

Desenvolupament d'una eina de càlcul de compatibilitat a partir de tests de portadors de mutacions genètiques

Antoni Requena

Resum — En els darrers anys, la investigació i els nous descobriments en el camp de la genètica humana estan augmentant de manera considerable. Malgrat que encara queda molt per descobrir, tot allò que a hores d'ara ja es coneix té una gran utilitat en el camp de la reproducció assistida. És ben sabut que la infertilitat ha estat quelcom present des de sempre, però des de ja fa uns anys existeixen solucions per tal que una parella amb aquest problema pugui tenir igualment descendència. Hi ha hospitals que disposen de bancs de donants per tal de poder fecundar artificialment als pacients que ho sol·licitin. En aquest escenari, el que busca el pacient és una descendència el més sana possible, i és en aquest moment on entra en joc aquest projecte.

A nivell genètic, els humans tenim un seguit de gens que poden presentar mutacions. Algunes d'elles provoquen malalties, mentre que d'altres no comporten cap problema. Tanmateix, el que sí que resulta problemàtic és el creuament entre dos humans, amb mutacions comunes en els seus respectius gens. En un banc de donants on a cadascú se l'hi ha fet un estudi genètic, a mesura que creix el número de donants es complica el procediment de selecció del donant òptim. Aquest projecte doncs es presenta com una automatització i optimització d'aquest procés de selecció.

Paraules clau — *genètica humana, test de portadors, mutacions genètiques, bancs de donants, donant òptim*

Abstract — Over the last few years, research and new discoveries in the field of human genetics are increasing a lot. Although there are still many unknown parts, everything that is known by now has a more than considerable utility in the field of assisted reproduction. It is well known that infertility has been something always present, but for the last few years there have been solutions for couples suffering from this problem. There are hospitals that have donor banks in order to artificially fertilize customers who request it. In this scenario, the client is seeking the healthiest donor for its case, and it is here where this project comes into play. At the genetic level, humans have a set of genes which may have mutations. Some of them cause diseases while others do not cause any problem. However, what is problematic is the crossing between two humans with common mutations in their genes. In a bank in which each donor has been genetically studied, the selection of the optimal offspring becomes more complicated as the donor bank grows. This project therefore comes as an automation and optimization of this selection process.

Index Terms — *human genetics, carrier test, genetic mutations, donor banks, optimal donor*



1 INTRODUCCIÓ

Q GENOMICS [8] és una empresa que se situa dins del sector de la genètica i la bioinformàtica. Ofereix serveis d'anàlisi genètica a hospitals.

Una de les anàlisis que ofereix s'anomena "qCarrier" [1], i consisteix en un test de portador de mutacions genètiques. Aquesta anàlisi serveix principalment a hospitals que tenen bancs de donants (tan d'ovaris, com d'esperma) i és molt útil ja que permet estudiar un gran nombre de gens del pacient, per detectar si en algun d'ells hi té alguna mutació.

Aquest estudi de mutacions és clau. La classificació de les mutacions és complex i l'efecte que té sobre l'individu també. Aquest coneixement va més enllà d'aquest projecte i per tant no s'hi entrarà en detall. Tanmateix, el que és necessari que quedi clar és que el creuament entre dos individus que tinguin mutació/ns comunes, seria fatal pel seu fill.

Per tal d'evitar aquestes circumstàncies, a tots els donants

d'un hospital se'ls ha fet un test de portadors per conèixer les seves mutacions. Tots els individus tenen mutacions, però algunes d'elles se sap que no tenen cap conseqüència. D'altres en canvi, fan que un donant es descarti directament com a tal.

Un individu pot estar sa malgrat tingui mutacions en gens concrets. Tot i així, si aquest individu es creua amb un altre que té una mutació comuna (o més), la seva descendència probablement presentaria problemes de salut.

L'objectiu per un hospital amb banc de donants és el d'oferir al client aquell donant que no té cap mutació comuna amb ell. Aquest donant serà considerat l'òptim.

Davant d'aquest escenari doncs, el que proposa aquest projecte és el fet d'oferir als hospitals que ho desitgin un servei que automatitzi aquest procediment de selecció del donant òptim. Un cop finalitzat aquest projecte, es disposarà d'un software capaç de calcular la compatibilitat entre persones a nivell genètic. Amb això els hospitals

guanyaran tan en rapidesa (del procés de selecció) com en fiabilitat (el donant que proposarà el software serà l'òptim per al client en qüestió).

En els següents apartats d'aquest document s'explica l'estat de l'art, on es defineixen els conceptes i les idees més rellevants que formen part del context del treball a més a més de citar treballs relacionats. A continuació s'expliquen els objectius del projecte, i en un següent apartat es detalla la metodologia que s'ha seguit.

Seguidament, s'inclou un apartat en el que s'explica el desenvolupament, explicant breument cada tasca executada.

Finalment, s'exposen els resultats obtinguts del projecte, i unes conclusions.

2 ESTAT DE L'ART

En l'apartat anterior s'ha parlat d'un test de portadors, que cal explicar una mica en detall per tal de que contextualitzat més el projecte.

El test de portadors permet identificar en els futurs pares d'un individu la presència de mutacions en gens causants de fins a 200 enfermetats, com per exemple la diabetis neonatal, la fibrosi quística, entre altres. Per tant és útil tant per un banc de donants, com per una parella que vol tenir fills. Les casuístiques possibles a l'hora de tenir descendència són tal i com es veu en la figura 1.

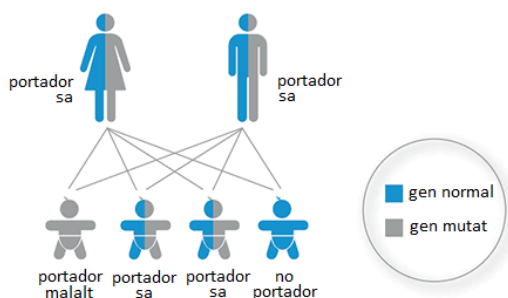


Figura 1. Casos possibles en la transmissió de mutacions a un fill

Una parella pot ser portadora d'una mutació, i a l'hora de tenir fills pot ser que es transmeti aquesta mutació, o bé pot ser que no es transmeti. I fins i tot, en cas que es transmeti, pot passar que el fill estigui lliure de malaltia genètica. Tanmateix, el risc de que un fill presenti malalties genètiques, si els pares tenen mutacions comunes en gens determinats és molt elevada.

Actualment, *qGenomics* no és l'únic laboratori que ofereix aquest tipus d'anàlisi als hospitals. Hi ha d'altres laboratoris que ofereixen aquest tipus de servei. Alguns exemples són *Ginemed* [6], *Igenomix* [7], entre altres.

En el cas d'una parella que vol tenir fills i es fa un estudi d'aquest tipus, el projecte que s'està proposant en aquest document no hi intervé. La utilitat del projecte apareix en el cas dels hospitals que disposen d'un banc de donants. En aquest cas, es parteix de la base en que l'hospital té tot el seu banc de donants estudiat a nivell genètic, i en el moment que un client es vol fecundar, l'hospital li ofereix el donant òptim per a ell a nivell genètic. Com ja s'ha explicat amb anterioritat, aquest projecte es presenta com un servei que s'oferirà a hospitals per a automatitzar i optimitzar aquest procés de selecció del donant òptim donat un receptor (client).

Cal dir doncs, que en aquest cas *qGenomics* tampoc és l'única empresa que ofereix aquest tipus de servei a hospitals. Els laboratoris *Igenomix* [7] també ofereixen un software que compleix aquesta funció. El motiu bàsic doncs del desenvolupament d'aquest projecte és la pressió de la competència. Més enllà d'*Igenomix* hi ha molts pocs laboratoris que ofereixin aquest servei addicional, i el fet de poder oferir un servei d'aquestes característiques als hospitals serà molt atractiu de cara a aconseguir nous clients i a mantenir els actuals.

