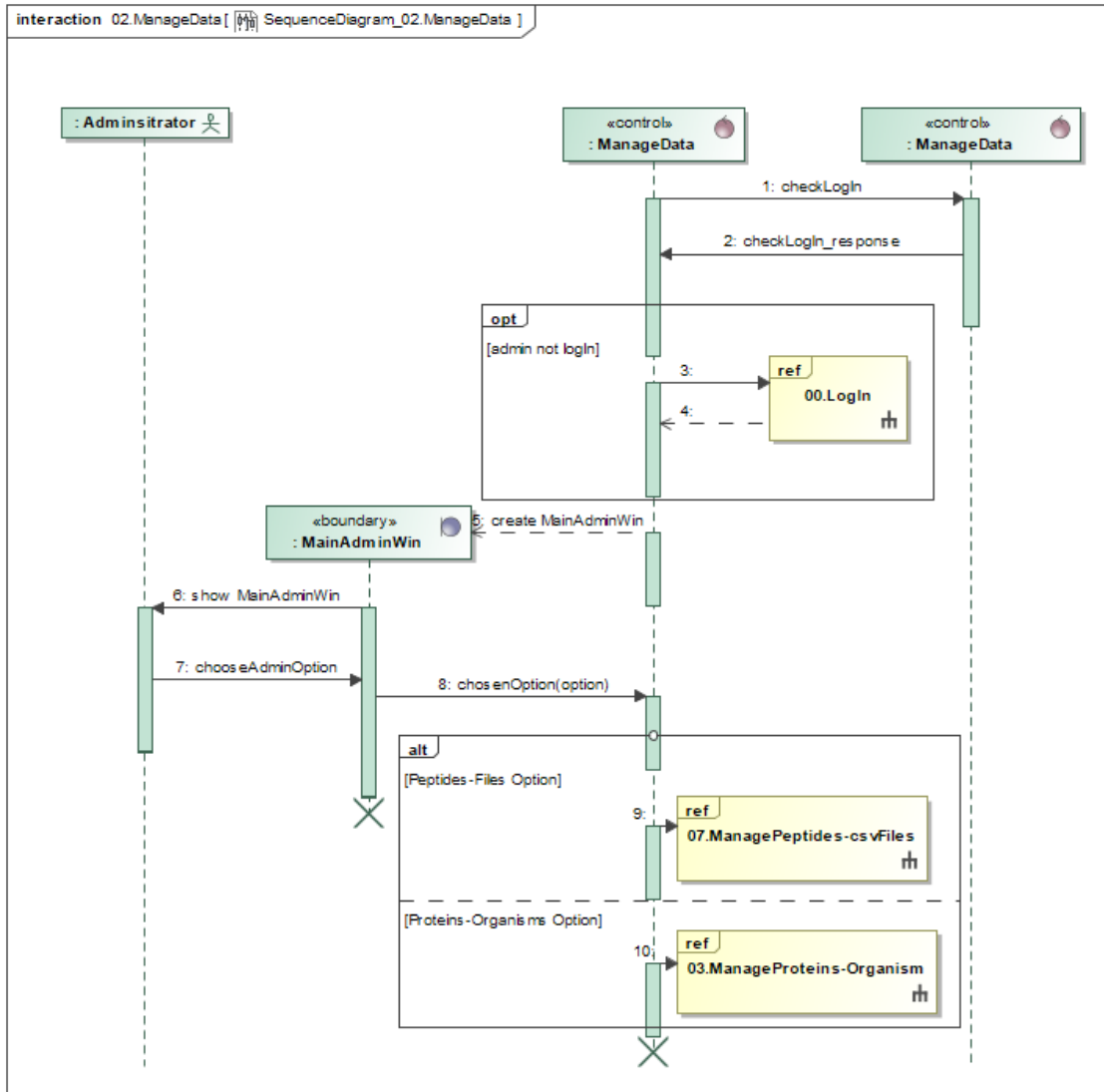


Figura 10. Diagrama de seqüències del cas d'ús 02.ManageData.

03.ManageProteins_Organism

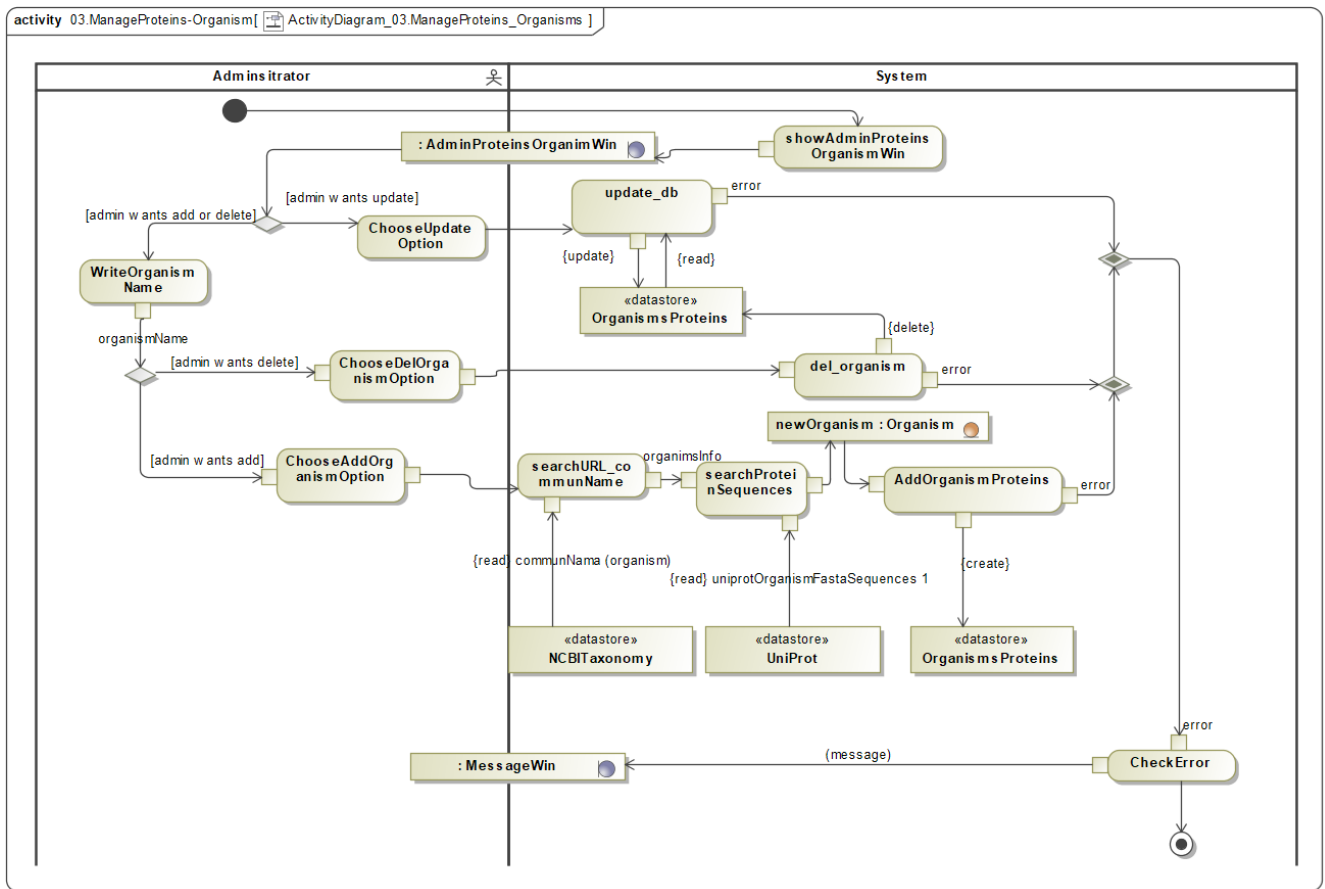


Figura 11. Diagrama d'activitats del cas d'ús 03.ManageProteins_Organism.

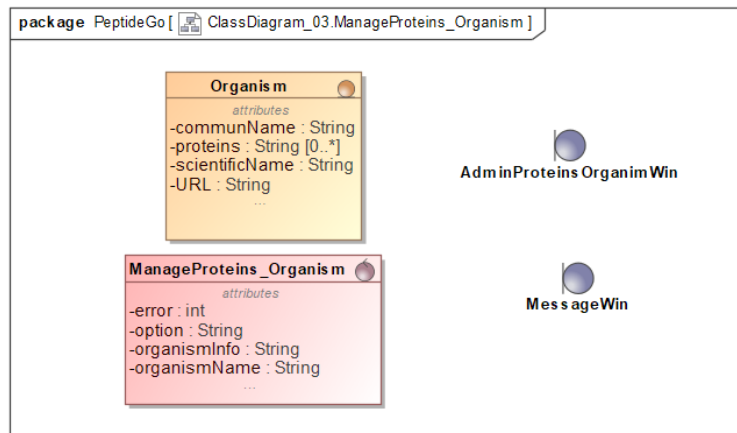


Figura 12. Diagrama de classes del cas d'ús 03.ManageProteins_Organism.

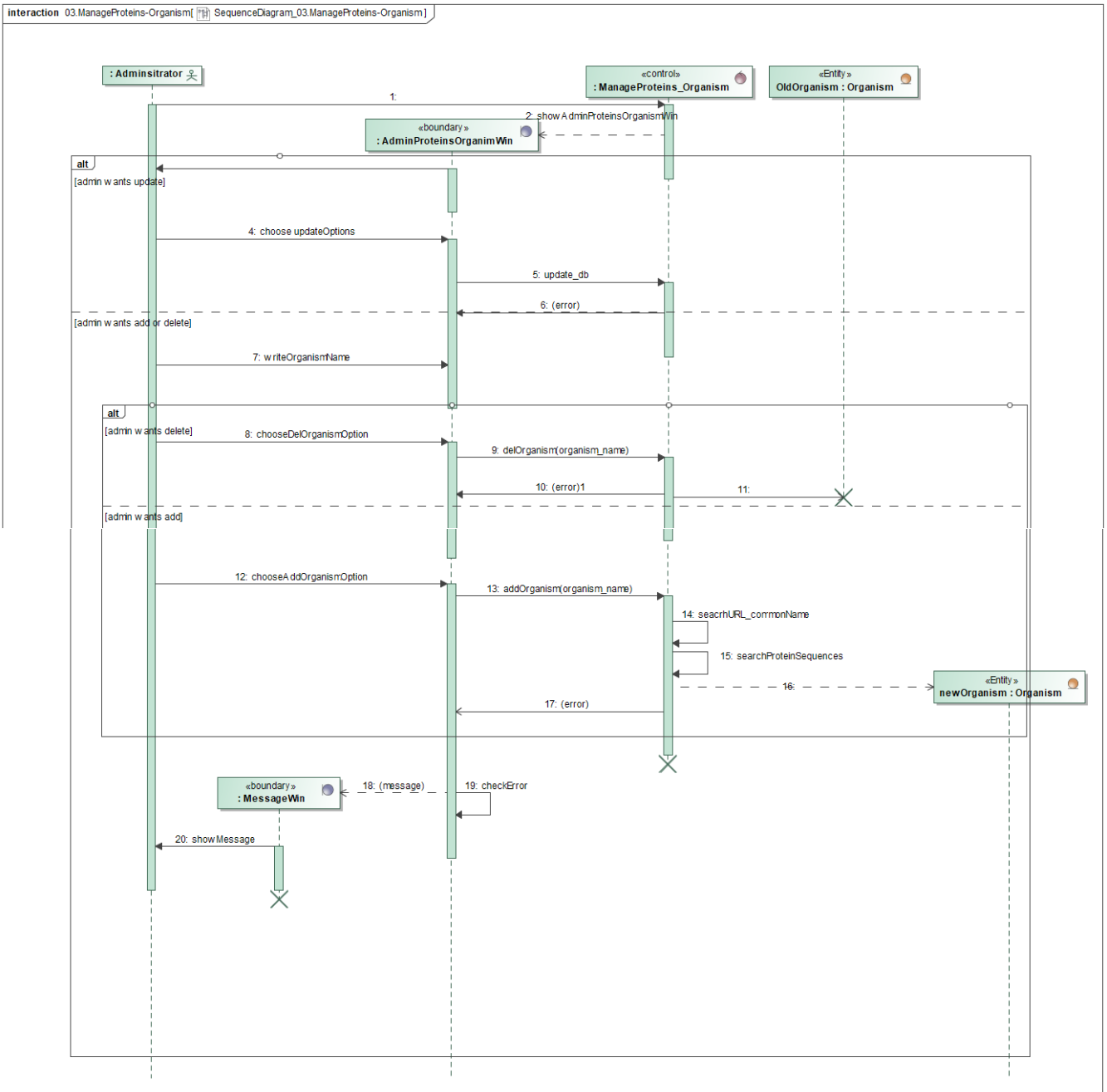


Figura 13. Diagrama de seqüències del cas d'ús 03.ManageProteins_Organism.

04. ManagePeptides_Files

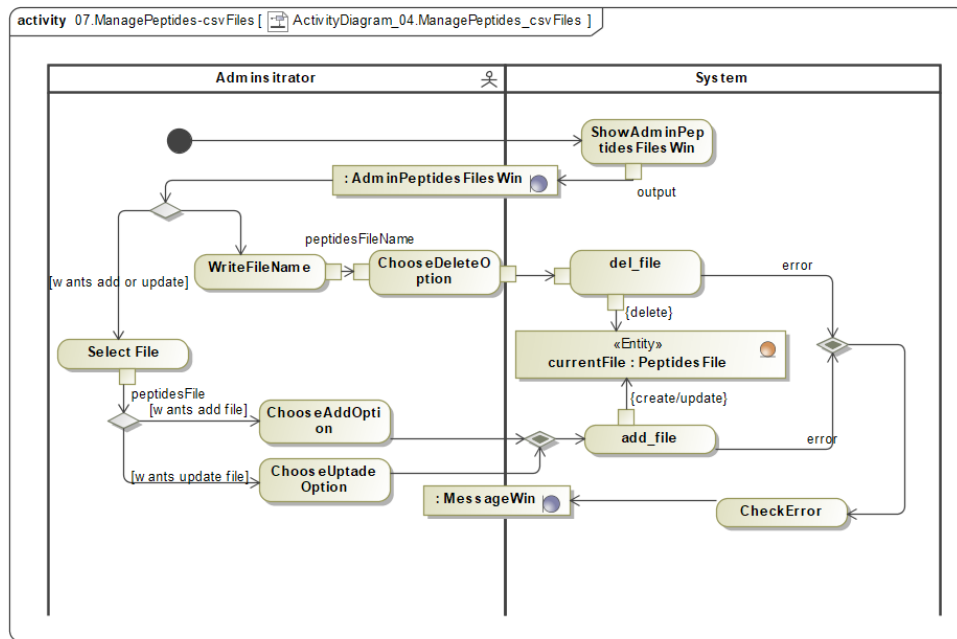


Figura 14. Diagrama d'activitats del cas d'ús 04.ManagePeptides_Files.

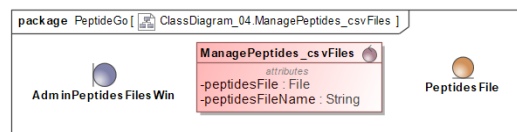


Figura 15. Diagrama de classes del cas d'ús 04.ManagePeptides_Files.

