

Eina de consulta dels registres d'Auditoria orientada a sistemes de salut

David Sánchez López

Resum— Avui en dia la gestió de dades i recursos empresarials depèn en gran mesura d'un sistema informàtic el qual pugui manejar grans quantitats d'informació adaptant-se a les necessitats de cada negoci. Entre els diferents sectors demandants d'aquests serveis es troba el de la sanitat, sent fonamental no només la capacitat de gestió d'informació adaptada a la empresa sinó també preservar la confidencialitat i l'ús apropiat de les dades pels usuaris de l'aplicació. Així doncs el propòsit d'aquest projecte es desenvolupar una funcionalitat addicional al producte sanitari del ERP que comercialitza la empresa UNIT4, el que a partir de registres d'auditoria que el sistema ja emmagatzema, aquest pugui normalitzar i processar la informació continguda amb la intenció de filtrar i fer estudis exhaustius segons diferents criteris del auditor per a obtenir qui es el responsable de determinades accions a l'aplicació.

Paraules clau— Auditoria, Sanitat, Protecció de Dades, Seguretat, Informació, Karat, Scrum

Abstract— Nowadays data and business resources management relies mostly on a computer system which should be able to deal with vast quantities of information arranging the business necessities. Among several demanding fields of these services there is the health one, not only being essential the data management adapted to the company but also preserve confidentiality and an appropriate use of the data by application users. Thus, this project purpose is to develop an additional functionality of the ERP's Health product that the company UNIT4 commercializes, using the audit registers that the system already stores, it should be able to normalize and process all the information contained in order to filter and create exhaustive studies according to the several rules established by the auditor which wants to know who is the person in charge of given application actions.

Index Terms— Audit, Health, Data Protection, Security, Data, Karat, Scrum



1 INTRODUCCIÓ

AQUEST projecte ha sigut realitzat en el marc de pràctiques universitàries a l'empresa UNIT4.

UNIT4 és una companyia que opera a nivell global la qual desenvolupa, comercialitza, dona suport i ofereix un extens ventall de software de gestió d'empreses amb el fi de proporcionar amb els seus productes una millora en la organització dotant de dinamisme i una gestió eficient dels negocis interessats.

Aquest tipus de software se'l coneix com Enterprise Resource Planning[1] (ERP) on el seu objectiu tal i com el seu nom indica es la planificació de recursos empresarials en base als sistemes d'informació. El seu us va ser popularitzat a principis dels anys 90 quan es va veure com una ampliació als *material/manufacturing resources planning* considerant-se com una eina independent va començar a representar la gran importància de la evolució de la integració d'aplicacions més enllà de la manufactura.

El principal avantatge d'aquests sistemes d'informació és

que permeten automatitzar moltes de les pràctiques de negoci associades amb aspectes operatius o bé productius de l'empresa. El ERP es distingeix de qualsevol altre software empresarial per dues característiques:

- **Modular:** La seva funcionalitat es troba dividida per mòduls la qual cosa permet al client adquirir només aquells que s'adaptin als seus requisits.
- **Parametritzable:** Permet ser configurat mitjançant nous desenvolupament que es puguin generar a partir de les necessitats canviants del negoci.

L'empresa Unit4 fa servir el seu propi framework anomenat *Karat* sobre el que es basen tots els productes que es van desenvolupant paral·lelament. Aquesta plataforma està basada en Java i està en continu creixement sempre adaptant-se a les necessitats canviants tant dels negocis com de les tendències tecnològiques que van sorgint amb el temps. Karat ofereix un ampli llistat de APIs que faciliten enormement la programació dins de l'entorn en el que es treballa, proporcionant classes i mètodes que s'adaptin a cada tipus de situació.

-
- E-mail de contacte: David.sanchez917@gmail.com
 - Menció realitzada: Enginyeria de Computació
 - Treball tutoritzat per: Xavier Roca (Centre de Visió per Computador)
 - Curs 2014/15

2 OBJECTIUS

Al ser una aplicació que constantment manega informació sensible, aquesta ha de ser accessible per un auditor intern o bé extern que sigui capaç de contrastar tant el que s'està emmagatzemant al sistema com l'ús que es fa d'aquesta informació i la seva seguretat.

Llavors l'objectiu del projecte serà crear dins del producte sanitari que comercialitza UNIT4 una funcionalitat que permeti realitzar estudis a un auditor a partir dels registres que es tenen a la Base de Dades (BBDD).

Aquest eina implica permetre a l'auditor crear estudis d'auditoria independents segons la seva finalitat, podent establir nombroses restriccions sobre els registres a buscar en base a la informació clínica o del sistema, normalitzar la informació que es rep de la BBDD, fer un post-filtratge per refinar encara més la informació que es vol visualitzar, que els estudis es generin de forma asíncrona o permetre exportar en diferents formats la informació obtinguda a raó d'informe.

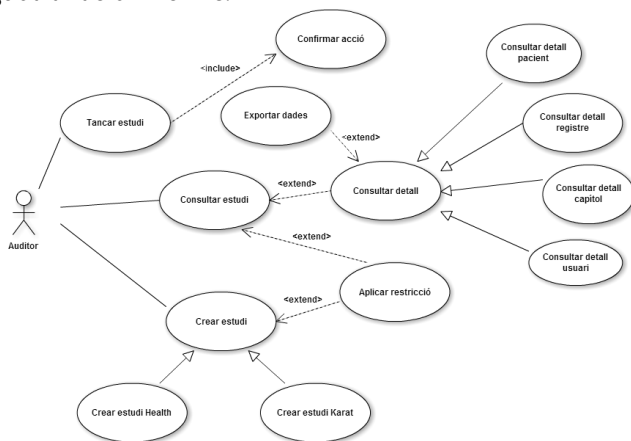


Fig1. Diagrama de casos d'us del programa

3 ESTAT DE L'ART

El software sanitari dona un ampli ventall de possibilitats tant per el sector públic com el privat, és vàlid per a qualsevol grandària d'organització sanitària amb múltiples centres, medicina especialitzada, salut mental o sociosanitària. Aquest software de gestió mèdica per a sanitat està desenvolupat i experimentat en col·laboració amb les diferents organitzacions i professionals del sector donant així una visió que permet realitzar una fàcil adaptació a la majoria de centres assistencials.

Diàriament qualsevol hospital o clínica ha de d'oferir servei a centenars de pacients que hi estan interns o bé que entren per una nova assistència. És essencial doncs l'ús d'aplicacions i bases de dades per gestionar la informació clínica dels pacients i així poder tenir des de un informe en detall, consultar l'historial d'intervencions, la assignació de llits, fins a designar les dietes diàries que rebrà cada pacient hospitalitzat. No obstant això, no només és important la integritat i consistència de les dades sinó que al ser informació sensible, segons està legislat a la LOPD[2] aquesta ha d'estar protegida en funció del

nivell en el que estigui categoritzada. Donat això queda en responsabilitat de l'empresa propietària del fitxer que prengui les mesures necessàries per complir amb tots els requisits legals que poden ser des de protegir l'accés al fitxer de dades amb un cert nivell de seguretat fins a registrar cada una de les accions que involucrin l'ús de dades personals .

La informació que manegen els usuaris del sector de la sanitat pot anar des de un nivell bàsic fins a l'alt en funció del seu contingut, com es pot apreciar a la Taula 1, les dades sanitàries estan ubicades al nivell més alt i per tant han d'estar controlades per un registre d'accessos [3].