3 OBJECTIUS

L'objectiu final del projecte és el d'obtenir un producte (software) que automatitzi el procés de selecció del donant òptim, donat un banc de donants d'un hospital concret.

Per tal d'assolir-ho, es defineix una llista de sub-objectius els quals ajudaran a aconseguir l'objectiu final, i es dona una prioritat a cadascun. A continuació hi ha una taula amb aquesta prioritització, i seguidament hi ha una descripció de cada objectiu marcat.

NOM	PRIORITAT
1. Definir els requeriments	Alta
2. Crear una estructura de dades per guardar la informació	Alta
3. Muntar un sistema d'exportació	Alta
4. Programar l'algorisme de càlcul de compatibilitat	Molt alta
5. Programar una interfície gràfica	Mitjana
6. Validació	Alta

Taula 1. Taula de prioritització dels objectius del projecte

A continuació s'hi troba una descripció de cadascun dels objectius que apareix a la taula 1 d'aquest document.

- 1. Definir els requeriments.** Els requeriments d'aquest nou software hauran de quedar ben definits, de tal manera que un cop finalitzat es pugui determinar si realment el software compleix amb aquests requeriments definits.
- 2. Crear una estructura de dades per guardar la informació.** Els pacients que vagin apareixent en aquest nou software, hauran de guardar-se d'una

- E-mail de contacte: antoni.requena@gmail.com
- Menció realitzada: *Tecnologies de la Informació*.
- Treball tutoritzat per: Ramon Martí (dEIC)
- Curs 2015/16

forma ordenada i estructurada. De cada pacient es guardaran dades de diferents tipus més enllà de l'anàlisi en si mateix. Es guardarà l'hospital al que pertany, el metge que va sol·licitar l'anàlisi, etc. És per això, que es dissenyarà una base de dades [3] que permeti guardar, per a cada pacient, tota la informació necessària.

3. **Muntar un sistema d'exportació.** S'hauran d'exportar les dades d'un pacient que s'ha acabat d'analitzar cap al nou software. El flux de la informació serà el següent: En primer lloc, l'hospital enviarà una mostra a *qGenomics* sol·licitant un anàlisi de "qCarrier". *qGenomics* ja disposa d'un software LIMS (*Laboratory information management system*), el qual permet portar el control de totes les mostres que s'estan analitzant i en la fase de l'anàlisi en que estan. Un cop acabat l'anàlisi d'aquesta mostra doncs, el LIMS tindrà un mòdul (la programació del qual també entra en el context d'aquest TFG) que exportarà les dades d'aquest pacient cap al nou software de càlcul de compatibilitat (que a partir d'ara en aquest document ens hi referirem com a *qMatching*).
4. **Programar l'algorisme de càlcul de compatibilitat.** És la part central de tot el projecte. Més endavant en aquest document s'explica en detall com funcionarà aquest algorisme.
5. **Programar una interfície gràfica.** S'haurà de dotar a aquest nou software d'una interfície gràfica el més amigable i intuïtiva possible per tal de fer que l'usuari s'hi adapti fàcil i ràpidament. Aquesta interfície gràfica es comunicarà amb la base de dades per obtenir la informació que s'haurà de mostrar a través de les diferents pantalles de l'aplicació.
6. **Validació.** S'hauran d'aplicar sistemes de test per tal de comprovar la correcta funcionalitat dels diferents mòduls a mesura que es vagi desenvolupant el projecte.

4 METODOLOGIA

A continuació es defineix una metodologia de treball específica que s'utilitzarà per dur a terme cada petita part amb la que es dividirà el projecte. La forma de treballar serà a partir de la definició de diferents tasques. El projecte anirà avançant a mesura que cada tasca es vagi implementant i validant. Per a cada tasca (que quedarà definida en la planificació), s'aplicarà la metodologia que s'exposa al final d'aquest apartat en forma de 4 punts. Cal dir que molt probablement algunes tasques requeriran més dedicació en algun d'aquests 4 punts, mentre que d'altres potser no tant.

1. Analitzar el problema a resoldre. Definir clarament els requeriments que tindrà el nou software, per tal de deixar clar què és allò que s'haurà de poder fer i què és allò que no. En un primer moment es definiran uns requeri-

ments generals per al projecte, però després per a cada tasca n'hi haurà uns d'específics.

2. Disseny. Informar-se sobre aquells conceptes desconeguts (i/o poc sòlids) que siguin necessaris per la part del projecte que s'estigui desenvolupant. Un cop es tingui clara la informació, i tots els requeriments que s'hauran definit en el punt anterior, fer el disseny per a després passar al següent pas.
3. Realitzar la implementació.
4. Test. És bàsic verificar els resultats en aquest projecte. S'haurà de poder garantir que l'*output* que donarà el programa en cap cas serà erroni.

Com a metodologia de desenvolupament (implementació) de software, s'utilitzarà el sistema RAD, Rapid Application Development [2], l'explicació del qual no s'inclou en aquest document, però per que en quedi clara la idea bàsica de funcionament, la filosofia del RAD consisteix en el desenvolupament i la construcció iteratius de prototipus. En cada iteració es va refinant el resultat que finalment es vol obtenir. En el projecte que s'està desenvolupant doncs, hi ha una llista de tasques, cadascuna de les quals representa una iteració que implicarà un anàlisi, un disseny, una implementació i un test.

Finalment, com que la major part del projecte consisteix en programació, es crearà un repositori (basat en *Git*) que servirà per documentar i portar un control de tots els canvis que hi vagin havent en el codi de forma organitzada.

5 DESENVOLUPAMENT

En aquest apartat s'inclou l'explicació de totes les tasques que s'han desenvolupat durant el projecte. En l'Annex A1 del present document s'hi pot trobar el diagrama de Gantt.

5.1 Anàlisi de requeriments

Aquesta primera tasca consisteix en la definició d'uns requeriments generals. Així doncs, i a grans trets, es vol programar un software (servidor web) amb una base de dades de pacients, i una interfície gràfica que interactui amb aquesta base de dades i sigui capaç de calcular la compatibilitat dels individus a nivell genètic. Els pacients s'aniran exportant en aquest nou sistema de forma automàtica quan s'hagin acabat d'analitzar en el laboratori.

5.2 Crear una estructura de dades per guardar la informació

Pel que fa aquesta tasca i seguint la planificació establerta, en primer lloc es detallen els requeriments que ha de tenir la base de dades deixant clar quines són les dades que haurà de guardar.

A partir d'això, se segueix amb el disseny de la base de dades, que en el cas d'aquest projecte consisteix en representar els requeriments en un diagrama entitat-relació (Apèndix A2) elaborat amb l'eina *Glyffy* [5]. Un cop s'ha

elaborat aquest diagrama, s'ha procedit a la traducció a taules d'aquest diagrama (base de dades relacional, que es troba a l'Apèndix A3), amb les seves relacions, claus primàries i claus externes.

5.2.1 Requeriments de la base de dades

En primer lloc, i com qualsevol sistema al qual s'hi accedeix amb unes credencials, el sistema tindrà un conjunt d'usuaris. Cada usuari haurà de tenir un nom d'usuari, una contrasenya, un mail, la data de creació de l'usuari i la data d'última modificació de la informació d'aquest usuari.

Cada usuari generarà *logs*, que també es guardaran a la base de dades. Pel que fa a els *logs*, el que es vol guardar és l'acció que s'ha fet, una descripció i una data (de creació i de modificació).