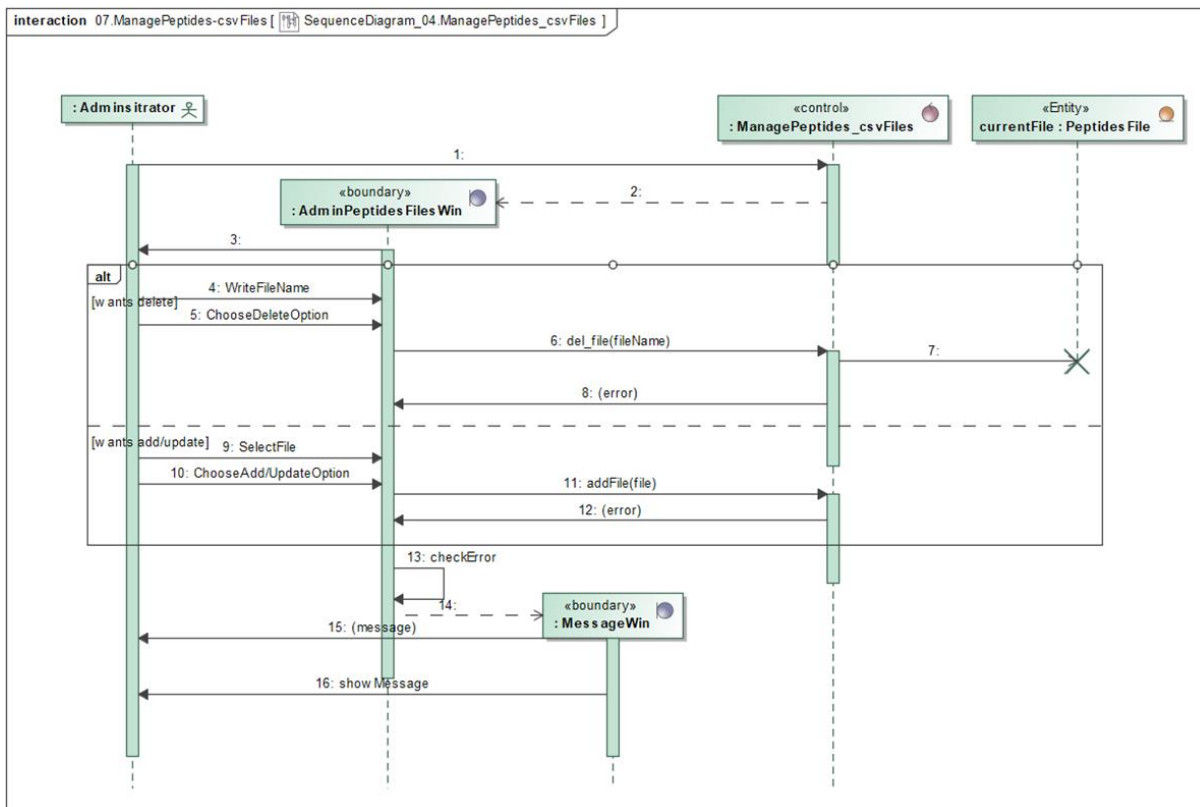


Figura 16. Diagrama de seqüències del cas d'ús 04.ManagePeptides_Files.

05.SearchProteins

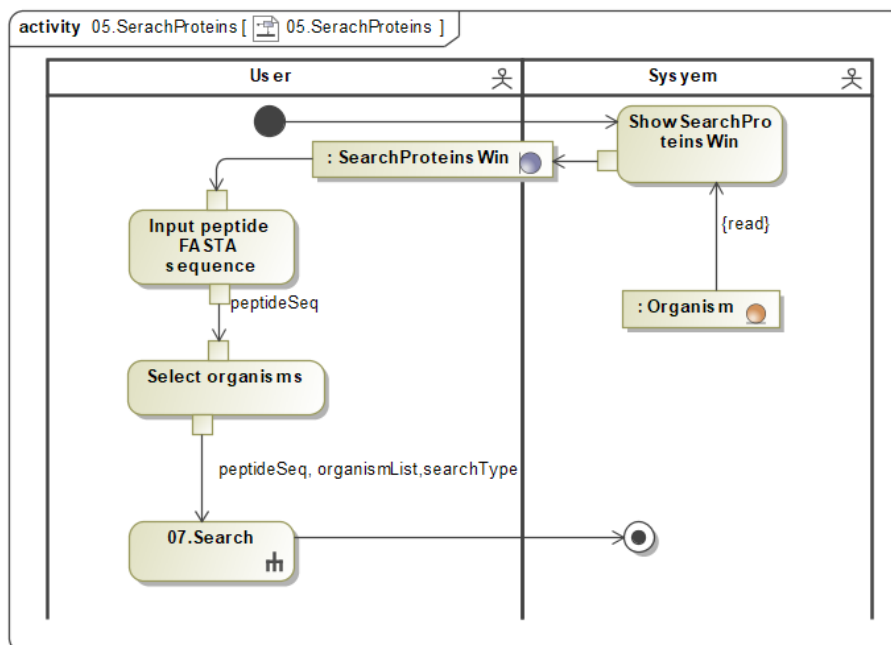


Figura 17. Diagrama d'activitats del cas d'ús 05.SearchProteins.

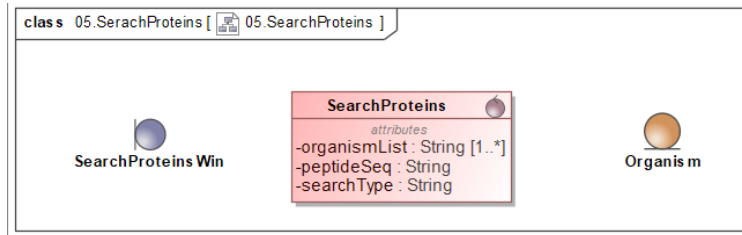


Figura 18. Diagrama de classes del cas d'ús 05.SearchProteins.

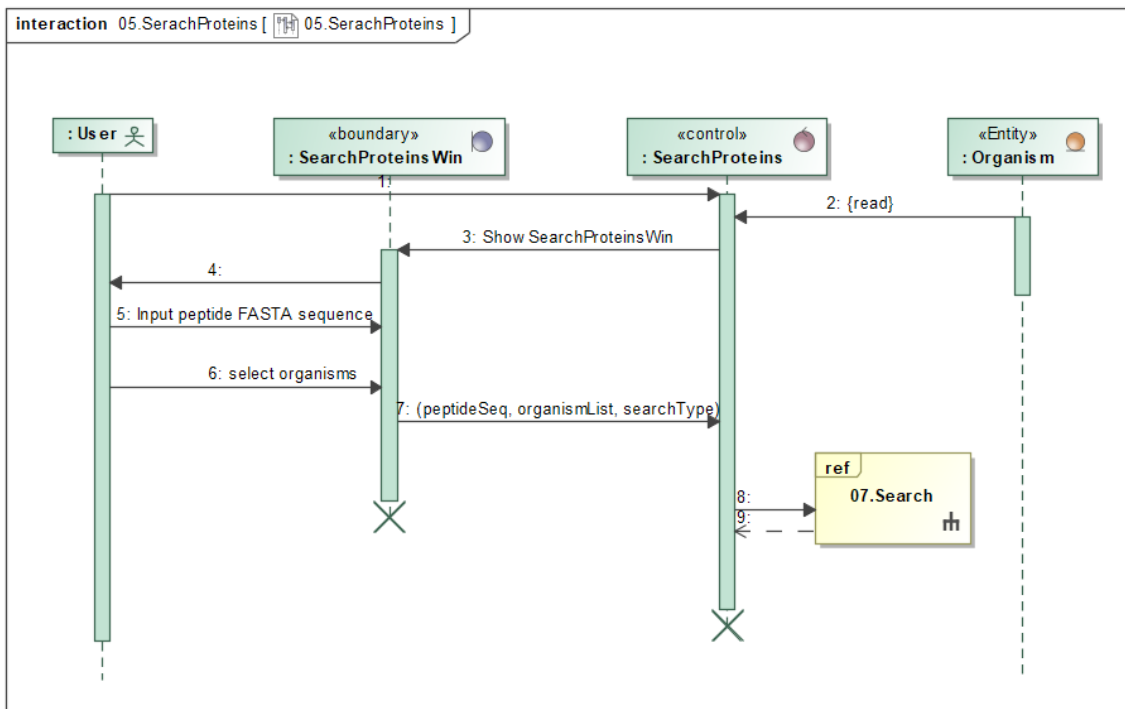


Figura 19. Diagrama de seqüències del cas d'ús 05.SearchProteins.

06.SearchBiopeptides

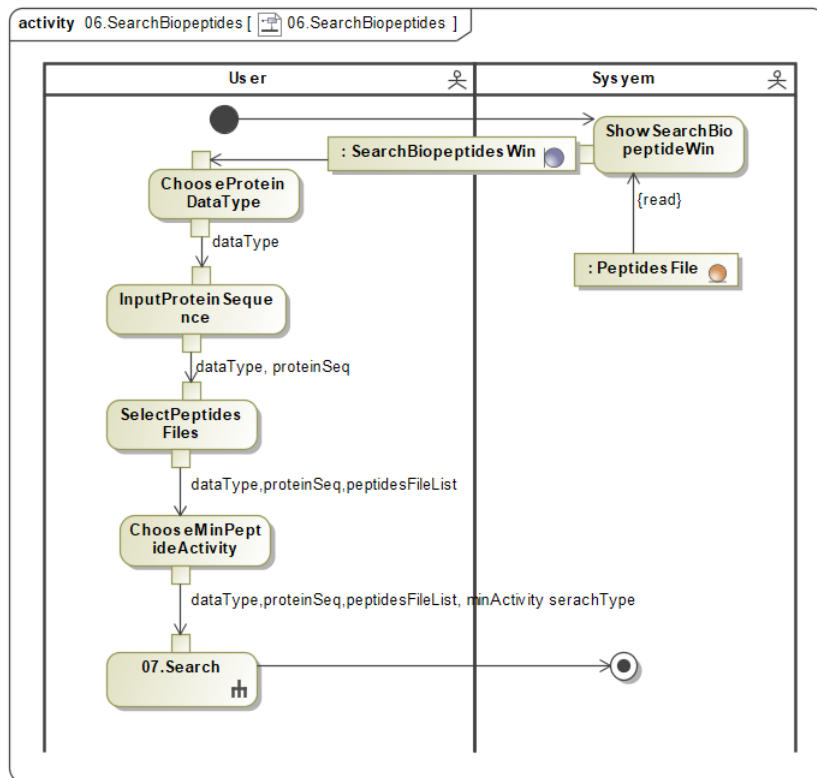


Figura 20. Diagrama d'activitats del cas d'ús 06.SearchBiopeptides.

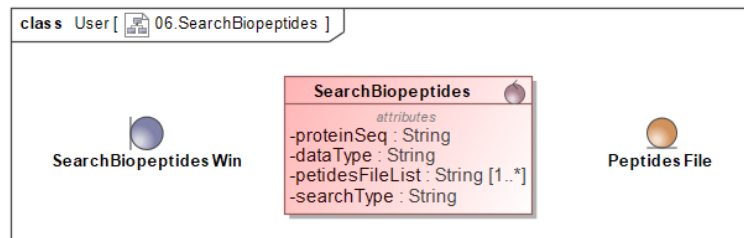


Figura 21. Diagrama de classes del cas d'ús 06.SearchBiopeptides.

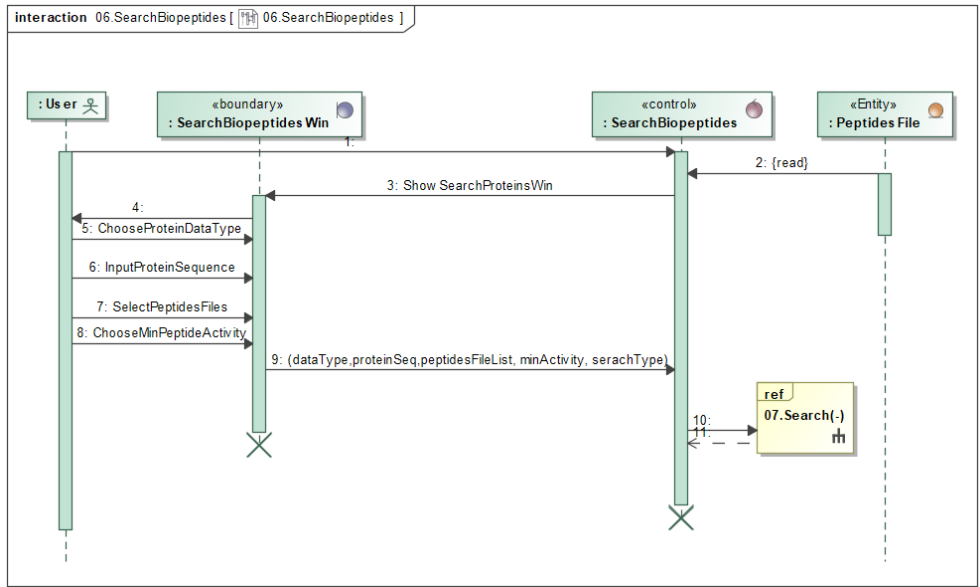


Figura 22. Diagrama de seqüències del cas d'ús 06.SearchBiopeptides.

07.Search

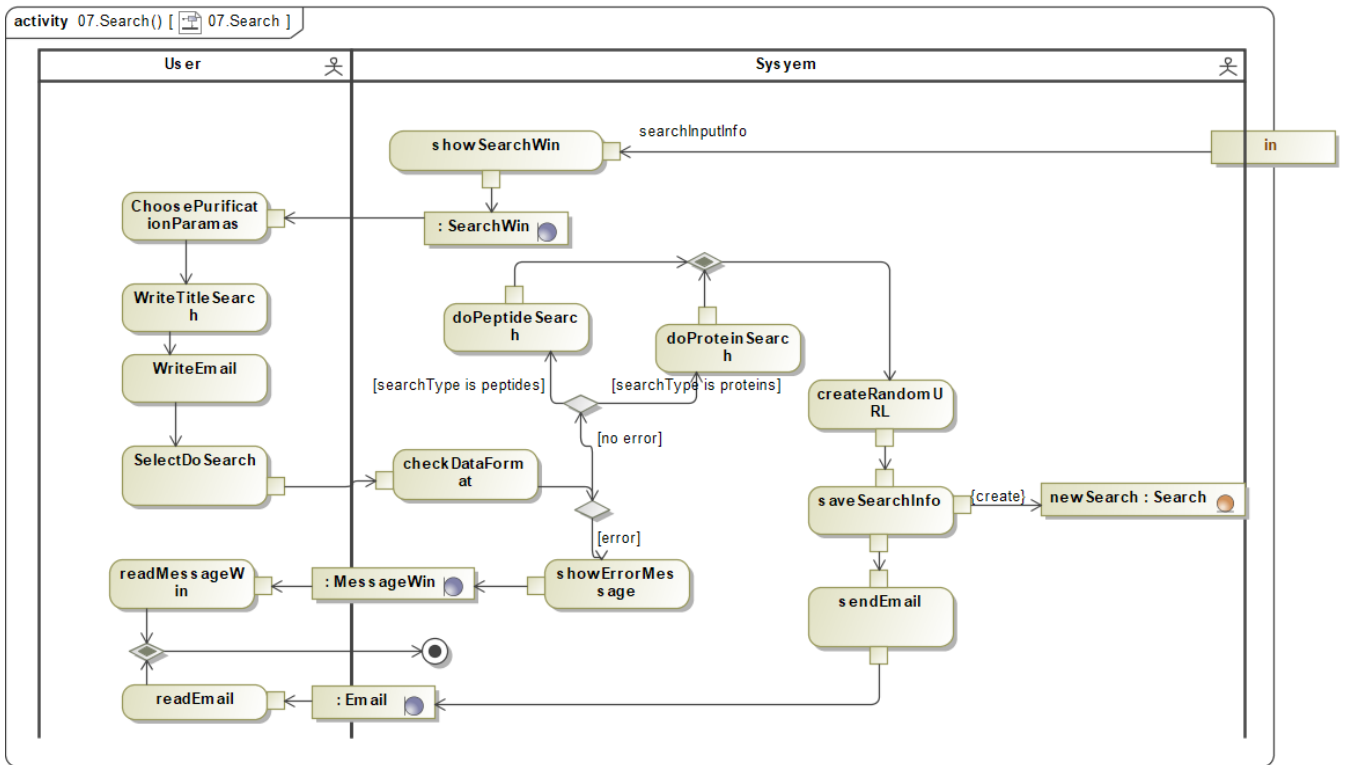


Figura 23. Diagrama d'activitats del cas d'ús 07.Search.