TAULA 1
NIVELLS DE SEGURETAT LOPD

Tipus de dades	Mesures de seguretat Obligatòries
Nivell Bàsic	
Nom	Document de seguretat
Cognoms	Règim de funcions i obligacions
Direccions de contacte	del personal
Telèfon	Registre d'incidències
Copies de seguretat	Control d'accés
	Gestió de suports
Nivell Mitjà	
Comissió d'infraccions penals	Mesures de nivell bàsic
	Responsable de seguretat
Comissió d'infraccions administratives	Auditoria bianual
Informació d'hisenda pública	Mesures addicionals d'identificació i autenticació d'usuaris
Informació de serveis financers	Control d'accés físic
Nivell Alt	
Ideologia	Mesures de nivell bàsic i alt
Religió	Seguretat en la distribució de suports
Creences	Registre d'accessos
Origen Racial	Mesures addicionals de copies de seguretat
Salut	
Vida	

Per al àmbit de la sanitat resulta un factor clau que tota creació, modificació i consulta quedi enregistrada automàticament per a cada esdeveniment de l'aplicació, de fet, la plataforma ja compta amb una tasca que s'encarrega d'emmagatzemar en registres tota acció efectuada en el sistema. En el cas de que algú vulgui consultar quins registres s'han guardat degut a certes accions es disposa d'un monitor de caràcter general[4] que permet consultar el detall dels registres d'auditoria, es poden trobar diferents tipus de registres d'auditoria segons el producte o aplicació sobre el que s'hagi esdevingut l'acció que els genera. Entre aquesta diversitat de tipus trobem els que pertanyen al producte sanitari Health, tot i que podem consultar el detall de tots els registres emmagatzemats, l'àmbit sanitari consta d'un grau alt d'especificitat i en conseqüència pot ser difícil treure con-

clusions d’anàlitzar registres individualment nomès consultant els seus detalls, per tant es pretèn crear una capa independent a l’actual monitor d’auditoria que maximitzi el detall visible i l’explotació de les dades on l’objectiu és que a més de facilitar l’accés a aquesta informació i fer cerques per certs paràmetres, això ho pugui fer qualsevol persona sense que tingui necessàriament coneixements tècnics informàtics més enllà del funcionament particular de la plataforma Karat i els productes que tingui instal·lats.

Durant la fase d’anàlisis s’ha estudiat la possible existència també d’una eina semblant per una altre empresa dedicada al desenvolupament de software. Tot i que no s’ha trobat cap aplicació amb la mateixa finalitat si que es troba present a alguns ERPs la opció d’Auditoria de registres[5], en un marc general permet monitoritzar en detall els canvis efectuats sobre taules mitjançant la interfície però cap destinada específicament per a un producte sanitari amb tals pretensions de detall i explotació.

4 METODOLOGIA

Des de mitjans de la passada dècada, al 1990, va començar a evolucionar la metodologia convencional de desenvolupament de software que es feia servir fins al moment, l’ús de mètodes molt estructurats i estrictes, extrets del model de desenvolupament en cascada feien cada vegada més difícil adaptar-se a projectes on els seus requeriments eren constantment canviants i necessitaven replantejar sovint la seva definició.

En conseqüència a aquest problema va començar a sorgir un nou model de treball anomenat Agile[6] el qual es basa principalment en un procés dinàmic que pretén adaptar-se als requeriments, canviants o no, en base a la col·laboració continua amb els stakeholders, les entregues periòdiques, la motivació del equip i emprar software funcional com a principal mesura de progrés.

Des de fa anys UNIT4 utilitza la metodologia àgil SCRUM[7] per a tot el els treballadors integrats en un entorn de desenvolupament, aquest projecte pertany al departament de Health el qual s’encarrega del desenvolupament i manteniment del mòdul sanitari i està format per un equip de 6 persones.

4.1 Rols

A cada equip sempre hi ha d’haver un rol assignat a cada integrant segons el qual portarà a terme unes tasques específiques i tindrà certes responsabilitats.

- **Product Owner:** Es el representant de totes les persones interessades en el projecte, tant de forma interna com externa. La presa de decisions radica en la seva paraula i posseeix un gran coneixement sobre el sector o mercat en el que treballa. És responsable de que s’assoleixin els objectius del projecte, de la planificació que segueix i que els resultats maximitzin el ROI.

- **Scrum Master:** Actua com a líder de l’equip però no en un sentit de directiu si no que vetlla per a que els integrants del equip segueixin les regles i processos del Scrum sense desviar-se del camí. D’aquesta manera pren un responsabilitat organitzativa per gestionar les reunions del Scrum i ensenyar al equip a autogestionar-se al igual que protegir i aïllar als membres de possibles impediments que els faci desviar-se de les tasques i difícilment aconseguir l’objectiu plantejat a cada sprint.
- **Equip:** Els membres han de desenvolupar les històries creades per tal de que siguin entregables a les dates previstes, per tant es funció seva estimar la complexitat que suposa cadascuna, planificar el treball a realitzar durant les reunions i sobretot la col·laboració entre tots els membres si és necessari per a completar una història. Els integrants del equip tindran diferents perfils de treball segons els quals es distribuiran les tasques d’acord amb els seus coneixements, els més coneguts són: Business Analyst, Developer, Quality Consultant i Technical Author.

4.2 Planificació

L’Scrum divideix el projecte en diferents Sprints al final dels quals el treball realitzat ha de ser entregable, comptint doncs amb uns mínims de qualitat acordada.

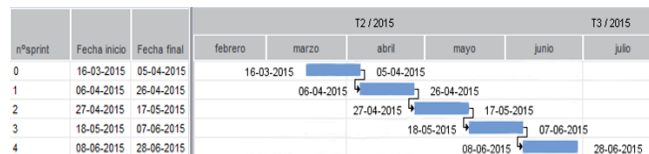


Fig 2. Planificació dels diferents sprints del projecte

Un Sprint representa un període de temps, normalment de 3 setmanes de duració, en el qual es plantegen uns objectius específics a assolir de cara a la finalització d’aquest. El contingut dels sprints es determina prèviament durant una reunió anomenada Sprint Planning on s’estima el seu abast dividint-lo en múltiples tasques i valorant cadascuna en punts, aquests seran una mètrica per determinar el temps previst de cada tasca.

A la [Fig 2] s’aprecia el diagrama de Gantt generat després de definir els sprints que delimiten el projecte, a mesura que es progressa es necessari imputar les hores dedicades per a cada tasca per tal de predir quin es el marge de temps que es disposa, les dilacions produïdes per necessitats sobrevingudes o l’assoliment del sprint. De forma paral·lela a mesura que s’imputen hores i es defineixen punts per a les tasques es genera un *Burn down chart* [Fig 3] que facilita la verificació del progrés i sobretot el treball pendent que hi ha durant el Sprint, aquest gràfic es molt usual al desenvolupament àgil de software especialment a Scrum. El diagrama de Gantt engloba el temps que avarca el projecte encara que no s’ha inclòs el mes previ al primer Sprint ja que aquest va ser dedicat exclusivament a la formació interna en la plataforma Karat per part de la empresa.