Tots els usuaris tindran associats un metge, que a la vegada formarà part d'un centre (hospital). Per cada metge s'ha de poder guardar el nom, el cognom, un telèfon i una data (de creació del metge i d'última actualització de la seva informació).

Tal i com s'ha dit abans, un metge formarà part d'un únic hospital, però cada hospital podrà tenir més d'un metge. Per cada hospital es vol guardar el nom, una data (de creació i de modificació), i també es vol representar d'alguna manera si un hospital determinat comparteix els pacients. D'entrada, quan un usuari entra al sistema, pot veure exclusivament els pacients de l'hospital on ell treballa, però podria ser que un hospital vulgués compartir els seus pacients.

Així doncs, un hospital tindrà un conjunt de pacients, però un pacient només podrà pertànyer a un únic hospital. El pacient, a més de formar part d'un hospital, també formarà part d'un metge. Igual que un hospital, un metge també tindrà un conjunt de pacients, però un pacient tampoc podrà pertànyer a més d'un metge. Per cada pacient que partanyi a un metge, el metge haurà de poder desfer comentaris de qualsevol tipus, però que majoritàriament serà informació clínica rellevant. Per cada comentari es vol guardar una data (de creació i de modificació) i el missatge (comentari).

Pel que fa als pacients, per cadascun es guardarà un número d'història clínica, i dos identificadors més (diferents). Addicionalment es guardarà el nom, els cognoms, el sexe, la data de naixement i una data (de creació i modificació). El que també es guardarà és si un pacient està habilitat i també si un pacient és donant. Això ens serveix ja que tal i com es planteja tot el projecte, quan s'exporti un nou pacient cap al *qMatching* perquè ja s'ha acabat el seu anàlisi al laboratori, ja podrà sortir com a mostra compatible per a una altra mostra. De totes maneres, podria ser que aquest nou pacient s'hagi fet el test de portadors per veure si és compatible amb la seva parella, i no pas per entrar al banc de donants de l'hospital en qüestió. Per aquest motiu, els metges d'un hospital que utilitzin el *qMatching* han de tenir la possibilitat de marcar un pacient com a donant o com a no-donant. Pel que fa al fet de deshabilitar-lo, podria donar-se el cas que a un pacient se l'hi hagués detectat una mutació que l'hospital

considera que ja no pot entrar en el seu banc de donants, doncs bé, també ha de poder deshabilitar-lo.

D'alguna manera també s'han de representar les parelles existents en el sistema.

Per a cada pacient, també s'ha de poder representar la seva informació fenotípica. S'entén com a informació fenotípica el color dels ulls, el color dels cabells, el grup sanguini, l'alçada i la raça. Tot i així, també es guardarà una data (de creació i de modificació d'aquesta informació fenotípica).

A continuació segueixen els requeriments que fan referència a la informació que es vol desfer del test de portadors que s'haurà fet a cada pacient.

Fins ara s'ha anat parlant del test de portadors (*qCarrier*), doncs bé, és molt important remarcar que no hi ha només un tipus de *qCarrier*. Tot i així, el que canvia d'un tipus de test a un altre és simplement la llista de gens que s'analitzen. Això és degut a que l'ADN humà conté una gran quantitat de gens (s'estima que uns 20.000) i seria molt costós buscar mutacions en tots ells. De fet no només seria costós, si no que en alguns casos no serviria de res, doncs es podrien trobar mutacions que a dia d'avui se'n desconeixen les implicacions.

És per aquest motiu que necessitarem poder representar diferents tipus de test a la base de dades, així com també la llista de gens que es mira en cadascun d'ells.

Per cada tipus de test haurem de poder guardar un nom, una data (de creació i de modificació), l'hospital al qual pertany, i una llista de gens. Per cada gen guardarem un nom, també guardarem si aquell gen es mira de forma completa, una data (de creació i de modificació), i el tipus de test al qual pertany.

Per cada pacient, es guardarà una prova (*qCarrier*). De la prova d'un pacient es guardarà una data (de creació i de modificació), el pacient al qual pertany la prova i el tipus de test de portadors de la prova.

A més a més, per la prova que s'haurà fet un pacient es guardarà per una banda totes les mutacions que s'hagin trobat en aquella prova, i també les regions dels gens que no hagin quedat cobertes (pel motiu que sigui, que a nivell de la base de dades igualment no ens interessa).

Per cada mutació guardarem el gen, un codi per identificar la mutació (cada mutació que pot aparèixer en un gen té un codi que l'identifica de forma única), el cromosoma, la posició d'inici i de fi en el cromosoma, una referència (que indicarà què és el que s'hauria hagut d'observar en aquell gen per no classificar-ho com a mutació), un observat (que representa allò que s'ha observat en el gen, i que difereix de la referència), una classificació (per indicar si la mutació és patotènica o bé *vous*, que significa que encara que sigui una mutació, no es classifica com a patogènica).

Pel que fa a les regions no cobertes guardarem el gen, i un codi intern i un d'extern d'identificació.

5.2.2 Diagrama entitat relació i traducció a taules

Fins aquí s'han exposat tots els requeriments que haurà de complir la base de dades. A partir d'aquí s'han traslladat tots aquests requeriments a un diagrama entitat-

relació, que es pot trobar en l'Apèndix A2 del present document. A continuació el que s'ha fet és traslladar aquest diagrama entitat-relació a un esquema de taules (base de dades relacional). Aquest esquema de taules es pot trobar en l'apèndix A3 del present document. Un cop amb aquest esquema, s'ha programat l'*script* en sql encarregat de crear la base de dades.

5.2.3 Test de la base de dades

Un cop amb la base de dades muntada s'ha procedit a fer-ne un test, per validar-ne el funcionament. Aquest test ha consistit bàsicament en dues fases:

- En primer lloc fer insercions a les diferents taules, i utilitzant les relacions entre les mateixes per representar informació del que podrien ser pacients reals. S'han creat mutacions per als diferents pacients de test, que a la vegada s'han assignat a un metge i un hospital. Un cop s'ha tingut la base de dades amb suficient informació per a omplir totes les taules (i les relacions) s'ha seguit al següent pas.
- El següent pas ha consistit en realitzar consultes, tenint en compte la informació que s'havia introduït prèviament. D'aquesta manera s'ha pogut validar si els resultats de les consultes eren consistents amb l'informació que s'hi havia introduït al principi.

5.3 Muntar un sistema d'exportació

5.3.1 Requeriments del sistema d'exportació

Anteriorment en aquest document, es fa referència a l'Apèndix A3, que conté la representació de la base de dades en taules i les relacions entre elles. Aquest esquema doncs es tindrà molt present en aquest apartat.

En primer lloc cal aclarir quina és la informació que s'exportarà cap a la *qMatching*. Els següents punts es poden considerar com a requeriments del mòdul d'exportació:

- Informació sobre el pacient (amb la qual omplirem la taula *patient* del *qMatching*). També s'entén com a informació del pacient l'hospital i el metge al qual "pertany".
- Informació de la parella del pacient (per omplir la taula *partner*).
- Informació del test de portadors. Serà informació genèrica del test i no pas dels resultats del mateix. La informació servirà per omplir la taula *carrier_test*.
- Informació de les mutacions. Tal com indica el nom, seran totes les mutacions que se l'hi hauran trobat a un pacient, i que s'hauran de desar a la taula *mutation*.
- Informació de les regions no cobertes. Igual que les mutacions, però en aquest cas s'haurà de guardar a la taula *not_covered_genes*.