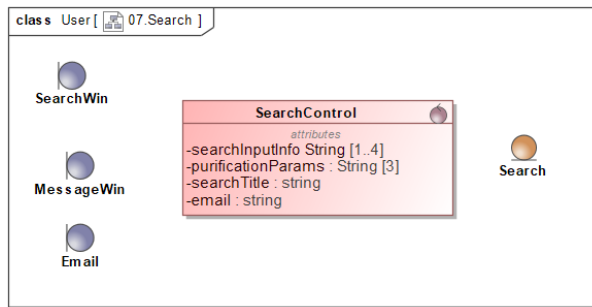


Figura 24. Diagrama de classes del cas d'ús 07.Search.

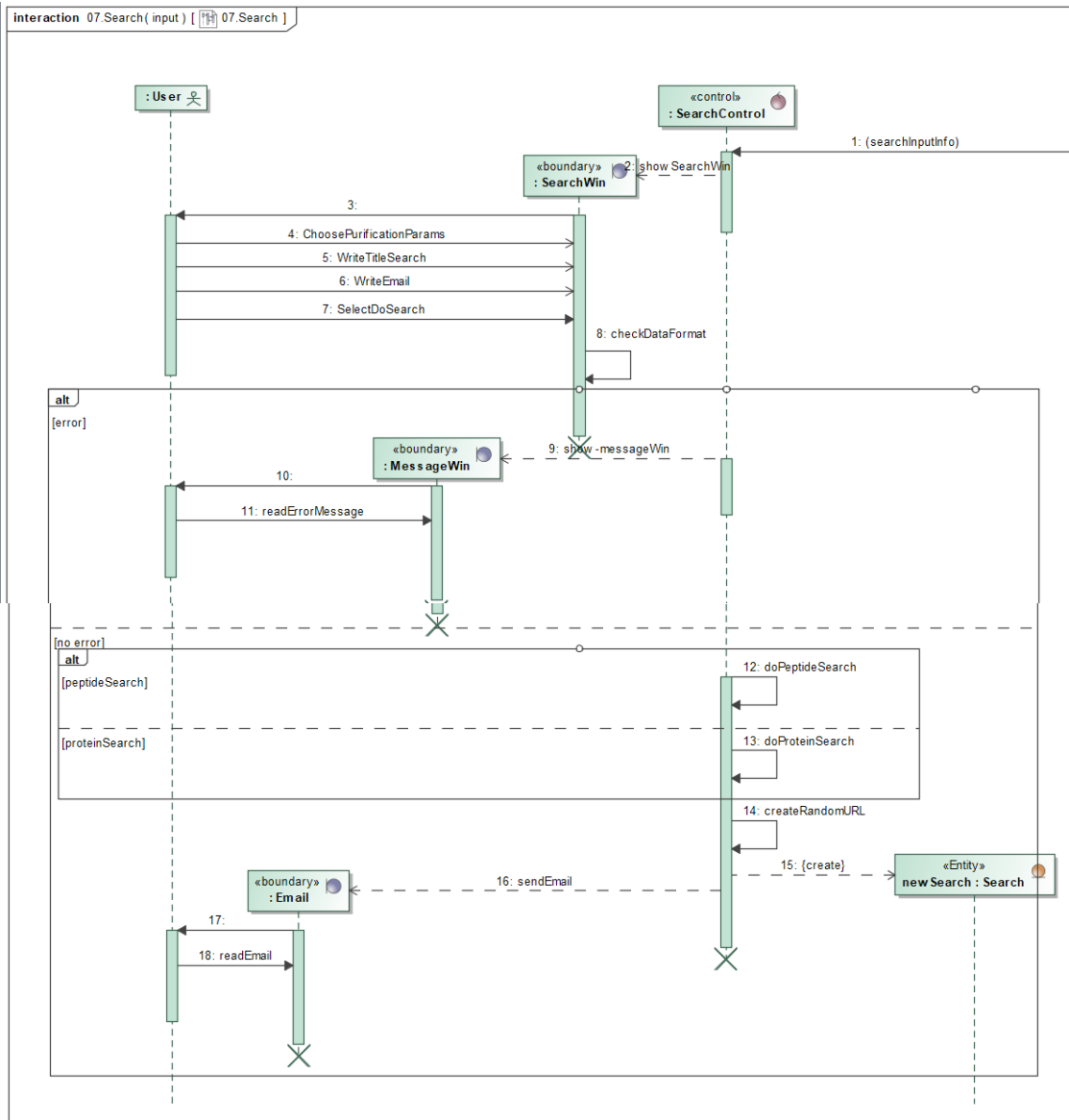


Figura 25. Diagrama de seqüències del cas d'ús 07.Search.

08.ViewManual

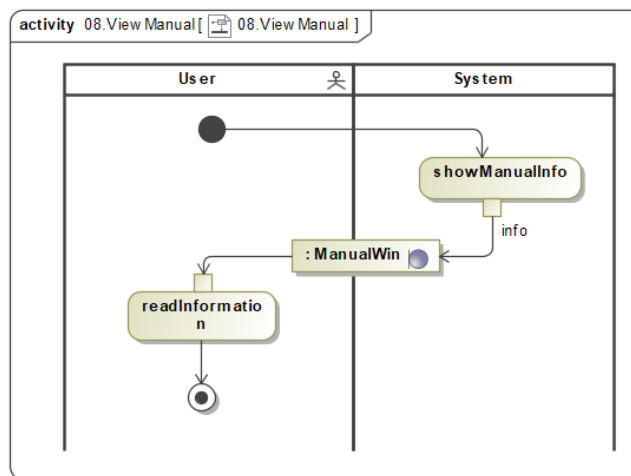


Figura 26. Diagrama d'activitats del cas d'ús 08.ViewManual.

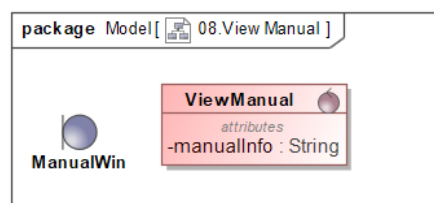


Figura 27. Diagrama de classes del cas d'ús 08.ViewManual.

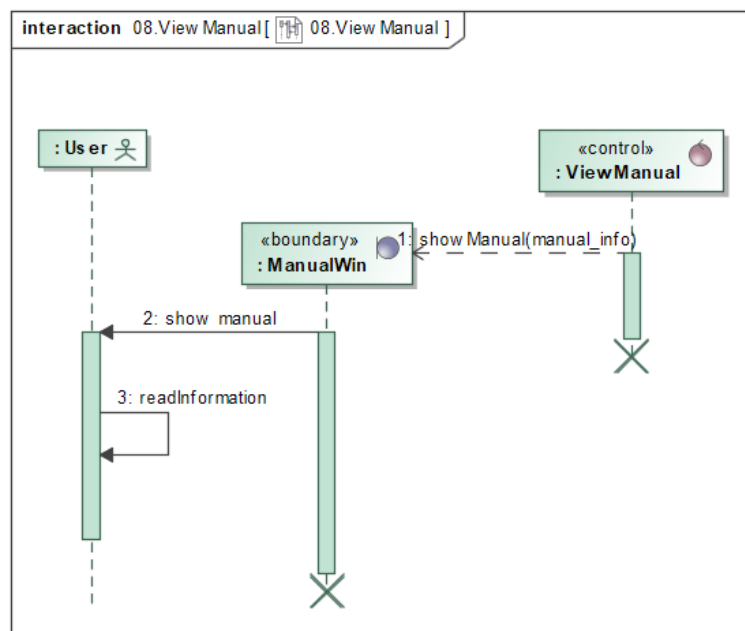


Figura 28. Diagrama de seqüències del cas d'ús 08.ViewManual.

09.ShowResults

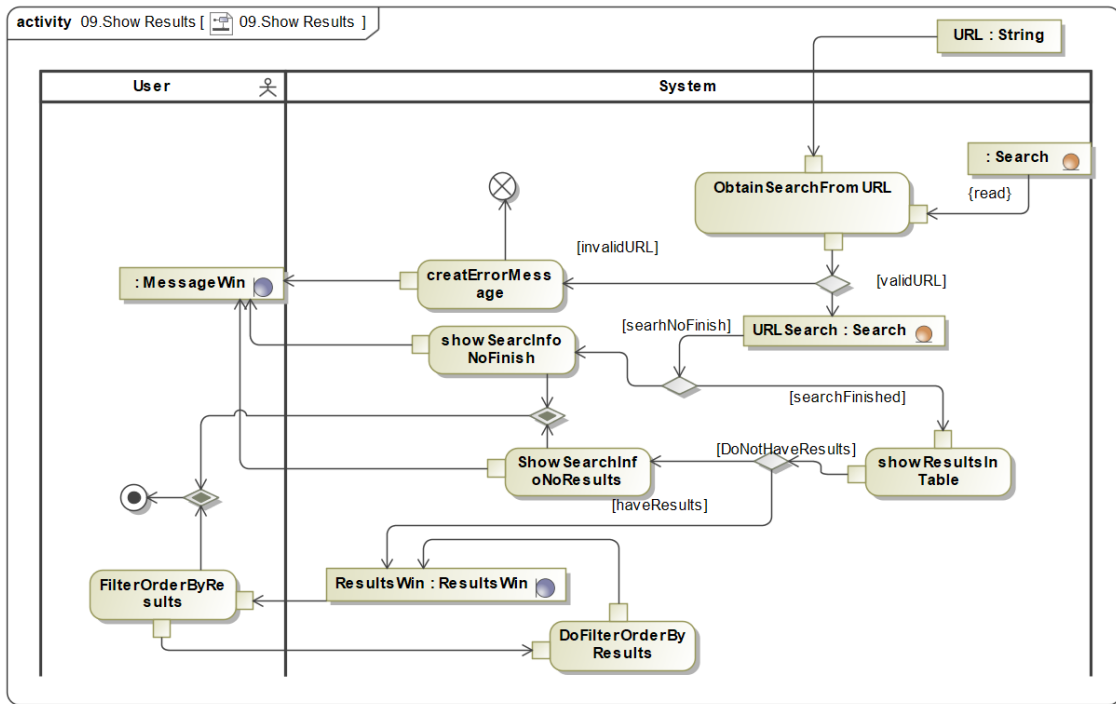


Figura 29. Diagrama d'activitats del cas d'ús 09.ShowResults.

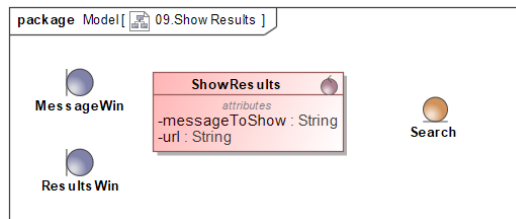


Figura 30. Diagrama de classes del cas d'ús 09.ShowResults.

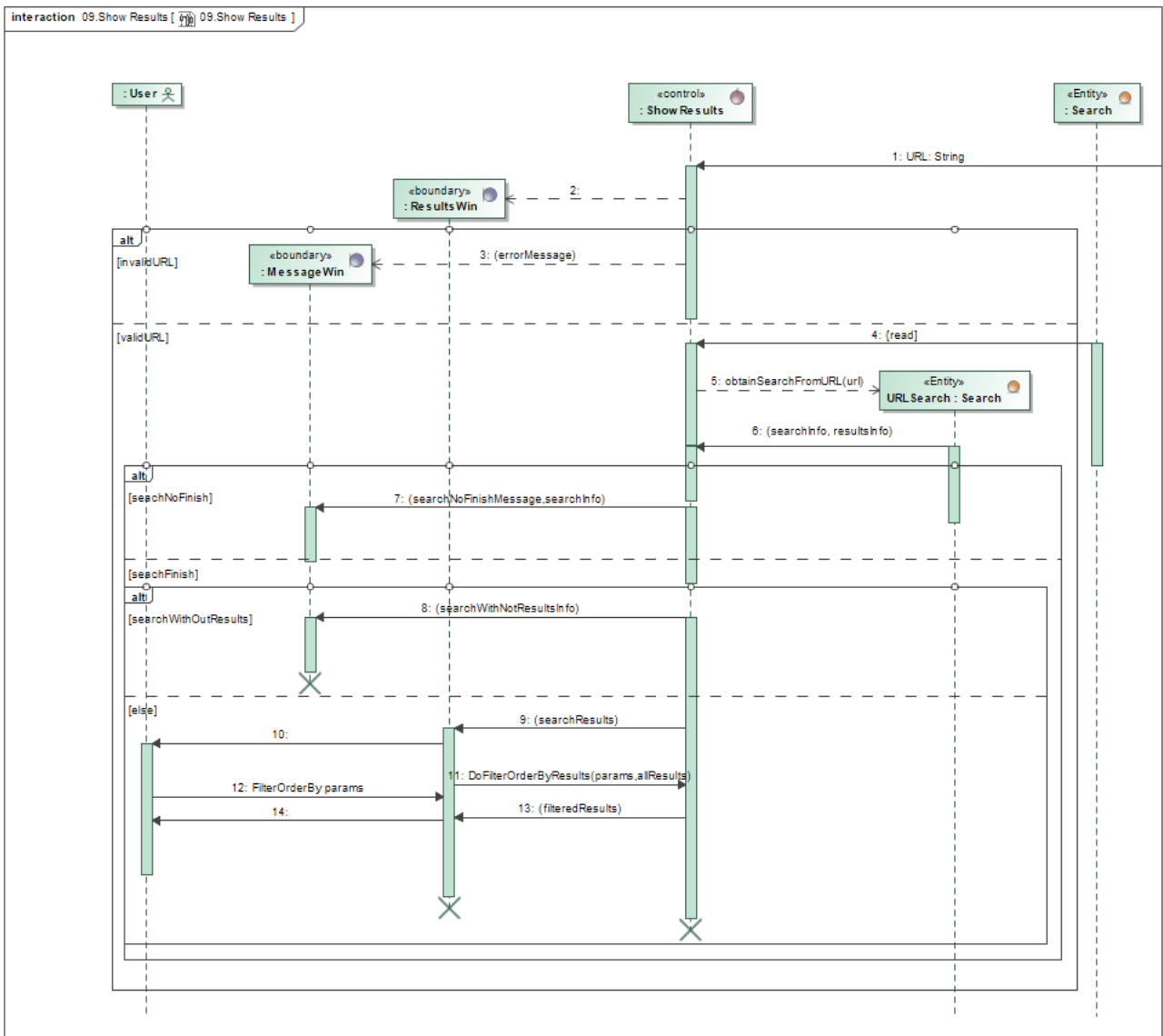


Figura 31. Diagrama de seqüències del cas d'ús 09.ShowResults.