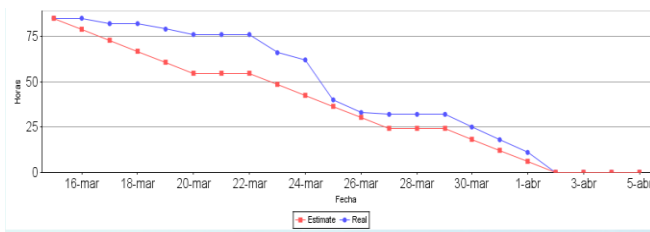


Fig 3. Burndown Chart del Sprint 1

Així doncs es parla de desenvolupament incremental ja que el treball es basa en unitats tancades, amb un sprint actual (sense importar el futur) i on les històries acostumen a tenir les fases de disseny, desenvolupament, documentació, proves i integració. Això es el que després al Sprint Review es confirmarà com un desenvolupament validat, entregable i demostrable ja que estem parlant de parts funcionals.

5 ARQUITECTURA I DESENVOLUPAMENT

Segons la política de treball a l'empresa com a fabricant de software, es disposen de diferents bases de dades depenent del context en el que s'hagi de treballar i també del departament en el que es vagi a fer servir. D'entre les que hi ha a l'abast es destaquen tres:

- **Desenvolupament:** Sobre aquesta BBDD l'equip treballa durant tot el procés de creació de software fins el seu estat com a entregable definitiu.
- **Test:** És complementària a la de desenvolupament però sobre tot es fa servir en etapes finals de un desenvolupament quan es vol posar a prova el software creat i sotmetre-ho a proves de càrrega i validació.
- **Publicació:** Una vegada el software ja està completament finalitzat i s'han fet totes les proves de validació i testeig necessàries es l'hora de exportar totes les novetats a aquesta BBDD per a que al proper Update aquestes noves implementacions les puguin gaudir els clients.

El sistema d'emmagatzematge de Karat es divideix principalment en dues BBDD per al funcionament de cada producte del ERP, per una banda està SDIC (o també denominat superdiccionari) la qual guarda la informació i definició d'objectes propis interns de Karat i d'altra banda està DATA sobre la qual es disposa tota la informació provinent de l'ús que genera qualsevol usuari en interacció amb la aplicació al igual que es fa servir per consultar la informació ja existent i des de on es farà servir també els objectes definits a SDIC. Gràcies a fer servir el driver ODBC[8], Karat permet dotar de heterogeneïtat al haver-hi llibertat per a triar el Sistema Gestor de Base de Dades (SGBD) que s'utilitzarà, aquest driver el que fa es inserir una capa intermediària denominada nivell d'interfície de client SQL entre l'aplicació i SGBD on el propòsit d'aquesta capa es traduir les consultes de dades de l'aplicació en commandes que el SGBD pugui entendre.

Entre els SGBD suportats per Karat tenim : Microsoft SQL Server, InterSystems Caché, Oracle Database i MySQL.

Deixant de banda la BBDD i centrant-nos en el framework, en el cas d'aquest projecte que està basat en el llenguatge Java tots els objectes es defineixen dins del framework Karat que està integrat com un plugin a Eclipse, d'entre els objectes propis a l'entorn Karat es disposa de l'anomenat "Objecte de Negoci", el qual actua com element clau dins de l'entorn de desenvolupament Karat, aquest actua com a una entitat única que gestiona diferents nivells d'informació contenint la lògica de negoci i les regles de validació. Sobre aquest nucli es basaran la resta de objectes amb el que se treballa, com per exemple formularis i llistats. Un objecte de negoci es compon d'un o més panells i aquests panells de ítems, els panells també tenen un paper important ja que alberguen camps d'informació que és directament obtinguda de consultes base o consultes a la BBDD generades a Karat. Cada panell pot tenir associada una consulta base, de la qual els ítems es poden nodrir i omplir els seus camps d'informació en funció de les seves necessitats.

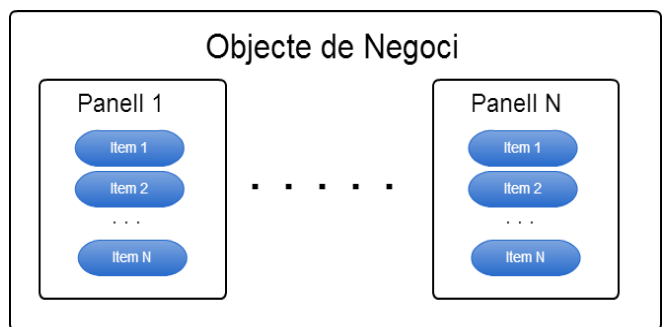


Fig 4. Composició d'un objecte de negoci

Així mateix, com s'acaba de mencionar, les consultes base o consultes de karat son un altre tipus d'objecte propi que actua com a una vista a la base de dades, es configuren al plugin d'Eclipse indicant quins camps seran els consultats i sobre quines taules i condicions es fa, l'avantatge d'això és que no es necessari escriure codi per a generar aquestes consultes ja que es facilita una mena de configurador de consultes amb tota la informació necessària a manegar. D'aquesta manera només cal assignar una consulta base al panell d'un objecte de negoci per tal de que cert ítem pugui obtenir el valor de un camp a la base de dades.

Un *formulari* es podria denominar com a una finestra específica d'interacció, els formularis es creen a través de la dissenyadora de formularis [Fig 5] i han d'estar basats obligatòriament en un objecte de negoci, la dissenyadora aporta les eines necessàries per a editar el formulari amb total llibertat, cal a dir doncs que encara que aquesta part es a nivell purament visual, des de aquesta eina es personalitza la interfície visual creant camps associats a ítems d'un panell i en conseqüència a un objecte de negoci.

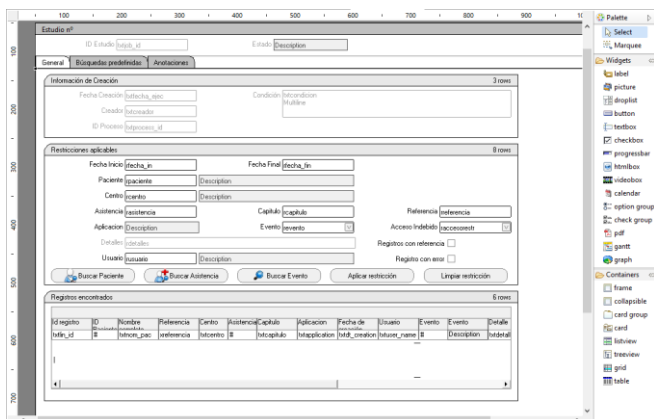


Fig 5. Diseñadora de formularios

Hi ha dos vies per a atorgar la operativitat de l'aplicació entre formulari i funcions, que no tenen per què ser incompatibles, per una banda està tot el framework de Karat integrat com a plugin a Eclipse com s'ha comentat anteriorment, d'altra banda la necessitat de personalització segons un esdeveniment i els objectes afectats insta a la creació d'Scripts fent servir l'ampli repertori de classes i mètodes proporcionades també per les APIs de Karat que ofereixen tractament i solucions precises a qualsevol pal liatiu.

Per a aquest segon cas de programació es fan servir les APIs de les que es nodreix Karat i que aporten un interessant plantejament per afrontar cada situació.

Existeixen diversos packages pensats per a cada entitat funcional de Karat, entre els més importants trobem un package per al Objecte de Negoci i altre per al Formulari, l'atractiu d'això és que cadascun posseeix entre d'altres una classe que s'encarrega de manejar els esdeveniments que afecten a temps real a la entitat en qüestió i sobre la que es pot sobreescriure els seus mètodes en funció de les necessitats que es tinguin per un cert moment i entitat. La possibilitat de treballar amb programació dirigida també per esdeveniments permet controlar qualsevol tipus de situacions que un entorn d'aquest tipus pugui generar.