5.3.2 Estudi de la base de dades i de l'estructura del codi del LIMS

Un cop aclarida la informació que s'exportarà, s'ha fet un estudi de la base de dades del LIMS per veure en quines taules s'ha d'anar a buscar la informació. En els apartats següents, on s'explicarà com s'ha programat el mòdul d'exportació, aquesta tasca quedarà també explicada. No es dedicarà un apartat exclusivament per a l'explicació de la mateixa.

A continuació, s'ha estudiat com estava estructurat el codi del LIMS per veure com s'havia de programar i integrar el nou mòdul d'exportació. Bé doncs, està programat en PHP en la seva totalitat i segueix el paradigma de programació estructurada (i no orientada a objecte). Seguint aquesta estructura doncs, s'ha creat un nou fitxer que a grans trets conté una funció anomenada *exportToqMatching(idMostra)* en la qual se li passa un identificador de la mostra del pacient (identificador en la BD del LIMS) i aquesta funció s'encarrega d'exportar tota la informació del pacient cap a la BD del *qMatching*. Un cop programat aquest nou mòdul, s'ha estudiat com s'integra en el LIMS. Per integrar-lo només s'ha hagut de decidir en quin punt es fa la crida a aquesta funció. Quan una mostra s'analitza, arriba un punt que es marca aquella mostra com a finalitzada (s'ha acabat el seu anàlisi) i també es marca com que ja s'ha enviat l'informe al metge sol licitant. És després de marcar la mostra com a "informe enviat", quan es farà la crida a la funció d'exportació.

5.3.3 Programació del mòdul d'exportació de dades

A continuació s'entra més en detall en el funcionament intern de la funció *exportToqMatching(idMostra)*.

Primer de tot es mira si la mostra ja existeix en la BD del *qMatching*, ja que si existeix, no es fa res.

A continuació s'exporta la informació sobre el pacient. Aquesta informació es va a buscar en varies taules del LIMS: *pacients*, *peticions_pendents*, *mostres*, *qMatching_assoc_hospitals* i *qMatching_assoc_metges*.

Pel que fa al metge al qual pertany la mostra, hi haurà una taula (*qMatching_assoc_metges*) que relacionarà els id's que tenen els metges en el LIMS amb els id's que tenen en el *qMatching*. El que també es fa en aquest punt és fer una inserció en la taula *sys_log* del *qMatching* indicant que s'ha exportat un pacient.

Seguidament el que es mira és si la mostra en qüestió té parella. Això es fa consultant la taula *relacioPacients* del LIMS. En cas que tingui parella, es fa una comprovació important a destacar. Es mira si la seva parella existeix també en el *qMatching*. En cas positiu, s'informa de la relació de parella en el *qMatching*, però en cas negatiu no es fa res. Aquesta situació podria donar-se si un membre de la parella s'ha marcat com a "informe enviat" però l'altre membre encara està en fase d'anàlisi de laboratori, per exemple.

A continuació s'exporta la informació del test de *qCarrier*. No dels resultats del test, si no la informació general del mateix. La informació es va a buscar a les taules *carrier*, *qMatching_assoc_tipus_prova* i *patient* de la base de dades del LIMS. Aquesta taula *qMat-*

ching_assoc_tipus_prova relaciona id's del *LIMS* amb id's del *qMatching*.

Per omplir la taula *mutation* del *qMatching* es va a buscar la informació a les taules *informearrier*, *carrierObservacio* i *mutació* de la base de dades del *LIMS*. El procediment és consisteix en iterar sobre totes les mutacions i per cadascuna, desar-ne la informació en el *qMatching*. Pel que fa a la taula *not_covered_genes* del *qMatching*, la informació no està desada directament a la base de dades si no a un fitxer. A la base de dades el que hi ha és el *path* d'aquest fitxer (per a cada mostra) i el que es fa és iterar les línies d'aquest fitxer i per cada línia (que representa una regió no coberta) s'exporta la informació cap al *qMatching*.

5.3.4 Test del mòdul d'exportació

Per fer aquest test el que s'ha fet és cridar manualment la funció *exportToqMatching(idMostra)*, amb diferents id's de mostres les quals s'havien marcat com a "informe enviat". A continuació s'han fet consultes a la base de dades (per cada mostra exportada) per veure si la informació s'havia exportat correctament (es contrastava amb la informació de la mostra que hi havia al *LIMS*). S'han fet un nombre considerable d'exportacions de test per cobrir diferents casos que podien donar-se.

5.4 Desenvolupament del nou software

5.4.1 Requeriments de la interfície gràfica

L'aplicació ha de tenir com a pantalla inicial un *login*, en el que l'usuari entrarà les seves credencials. Si són errònies l'aplicació continuarà en la mateixa pantalla de *login*. En cas que siguin vàlides, es passarà a una pantalla on l'usuari tindrà un menú (que es mantindrà per a totes les diferents pantalles) i una llista de pacients. Aquesta llista de pacients es mostrarà en una taula i per cada pacient apareixerà el seu número d'historial clínica, un identificador de la mostra, i el gènere de la mostra.

L'usuari podrà clicar qualsevol dels pacients que apareguin en la taula. Quan en cliqui un, anirà a parar a una altra pantalla en la que podrà veure informació detallada del pacient. Per una banda informació personal, i per l'altra informació de l'anàlisi del pacient. Com a informació personal haurà d'aparèixer el sexe, la data de naixement, el gènere, i l'identificador de la seva parella (si en té). Pel que fa a l'anàlisi del pacient, haurà d'aparèixer la versió del test que es va fer el pacient (i la data), i una llista amb les mutacions (patogèniques i vus) i una altra llista de les regions no cobertes.

En aquesta mateixa pantalla haurà d'aparèixer informació fenotípica del pacient. Aquesta informació no s'exportarà del *LIMS*, i per tant haurà de poder informar-se en aquesta pantalla. També haurà d'aparèixer un botó per deshabilitar el pacient, i un altre per marcar-lo com a donant (o no).

Per passar a la següent pantalla, s'haurà de clicar un botó amb el nom de *Matching*. En la següent pantalla apareixerà un conjunt d'opcions que es tindran en compte posteriorment en l'algorisme de *matching*. Aquestes opcions seran un filtre fenotípic, en que l'usuari podrà selec-

cionar les característiques fenotípiques que hauran de complir les mostres objectiu amb les quals es creuarà la mostra índex. Aquest filtrat quedarà com es veu en la figura 2. Quan l'usuari hagi seleccionat les opcions (pot no seleccionar-ne cap) anirà a una següent pantalla en la que s'haurà aplicat l'algorisme de *matching* contra tots els

Select options

Ethnicity:	Eye color:	Blood type:
<input type="checkbox"/> Caucasian	<input type="checkbox"/> Brown	<input type="checkbox"/> A+
<input type="checkbox"/> Asian	<input type="checkbox"/> Black	<input type="checkbox"/> A-
<input type="checkbox"/> European	<input type="checkbox"/> Green	<input type="checkbox"/> B+
	<input type="checkbox"/> Blue	<input type="checkbox"/> B-
Height:	Hair color:	<input type="checkbox"/> AB+
Any	<input type="checkbox"/> Brown	<input type="checkbox"/> AB-
	<input type="checkbox"/> Blond	<input type="checkbox"/> O+
	<input type="checkbox"/> Dark	<input type="checkbox"/> O-
	<input type="checkbox"/> Redhead	

Figura 2. Opcions de filtrat fenotípic de pacients del *qMatching*.

pacients del sexe contrari (i que compleixin amb el filtre fenotípic de la pantalla anterior). A més, estaran ordenats amb el que anomenarem "*risk ratio*" que representarà el risc de reproducció.