10.RemoveOldData

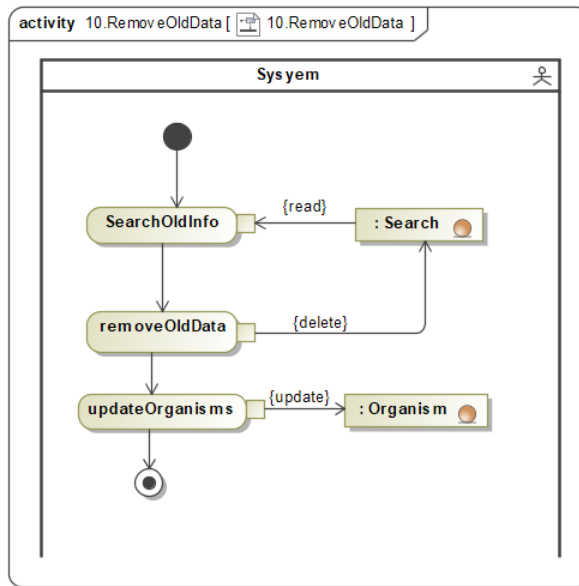


Figura 32. Diagrama d'activitats del cas d'ús 10.RemoveOldData.

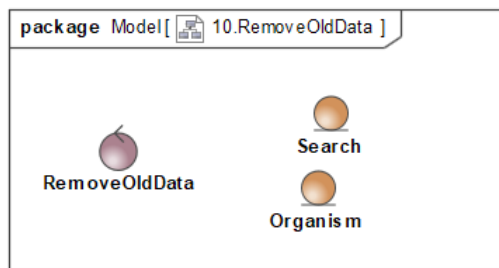


Figura 33. Diagrama de classes del cas d'ús 10.RemoveOldData.

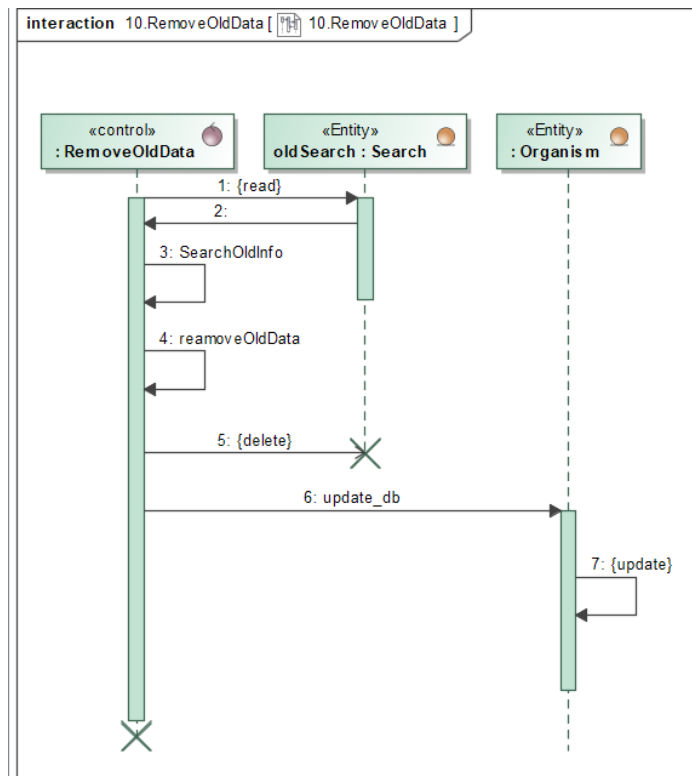


Figura 34. Diagrama de seqüències del cas d'ús 10.RemoveOldData..

5 Disseny

5.1 Arquitectura i organització de fitxers de l'aplicació

L'aplicació web implementada en el framework web2Py té una arquitectura model-vista-controlador (MVC) intrínseca en aquesta. El MVC és un patró de disseny utilitzat per a la implementació d'interfícies d'usuari. Aquest patró de desenvolupament de programari divideix l'aplicació en tres parts interconnectades: el model de dades, la interfície usuari i la lògica de control.

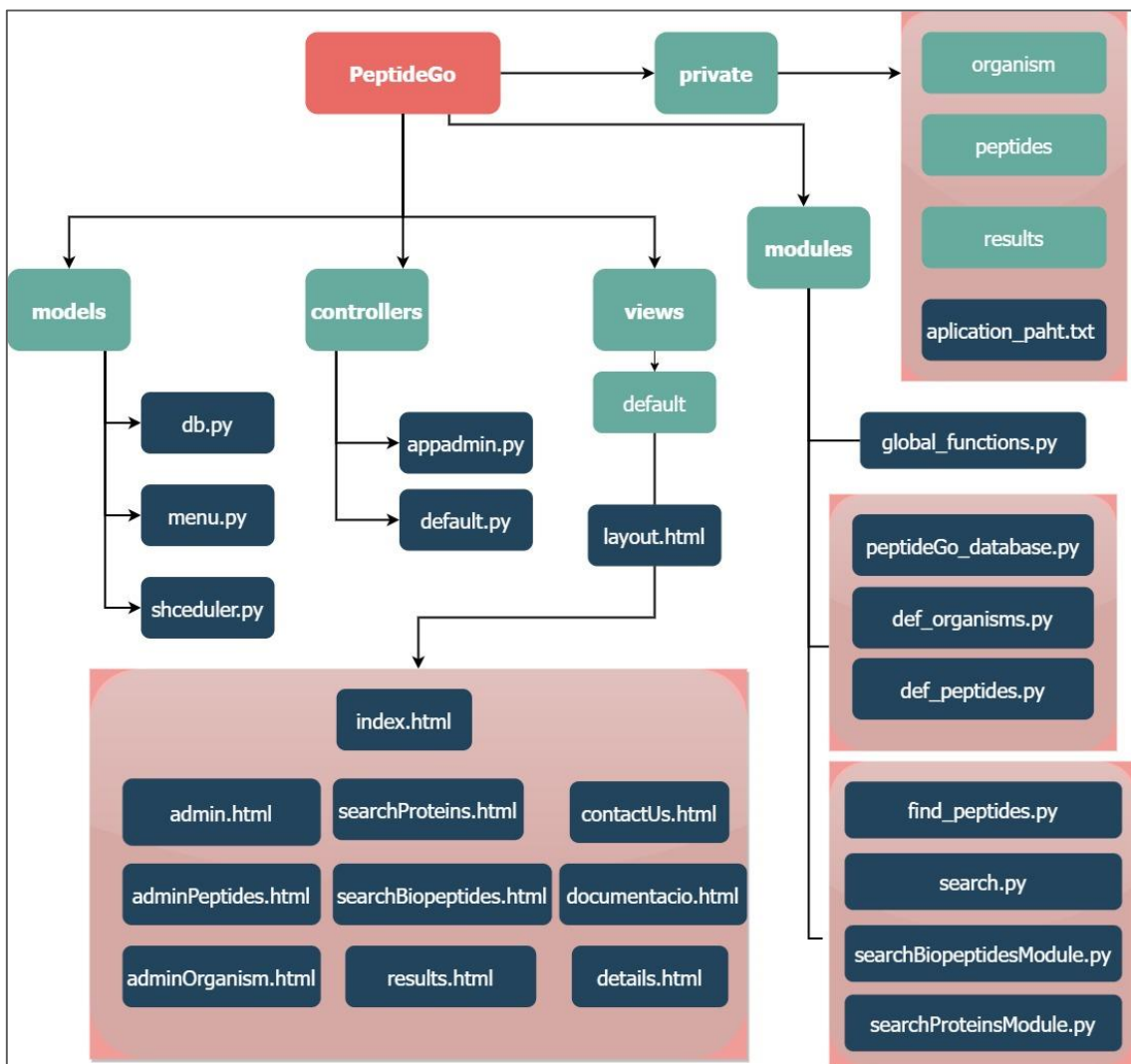


Figura 35. Estructura dels directoris i fitxers de l'aplicació PeptideGo.

A continuació es defineixen, a grans trets, els fitxers i directoris que inclou l'aplicació per tal de poder complir els requisits funcionals i no funcionals definits en la secció de Requisits. Posteriorment es comentarà amb més detall en la secció de Implementació algunes de les seves característiques.

En els models (directori *models*), trobem els fitxers python:

- *db.py* predeterminat per web2py per al creació i maneig de la base de dades dels usuaris registrats i els seus permisos (*auth*) amb funcions utilitzades pels controlador *appadmin.py* i *default.py*. Aquest fitxer forma part de la solució dels casos de ús 00. logIn i 01. logOut, així com per a complir el requisits no funcionals de seguretat de control d'accés per a la gestió de les bases de dades.
- *menu.py* on es defineix el contingut i estructura de la barra de navegació mostrada en les vistes (consultat pel fitxer de vistes *layout.html*) per a millorar la usabilitat i l'accés a les diferents seccions.
- *shceduler.py* per a les funcions realitzades periòdicament pel *cron* referents a la base de dades.

Dins de les funcions que s'haurien d'incloure als models, cal destacar que els fitxers de gestió de dades de les cerques (*peptideGo_database.py* *def_organism.py* i *def_peptides.py*) s'han decidit implementar com a mòduls externs no inclosos en el directori *models* sinó en *modules*.

Per altra banda, dins dels controladors (directori *controllers*) trobem el fitxers python:

- *appadmin.py* predeterminat per web2Py per al control de les dades referides als usuaris i associat a les vistes predefinides de web2Py per al maneig d'usuaris (part de la solució dels casos de ús 00. logIn i 01. logOut).

- *default.py* amb al control d'accés i accions a realitzar en les vistes de l'aplicació detallades en l'apartat Implementació Aquest conté una funció per cada una de les vistes amb el mateix nom que la vista.

Finalment, en les vistes (directori *views*) trobem el directori *default* que conté el fitxers *html* (amb vistes realitzades amb HTML5 i python i funcions de suport en javaScript):

- *layout.html*: capa que conté la base de les diferents vistes (inclouen estil utilitzat, capçalera, peus i la barra de menú).
- *index.html*: vista de la pàgina principal de l'aplicació (Figura 36).
- *user.html*: vista per a que l'usuari pugui realitzar logIn (Figura 38) referent als casos d'ús 00.logIn i 01.logOut.
- *admin.html*: vista de la pàgina principal d'administració de les dades (Figura 37). Aquesta forma part de la solució dels casos d'ús 02.ManageData i 01.logOut.
- *adminOrganism.html*: vista de la pàgina d'administració de les dades referents als organismes (Figura 39). Aquest forma part de la solució dels casos d'ús 03.ManageProteins Organisms i 01.logOut.
- *adminPeptides.html*: vista de la pàgina d'administració de les dades referents als fitxers de pèptids (Figura 40) per a la solució dels casos d'ús 04.ManagePeptides Files i 01.logOut.
- *searchBiopeptides.html*: vista del formulari per a la cerca de pèptids (Figura 41 i Figura 43) per a la solució dels casos d'ús 06.SearchBiopeptides i 07.Search.
- *searchProteins.html*: vista del formulari per a la cerca de proteïnes (Figura 42 i Figura 43) per a la solució del casos d'ús 05.SearchProteins i 07.Search.
- *results.html*: vista per a mostrar tots els resultats d'una cerca determinada (Figura 44) per a la solució del cas d'ús 09.ShowResults.

- *details.html*: vista dels detalls d'un resultat determinat (Figura 45) per a mostrar la traçabilitat d'aquell resultat (adicional al cas d'ús 09.ShowResults).
- Addicionalment s'ha creat els fitxers *conctacUs.html*, *documentacio.html* i *help.html* per a unes vistes annexes complint amb els requisits no funcionals d'usabilitat de l'aplicació i permetre al usuari consultar informació de ús i contacte. (cas d'ús 08.ViewManual)

També hi ha un seguit de mòduls (directori *modules*) amb les funcionalitats necessàries per les cerques i la gestió de les dades. Dins de les funcionalitats necessàries s'han creat els mòduls en python:

- *find_peptides.py*: amb les funcions proporcionades pel codi inicial amb les modificacions pertinents (per al cas d'ús 07.Search).
- *search.py*: amb funcions globals per a totes les cerques (per al cas d'ús 07.Search)
- *searchBiopetidsModules.py* i *searchProteinsModule.py*: amb funcions exclusivament utilitzades en cada una de les cerques per a la solució dels casos d'ús 06.SearchBiopeptides i 05.SearchProteins respectivament.

Per altra banda per a les funcions de administració i accés de dades s'han creat els mòduls:

- *peptideGo_database.py*: per a crear i administrar la base de dades de PeptideGo (crear i inicialitzar les taules definides; inserir, eliminar i seleccionar la informació més freqüent necessària per a la solució dels casos d'ús 03.ManageProteins Organisms, 04.ManagePeptides Files, 05.SearchProteins, 06.SearchBiopeptides, 07.Search i 09.ShowResults).
- *def_organism.py* i *def_peptides.py*: per a la implementació de les principals funcions de les dades d'organismes-proteïnes i activitats-pèptids respectivament necessaris per a la solució dels casos d'ús 03.ManageProteins Organisms, 04.ManagePeptides Files, 05.SearchProteins, 06.SearchBiopeptides i 07.Search)

Finalment, algunes de les dades s'emmagatzemen en el directori *private* (el qual web2py manté segur referent a l'accessibilitat a les seves dades). En ell es troben els directoris *organism*, *peptides* i *results* amb les dades referents a aquests camps a part de tenir-les emmagatzemades en la base de dades local (complint així el requisits no funcionals de gestió de guardat/accés a les dades i seguretat).

Addicionalment, per tal de tenir una gestió més fàcil en ús dels fitxers guardats en el directori *private*, també s'ha creat un fitxer *aplicacion_paht.txt* el qual es pot interpretar com un diccionari que relaciona una paraula clau i el *path* del fitxer o directori associat. Aquest està localitzat en el directori *private* i es pot accedir al seu contingut amb les funcions del mòdul *global_functions.py* per tal de obtenir la informació d'aquest fitxer de direccions. Aquests dos fitxers han estat creats per a facilitar la programació i la flexibilitat d'aquesta aplicació al canvi de organització dels directoris.

5.2 Disseny de la Interfície Gràfica

Com es menciona en els requisits no funcionals, s'espera que l'aplicació serà utilitzada principalment per persones familiaritzades en altres aplicacions de caràcter biotecnològic, com per exemple, UniProt o NCBI. Per aquest motiu es demana un disseny similar per a facilitar la seva usabilitat i ajudes contextuais per tal de guiar l'usuari en aspectes, com per exemple, el format de les dades a introduir. Addicionalment, s'ha decidit utilitzar una plantilla d'estils amb colors neutres referents a l'escala de grisos amb un efecte senzill però elegant. S'ha mantingut un format similar dins del procés cognitiu (per exemple a l'hora d'elegir entre dues opcions o seleccionar quelcom en una llista) per tal de poder ressaltar les parts importants i que sigui el més intuïtiu i fàcil possible per a l'usuari.