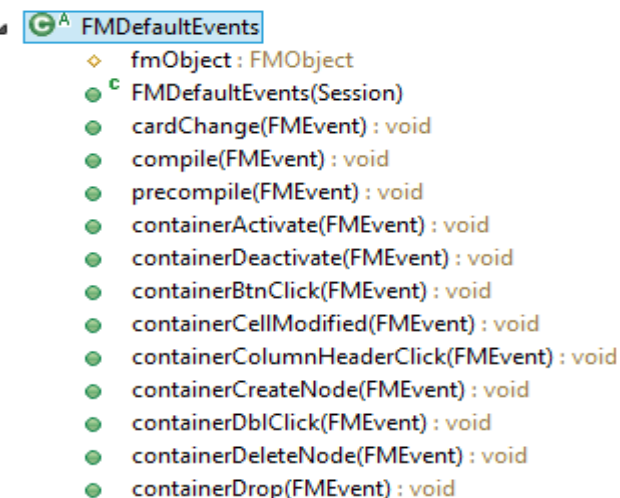


Fig 6. Alguns mètodes de la classe FMDefaultEvents (Formulari)

6 RESULTATS

Com a resultat dels 5 sprints planificats al Scrum s'obté una eina totalment funcional basada en el producte sanitari del ERP. A continuació s'exposen cadascun dels sprints amb el seus objectius, el seu procediment i canvis de requeriments si han sigut necessaris.

6.1 Sprint 1: Anàlisi de requisits i prova de concepte

Prèviament a començar amb el desenvolupament és necessari en aquest sprint inicial fer un anàlisi de requisits que presenta el projecte i una vegada s'ha pogut definir aproximadament l'abast i dificultat d'aquest es pot iniciar el desenvolupament amb una prova de concepte. La prova de concepte ajuda alhora sent com una doble validació que atorgarà si s'escau requisits addicionals al projecte i serveix com una base inicial sobre la qual s'inicia el desenvolupament que posteriorment serà escalable.

La prova de concepte que comprèn aquest sprint consistirà en una versió bàsica que sigui capaç d'extreure la informació important de registres a la BBDD d'origen i mostrar-la per pantalla en funció de determinades condicions que l'usuari estableixi a la interfície.

Els registres d'auditoria es generen automàticament per cada esdeveniment i s'emmagatzemen a SDIC, però aquesta BBDD és exclusivament per definició d'objectes karat pel que s'ha de treballar a DATA i resulta necessari interactuar de forma paral·lela SDIC-DATA per a poder nodrir-se de la taula de BBDD origen i posteriorment operar sobre la que li correspon, DATA.

La informació continguda a la taula d'origen no té cap organització concreta i hi ha camps de dades no rellevants o bé que aglomeren informació la qual resulta impossible de processar sense prèviament tractar-la.

Per aquest motiu una vegada es generi la petició de creació d'un estudi s'ha de consultar la base de dades origen i mitjançant un analitzador sintàctic o parser, normalitzar i filtrar només la informació necessària [Fig7].

El procés de normalització que realitza el parser es basa en filtrar les dades dels camps descartant la que no sigui útil o bé sigui redundant, és molt important la desnormalització de aquella informació que està aglomerada a un mateix camp ja que serà essencial per després utilitzar-la com a paraula clau de cerca en futurs estudis d'auditoria.

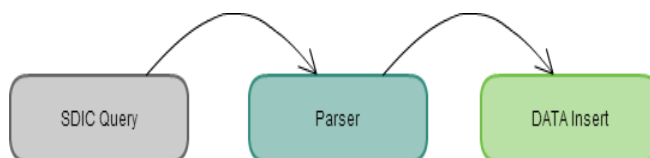


Fig 7. Seqüència per obtenir registres de SDIC

D'aquest escenari esdevé a la necessitat de creació de dues taules [Fig 8] correlacionades dins de DATA, l'ús d'una és totalment dependent de l'altra des del moment que han rebut les dades de SDIC i on la clau que uneix aquestes dues taules es l'identificador per a cada estudi d'auditoria que es realitzi, la necessitat d'una segona taula garanteix la integritat i independència de les dades

d'un estudi en front a la resta.

La primera taula *vh_audit_det* preserva cada un dels registres normalitzats obtinguts de SDIC posseint dues claus, l'identificador de registre dins d'un estudi d'auditoria i l'identificador de estudi al que correspon aquest. Com s'ha explicat, aquesta clau primària es comunica amb la segona taula *vh_audit_cab* que conté la informació de capçalera essencial per identificar cada estudi com és l'ID de l'estudi, usuari creador, data de creació, les condicions establertes per realitzar l'estudi i l'estat actual...

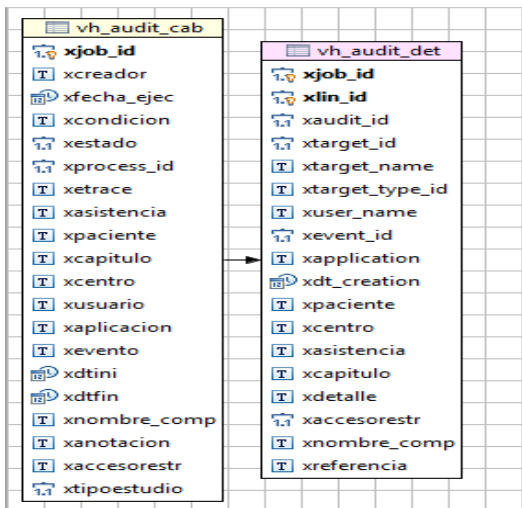


Fig 8. Relació de les dues taules a DATA

Per motius tècnics es van haver d'afegir els camps que treballen com a restricció dins de la taula que fa de capçalera per què es pretén que el sistema sigui multi llenguatge fent servir mòduls de missatges que es visualitzen en funció del idioma per defecte que tingui instal·lat l'usuari, però les dades han de ser emmagatzemades a la Base de Dades per separat si es vol després visualitzar totes les condicions juntes mostrant nom del camp traduït i el seu corresponent valor.

Una vegada s'ha codificat el parser i s'han integrat les taules a la BBDD cal vincular les dades obtingudes amb la definició visual que crearem a través de la dissenyadora de formularis. A l'hora de crear un nou estudi d'auditoria és essencial que es pugui parametritzar [Fig 9] aquest amb diferents paraules claus que siguin suficientment exhaustives per a buscar en qualsevol àmbit.

Fig 9. Paràmetres possibles a la creació d'un estudi

6.2 Sprint 2: Execució asíncrona i millora d'usabilitat

El sprint 2 comprèn una sèrie d'adaptacions del projecte per tal de facilitar l'experiència del usuari, principalment es basa en proporcionar la independència de generació entre estudis d'auditoria i el mateix programa, d'altra banda també inclou la navegació interactiva entre dades clíniques per a facilitar al auditor a fer una cerca exhaustiva durant la definició d'un estudi.

La creació i existència de registres per a cada acció amb informació sensible és habitual donat el cas que els usuaris que utilitzen l'aplicació ho faran en el context de la sanitat, per això s'ha de considerar la gran quantitat de assistències que pot generar un centre sanitari com un hospital o clínica, segons el cas pot arribar a generar milers de registres diaris al sistema sobre qualsevol acció d'un usuari amb l'aplicació.