Per cada pacient d'aquesta llista haurà de sortir el seu historial clínic, un identificador de la mostra, el tipus de test que s'ha fet i el *risk ratio*. A més, per cada pacient haurà d'aparèixer un botó que portarà l'usuari a una altra pantalla en la que podrà veure els detalls del *matching* de la mostra índex contra la que haurà clicat.

En aquesta pantalla, apareixerà una taula resumint les coincidències de mutacions del pacient índex contra l'altra, i haurà d'aparèixer també una taula amb els detalls d'aquestes coincidències.

La taula resum que hem dit abans, simplement mostrarà un valor que indiqui el número de coincidències per a cada tipus de mutació (patogència, vus) i regions no cobertes.

En la taula detallada, apareixerà una llista de totes les coincidències que s'hagin trobat. Per cada coincidència sortirà el cromosoma, el gen, el codi de la mutació, i si aquell gen s'està mirant de forma completa.

Aquestes són totes les pantalles que ha de tenir l'aplicació. Al principi de l'apartat s'ha comentat que hauria d'aparèixer un menú en totes elles. Bé doncs en aquest menú sortirà un botó per poder fer *logout*, un botó de *My Account* per veure el perfil de l'usuari i un botó per tornar a la pantalla inicial, que serà on l'usuari veurà la llista de tots els seus pacients.

5.4.2 Disseny de la interfície gràfica

Un cop definits tots els requeriments de la interfície gràfica, es procedeix al disseny començant per un prototipus en paper (*paper prototyping*). Un cop validat aquest model en paper, es segueix amb la programació de la plantilla en *html*.

5.4.3 Test de la interfície gràfica

El test de la interfície gràfica s'ha dut a terme validant en primer lloc que tots els botons de la plantilla html tinguessin la funcionalitat desitjada.

A més, també s'ha validat que la plantilla es renderitzés correctament en els navegadors més habituals (safari, firefox, chrome i internet explorer).

5.4.4 Programació del sistema de login (control d'usuaris)

La pantalla de login ja estava feta (*html*), però en aquesta tasca se l'hi ha donat capacitat d'interacció amb la base de dades (*php*). Les credencials entrades per un usuari es validen a la taula *user* de la base de dades. En cas que siguin vàlides, es redirigeix a la pantalla inicial on l'usuari pot veure els seus pacients. Tot i així, en aquesta tasca no s'ha desenvolupat aquest fet de que un usuari vegi únicament els seus pacients. En aquesta tasca simplement es valida l'usuari. En cas de credencials incorrectes, es torna a mostrar la pàgina de *login* amb un missatge d'error.

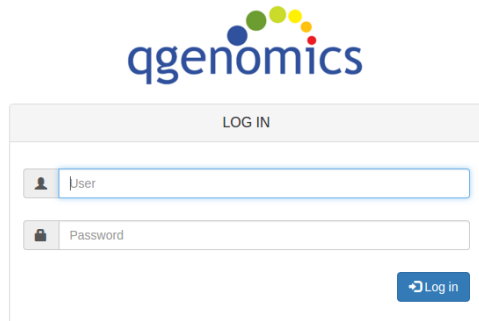


Figura 3. Pantalla de login del *qMatching*

5.4.5 Test del sistema de login

Aquesta tasca ha consistit en intentar fer *login* amb unes credencials incorrectes. Un cop vist que el login no tenia èxit, s'ha donat per validat el sistema.

5.4.6 Programació del llistat de pacients tenint en compte l'usuari

Com s'ha explicat amb anterioritat, en la base de dades un usuari està relacionat amb un metge, i aquest metge pertany a un hospital. A més, un hospital té un conjunt de pacients. Així doncs, el que s'ha fet ha estat consultar aquest llistat de pacients per mostrar-lo.

5.4.7 Test del llistat de pacients

Per dur a terme el test d'aquesta part, el que es va fer va ser crear un conjunt de pacients inventats a la base de dades, i assignar-los a diferents hospitals també inventats. A continuació es van crear metges pels hospitals, i per cada metge un usuari.

Un cop fet això, es va entrar a l'aplicació amb les diferents credencials creades per veure si amb cadascuna es mostren els pacients de l'hospital correcte.

5.4.8 Programació de la resta de pantalles (interacció amb la BD)

En la figura 3 es pot veure la primera pantalla a la qual s'accedeix en iniciar l'aplicació. És una pantalla amb un formulari de login on l'usuari entra amb les seves credencials. En cas que no siguin vàlides, es torna a mostrar la mateixa pantalla de login amb un missatge d'error. En cas contrari, es passa a la pantalla de la figura 4 (llistat de pacients). Cada usuari, que representa un metge, i forma part d'un hospital, podrà veure únicament un llistat amb tots els pacients del seu hospital. Els pacients de d'altres

Patient list

Sample id	Historial number	Gender
016327	016327	Female
016329	016329	Female
016381	016381	Male
016382	016382	Female
016383	016383	Female
016439	016439	Female
016570	016570	Female
016572	016572	Male
016615	016615	Male
016629	016629	Male

Figura 4. Pantalla de llistat de pacients del *qMatching*

hospitals no els podrà veure.

Tots els pacients que apareguin en aquesta llista es podran clicar, i en fer-ho anirem a parar a la pantalla de veure la informació del pacient. En aquesta pantalla apareix molta informació sobre el pacient en qüestió.

En primer lloc, apareix un *switch* que permet habilitar i deshabilitar el pacient. Aquesta informació està guardada a la taula *patient* de la base de dades.

També hi apareix un apartat d'informació personal. La informació d'aquest apartat també està extreta de la taula *patient*. Pel que fa a la informació personal, hi ha l'apartat de la parella del pacient. En cas que a la base de dades del *qMatching* un pacient tingui parella, en aquest apartat sortiria l'identificador de la parella i un botó de *matching* directe.

En cas que el pacient no tingui parella, apareix un botó per tal que l'usuari pugui informar d'aquesta parella. En clicar-lo, s'obre un *modal*, en el qual l'usuari entra l'id d'un pacient. Quan després clica a *save*, es comprova que aquest pacient existeixi a la base de dades, que sigui del sexe contrari i que no tingui ja una parella assignada. Al mateix nivell que la informació personal, hi ha un apartat de fenotip. Quan un hospital sol licita un anàlisi a *qGenomics*, no envia la informació fenotípica. És per aquest motiu que des del *qMatching* l'hospital podrà editar la informació fenotípica dels seus pacients en cas que la vulgui utilitzar. La pantalla d'edició del fenotip és tal i com es veu en la figura 5. Consisteix en un *modal* que s'obre quan es clica el botó d'edició. Quan l'usuari clica *save*, la informació fenotípica es queda guardada a la taula *phenotype* de la base de dades.

Phenotype information

Ethnicity: not informed

Blood type: not informed

Hair color: not informed

Eye color: not informed

Height (cm):

Save! Close

Figura 5. Pantalla d'edició del fenotip d'un pacient

A més del fenotip i de la informació personal del pacient, en aquesta pantalla també hi apareix informació sobre l'anàlisi que se li va fer al pacient. La informació que es mostra de l'anàlisi està guardada a les taules *carrier_test*, *carrier_test_version*, *not_covered_genes*, *disease* i *mutation* de la base de dades. En primer lloc es mostra la data en que es va fer l'anàlisi i la versió de l'anàlisi. A continuació, hi apareixen totes les mutacions que se l'hi van trobar en el pacient. Estan separades en tres apartats. En primer lloc apareixen les mutacions patogèniques, en les que es mostra el gen on apareix la mutació, el codi de la mutació i la malaltia associada a la mutació en qüestió. En segon lloc, apareixen les mutacions classificades com a *vous*, que significa que per la mutació trobada, se'n desconeixen les conseqüències.