A continuació es mostren les captures de la interfície gràfica i s'esmenten alguns dels detalls tinguts en compte principalment per a la usabilitat de l'usuari i estètica de l'aplicació.

Per tal de navegar per les diverses seccions, s'ha inclòs una barra de menú amb les opcions principals de *home*, cerca de biopèptids, cerca de proteïnes i ajuda. A més d'un botó annex per accedir a l'administració de la base de dades.

Per al disseny de selecció de dues opcions en les pàgines principals de l'aplicació i d'administració (Figura 36 i Figura 37) s'ha decidit mantenir un format similar amb un quadre centrat amb botons destacats per a les dues opcions a elegir.

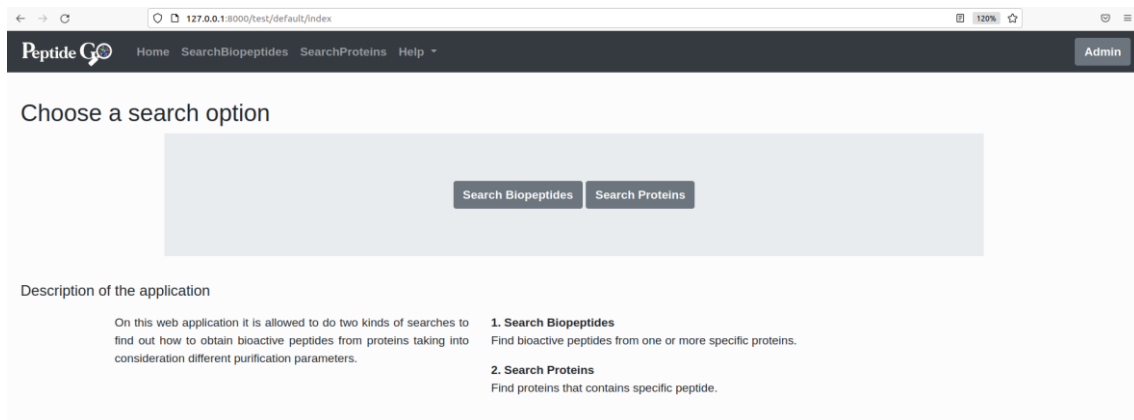


Figura 36. Pàgina principal de l'aplicació (model selecció de 2 opcions principals)



Figura 37. Pàgina principal de l'administració de dades (model selecció de 2 opcions principals)

Per a l'administració de la base de dades, s'ha utilitzat la estructura de *logIn* implementada per defecte per web2py en la qual es demanen les dades per accedir (Figura 38). En les pàgines d'administració de cada una de les parts de la base de dades es mostra un botó per a fer *logOut*, les opcions que es poden realitzar en les dades i un llistat de la informació que es té amb enllaços per a tenir més detalls (en el cas dels organismes enllaç

al *NCBI taxonomy* i en el cas dels fitxer enllaç per a poder descarregar-se el contingut) (Figura 39 i Figura 40).



Figura 38. Pàgina de logIn.

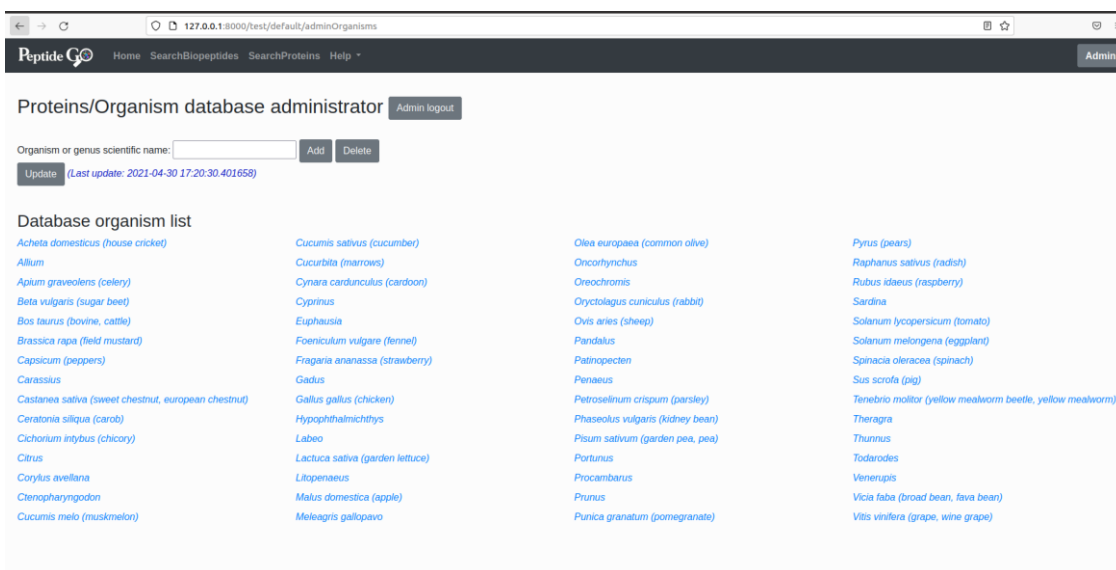


Figura 39. Pàgina d'administració de dades referents a organismes-proteïnes (model accions bases de dades).

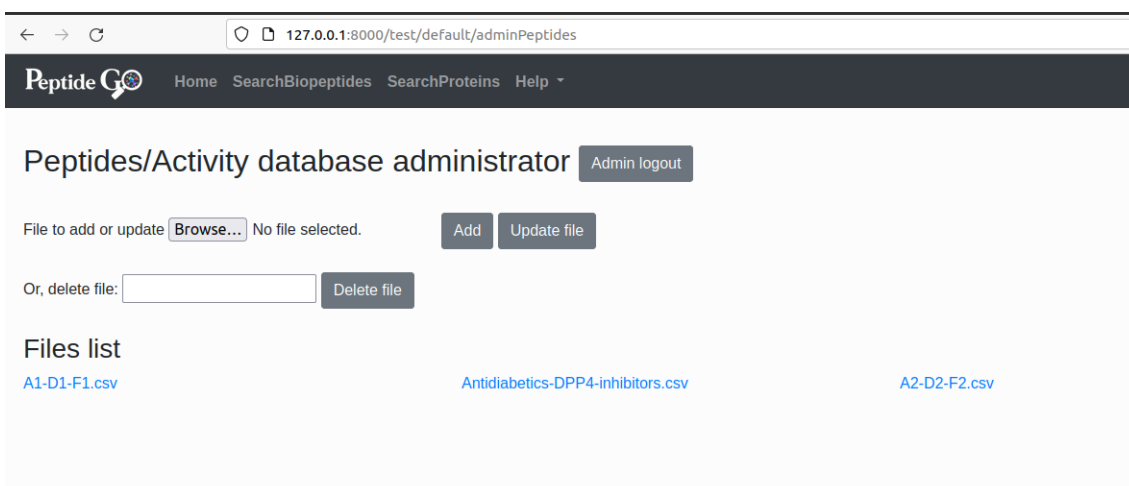


Figura 40. Pàgina d'administració de dades referents a pèptids (model accions bases de dades).

Per altra banda, en els formularis per a introduir les dades les cerques (Figura 41, Figura 42 i Figura 43) s'ha elegit el tipus de figura gràfica idònia per a cada tipus de informació: Tipus *radio* per a la selecció entre els formats de les dades a introduir (Figura 41); quadre de text per a la introducció de seqüències (però també la possibilitat d'afegir la informació a partir d'un fitxer i actualitzant posteriorment el contingut del quadre de text per a no crear ambigüitats) (Figura 41 i Figura 42); quadre amb llistat de termes en *checkbox* per a l'elecció de subconjunt de molècules de les base de dades (organismes o funcions de pèptids) amb les opcions de seleccionar tots o cap mitjançant un botó annex (Figura 41 i Figura 42) i selecció mitjançant desplegable d'una única opció per cada un dels paràmetres de purificació (Figura 43). En aquest camp també cal destacar la introducció del valor mínim d'activitat en la cerca de pèptids amb la restricció d'indicar un valor numèric amb 3 decimals (Figura 41) i l'elecció dels valors per defecte de tots els camps amb el valor més comú o d'interès.

The screenshot shows a web browser window with the URL `127.0.0.1:8000/test/default/searchBiopeptides`. The page title is "Search Biopeptides" and the subtitle is "Find bioactive peptides from one or more specific proteins." The interface is divided into sections:

- 1. Input**
 - Choose protein's input data format (UniProt code or FASTA sequence):
 - UniProt code
 - FASTA sequence
 - Fill the text area with the protein's information (with protein input data format selected):
 - A large empty text area.
 - Or, upload file/s with the protein's information (with protein input data format chosen):
 - A "Browse..." button and the text "No files selected."
 - Choose the file names with activities, targets and effects of the peptides you want to search and write the minimum peptide bioactivity:
 - A list of files with checkboxes:
 - A1-D1-F1.csv
 - Antidiabetics-DPP4-inhibitors.csv
 - A2-D2-F2.csv
 - Buttons: "Select All" and "Clear All".
 - Filter by minimum peptide bioactivity: A dropdown menu.

Figura 41. Formulari per a introduir la informació referents als pèptids en la cerca de pèptids. (model formulari).

Peptide GO Home SearchBiotopes SearchProteins Help Admin

Search Proteins

Find proteins that contains specific peptide.

1. Input

- Fill the text area with the sequence of the peptide in FASTA format.
- Or, upload file with the sequence of the peptide in FASTA format. No file selected.
- Choose the organism and/or genus where you want to search proteins containing the peptide.

<input checked="" type="checkbox"/> Acheta domestica (house cricket)	<input checked="" type="checkbox"/> Foeniculum vulgare (fennel)	<input checked="" type="checkbox"/> Pisum sativum (garden pea, pea)
<input checked="" type="checkbox"/> Allium	<input checked="" type="checkbox"/> Fragaria ananassa (strawberry)	<input checked="" type="checkbox"/> Portunus
<input checked="" type="checkbox"/> Apium graveolens (celery)	<input checked="" type="checkbox"/> Gadus	<input checked="" type="checkbox"/> Procambarus
<input checked="" type="checkbox"/> Beta vulgaris (sugar beet)	<input checked="" type="checkbox"/> Gallus gallus (chicken)	<input checked="" type="checkbox"/> Prunus
<input checked="" type="checkbox"/> Bos taurus (bovine, cattle)	<input checked="" type="checkbox"/> Hypophthalmichthys	<input checked="" type="checkbox"/> Punica granatum (pomegranate)
<input checked="" type="checkbox"/> Brassica rapa (field mustard)	<input checked="" type="checkbox"/> Labeo	<input checked="" type="checkbox"/> Pyrus (pears)
<input checked="" type="checkbox"/> Capsicum (peppers)	<input checked="" type="checkbox"/> Lactuca sativa (garden lettuce)	<input checked="" type="checkbox"/> Raphanus sativus (radish)
<input checked="" type="checkbox"/> Carassius	<input checked="" type="checkbox"/> Litopenaeus	<input checked="" type="checkbox"/> Rubus idaeus (raspberry)

Figura 42. Formulari per a introduir la informació referents als pèptids en la cerca de proteïnes (model formulari).

2. Purification params

Choose:

- Maximum number of purification steps.
- Minimum molecular weight difference (Da).
- Proteases

Max steps

Minimum MW diference (Da)

Proteases

3. Run

- Write a descriptive title for your search.
- Write and confirm the email where you want to receive the results.
- Click on RUN button to start search.

Search Title (descriptive title for your search)

Email

Repeat Email

Figura 43. Formulari per a introduir la informació referents als paràmetres de purificació i la (model formulari).

Finalment, en les figures Figura 44 i Figura 45 es mostra una captura amb un exemple de disseny de la finestra per a visualitzar els resultats en una taula i amb un cercador per tal de buscar, ordenar i filtrar aquest resultats pels camps que es mostren. Addicionalment, en aquesta taula s'ha afegit un botó de més informació que porta a l'usuari a una pàgina

amb text on principalment es mostra de forma organitzada en etapes la informació necessària per conèixer la traçabilitat d'aquell resultat en qüestió (Figura 45).

Results of the search PA

Search information

searchName = searchPA
date of the search = 2020-08-03 11:40:23.049610
Peptide = PA
Organisms = Allium - Bos taurus
Maximum number of purification steps = 3
Minimum molecular weight difference (Da) = 3.000
Proteases = All

Filtering options

Results

112 records found

Accession	Accession Name	Protein Name	Organism	Peptide Occurrences	Number Of End Fragments	End Fragments	Steps	Molecular Weight Difference	Peptidases	Peptidase1	Ec	Peptidase1	Peptidase
More details	Q32PD7	CSN9_BOVIN	COP9 signalosome complex subunit 9 ...	Bos taurus	1	1	[PA]	2	<10.0kDa-><3.0kDa	S01.151->S01.135	trypsin 1	3.4.21.4	granzyme
More details	Q32PD7	CSN9_BOVIN	COP9 signalosome complex subunit 9 ...	Bos taurus	1	1	[PA]	2	<10.0kDa-><3.0kDa	S01.151->S01.223	trypsin 1	3.4.21.4	acrosin
More details	Q32PD7	CSN9_BOVIN	COP9 signalosome complex subunit 9 ...	Bos taurus	1	1	[PA]	2	<10.0kDa-><3.0kDa	S01.151->S01.212	trypsin 1	3.4.21.4	plasma kallikrein

Figura 44. Pàgina de taula de resultats i informació de la cerca.

Results search [id=1]

Accession= Q32PD7
Protein Name=COP9 signalosome complex subunit 9
Organism= Bos taurus
Steps= 2
Proteases= trypsin 1 [S01.151 (EC: 3.4.21.4)] -> granzyme A [S01.135 (EC: 3.4.21.78)]
Molecular weight difference = <10.0kDa-><3.0kDa
Peptide_Occurrences= 1
End_Fragments= [PA]

Step 1

Protease: S01.151
Cleavage sites: [2-3] [37-38] [40-41]
Number of fragments: 4

Fragment 1:
Sequence: MK
Molecular weight: 277.38 Da
Size: 2 AA
Has peptide: No

Fragment 2:
Sequence: PAVDEMFPPEGAGPYVDLDEAGGSTGLLMDLAANEK
Molecular weight: 3580.90 Da
Size: 35 AA
Has peptide: Yes

Fragment 3:
Sequence: AVH
Molecular weight: 325.36 Da
Size: 3 AA
Has peptide: No

Fragment 4:

Figura 45. Pàgina de més informació sobre un resultat.

5.3 Disseny de la base de dades

Per a realitzar la base de dades primer de tot s'ha dut a terme un esquema entitat-relació de totes les dades de les quals es vol tenir persistència. En aquest s'indiquen les entitats i relacions entre elles així com la multiplicitat esperada i la obligatorietat/opcionalitat d'aquesta per cada relació/entitat. També s'indiquen els atributs de les relacions en cas de haver-hi.