En aquest context la petició de generar un estudi d'auditoria, encara que sigui entre un rang de dies curt pot significar l'espera de l'usuari fins que acabi la tasca de consulta i processament. Per dotar de fluïdesa i completa operativitat de l'aplicació, aquest procés de generar un nou estudi d'auditoria es llança de manera asíncrona permetent així navegar per el sistema i consultar altres estudis disponibles si és necessari.

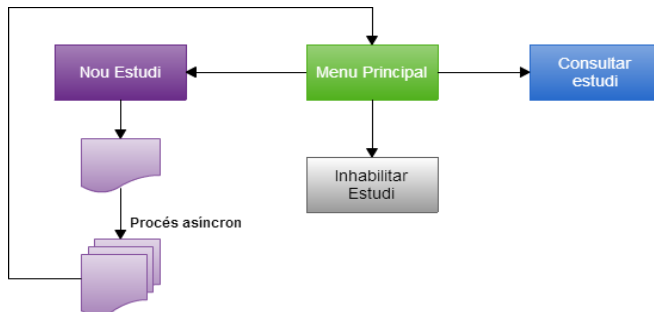


Fig 10. Workflow d'un estudi d'auditoria

Independentment de la creació d'un estudi que ja s'ha comentat existeix la possibilitat de consultar un d'actiu o bé de inhabilitar un quan la seva finalitat d'ús ja hagi sigut satisfeta. La consulta d'un estudi actiu mostra en totalitat, mitjançant una taula, tots els registres trobats sota les condicions que es van preestablir en la seva generació. Aquesta taula té un format interactiu que permet la ordenació de registres i translació de camps existents. En aquest formulari resulta important la presència també de un post filtratge sobre els registres continguts que es volen visualitzar en aquell moment donant la possibilitat de fer més exhaustiu l'estudi i també reduir el volum de dades amb el que treballar. D'altra banda una peça fonamental per a facilitar a l'auditor l'ús de l'aplicació és la llibertat de navegació entre l'aplicació i les dades del sistema. Certs camps com per exemple la assistència, centre o pacient estan identificats per una certa numeració o nomenclatura, la qual cosa un auditor extern no ha de per què conèixer-la.

Els tres primers botons que apareixen a l'extrem inferior [Fig11] permeten al auditor navegar i triar el nom concret a través d'una sèrie de formularis de cerca avançada de pacient, assistència i esdeveniment [Annex 1,2,3].

Fig 11. Pestanya per a filtrar segon restriccions

En una estada hospitalària resulta crucial per al pacient la correcta estructuració de les dades, per això és destacable la existència de dues dependències de dades entre camps rellevants, son els següents :

- Pacient, centre, assistència, capítol
- Aplicació, esdeveniment, detall

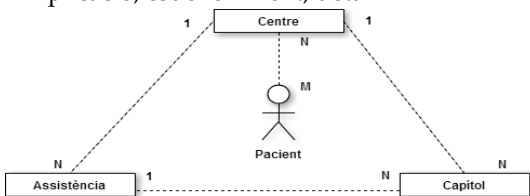


Fig 12. Relació Pacient – Centre – Assistència - Capítol

Les assistències es creen per a cada ingrés d'un nou pacient donat un centre determinat, el capítol s'originarà quan a partir d'aquesta assistència s'ha de derivar a una altre intervenció hospitalària tenint una assistència com a mínim un capítol associat. Resulta necessari doncs per exemple pacient i centre per a obtenir o buscar una assistència, per tant és determinant que el programa disposi de navegació lliure per formularis que proporcionen vies d'accés a totes aquestes dades [Annex 4].

De la mateixa manera, el camp aplicació especifica sobre quin formulari està vinculat un registre d'auditoria, aquest tindrà una relació directa amb un esdeveniment i proporcionarà un detall d'esdeveniment. La navegabilitat no només es possible durant el procés de restricció o filtratge si no que en els registres resultants hi haurà un enllaç sobre els camps clau que puguin aportar informació més detallada d'aquest, un exemple pot ser la fitxa de pacient, les assistències amb el seu seguiment, dades personals de l'usuari...

La suma de totes aquestes funcionalitats dona total llibertat a l'auditor de consultar les dades necessàries al sistema com per a valorar i treure conclusions de les accions que s'han realitzat fent us de l'aplicació.

6.3 Sprint 3: Exportació de dades i detecció d'accions incongruents

Després de realitzar l'anterior sprint review, es van establir una sèrie de canvis de requisits que van prioritzar la realització d'aquests abans de la continuació en desenvolupament de noves funcionalitats.

Per una banda, a mesura que han anat progressant els anteriors sprints i es veia el potencial del programa, s'ha començat a valorar informació addicional com a camps clau i per tant la llista de possibles restriccions ha anat creixent i adaptant-se en funció de les necessitats.

Té també un paper important que la aplicació realitzi una validació de les dades introduïdes pel usuari, s'efectuen des de consultes bàsiques que comproven la existència d'un valor a la base de dades fins a consultes més complexes que a més de comprovar l'existència d'uns camps determinats verifica si és coherent en funció de la seva dependència amb un altre.

Com a eina de comunicació entre el usuari i la plataforma, Karat proporciona l'ús de mòduls de missatges que son totalment personalitzables podent crear missatges d'informació i alerta predeterminats o bé basats en algun paràmetre i que són realment útils per al procés de validació de dades.

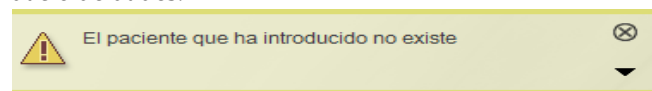


Fig 13. Missatge d'avís al validar les dades

Adicionalment de que l'aplicació permeti mostrar els registres i també fer un post-filtratge més exhaustiu d'aquests, s'ha creat una pestanya al mateix formulari que es divideix en tres àrees d'informació que agrupen dades diferents en funció del resultat de l'estudi. S'ha definit que automàticament quan l'estudi d'auditoria és generat, a partir de consultes als registres propis del estudi proporcionari les 3 àrees següents d'informació útil:

- Usuaris els quals estan relacionats amb els registres resultants.
- Relació d'un registre amb usuari – pacient – capítol
- Relació d'un registre amb usuari – esdeveniment i capítol

A més de la interacció per els formularis ja mostrats, s'ha incorporat també la opció de exportar els registres obtinguts en un estudi a través d'algun format d'arxiu com Excel o PDF, aquesta possibilitat es una altra opció d'estudi per a qui l'interessa manegar o analitzar les dades d'una forma alterna a la aplicació.

Segons està dissenyat el ERP, un professional que faci servir l'aplicació ha d'estar registrat amb un usuari el qual ha de contenir la seva informació personal i per tant cada usuari tindrà permisos d'accés a certs centres o determinades intervencions i assistències en funció de la seva especialitat. Amb la premissa de que hi pugui existir una fallada de seguretat desconeguda en el sistema i algun professional l'exploti accedint a dades que no li corresponen, aquesta acció també quedarà emmagatzemada en els registres i es podrà rastrejar fàcilment si per a una acció el seu accés no hauria de ser possible. Donant per suposat doncs que qualsevol acció que compleixi amb aquest requisit el registre serà destacat respecte els altres per tal de notificar al auditor una possible infracció.