En tercer i últim lloc, apareixen les regions no cobertes. Com ja s'ha explicat amb anterioritat, cada versió de *qCarrier* estudia una llista finita de gens. A vegades, i per culpa de la complexa tècnica que s'utilitza al laboratori el que passa és que queden regions amb insuficient informació com per determinar si hi ha mutacions o no. Tota la informació que queda en aquestes zones, es classifica com a regions no cobertes.

En l'apartat d'anàlisi, també hi apareix un botó de *matching* el qual porta a l'usuari cap a la pantalla de filtrat fenotípic de pacients per fer el *matching*, tal i com es veu en la figura 2.

Permet a l'usuari fer un filtrat previ a nivell fenotípic. En cas que l'usuari no seleccioni cap tipus de filtre, l'únic filtre que s'aplicarà serà el de pacients del sexe contrari (com és lògic i evident). En cas de que seleccioni algun tipus de filtre, per exemple que sigui de raça europea, doncs els pacients que es mostraran a continuació seran tots els del sexe contrari de raça europea.

Quan l'usuari tingui seleccionats tots els criteris desitjats, clicar al botó de *matching* i anirà a parar a la pantalla on apareixen tots els *matchings* calculats amb els pacients del sexe contrari (filtrats per fenotip, si és el cas).

En aquesta pantalla el que apareix és una taula de resultats amb tots els pacients del sexe contrari que compleixien els criteris de filtrat de la pantalla anterior. Aquesta taula és tal i com es veu en la figura 6.

En aquesta pantalla ja s'ha aplicat l'algorisme de *matching* contra tots els pacients. En apartats posteriors ja s'explicarà en detall el funcionament d'aquest algorisme. Per a cada pacient, apareix el seu número d'historial clí-

nic, un identificador de la mostra, el nom de la versió d'anàlisi que es va aplicar al pacient, i un índex que hem anomenat *risk ratio*. Aquest índex és el que es fa servir com a criteri d'ordenació dels pacients en aquesta taula. Els que tenen un *risk ratio* més baix apareixen més amunt en la taula.

Aquest índex es un dels *outputs* de l'algorisme de *matching*. Es calcula seguint la següent metodologia: Cada mutació en una regió no coberta val 1. Cada mutació classificada com a *vous* val per 100, i cada mutació classificada com a patogènica val per 1000. Tenint en compte això, quan s'aplica l'algorisme de *matching* entre dos pacients, es multiplica cada mutació en comú que es tro-

Results

Show 10 entries Search:

History Num.	Sample Id	Panel	RISK RATIO	Actions
016572	016572	qCarrier	3	Matching Details
016615	016615	qCarrier	4	Matching Details
016629	016629	qCarrier	5	Matching Details
016631	016631	qCarrier	5	Matching Details
016636	016636	qCarrier	5	Matching Details
016637	016637	qCarrier	5	Matching Details

Figura 6. Pantalla de tots els resultats del *qMatching*

ba. És a dir; si un pacient té una mutació patogènica, i en l'altre pacient la mateixa mutació està com a *vous*, el valor que se sumarà al *risk ratio* serà 100×1000 , i així per totes les mutacions.

D'aquesta manera doncs, els pacients més compatibles sempre sortiran a la part de dalt de la taula. Cada fila (que correspon a un pacient) tindrà un botó de *Matching details*, que en clicar-lo portarà a l'usuari a una pantalla per veure el detall d'un *matching* concret.

En aquesta pantalla es veu el detall del *matching* entre dos pacients. Hi apareix una taula de resultat del *matching* molt més detallada tal i com es pot veure en la figura 7. En aquesta taula, apareixen totes les mutacions que té el donant i totes les mutacions que té el receptor. La gràcia però, està en que en cas que tinguin una mutació en co-

Receiver 016629				Donor 016570	
Chr	Gene	Mutation code	Gene completeness	Mutation code	Gene completeness
chr7	CFTR	c.C4277T;p.S1426F (Pat)	complete	c.F75C;p.Z32S (Vous)	complete
chr19	BCKDHA	c.T910C;p.F304L (Vous)	complete		
chr21	HLCS			c.G520A;p.E174K (Pat)	targeted

Figura 7. Pantalla de resultats d'un *matching* en concret

mú, la fila del gen afectat per aquesta mutació sortirà marcada en vermell. Per cada mutació, apareixerà el cromosoma afectat, el gen, el codi de la mutació, i si el gen en qüestió s'ha mirat de forma completa o no.

Aquest apartat, realment està separat en dues taules (tal i com ja es veu en la imatge). En primer lloc la taula de mutacions patogèniques i *vous*, i en segon lloc la taula de regions insuficientment cobertes.

5.4.9 Test de la tasca anterior

Per tots els pacients que s'analitzen al laboratori, s'acaba obtenint un informe, on apareixen entre moltes altres coses totes les mutacions trobades en el pacient en qüestió. És per aquest motiu que per dur a terme el test de la tasca anterior s'han agafat un nombre considerable de pacients ja analitzats al laboratori, i s'han exportat cap al *qMatching* utilitzant l'algorisme d'exportació. S'han assignat tots a un mateix hospital, i s'ha agafat l'informe de cadascun d'ells. A continuació, s'han començat a fer *matching* entre ells, comprovant que les dades resultants de l'algorisme de *matching* són coherents amb les dades de l'informe. És a dir, si en la pantalla de resultats apareix una mutació patogènica en ambdós pacients (i per tant marcada en vermell), s'ha mirat l'informe dels dos pacients per comprovar que en els dos casos aparegués aquesta mutació i estigués marcada com a patogènica. Aquest procediment s'ha seguit per a cadascun dels pacients exportats, creuant-los tots amb tots.

5.4.10 Programació de l'algorisme de compatibilitat

La informació que utilitza l'algorisme de compatibilitat és la que està guardada a les taules *patient*, *carrier_test*, *carrier_test_version*, *not_covered_genes*, *mutation* de la base de dades. La programació d'aquest algorisme s'ha plantejat com a una única funció que rep dos paràmetres, que són dos identificadors de pacient. El primer és el receptor i el segon és el donant.

A continuació el que es fa és accedir a la taula *mutation* i la taula *not_covered_genes* per obtenir tres arrays per al donant i tres arrays pel receptor. El primer array tindrà totes les mutacions patogèniques, el segon totes les mutacions classificades com a *vous*, i el tercer array tindrà les mutacions classificades com a no cobertes.

Ara el que es vol obtenir és la següent informació:

- Patogèniques del receptor, comunes amb les patogèniques del donant
- Patogèniques del receptor, comunes amb les *vous* del donant
- Patogèniques del receptor, comunes amb les no cobertes del donant
- *Vous* del receptor, comunes amb les patogèniques del donant
- *Vous* del receptor, comunes amb les *vous* del donant
- *Vous* del receptor, comunes amb les no cobertes del donant
- No cobertes del receptor, comunes amb les patogèniques del donant
- No cobertes del receptor, comunes amb les *vous* del donant
- No cobertes del receptor, comunes amb les no cobertes del donant

Tenint en compte això, i sabent que ja es tenen els tres arrays per separat, s'ha hagut d'iterar sobre els arrays per mirar els elements que tenen en comú, que en definitiva són les mutacions que tenen en comú.

L'output d'aquesta funció doncs consisteix en els 9 punts

que s'han detallat algunes línies més amunt. A més d'aquests 9 punts, aquesta funció també fa el càlcul del *risk ratio*.