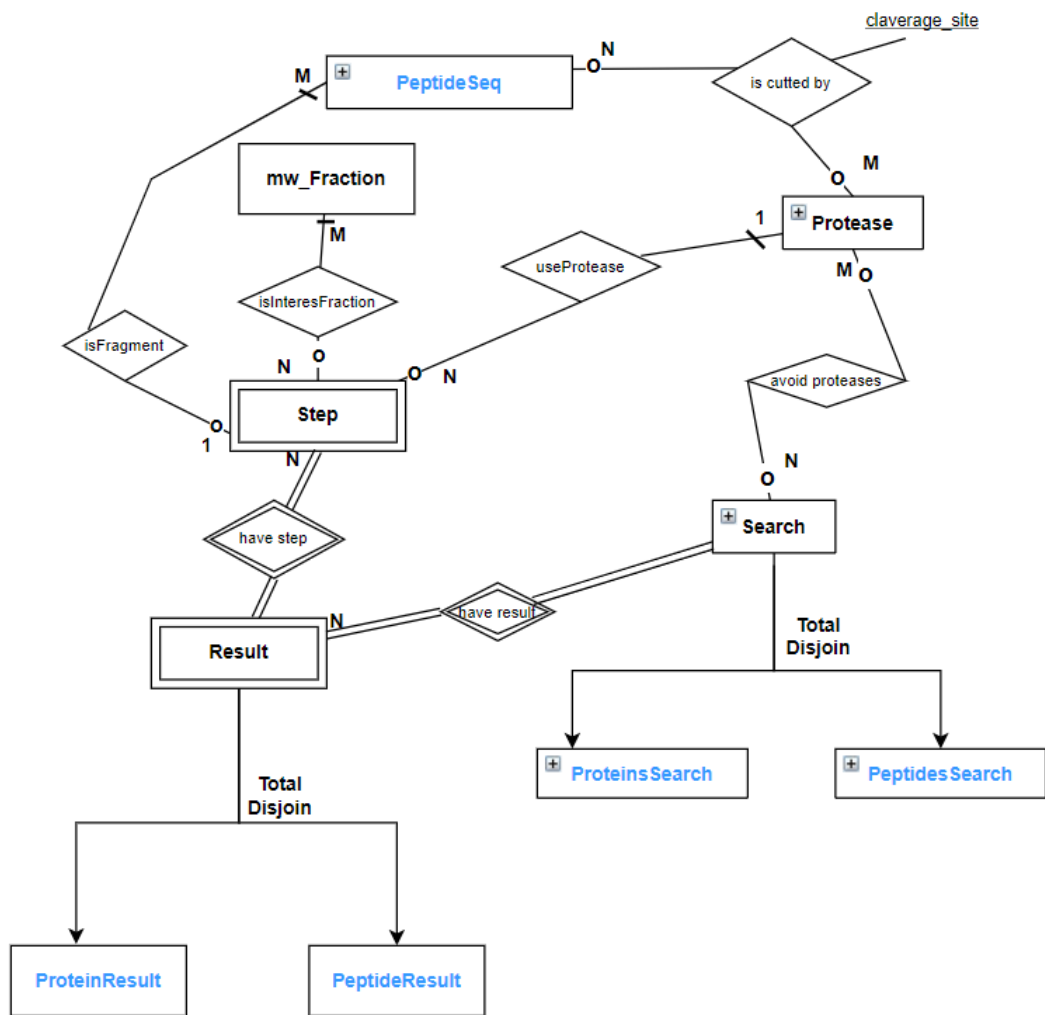


Figura 46. Diagrama entitat-relació en el que es representen les entitats i relacions comuns en totes les cerques.

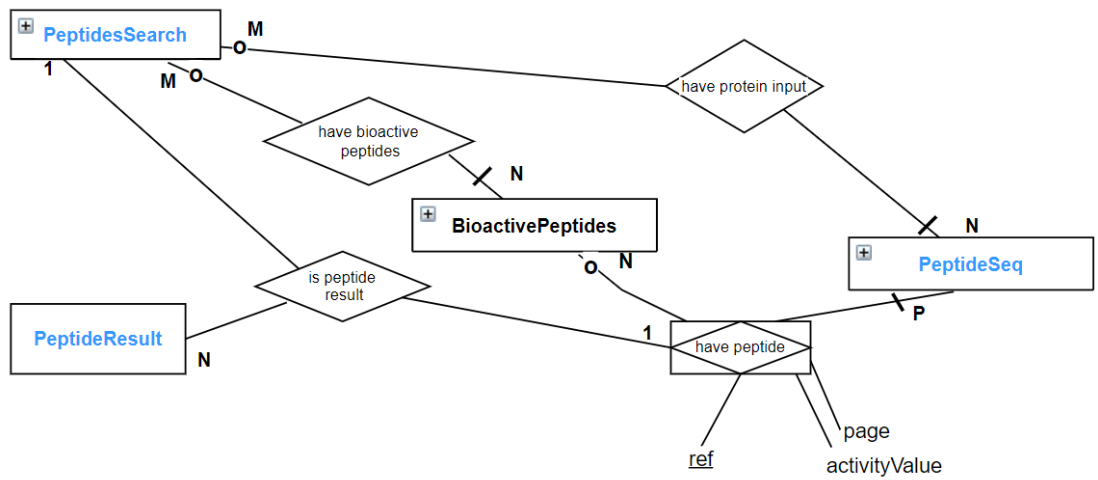


Figura 47. Diagrama entitat-relació en el que es representen les entitats i relacions particulars de les cerques de pèptids.

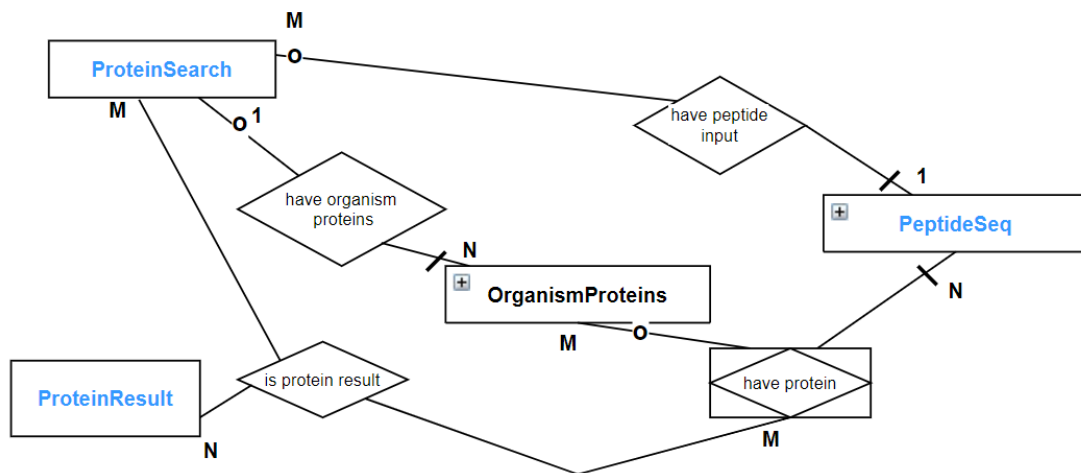


Figura 48. Diagrama entitat-relació en el que es representen les entitats i relacions particulars de les cerques de proteïnes.

Les dades de les quals es vol tenir persistència són principalment les dades relacionades amb les entrades i els resultats de les cerques. Per a tenir una visió més simplificada de l'esquema s'han realitzat 3 esquemes parcials. Per una banda, la representació de les dades comunes per a les cerques i els resultats dels dos tipus de cerca i per l'altra banda, un esquema per cada una de les cerques amb les dades particulars de cada una. A més a més, com es pot observar en les Figura 46, Figura 47 i Figura 48 s'han destacat en color blau les entitats en comú d'aquest diagrames.

A partir d'aquests esquemes de models-entitats i els atributs que s'han anat mencionat en els requisits funcionals s'ha definit la necessitat de les següents taules amb els respectius atributs i claus primàries i foranes:

search(id, search_name, url, min_mw, max_steps, search_type, proteases, user_email, date)
Primary Key: id

result(search, id_result)
Primary Key: search, id_result
Foreign Key: search REFERENCES search(id)

step(search, id_result, step, protease, claverages, all_fragments, final_fragments, fractions)
Primary Key: search, id_result, step
Foreign Key: search, id_result REFERENCES result(search, id_result)

peptideSeq(sequence, aa_seq, uniprotCode, all_cuts)
Primary Key: sequence

bioactivePeptides(id, bioactivity, target, effect)
Primary Key: id
Unique: bioactivity, target, effect

havePeptide(ref, sequence, activity, original_bioactivity_value, peptide_info_place, inhibition_type, activityValue)
Primary Key: ref, sequence, activity
Foreign Key: sequence REFERENCES peptideSeq(sequence)
Foreign Key: activity REFERENCES bioactivePeptides(id)

peptidesSearch(search, min_activity)
Primary Key: search
Foreign Key: search REFERENCES search(id)

haveProteinInput(search, sequence)
Primary Key: search, sequence
Foreign Key: search REFERENCES peptidesSearch(search)
Foreign Key: sequence REFERENCES peptideSeq(sequence)

haveBioactivePeptides(search, activity)
Primary Key: search, activity
Foreign Key: search REFERENCES peptidesSearch(search)
Foreign Key: activity REFERENCES bioactivePeptides(id)

isPeptideResult(doi, sequence, activity, search, id_result)
Primary Key: doi, search, sequence, id_result, activity
Foreign Key: search, id_result REFERENCES result(search, id_result)
Foreign Key: doi, sequence, activity REFERENCES havePeptide(ref, sequence, activity)
Foreign Key: search REFERENCES peptidesSearch(search)

organismProteins(scientificName, communName, ncbiURL)
Primary Key: scientificName

haveProteins(sequence, organismName)

Primary Key: sequence, organismName

Foreign Key: sequence REFERENCES peptideSeq(sequence)

Foreign Key: organismName REFERENCES organismProteins(scientificName)

proteinsSearch(search)

Primary Key: search

Foreign Key: search REFERENCES search(id)

haveOrganismProteins(organismName, search)

Primary Key: organismName, search

Foreign Key: search REFERENCES proteinsSearch(search)

Foreign Key: organismName REFERENCES organismProteins(scientificName)

havePeptideInput(search, sequence)

Primary Key: search, sequence

Foreign Key: search REFERENCES peptidesSearch(search)

Foreign Key: sequence REFERENCES peptideSeq(sequence)

isProteinResult(search, id_result, sequence, organismName)

Primary Key: search, sequence, id_result, organismName

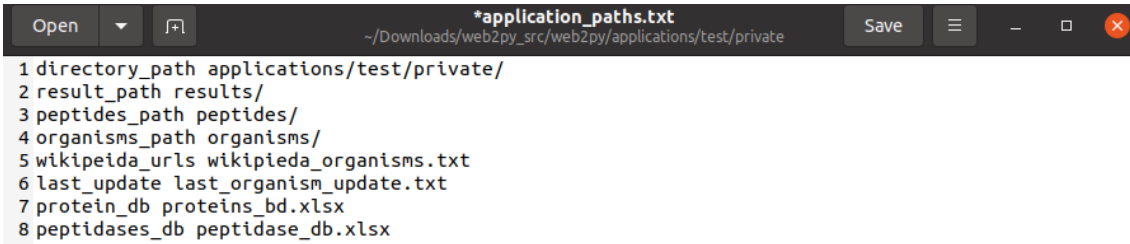
Foreign Key: search, id_result REFERENCES result(search, id_result)

Foreign Key: organismName REFERENCES organismProteins(scientificName)

Foreign Key: search REFERENCES proteinsSearch(search)

6 Implementació

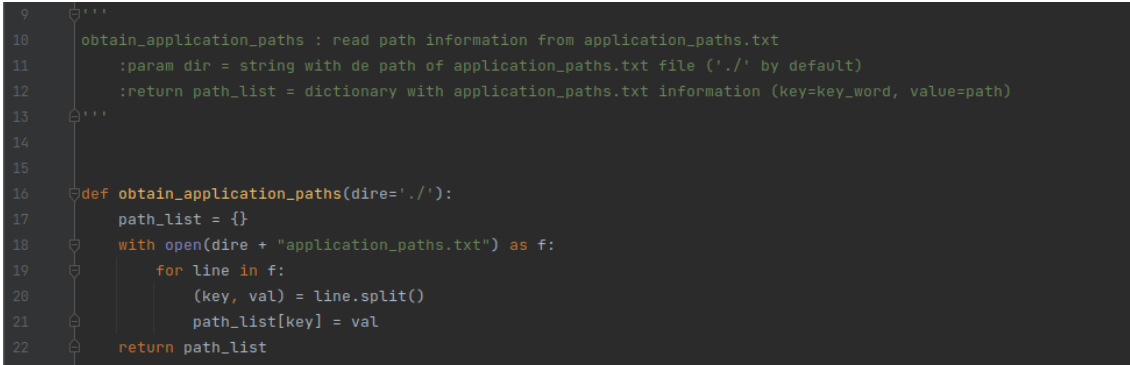
En aquest apartat s'esmenten els detalls més destacables de la implementació. Com es comenta en l'apartat Arquitectura i organització de fitxers de l'aplicació, s'ha generat el fitxer de text *application_paths.txt* el qual es pot interpretar com un diccionari que relaciona una paraula clau i el *path* del fitxer o directori associat. Aquest fitxer està localitzat en el directori *private* i es pot accedir al seu contingut amb les funcions del mòdul *global_functions.py*. L'estructura del fitxer *application_paths.txt* exemplificada en la Figura 49 es defineix amb una paraula “clau[espai]path” separats per salt de línia. Aquest fitxer es interpreta com un diccionari python en la funció *obtain_application_path(dire)* del fitxer *private/global_functions.py* (Figura 50) utilitzada pels diversos mòduls que necessiten accés a aquest directoris i fitxers.



```
*application_paths.txt
~/Downloads/web2py_src/web2py/applications/test/private
Save

1 directory_path applications/test/private/
2 result_path results/
3 peptides_path peptides/
4 organisms_path organisms/
5 wikipeda_urls wikipeda_organisms.txt
6 last_update last_organism_update.txt
7 protein_db proteins_bd.xlsx
8 peptidases_db peptidase_db.xlsx
```

Figura 49. Exemple de Contingut fitxer *application_paths.txt*.



```
9
10 obtain_application_paths : read path information from application_paths.txt
11 :param dir = string with de path of application_paths.txt file ('./' by default)
12 :return path_list = dictionary with application_paths.txt information (key=key_word, value=path)
13
14
15
16 def obtain_application_paths(dire='./'):
17     path_list = {}
18     with open(dire + "application_paths.txt") as f:
19         for line in f:
20             (key, val) = line.split()
21             path_list[key] = val
22     return path_list
```

Figura 50. Funció *obtain_application_path(dire)* del fitxer *private/global_functions.py*