Així doncs, per als registres que compleixin la anterior premissa, aquests apareixeran marcats per un color diferent per tal de facilitar al auditor la localització d'una possible infracció en el sistema.

6.4 Sprint 4: Propagació de restriccions i altres millores

Després de l'assoliment dels altres sprints i mentre es comprovava la correcta funcionalitat del programa va sorgir la petició de dos nous requisits que eren directament relacionats amb el rendiment del programa i la facilitat del usuari per a navegar:

- Les restriccions de post-filtratge han de ser propagades també a les taules de la segona pestanya.

Al sprint anterior es van elaborar 3 àrees que mostrarien informació útil a consultar referent als registres obtinguts dins d'un estudi i amb la finalitat de sintetitzar al màxim la informació per tal de que pugui ser analitzat sense confusions. Fins al moment la informació mostrada en aquestes àrees era referent a tots els registres obtinguts de l'estudi sense donar opció a una tria o forma de cerca més practica que no sigui navegar per tota la taula.

Per aquest motiu s'ha decidit que les restriccions que s'apliquin al estudi d'auditoria sobre la taula general (primera pestanya) també afectin a la informació continguda en les 3 noves taules i així poder consultar relacions de camps específics sense renunciar a les restriccions sobre altres.

- L'obertura de estudis es veu penalitzada notablement per consultes extenses.

Encara que és un factor que es tenia en compte que passaria considerant el gran volum de dades amb el que treballen els clients actualment, no obstant s'ha procurat de buscar solucions que no alterin el lliure us de l'aplicació. Les taules compten amb un sistema de càrrega progressiva per a la visualització de registres, la qual anirà mostrant de 60 en 60 a mesura que el usuari es desplaça per els seus límits evitant la càrrega sencera de tots els registres des de un inici. Altrament es va identificar que alguns camps existents a les taules de la base de dades finalment no eren utilitzats, al igual que a les consultes SQL es projectaven camps que per canvis de requisits no s'utilitzaven finalment.

A més de camps sense utilitzar també hi ha una altre penalització de temps referent als accessos a BBDD, la plataforma Karat proporciona una opció molt útil quan es vol informar un ítem al objecte de negoci que depèn directament d'un altre valor, el qual s'anomena Multi-Dependent Query Object (MDQO), de manera gràfica i molt simple es poden establir relacions per a que en funció de les dades vagi a buscar el valor d'un camp a certa taula de la base de dades. L'inconvenient d'aquest mètodes es que necessita de dos accessos a BBDD per a obtenir el valor final, un per a obtenir el valor "clau" amb el que es treballa i altre per fer servir aquest valor com a restricció en la cerca del camp finalment desitjat.

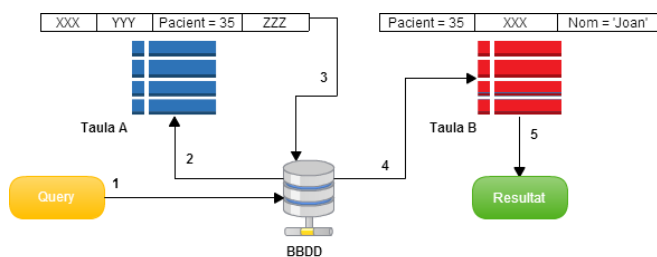


Fig 12. Exemple de una MDQO amb nom de pacient

En casos en que la quantitat d'informació no sigui gaire abundant és una opció realment útil per a estalviar temps i línies de codi, encara que quan el volum de dades és de grans dimensions el fet de que necessiti 2 consultes per a obtenir el valor de cada camp que ho tingui assignat pot ser un desavantatge i acabar penalitzant en temps el seu funcionament. Aquesta penalització es va poder apreciar lleugerament en el funcionament del programa.

Donat un estudi d'auditoria que conté 2623 registres diferents el seu temps de càrrega de la finestra principal era de aproximadament 2020ms i per carregar la segona pestanya la qual conté 3 tipus diferents de taules (comentades al Sprint 3) el seu temps rondava els 2130ms. Considerant que la quantitat de registres és elevada, per a cada registre consultat son necessaris 2 accessos a base de dades, considerant que no només a la finestra principal si no que a la segona pestanya es tornen a fer servir per a diferents casos.

Una alternativa viable per a prescindir d'aquest tipus de consultes és fer un *left outer join* entre les taules indicades, així tindrem ambdós valors a la mateixa consulta des de un inici i podrem prescindir de una segona que només ens penalitzaria innecessàriament en temps amb MDQO.

TAULA 2
TEMPS DE RESPOSTA APERTURA

	MDQO	Left outer join
Obertura estudi	2057ms	1425ms
Transició 2a pestanya	2131ms	522ms

Aquesta suma de factors penalitzaven en part els temps de càrrega, donada l'obertura d'un estudi d'auditoria amb un total de 2623 registres els temps d'espera han millorat des de un 30% fins a un 75% depenent del tipus de informació que s'estigui sol·licitant i la pestanya sobre la que estiguem visualitzant.

Més enllà de les millores que afecten al rendiment del programa, durant aquest sprint també es va proposar la investigació en una nova línia de treball que podria ser de profit per al projecte. Fins el moment s'havia treballat amb els registres generats per les accions que es porten a terme dins del mòdul Health a Karat, la generació d'aquests registres és directament cridada per codi dins de les classes que s'encarreguen de proveir funcionalitat

al mòdul, per tant és semi-automàtic i ha d'estar prèviament contemplat el cas per a que es generi.

De manera paral·lela, la mateixa plataforma disposa d'un sistema addicional que genera també registres d'auditoria independentment del mòdul per a cada acció en la que un objecte de negoci sigui utilitzat, la qual cosa pot ser interessant si de manera complementària es pot obtenir més informació referent a un esdeveniment. Donat que es una via d'investigació massa extensa, és difícil poder abastar-la sencera ja que a diferència dels registres processats anteriorment, aquests últims contenen la informació fent servir patrons diferents depenent de cada cas, la tasca s'ha centrat en només un objecte de negoci que era bastant comú a ser utilitzat per al mòdul de Health.

Durant aquesta tasca es va apreciar que aquests registres son generats de forma automàtica pel sistema donant com a informació rellevant els camps clau que representen cada objecte de negoci la qual cosa de vegades no arribava a aportar suficientment informació com per a mostrar-la igual que el registres de Health.

Encara que directament no proporcioni el valor de tots els camps que es volen mostrar per a un registre, si que es proporciona els camps suficients com per a buscar a la Base de Dades segons la seva dependència i connectivitat entre taules. Per exemple un cas comú son els registres en els que es proporciona centre, assistència i capítol, tenint aquests valors es pot obtenir altres com el pacient al qual està associat o el tipus d'esdeveniment.

6.5 Sprint 5: Testeig i documentació

Una vegada la fase de desenvolupament s'ha donat per finalitzada i la aplicació compleix amb tots els requisits que es van plantejar en la fase d'anàlisi, és essencial que el programa resultant passi per una sèrie de tests i validacions que assegurin el correcte funcionament exempt de errors i casos inesperats.

Fins al moment tot el desenvolupament i proves s'han realitzat fent servir una base de dades habilitada únicament per al equip de desenvolupament, la qual manega dades fictícies per a poder treballar. Durant la fase de test, és necessari aplicar el programa a un cas real i per això es va preparar un entorn simulat amb dades reals per a que es puguin fer totes les proves necessàries de la fase, per a això es va fer servir una rèplica de la base de dades d'un hospital que és client de Unit4.