5.4.11 Test de l'algorisme de compatibilitat

Per fer el test d'aquest algorisme el que s'ha fet és executar-lo per a molts pacients diferents. En finalitzar l'execució es mirava l'*output* que donava, i es comprovava que fos coherent amb les dades dels informes dels pacients. És a dir, si en l'algorisme ha donat com a output una mutació comuna, que és patogènica en el receptor i *vous* en el donant, s'ha consultat l'informe de tots dos per mirar que realment aquesta mutació existís i estés classificada correctament.

6 PRESENTACIÓ DE RESULTATS

Tal i com s'ha anat explicant en els anteriors apartats, aquest projecte ha consistit en el desenvolupament d'un sistema que automatitza i optimitza el càlcul de compatibilitat entre pacients a partir de tests de portadors de mutacions genètiques.

El que s'ha acabat obtenint és una aplicació web amb una base de dades de *backend* amb la informació genètica de tots els pacients que han estat analitzats a *qGenomics*.

En un primer moment es van definir uns requeriments generals de l'aplicació, i per cada tasca se n'han definit de més específics, aclarint què és allò que l'aplicació havia de poder fer i què és allò que no.

Les dades resultants de les anàlisis ja s'estaven guardant a la base de dades del *LIMS* de *qGenomics*, però s'ha desenvolupat un mòdul que exporta aquestes dades cap a la base de dades de l'aplicació *qMatching*.

Els hospitals que han enviat (o bé que estan enviant) mostres de pacients per analitzar, sol llicitant un *qCarrier*, ara ja disposaran d'un sistema on podran no només veure els anàlisis de tots els pacients d'una forma centralitzada, si no que també podran veure compatibilitats a nivell genètic entre ells.

Tota l'aplicació s'ha desenvolupat pensant en tot moment amb la interfície gràfica, per tal que sigui el més intuïtiva i amigable possible.

A més, s'ha dut a terme un procés de test per a totes les tasques que s'han anat desenvolupant, per tal de validar el correcte funcionament del codi.

Per un hospital que disposa de bancs de donants molt gran, el *qMatching* li suposa una millora més que considerable, ja que la selecció del donant òptim passa a fer-se de forma automatitzada, guanyant rapidesa de servei i anul·lant possibles errors humans.

El resultat obtingut d'aquest projecte és un quelcom merament tècnic, en el sentit de que consisteix en un software, i res més enllà d'això. Tanmateix, tal i com ja s'explicarà en el següent apartat, en un futur proper aquest projecte es vestirà amb un marc legislatiu, per un motiu bàsic. Les dades amb les que es tracten són de caire genètic, i això té unes implicacions legals molt importants, que si bé en l'actual projecte no estan cobertes, en un període de temps curt no hauria de ser així.

7 CONCLUSIONS

En la realització d'aquest treball s'ha pogut observar tot el procés necessari per a desenvolupar una aplicació web amb *backend*. És evident que no totes les aplicacions web es desenvolupen de la mateixa manera i utilitzant les mateixes tecnologies, però la que s'ha desenvolupat en aquest projecte no deixa de ser-ne un exemple.

Pel *qMatching* doncs s'han tractat varis conceptes. En primer lloc, bases de dades. S'ha hagut de fer un disseny intentant optimitzar l'emmagatzematge de dades, i també s'ha hagut d'entendre un disseny ja fet, on s'havien d'anar a buscar dades per exportar-les.

Pel que fa al codi, s'ha hagut d'estructurar en diferents funcions, arxius i degudament comentat de tal manera que si en un futur apareixen requeriments nous, ha de ser relativament senzill adaptar-los a l'aplicació existent.

Addicionalment, per dur a terme aquest projecte també s'ha après a utilitzar la metodologia adequada de planificació i organització adequades per tal de facilitar el desenvolupament, seguiment i correcció d'errors.

Finalment, hi ha alguns aspectes que són susceptibles a ampliar. La primera és l'organització del codi. En la finalització del projecte el codi està estructurat d'una forma que resulta senzill fer-ne un manteniment, però en cas que apareguessin requeriments que impliquin grans canvis, probablement s'hauria de fer una reestructuració del codi doncs si no acabaria esdevenint impossible de mantenir.

En segon lloc, la part de test del codi. La majoria de test s'ha dut a terme de forma manual, com ja s'ha anat explicant al llarg del document. Per exemple, el test de l'algorisme de *matching* s'ha dut a terme comprovant que el resultat que ofereix concorda amb la informació dels informes que es van fer en el seu moment. Tot i que tal i com està ara l'aplicació, aquest tipus de test ha servit perfectament, s'hauria de valorar la possibilitat de desenvolupar un mòdul a part de test del producte, que s'executés sempre després d'afegir canvis en el codi, per validar que segueix funcionant correctament.

A més a més d'això, hi ha un tercer aspecte susceptible també a millora. La comunicació amb el servidor on hi haurà allotjada l'aplicació *qMatching* es fa a través del protocol HTTP. Les dades que tracta l'aplicació són de caràcter genètic, i per tant de caire sensible (confidencials). És per aquest motiu que en un futur proper s'haurà de valorar la possibilitat d'afegir un certificat digital al servidor, per tal que la comunicació a nivell d'aplicació es faci de forma xifrada (HTTPS).

D'aquest últim aspecte, se'n deriva un altre. Aquest projecte, s'ha enfocat des de bon principi cap a una vessant molt tècnica. De fet, ja està bé que sigui així ja que és un TFG d'enginyeria informàtica. Tot i així, el producte final que s'ha obtingut no deixa de ser un servei que estarà a la venda de cara a possibles clients. Per aquest motiu,

i al tractar-se de dades confidencials, s'hauria de buscar la manera de transmetre seguretat al client, i és per això que en un futur també proper, s'haurà de valorar la possibilitat de vestir aquest projecte amb un apartat legislatiu, donant a entendre bàsicament que ningú té accés a les dades més enllà que el seu propietari.

AGRAÏMENTS

No m'agradaria tancar aquest treball de final de grau sense fer abans un petit reconeixement a les persones que m'han ajudat a tirar endavant i fer-lo possible.

En primer lloc vull mostrar el meu agraïment a Ramon Martí, tutor d'aquest treball de fi de grau. Gràcies a la seva ajuda en forma d'idees, recomanacions i valuosos consells he pogut aprendre molt més i augmentar la qualitat d'aquest treball.