6.1 Implementació de la base de dades

La base de dades s'ha implementat amb `mysql` i executat mitjançant la llibreria `pymysql` de `python`. Aquesta està definida en el fitxer `peptideGo_database.py` i les definicions explícites es poden consultar en les següents captures de codi. Tenint en compte, per exemple, atributs no nuls, identificadors utilitzant `AUTO_INCREMENT` en algunes taules, restriccions de valors en alguns atributs i `ON DELETE CASCADE` en les relacions de claus foranes on es necessari especificar.

```
30 def createGeneralTables(cursor=cursor):
31     cursor.execute("CREATE TABLE search ( id int AUTO_INCREMENT,"
32                   "search_name varchar(40) NOT NULL, "
33                   "url varchar(40) UNIQUE NOT NULL,"
34                   "min_mw int NOT NULL CHECK (max_steps>0),"
35                   "max_steps int NOT NULL CHECK (min_mw>0), "
36                   "search_type int NOT NULL CHECK (search_type = 1 or search_type = 2),"
37                   "proteases int NOT NULL CHECK (proteases = 1 or proteases = 2),"
38                   "user_email varchar(30) NOT NULL, "
39                   "date datetime NOT NULL, "
40                   "PRIMARY KEY (id))"
41                   "ENGINE=InnoDB;")
42
43     cursor.execute("CREATE TABLE result (search int, id_result int, "
44                   "PRIMARY KEY (search, id_result), "
45                   "FOREIGN KEY (search) REFERENCES search(id)) "
46                   "ENGINE=InnoDB;")
47
48     cursor.execute("CREATE TABLE step (search int, id_result int, step int,"
49                   "protease char(7) NOT NULL, claverages char(30) NOT NULL,"
50                   "all_fragments text NOT NULL, final_fragments text NOT NULL, fractions char(20) NOT NULL,"
51                   "PRIMARY KEY (search, id_result, step),"
52                   "FOREIGN KEY (search, id_result) REFERENCES result(search, id_result))"
53                   "ENGINE=InnoDB;")
54
55     cursor.execute("CREATE TABLE peptideSeq ( sequence int AUTO_INCREMENT,"
56                   "aa_seq text NOT NULL,"
57                   "uniprotCode varchar(10) UNIQUE, "
58                   "all_cuts text,"
59                   "PRIMARY KEY (sequence)) "
60                   "ENGINE=InnoDB;")
61
62
63 def createBioactivePeptidesTables(cursor=cursor):
64     cursor.execute("CREATE TABLE bioactivePeptides ( id int AUTO_INCREMENT UNIQUE,"
65                   "bioactivity varchar(30), target varchar(30), effect varchar(30),"
66                   "PRIMARY KEY (id), "
67                   "UNIQUE (bioactivity,target,effect))"
68                   "ENGINE=InnoDB;")
69
70     cursor.execute(
71         "CREATE TABLE havePeptide ( ref varchar(40), sequence int, activity int, original_bioactivity_value varchar(20),"
72         "peptide_info_place varchar(20), inhibition_type varchar(15),"
73         "activityValue float NOT NULL,"
74         "PRIMARY KEY ( ref, sequence, activity), "
75         "FOREIGN KEY (sequence) REFERENCES peptideSeq(sequence),"
76         "FOREIGN KEY (activity) REFERENCES bioactivePeptides(id) ON DELETE CASCADE )"
77         "ENGINE=InnoDB;")
```

```

79 def createSearchPeptideTables(cursor=cursor):
80     cursor.execute("CREATE TABLE peptidesSearch ( search int, "
81                   "min_activity float NOT NULL, "
82                   "PRIMARY KEY (search), "
83                   "FOREIGN KEY (search) REFERENCES search(id))"
84                   "ENGINE=InnoDB;")
85
86     cursor.execute("CREATE TABLE haveProteinInput ( search int, sequence int,"
87                   "PRIMARY KEY ( search, sequence),"
88                   "FOREIGN KEY (search) REFERENCES peptidesSearch(search),"
89                   "FOREIGN KEY (sequence) REFERENCES peptideSeq(sequence))"
90                   "ENGINE=InnoDB;")
91
92     cursor.execute("CREATE TABLE haveBioactivePeptides ( search int, activity int,"
93                   "PRIMARY KEY ( search, activity), "
94                   "FOREIGN KEY (search) REFERENCES peptidesSearch(search),"
95                   "FOREIGN KEY ( activity) REFERENCES bioactivePeptides(id) ON DELETE CASCADE )"
96                   "ENGINE=InnoDB;")
97
98     cursor.execute(
99         "CREATE TABLE isPeptideResult ( doi varchar(30), sequence int, activity int, search int, id_result int,"
100        "PRIMARY KEY ( doi, search, sequence, id_result, activity ),"
101        "FOREIGN KEY (search, id_result) REFERENCES result(search, id_result),"
102        "FOREIGN KEY (doi, sequence, activity ) REFERENCES havePeptide(ref, sequence, activity ),"
103        "FOREIGN KEY (search) REFERENCES peptidesSearch(search))"
104        "ENGINE=InnoDB;")
105
106
107 def createOrganismProteinsTables(cursor=cursor):
108     cursor.execute("CREATE TABLE organismProteins ( scientificName varchar(50), communName varchar(50), "
109                   "ncbiURL varchar(120),"
110                   "PRIMARY KEY (scientificName)) "
111                   "ENGINE=InnoDB;")
112
113     cursor.execute("CREATE TABLE haveProteins ( sequence int, organismName varchar(30),"
114                   "PRIMARY KEY ( sequence, organismName), "
115                   "FOREIGN KEY (sequence) REFERENCES peptideSeq(sequence) ON DELETE CASCADE, "
116                   "FOREIGN KEY (organismName) REFERENCES organismProteins(scientificName) ON DELETE CASCADE )"
117                   "ENGINE=InnoDB;")
118
119     cursor.execute("CREATE TABLE proteinsSearch ( search int, "
120                   "PRIMARY KEY ( search), "
121                   "FOREIGN KEY (search) REFERENCES search(id) ON DELETE CASCADE )"
122                   "ENGINE=InnoDB;")
123
124
125 def createSearchProteinTables(cursor=cursor):
126     cursor.execute("CREATE TABLE haveOrganismProteins ( organismName varchar(30), search int,"
127                   "PRIMARY KEY ( organismName, search), "
128                   "FOREIGN KEY (search) REFERENCES proteinsSearch(search),"
129                   "FOREIGN KEY (organismName) REFERENCES organismProteins(scientificName))"
130                   "ENGINE=InnoDB;")
131
132     cursor.execute("CREATE TABLE havePeptideInput ( search int, sequence int,"
133                   "PRIMARY KEY ( search, sequence), "
134                   "FOREIGN KEY (search) REFERENCES peptidesSearch(search),"
135                   "FOREIGN KEY (sequence) REFERENCES peptideSeq(sequence))"
136                   "ENGINE=InnoDB;")
137
138     cursor.execute("CREATE TABLE isProteinResult (search int, id_result int, sequence int, organismName varchar(30),"
139                   "PRIMARY KEY ( search, sequence, id_result,organismName), "
140                   "FOREIGN KEY (search, id_result) REFERENCES result(search, id_result),"
141                   "FOREIGN KEY (organismName) REFERENCES organismProteins(scientificName),"
142                   "FOREIGN KEY (search) REFERENCES proteinsSearch(search))"
143                   "ENGINE=InnoDB;")
144

```

6.2 Implementació de la solució dels casos d'ús

Referent a la solució dels casos d'ús 00.logIn i 01.logOut així com el requisit no funcional d'autenticació, s'ha utilitzat la base de dades d'usuaris i estructures de control d'accés i autorització (*auth*) definida per l'eina web2py en el model *db.py*. Per al control d'aquest accés s'utilitza el controlador referent a la vista *user.html*, es a dir, la funció *user()* en el fitxer *controllers/default.py*. Aquesta funció pot rebre per argument les opcions *login* i *logout* i redirigeix a l'usuari segons la opció i si està identificat o no (Figura 51).

En les vistes en les que cal estar autenticat per a poder accedir s'ha afegit l'etiqueta `@auth.requires_login` en les funcions del controlador *default.py* associades a aquestes vistes. Per exemple, la mostrada en la captura

Figura 52 en la funció del controlador de la pàgina principal d'administració. Amb aquesta acció es comprova prèviament si l'usuari ha realitzat *login* abans d'accedir a aquella vista, en cas negatiu es mostra la pàgina associada a *user.html* restringint l'accés a usuaris no administradors.

```
8 def user():
9     if request.args(0)=='login':
10         if auth.user is None:
11             return dict(form=auth())
12         else:
13             redirect(URL('admin'))
14
15     if request.args(0)=='logout':
16         if auth.user is None:
17             redirect(URL('index'))
18         else:
19             return dict(form=auth.logout())
20     return dict(form=auth())
21
```

Figura 51. Funció del controlador referent a la vista *user.html*

```
330
331 @auth.requires_login()
332 def admin():
333     return locals()
334
```

Figura 52. Captura del codi referent a la funció del controlador de la vista *admin.html* (pàgina principal d'administració).

Respecte la solució del cas d'ús 03.ManageProteins Organism localitzem el mòdul *def_organisms.py* el qual és una capa d'abstracció que utilitza algunes de les funcions del mòdul *peptideGo_database.py*. Entre aquestes funcions de *def_peptides.py* trobem les funcions *show_organisms*, *add_organism*, *delete_organism* i *update_db* utilitzades per la funció del controlador associada a vista de gestió de la informació de organismes/proteïnes (*adminOrganisms.html*) segons la interacció del usuari amb la interfície gràfica per a resoldre els requisits funcionals plantejats. També cal destacar que aquest mòdul conté altres funcions de suport per a realitzar aquestes accions, com per exemple per la obtenció de les seqüències proteïques i noms comuns del organisme a afegir o la creació de *threads* per a reduir el temps d'obtenció d'aquestes dades (complint amb els requisits no funcionals de rendiment) amb l'ús de un semàfor local per evitar conflictes. Aquest últim cas, exemplificat en la Figura 53 amb la funció *add_protein(code)*, s'executa per cada *thread* on es realitza la petició de la seqüència a afegir i posteriorment s'afegeix aquesta al *dataframe* global (*df*) amb control d'escriptura mitjançant el semàfor local *local_sema*.

En aquestes funcions cal destacar l'ús de les llibreries *os* i *openpyxl* per al maneig dels fitxers de proteïnes que estan en format excel; la llibreria *threading* per a la creació i gestió dels *threads*; *pandas* per a la gestió de dades i *biopython* per a la realització de diverses accions amb les dades biotecnològiques amb formats estàndards. Finalment, també cal destacar *request* per a la connexió amb les bases de dades externes (principalment UniProt).

```

143 '''
144     add_protein : search a protein on UniProt and add code+sequence on the global dataframe (df)
145     :param code = UniProt code of the protein to add
146     :return global dataframe with the add protein
147 '''
148
149 def add_protein(code):
150     try:
151         global df
152         data = peptids.get_up_sequence(code)
153         # Add protein to dataframe
154         local_sema.acquire() # critical section when add a new protein on the dataframe
155         df = pd.concat([df, pd.DataFrame({'UniProtCode': [code], 'Sequence': [data]})])
156         local_sema.release()
157     except:
158         print('***def_organisms.add_protein: Error on add/search ' + code + ' protein.')

```

Figura 53. Funció *add_protein* del mòdul *modules/def_organism.py*.

Respecte la solució del cas d'ús 04.ManagePeptides Files localitzem el mòdul *def_peptides.py* el qual és una capa d'abstracció que utilitza algunes de les funcions del mòdul *peptideGo_database.py* per tal de donar suport al controlador de la vista associada (*adminPeptides.html*). En *def_peptides.py* trobem els mètodes *show_files_names*, *add_file* i *delete_file* utilitzades per la funció del controlador associada a vista de gestió de la informació de pèptids/fitxers (*adminPeptides.html*) segons la interacció del usuari amb la interfície gràfica per a resoldre els requisits funcionals plantejats. En aquestes funcions cal destacar l'ús de les llibreries *csv* i *os* per al maneig dels fitxers de pèptids que estan en format *csv*.

Referent a la solució dels casos d'ús 05. SearchProteins i 06.SearchBiopeptides consisteixen en els fitxers vista *searchProteins.html* i *searchBiopeptides.html* respectivament. Aquests contenen també el cas d'ús 07. Search. Referent a aquestes vistes, cal mencionar el control de les dades introduïdes per l'usuari en prémer el botó RUN en javascript, mostrant missatges d'alerta en el cas de no ser vàlides (format incorrecte, no selecció, text en blanc...). En el cas de l'anàlisi de format FASTA o codis UniProt, s'ha utilitzat la cerca mitjançant correspondència amb expressions regulars amb el contingut de l'àrea de text. Primer s'han definit aquestes expressions a partir de la definició del format i el codis respectivament tenint en compte tots els possibles casos i,

posteriorment, mitjançant *RegExp* de JavaScript, s'ha analitzat si el contingut de l'àrea de text coincideix amb el format esperat. Un exemple d'aquesta solució es mostra en la Figura 54 on es veu una captura de la funció utilitzada en la vista *searchBiopoetides.html* per a comprovar si el format és correcte.

```

23
24 function validateText(){
25     var inputProteins =document.querySelector('input[name="inputProteins"]:checked').value;
26     var textArea = document.getElementById("inputText");
27     var text=textArea.value
28     var splittText = text.split('\n');
29     if (inputProteins.localeCompare('fasta')==0)
30     {
31         var textArea = document.getElementById("inputText");
32         var text=textArea.value
33         var splittText = text.split('\n');
34
35         if (splittText.length<2 )
36             textArea.setCustomValidity("Input format is not correct");
37         else
38         {
39             let regularFirstLine = new RegExp ("^>. *");
40             let regularAA = new RegExp ("^[ACDEFGHIKLMNPQRSTVWY]+$");
41             var correct= regularFirstLine.test(splittText[0]);
42
43             for(var i = 1;i < splittText.length-1;i++)
44                 correct = correct && regularAA.test(splittText[i]);
45             if (correct)
46                 textArea.setCustomValidity("");
47             else textArea.setCustomValidity("Input format is not correct");
48         }
49     }
50
51     if (inputProteins.localeCompare('uniprotCode')==0)
52     {
53         if (splittText.length!=1)
54             textArea.setCustomValidity("Input format is not correct (check if there are unnecessary spaces or line break)");
55         else
56         {
57             let regularFirstLine = new RegExp ("^[OPQ][0-9][A-Z0-9]{3}[0-9]$");
58             var correct= regularFirstLine.test(splittText[0]);
59             if (!correct)
60             {
61                 let regularFirstLine = new RegExp ("^[A-NR-Z][0-9]([A-Z][A-Z0-9]{2}[0-9]){1,2}$");
62                 var correct= regularFirstLine.test(splittText[0]);
63             }
64             if (correct)
65                 textArea.setCustomValidity("");
66             else textArea.setCustomValidity("Input format is not correct (check if there are unnecessary spaces or line break)");
67         }
68     }
69 }
70

```

Figura 54. Funció *validateText()* en el fitxer */views/default/searchBiopoetids.html*

En aquestes vistes també és important destacar la incorporació dels botons *selectAll* i *clearAll* i la relació en les opcions de pujar un fitxer o introduir text de forma manual. On si es puja un fitxer el contingut del àrea de text s'actualitza amb el contingut del fitxer seleccionat per tal d'evitar ambigüitats.

Respecte la solució del cas d'ús 07.Search inclòs en els casos d'ús 05. SearchProteins i 06.SearchBiopeptides s'ha decidit crear un mòdul *search.py* amb les funcions *search_proteins* i *search_peptides*. Aquestes realitzen les adaptacions de format necessàries dels paràmetres i utilitzen les funcions comunes definides del mòdul *find_peptides.py* per tal de realitzar les cerques amb els paràmetres introduïts per l'usuari.

A més, cal destacar que el mòdul *find_peptides.py* conté part del codi previ al projecte amb algunes correccions del algoritme que afectaven el correcte resultat de les cerques de més d'una etapa i adaptacions per a les especificacions de la plataforma.