Amb la finalitat de seguir alguna pauta organitzativa, primerament es redacta tot un llistat de casos de test als que es podria sotmetre el programa, sempre buscant de trobar la seva fallada ja sigui provant amb valors que possiblement surtin d'un rang previst o bé sotmetre un procés a una prova de estrès amb gran volum de dades.

Aquesta fase de test va ser realment útil per a comprovar el comportament del programa sota un cas real i on la quantitat d'informació es molt més gran. Encara que el funcionament general era el esperat, tal i com estava previst també en aquesta fase es van detectar alguns error en

el funcionament que es van originar per no considerar certes situacions per les que podria passar el programa.

- **Camps no informats** : Tal i com es menciona al sprint 1 els registres d'origen a SDIC emmagatzemen la informació aglomerada en un String concret i separada per determinats tokens. Es va cometre l'error de que per al camp pacient i centre es va obviar que sempre contindrrien algun valor, la qual cosa no es necessàriament certa depenent del cas i impedia el correcte funcionament del parser.
- **Usuari no existent** : Encara que la millor opció per a impedir a un usuari l'accés és deshabilitar-li els seus permisos, també es possible de forma menys neta eliminar-lo de la base de dades la qual cosa pot implicar la pèrdua d'informació per segons quines situacions. Al igual que el cas anterior, no es va considerar de que un usuari hagués sigut eliminat i quan s'intentava accedir a un cert usuari a través del identificador d'usuari que queda guardat als registres d'auditoria, la acció era impossible per que aquest no existia a la base de dades.

L'aparició de errors no controlats pot penalitzar, fins i tot inhabilitar un estudi si es troba amb algun error del que no esta previst, això era un inconvenient ja que durant els errors que es van trobar al testeig es va poder apreciar de que no era eficient que per un registre que no es pogués processar bloquegés la execució i lectura de tota la resta com a conseqüència. Això va implicar no fer un parser tan robust i contemplar la possibilitat de que un registre pot estar corrupte o bé que alguna circumstancia o situació no considerada impedeixi el seu processament, fent servir la captura d'excepcions s'assegura de que es detecti quin registre no es pot processar per almenys gravar-lo amb un estat de error i poder continuar amb l'execució.

Altrament una prova de estrès va ser un gran exemple per a poder quantificar la càrrega a la que es pot sotmetre l'aplicació i el servidor depenent de la petició que es faci. Malgrat que el creador d'estudis està pensat per a que s'utilitzi establint certes restriccions sobre aquest i així estalviar temps de processament i registres a visualitzar per l'estudi, es va suposar al testeig el pitjor cas possible processant tota la base de dades per a un any sencer d'activitat. Aquest estudi va suposar processar un total de 1.352.505 registres de Health que va trigar 2h 20m 40s en completar-se, una quantitat de temps que es comprensible, per tal motiu es va decidir fer que la generació de estudis s'executés de manera asíncrona per no bloquejar a l'usuari de fer altres accions simultàniament.

Com a fase final d'un projecte software, s'ha hagut de documentar internament en detall l'aplicació.

Per una part ha d'haver una documentació tècnica per a que si es decideix dotar de continuïtat al projecte amb nous desenvolupaments, els futurs treballadors que reprenquin el projecte puguin assabentar-se de l'arquitectura i les incidències tècniques que s'han hagut

de pal·liar durant el procés al igual que les mancances de les que pateix en l'actualitat i que haurien de ser millorades. Per altra banda també és essencial elaborar la documentació d'usuari, aquesta ha d'explicar en detall l'estructura del programa, la seva composició en funcionalitats i finestres d'interacció juntament amb les instruccions i consells de com s'ha d'utilitzar per a que l'usuari final tingui la millor experiència possible.

7 CONCLUSIÓ

S'ha aconseguit crear una eina intuïtiva i navegable entre els mòduls ja existents del ERP atorgant la completa facilitat per a un auditor accedir a dades essencials en un estudi exhaustiu. La divisió de les dades en pestanyes, la opció a establir condicions i el us de objectes propis de Karat com taules fan més dinàmica la consulta de dades a temps real. Actualment aquest projecte ja es pot incorporar a publicació en les properes setmanes per a que els clients tinguin a l'abast aquesta eina per a la seva lliure explotació, la qual cosa era una funcionalitat bastant demandada entre els clients del sector.

El projecte s'ha pogut realitzar i acabar dins del temps previst, com a primera vegada treballant amb un equip fent servir una metodologia del tipus àgil trobo enriquidor per al treballador i de la mateixa forma per a l'empresa emprar aquesta mena de bones pràctiques per al desenvolupament software que en molt poc temps genera bons resultats de treball. La realització del projecte m'ha proporcionat una gran dosis d'experiència i flexibilitat desenvolupant sobre un ERP tenint en compte els requisits especials que té el sector de la sanitat.

8 VIES DE CONTINUACIÓ

Em trobo satisfet amb el nivell de completesa que te el projecte una vegada finalitzat, encara que després de l'experiència realitzant aquest projecte orientat al sector de la sanitat trobo que degut a la seva complexitat se li podrien seguir afegint encara més funcionalitats de les que té actualment ja que mai es suficient per a un software que pot estar en continu creixement. Com a futures continuacions podria esser una bona opció continuar amb el desenvolupament de estudis de registres aliens al mòdul de Health, generar un log a disc per a tenir constància de si hi ha agut algun problema durant la generació i en quin moment o bé implantar sistemes experts per a poder guiar un auditor de quins usuaris estan fent possiblement un us incorrecte en funció de la informació manegada i el seu contrast amb altres registres, en conseqüència un us il·legal de les dades.

AGRAÏMENTS

Vull donar especials agraïments al equip que pertany al departament de Health, el qual ha sigut dia a dia el meu suport per afrontar qüestions tècniques i per saber encaixar la trajectòria del projecte. A Fernando Leeson per ser el meu tutor dins de l'empresa, el qual a més de fer el paper de Product Owner ha sabut fer que la meva estància a la empresa sigui lo més còmode i satisfactòria per a poder desenvolupar el meu projecte.

A Xavier Roca per fer un seguiment continu juntament amb la dedicació, correcció i assessorament durant la realització del projecte per a que la meva documentació sigui l'adequada.

A Jordi Pons per fer de tutor extern a les pràctiques i preocupar-se per la meva estància i la seva evolució.