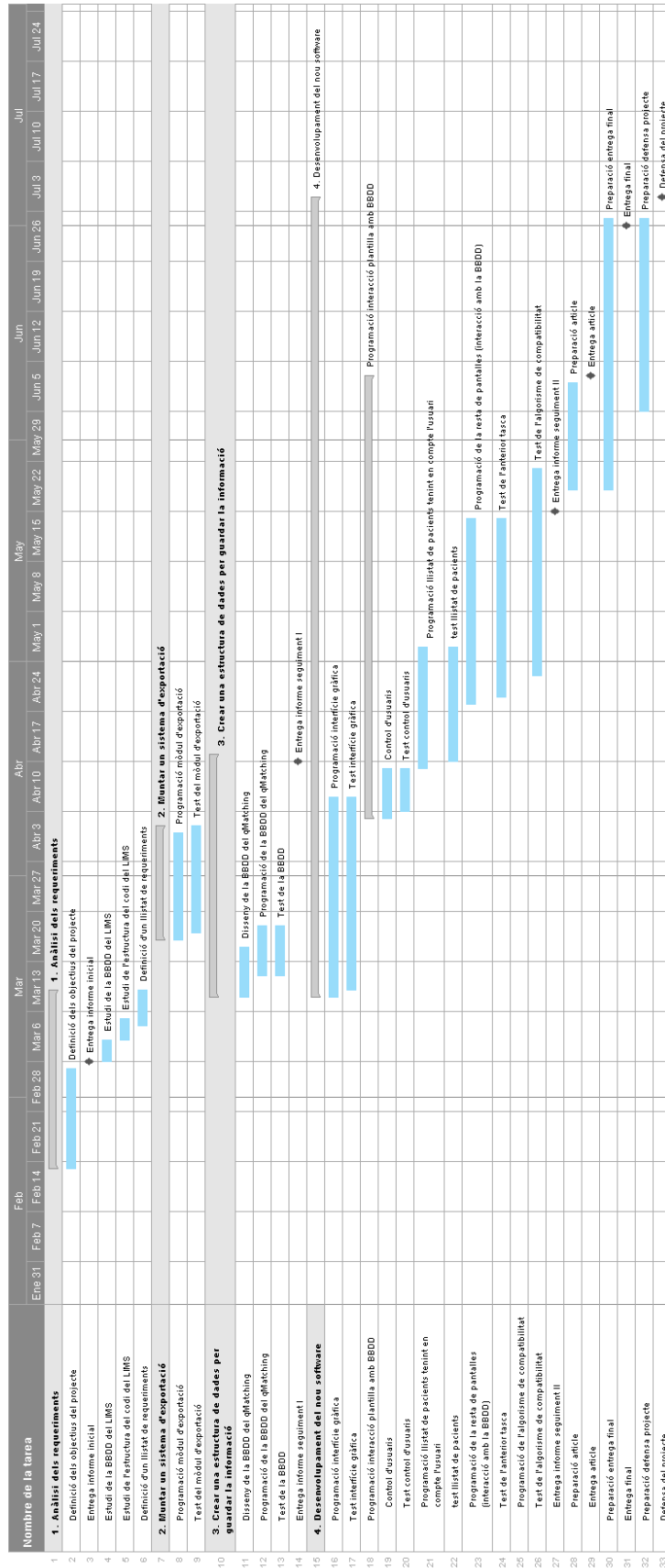
També m'agradaria donar les gràcies a l'empresa *qGenomics*, que és on he desenvolupat aquest projecte i és també on va néixer la idea. Els hi vull agrair sobretot el seu suport i ajuda en les parts de caire més aviat genètic, i que s'escapaven del meu coneixement.

Finalment, però no per això menys important, agrair també el suport d'amics, familiars, i de totes les persones que en major o menor mesura m'han donat suport per a dur a terme aquest projecte.

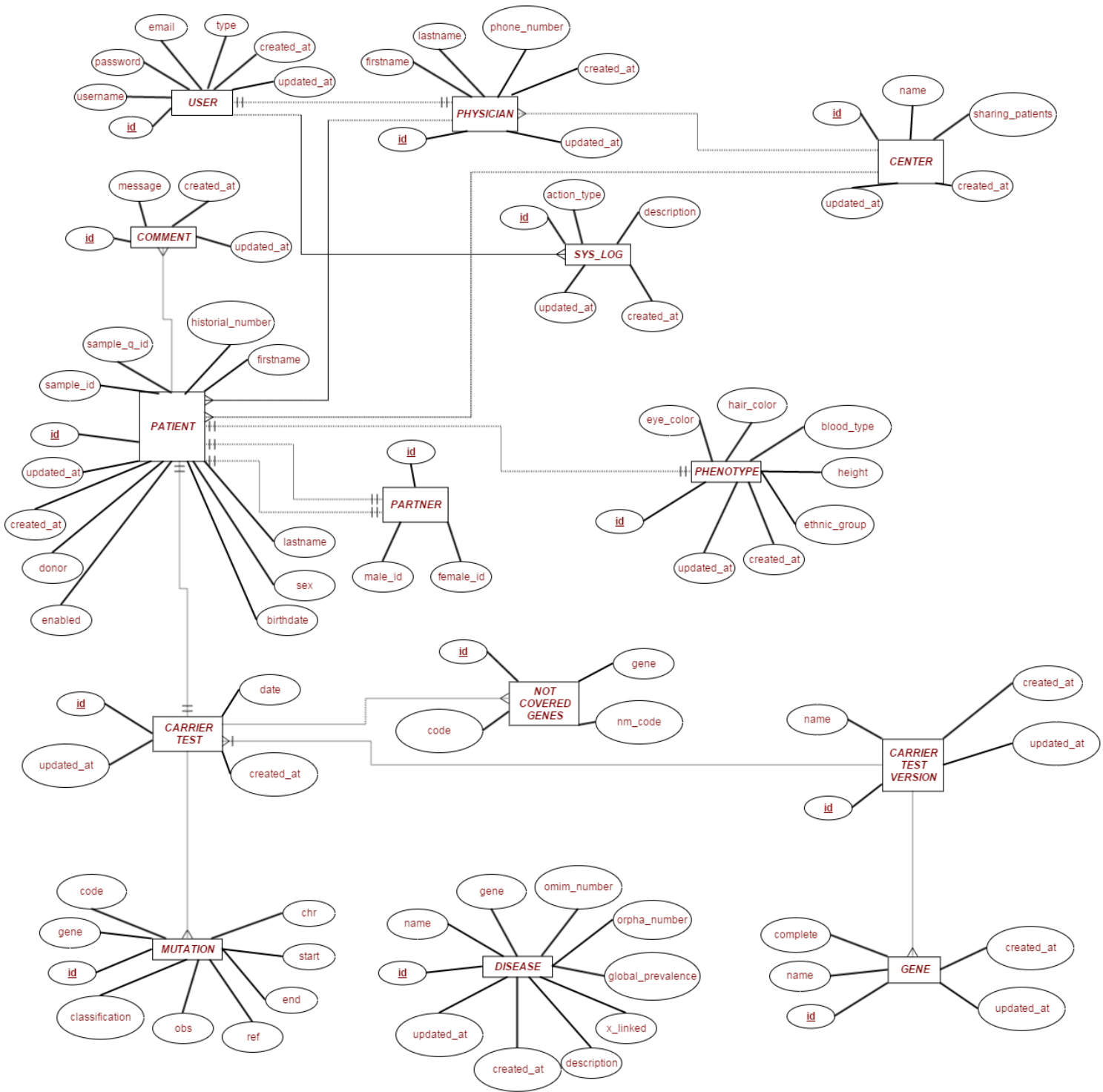
BIBLIOGRAFIA

- [1] qCarrier Test, <http://www.qgenomics.com/ca/node/338> (15/02/2016)
- [2] Martin, James (1990). MacMillan Publishing Co., ed. *Rapid Application Development*
- [3] MySQL Documentation, <https://dev.mysql.com/doc/> (02/03/2016)
- [4] Getting started with Bootstrap library, <http://getbootstrap.com/getting-started/> (02/03/2016)
- [5] Online Diagram and Flowchart Software, <https://www.glify.com/> (10/03/2016)
- [6] Laboratoris Ginemed, <https://www.ginemed.es/reproduccion-asistida/tecnicas-de-laboratorio/test-de-compatibilidad-genetica-matching-genetico> (08/06/2016)
- [7] Laboratoris Igenomix, <http://www.igenomix.com/es/tests/test-genetico-portadores-cgt/> (08/06/2016)
- [8] Laboratoris qGenomics, <http://www.qgenomics.com> (19/06/2016)

A1 APÈNDIX - DIAGRAMA DE GANTT



A2 APÈNDIX – DIAGRAMA ENTITAT RELACIÓ



A3 APÈNDIX – ESQUEMA DE TAULES DE LA BASE DE DADES