Un cop realitzada la cerca es genera l'URL dels resultats d'aquesta, utilitzant el mòdul de python *secrets*, amb la funció `secrets.choice(range(20,40))` per obtenir un nombre aleatori entre 20 y 40 que serà la llargada (variable *len*) i `secrets.token_urlsafel(len)` per obtenir el URL. Posteriorment, es realitza una codificació d'aquest mitjançant SHA256 amb la llibreria de python *hashlib* per tal d'emmagatzemar aquest codi i utilitzar-lo per a poder reconèixer si el URL correspon a la cerca. Gràcies a aquest mecanisme s'obté un URL independent a les dades de la cerca i no s'emmagatzema directament aquest a la base de dades per tal de tenir més seguretat. Si algú arriba a tenir accés a elles no obtindrà el URL sinó el resultat de la funció hash.

Finalment, en aquest apartat de implementació cal mencionar la solució realitzada per al cas d'ús 09. ShowResults. S'ha creat una vista */default/results.html* on s'obtenen els resultats referent a la cerca a través del seu controlador, consultat la base de dades amb les funcions del mòdul *peptideGo_database*. Aquestes dades es mostren en forma de taula mitjançant una estructura predefinida en web2py anomenada `SQLFORM.smartgrid` que permet mostrar les dades en una taula i ordenar i filtrar segons els diferent paràmetres (Figura 44).

7 Avaluació.

Per a la realització de l'avaluació s'han realitzat proves unitàries de les funcions i accions implementades dels mòduls i seguidament s'han dissenyat proves relacionades amb els requisits funcionals. S'ha decidit separar aquest diferents proves per casos d'ús i especificar les diverses interaccions esperades (correctes i incorrectes) que pot tenir l'usuari amb la interfície gràfica. En les següents taules s'inclou el disseny i resultat de les proves amb una descripció d'aquestes, juntament amb la sortida esperada i la validació o comentari pertinent referent a si es compleix.

Taula 1. Proves referents a l'accés de l'usuari administrador.

00. LogIn		
Usuari introdueix nom d'usuari i contrasenya vàlids.	El sistema mostra la pàgina principal d'administració i l'usuari pot realitzar tasques administratives.	Correcte
Usuari introdueix nom d'usuari i contrasenya no vàlids.	El sistema mostra un missatge d'error.	Correcte
01. LogOut		
Usuari indica que vol fer logOut.	El sistema mostra la pàgina principal de l'aplicació i l'usuari ha de tornar a fer logIn per a poder realitzar tasques administratives.	Correcte

Taula 2. Proves referents a l'administració de les bases de dades.

02.ManageData		
L'usuari no té permisos d'administrador.	El sistema mostra la pàgina de logIn (ja sigui accedint a través de la interfície o accedint directament a qualsevol de les URL de la secció).	Correcte
L'usuari té permisos d'administrador.	El sistema mostra les opcions.	Correcte
L'usuari seleccionar administrar proteïnes	El sistema mostra la pàgina per administrar proteïnes.	Correcte
L'usuari seleccionar administrar pèptids	El sistema mostra la pàgina per administrar pèptids.	Correcte
03.ManageProteins_Organisms		
L'usuari intenta afegir un organisme nou vàlid que no es troba en la base de dades.	El sistema cerca i afegeix a la base de dades les proteïnes i el nom comú de l'organisme afegit.	Correcte
L'usuari intenta afegir un organisme nou no vàlid.	El sistema mostra un missatge d'error indicant que hi ha hagut un error en afegir.	Correcte

L'usuari intenta afegir un organisme es ja troba en la base de dades.	El sistema cerca i actualitza la base de dades les proteïnes i el nom comú de l'organisme afegit.	Correcte
L'usuari intenta eliminar un organisme existent.	El sistema esborra les dades referents a l'organisme.	Correcte
L'usuari intenta eliminar un organisme que no es troba en la base de dades.	El sistema mostra un missatge d'error indicant que hi ha hagut un error en eliminar.	Correcte
L'usuari intentar realitzar una actualització de les dades.	El sistema actualitza les dades de tots els organismes continguts a la base de dades.	Correcte
L'usuari intenta afegir sense escriure el nom del organisme.	El sistema mostra un missatge avisant a l'usuari que no s'ha introduït cap nom.	Correcte
L'usuari intenta eliminar sense escriure el nom del organisme	El sistema mostra un missatge avisant a l'usuari que no s'ha introduït cap nom.	Correcte
04.ManagePeptides_Files		
L'usuari intenta afegir un fitxer amb nom vàlid i no es troba a la base de dades.	El contingut del fitxer s'afegeix a la base de dades.	Correcte
L'usuari intenta afegir un fitxer amb nom no vàlid.	El sistema mostra un missatge avisant a l'usuari que el nom del fitxer no és vàlid.	Correcte
L'usuari intenta afegir un fitxer amb nom ja existent.	El sistema mostra un missatge avisant a l'usuari que ja existeix un fitxer amb aquest nom.	Correcte
L'usuari intenta actualitzar un fitxer amb nom que es troba en la base de dades.	El sistema actualitza el fitxer amb les dades contingudes al nou fitxer.	Correcte
L'usuari intenta actualitzar un fitxer amb nom que no es troba en la base de dades.	El sistema mostra un missatge avisant a l'usuari que no existeix un fitxer amb aquest nom.	Correcte
L'usuari intenta eliminar un fitxer que es troba en la base de dades.	Les dades contingudes al fitxer s'eliminen de la base de dades.	Correcte
L'usuari intenta eliminar un fitxer que no es troba en la base de dades.	El sistema mostra un missatge avisant a l'usuari que no existeix un fitxer amb aquest nom.	Correcte

L'usuari intenta actualitzar sense seleccionar cap fitxer.	El sistema mostra un missatge avisant que no s'ha seleccionat cap fitxer.	Correcte
L'usuari intenta afegir sense seleccionar cap fitxer.	El sistema mostra un missatge avisant que no s'ha seleccionat cap fitxer.	Correcte
L'usuari intenta eliminar sense escriure el nom del fitxer.	El sistema mostra un missatge avisant que no s'ha introduït cap nom de fitxer.	Correcte

Taula 3. Proves referents a la cerca de proteïnes

05. SearchProteins		
Validació paràmetres d'entrada (pèptid/organismes)		
L'usuari no introdueix cap seqüència de pèptid.	El sistema mostra un missatge avisant que no s'ha introduït cap seqüència de pèptids.	Correcte
L'usuari no introdueix una seqüència correcta en format FASTA.	El sistema mostra un missatge avisant que el format de la seqüència no és correcte.	Correcte
L'usuari introdueix una seqüència correcta seleccionant un fitxer.	El sistema llegeix i interpreta la informació del fitxer i la mostra en el quadre de text.	Correcte
L'usuari introdueix una seqüència correcta escrivint en el quadre de text.	El sistema llegeix i interpreta la informació del quadre de text.	Correcte
L'usuari selecciona tots els organismes.	El sistema no dona missatge d'error i selecciona tots els organismes.	Correcte
L'usuari no selecciona cap organisme.	El sistema mostra un missatge avisant que no s'ha seleccionat cap organisme.	Correcte
L'usuari selecciona un subconjunt d'organismes (diversos test: després de <i>ClearAll</i> , <i>SelectAll</i> i simplement interactuant amb les <i>checkbox</i>).	El sistema no dona missatge d'error.	Correcte

Cerques amb diferents paràmetres de purificació		
Cerca amb 1 <i>max steps</i> .	Els resultats de la cerca són els esperats i només tenen una etapa.	Correcte
Cerca amb 2 <i>max steps</i> .	Els resultats de la cerca són els esperats i tenen una o dues etapes.	Correcte
Cerca amb 3 <i>max steps</i> .	Els resultats de la cerca són els esperats i tenen una, dues o tres etapes.	Correcte
Cerca amb 3.000 min MW.	Els resultats de la cerca són els esperats i només s'obtenen resultats en els quals la mínima diferència de pes molecular és de 3.000 Da.	Correcte
Cerca amb 30.000 min MW.	Els resultats de la cerca són els esperats i només s'obtenen resultats en els quals la mínima diferència de pes molecular és de 30.000 Da.	Correcte
Cerca amb proteases comercials.	Els resultats de la cerca són els esperats i només inclouen els que s'utilitzen proteases comercials a totes les etapes.	Correcte
Cerca amb totes les proteases.	Els resultats de la cerca són els esperats i inclouen qualsevol proteasa de la base de dades (comercial i no comercial).	Correcte
Altres (nom cerca, correu i ajudes contextuais)		
L'usuari no introdueix títol de la cerca.	El sistema mostra un missatge avisant que no s'ha introduït títol de la cerca.	Correcte
L'usuari no introdueix el correu.	El sistema mostra un missatge avisant que no s'ha introduït el correu.	Correcte
L'usuari introdueix quelcom que no té format de correu en el correu.	El sistema mostra un missatge avisant que el format de correu no és correcte.	Correcte
La repetició del correu no es vàlida.	El sistema mostra un missatge avisant que la repetició del correu no es correspon.	Correcte
L'usuari consulta les ajudes contextuais.	El sistema mostra les ajudes contextuais.	Correcte

Taula 4. Proves referents a la cerca de pèptids.

05. SearchPeptides		
Validació paràmetres d'entrada (proteïnes/fitxers)		
L'usuari no introdueix cap seqüència de proteïna.	El sistema mostra un missatge avisant que no s'ha introduït cap seqüència de proteïna.	Correcte
L'usuari selecciona que introduirà la seqüència en format FASTA i no introdueix una seqüència correcta en format FASTA.	El sistema mostra un missatge avisant que el format FASTA de la seqüència no és correcte.	Correcte
L'usuari selecciona que introduirà els codis de UniProt i no introdueix dades vàlides.	El sistema mostra un missatge avisant que el format dels codis UniProt no és correcte.	Correcte
L'usuari introdueix una seqüència FASTA correcta a partir d'un fitxer	El sistema llegeix i interpreta la informació del fitxer i la mostra en el quadre de text.	Correcte
L'usuari introdueix un llistat de codis de UniProt correctes a partir d'un fitxer.	El sistema llegeix i interpreta la informació del fitxer i la mostra en el quadre de text.	Correcte
L'usuari introdueix una seqüència FASTA correcta escrivint en el quadre de text.	El sistema llegeix i interpreta la informació del quadre de text.	Correcte
L'usuari introdueix un llistat de codis de UniProt correctes escrivint en el quadre de text.	El sistema llegeix i interpreta la informació del quadre de text.	Correcte
L'usuari selecciona tots els fitxer de pèptids.	El sistema no dona missatge d'error i selecciona tots els noms de fitxers.	Correcte
L'usuari no selecciona cap fitxer de pèptids.	El sistema mostra un missatge avisant que no s'ha seleccionat cap fitxer de pèptids.	Correcte
L'usuari selecciona un subconjunt els fitxer de pèptids. (diversos test: després de <i>ClearAll</i> , <i>SelectAll</i> i simplement interactuant amb les <i>checkbox</i>).	El sistema no dona missatge d'error.	Correcte
Cerques amb diferents paràmetres de purificació		
Cerca amb 1 <i>max steps</i>	Els resultats de la cerca són els esperats i només tenen una etapa.	Correcte

Cerca amb 2 <i>max steps</i>	Els resultats de la cerca són els esperats i tenen una o dues etapes.	Correcte
Cerca amb 3 <i>max steps</i>	Els resultats de la cerca són els esperats i tenen una, dues o tres etapes.	Correcte
Cerca amb 3.000 min MW	Els resultats de la cerca són els esperats i només s'obtenen resultats en els quals la mínima diferència de pes molecular és de 3.000 Da.	Correcte
Cerca amb 30.000 min MW	Els resultats de la cerca són els esperats i només s'obtenen resultats en els quals la mínima diferència de pes molecular és de 30.000 Da.	Correcte
Cerca amb proteases comercials.	Els resultats de la cerca són els esperats i només inclouen els que s'utilitzen proteases comercials a totes les etapes.	Correcte
Cerca amb totes les proteases.	Els resultats de la cerca són els esperats i inclouen qualsevol proteasa de la base de dades (comercial i no comercial).	Correcte
Altres (nom cerca, correu i ajudes contextuais)		
L'usuari no introdueix títol de la cerca	El sistema mostra un missatge avisant que no s'ha introduït títol de la cerca.	Correcte
L'usuari no introdueix el correu	El sistema mostra un missatge avisant que no s'ha introduït el correu.	Correcte
L'usuari introdueix quelcom que no té format de correu en el correu	El sistema mostra un missatge avisant que el format de correu no és correcte.	Correcte
La repetició del correu no es vàlida.	El sistema mostra un missatge avisant que la repetició del correu no es correspon.	Correcte
L'usuari consulta les ajudes contextuais	El sistema mostra les ajudes contextuais.	Correcte

8 Conclusions.

A grans trets la valoració del projecte ha estat positiva. S'ha pogut realitzar la totalitat de l'aplicació web per a resoldre la necessitat del grup de recerca d'agilitzar el procés de cerca a de pèptids amb bioactivitat en proteïnes.

Dins d'aquesta aplicació, acadèmicament s'han reforçat conceptes d'anàlisi dels requisits funcionals i no funcionals i disseny i implementació de base de dades de forma autònoma. Un dels grans reptes d'aquest projecte ha estat saber interpretar el que volien els futurs usuaris per tal d'adaptar-se a les seves necessitats i el procés d'aprenentatge de com funciona el model vista-controlador, així com la programació amb HTML5 i javaScript de la qual no tenia coneixements previs. Addicionalment, també s'han ampliat els coneixements relacionats amb el maneig de dades biotecnològiques.

A nivell personal ha estat satisfactori formar part d'un grup de recerca per a realitzar un projecte com aquest, que té una repercussió en el món de la bioinformàtica. Així com formar part de en la presa de decisions de totes les parts de l'aplicació.

Realitzant aquest treball, he pogut apreciar que un projecte informàtic en la seva totalitat és un procés complex en el qual s'ha de realitzar un anàlisi previ a la implementació, el qual pot arribar a ser fins i tot més important que aquesta.

9 Recursos Utilitzats

1. UniProt Available online: <https://www.uniprot.org/>.
2. Rawlings, N.D.; Barrett, A.J.; Thomas, P.D.; Huang, X.; Bateman, A.; Finn, R.D. The MEROPS database of proteolytic enzymes, their substrates and inhibitors in 2017 and a comparison with peptidases in the PANTHER database. *Nucleic Acids Res.* **2018**, *46*, D624–D632, doi:10.1093/NAR/GKX1134.
3. PubMed Available online: <https://pubmed.ncbi.nlm.nih.gov/>.
4. www.cheminformatics-nutrition.recerca.urv.cat | Portada Available online: <http://www.cheminformatics-nutrition.recerca.urv.cat/ca/>.
5. UML 2 diagramming, OO software modeling, Source code engineering Tool MagicDraw UML from No Magic Available online: https://www.magicdraw.com/main.php?ts=navig&cmd_go_to_login_custom_pages=1&menu=download_no_cost_plugins&back_cmd=cmd_show.
6. PyCharm: the Python IDE for Professional Developers by JetBrains Available online: <https://www.jetbrains.com/pycharm/>.
7. XAMPP Installers and Downloads for Apache Friends Available online: <https://www.apachefriends.org/es/index.html>.
8. Home - Taxonomy - NCBI Available online: <https://www.ncbi.nlm.nih.gov/taxonomy>.