BIBLIOGRAFIA

- [1] Resgroup, "The history of Enterprise Resource Planning", <http://www.resgroup.com/accounting-software-history-enterprise-resource-planning-glance>
- [2] Reial Decret 1720/2007, del 21 de desembre, pel qual s'aprova el Reglament de desenvolupament de la Llei Orgànica 15/1999, del 13 de desembre, de protecció de dades de caràcter personal.
- [3] "Mesures de Seguretat a complir segons RD 1720", Unit4 Business Software - 2012
- [4] "Arquitectura de seguretat i Auditoria de la plataforma Karat", Unit4 Business Software - 2013
- [5] "Openbravo Audit Trail", Openbravo - 2011
<https://www.youtube.com/watch?v=cEsz9aKgUAI>
- [6] Agilesherpa, "A brief history of Agile", <http://www.agilesherpa.org/intro-to-agile/a-brief-history-of-agile/>
- [7] "Metodologia SCRUM", Unit4 Business Software - 2012
- [8] "ODBC - Open Data Base Connectivity Overview", Microsoft <https://support.microsoft.com/en-us/kb/110093>

APÈNDIX

A1. BUSCADOR DE PACIENT

Selección pacientes (dasanche)

Búsqueda automática
 Tarjeta sanitaria |

Búsqueda manual

1º apellido | CIP | Episodio |
 2º apellido | NHC |
 Nombre Jose | Fecha nacimiento | 29 |
 Sexo | Documento identidad |

1º apellido	2º apellido	Nombre	Fecha nacimiento	Sexo	Documento identidad	Tarjeta sanitaria	NHC	Número expediente
ADAME	LIZA	JOSEFINA	15/02/1964	Hombre				129
BALL		JOSEPH	10/01/1965	Mujer				1426
COX		JOSEPH	15/10/1941	Hombre		QUBO04110150...		1171
GRIFFITHS		JOSEPH		Hombre				1515
LAWRENCE		JOSEPH	23/09/1928	Hombre		RIVE0280923015		1209
Martinez	Albala	Jose	10/02/1968	Hombre	50000556M			222
NICHOLSON		JOSEPH	19/01/1972	Hombre		BOUL07201190...		1317
SHARP		JOSEPH	10/12/1982	Hombre				1467
WRIGHT		JOSEPH	01/08/1975	Hombre				1519

+ Nuevo paciente Seleccionar Ficha paciente Salir

El buscador de pacient permet fer una cerca segons les dades personals d'aquest com per exemple nom, cognoms, targeta sanitària, nº de document o data de naixement.

A2. BUSCADOR D'ASSISTÈNCIA SEGONS PACIENT I CENTRE

Buscador de asistencias

Edición Acciones Ayuda

Paciente 1426 | BALL, JOSEPH |
 Centro | |

Mostrar asistencias anuladas
 Mostrar asistencias finalizadas

Asistencias									
Sel.	* Asistencia	* Centro	Nombre centro	* Tipo asistencia	Prestaci...		Fecha cita	Fecha ingreso	Fec
<input type="checkbox"/>	2009A0727	GSS01	Hospital GSS01	Ambulantes	5	Ingreso ambulatorio			
<input type="checkbox"/>	2012C01198	GSS01	Hospital GSS01	Citaciones	101	Primera visita especia...	30/01/2012 9:00		
<input type="checkbox"/>	2013C07620	GSS01	Hospital GSS01	Citaciones	310	Ecografia simple	22/10/2013 10:45		

Detalle asistencia Aceptar Salir

Donat un pacient i un centre informats, al prémer el botó de cercar assistència mostrarà una finestra amb totes les assistències existents per a aquest pacient amb la intenció de que es triï una com a restricció a l'estudi d'auditoria.

A3. BUSCADOR D'ESDEVENIMENTS SEGONS L'APLICACIÓ I EL SEU DETALL

Aplicación	Evento	Descripción
1 vh_a_registro	Nuevo	Registro ambulantes-Nuevo
2 vh_a_registro	Modificación	Registro ambulantes-Modificación
3 vh_a_registro	Anulación	Registro ambulantes-Anulación
4 vh_as_activgst_reh	Ejecutar 1	Gestión de grupos. Ejecutar cambio de estado
5 vh_as_actividades	Nuevo	Actividades planificadas-Nuevo
6 vh_as_actividades	Modificación	Actividades planificadas-Modificación
7 vh_as_actividades	Anulación	Actividades planificadas-Anulación
8 vh_as_actividades	Ejecutar 1	Actividades planificadas-Eliminar varias
9 vh_as_actividades	Ejecutar 2	Actividades planificadas-Planificar grupo
10 vh_as_actividades	Ejecutar 3	Actividades planificadas-Planificar individual
11 vh_as_actividades_gest	Ejecutar 1	Gestión de las actividades. Ejecutar cambio de estado
12 vh_as_annotaciones	Nuevo	Anotaciones de curso clínico-Nuevo
13 vh_as_annotaciones	Modificación	Anotaciones de curso clínico-Modificación
14 vh_as_annotaciones	Consulta	Anotaciones de curso clínico-Consulta
15 vh_as_anulacion	Anulación	Anulación de la asistencia-Anulación
16 vh_as_asis_episodios	Nuevo	Registro datos de episodio-Nuevo
17 vh_as_asis_episodios	Modificación	Registro datos de episodio-Modificación
18 vh_as_asis_episodios	Anulación	Registro datos de episodio-Anulación
19 vh_as_asistencias	Nuevo	Registro datos de asistencia-Nuevo
20 vh_as_asistencias	Modificación	Registro datos de asistencia-Modificación
21 vh_as_asistencias	Anulación	Registro datos de asistencia-Anulación
22 vh_as_cambios	Ejecutar 1	Cambios de entidad y plan de cobertura. Ejecutar cambio
23 vh_as_censo_retro	Ejecutar 1	Censo anterior. Recuperar asistencia
24 vh_as_codificaciones	Consulta	Codificación diagnósticos y procedimientos-Consulta
25 vh_as_consumos	Consulta	Registro de consumos-Consulta
26 vh_as_contactos	Consulta	Personas de contacto-Consulta
27 vh_as_custodia	Nuevo	Registro objetos en custodia-Nuevo
28 vh_as_custodia	Modificación	Registro objetos en custodia-Modificación

El botó de Buscar Esdeveniment deriva a una finestra de cerca on es faciliten els tres camps relacionats aplicació, esdeveniment i descripció per tal de que es pugui trobar amb facilitat l'acció que es desitja.

A4. FITXA HOSPITALÀRIA DE PACIENT SEGONS UN CAPÍTOL

Identificador

* N° asistencia 2013H01198 Paciente MÚJICA MONAGAL, MARTA Referencia

N° expediente 176 Sexo 2 Mujer Diag. ppal. Edad 24 años

NHC 20140572 Ubicación 0038 Cama 0038 2 F. nacimiento 20/05/1991

Registro Asistencia Alta Movimientos

Registro de entrada

* Fecha ingreso 12/12/2013 * Hora 10:00

Tipo ingreso 1 NORMAL

* Motivo de entrada 1 INICIATIVA DEL PACIENTE

Aislamiento Reingreso

Tipo aislamiento

Fecha previsión alta 25/12/2013 Hora 14:00

Procedencia 0 DOMICILIO

Centro externo

Tipo de transporte 11 Sillas de ruedas

Transporte

Perfil de la asistencia

Asistencial 1 Enfermedad

Paciente 1 General

Niv. dependencia 2 Necesita cierta ayuda

* Motivo Caída

* Servicio ADM ADMISIONES

Especialidad E21 Traumatología

Profesional MASA Soler Adell, Miquel Angel

Datos económicos

Sociedad 00 Sociedad Unit4 HG

Pendiente confirmación Forfait Datos accidentes

Grupo de cobertura

Entidad 101495 EURO INSURANCES LIMITED

Plan de cobertura

Póliza

Tarjeta

* Prestación 2 Ingreso Hospitalización -

Notificación electrónica

Fecha Hora N° volantes entregados

N° referencia confir...

Observaciones

Etiquetas Hoja asistencia Opciones Lista de espera Aceptar Salir

En el moment en que l'usuari prem l'enllaç que apareix sobre el identificador de capítol, el sistema li derivarà a aquest formulari el qual conté tota la informació hospitalària del pacient en referència a la seva assistència i capítol.