

Projecte en administració de xarxes i sistemes operatius en entorns de programari lliure

Jordi Serra Ruiz
Remo Suppi Boldrito
Josep Jorba Esteve

P08/M2018/02095



Universitat Oberta
de Catalunya

www.uoc.edu

Índex

Introducció	5
Objectius	9
1. Estudi de viabilitat	11
1.1. Necessitats i requisits del client	13
1.2. Anàlisi de la situació actual	15
1.3. Definició de requisits del sistema	16
1.4. Estudi d'alternatives de solució	17
1.5. Valoració i elecció de les possibles solucions	20
2. Anàlisi del sistema	22
2.1. Definició del sistema	22
2.2. Requisits exactes del projecte	22
2.3. Establiment de requisits	26
2.4. Definició d'interfícies d'usuari	27
2.5. Especificació del pla de proves	28
3. Disseny del sistema	30
3.1. Arquitectura	31
3.2. Definició de nivells d'arquitectura	31
3.3. Especificació d'estàndards, normes de disseny i construcció	33
3.4. Identificació de subsistemes	34
3.5. Casos d'ús reals	35
3.6. Revisió de casos d'ús per subsistema	36
3.7. Especificacions de desenvolupament i proves	37
3.8. Requisits d'implantació	40
4. Desenvolupament	43
4.1. Planificació de les activitats d'integració del sistema	44
4.2. Com es tria la llicència més adequada	47
4.3. Entorn de desenvolupament	48
4.4. Documentació	49
5. Implantació	51
5.1. Formació	52
5.2. Implantació del sistema, proves i nivell de servei	53
5.3. Acceptació del sistema	54
5.4. Manteniment	55
Resum	56

Bibliografia	59
Annex	60

Introducció

El desenvolupament de projectes en administració de xarxes i sistemes operatius en entorns de programari lliure és similar a qualsevol altre tipus de projecte amb algunes subtils diferències. Els projectes no només han de contenir les eines desenvolupades, sinó també la formació als usuaris que hi treballaran.

Per a tenir un bon finançament del projecte, és necessari seguir un procés que passi per la definició clara i acurada de l'abast del problema que es vol solucionar, el disseny de la possible solució, el desenvolupament d'aquesta, la posada en funcionament i el manteniment de la solució escollida. És important, segons el tipus de projecte, tenir en compte els aspectes formatius per als usuaris i administradors que hauran de posar en producció la solució desenvolupada. Aquests aspectes no s'han d'oblidar, ja que la bondat de la solució proposada passa pel fet que els usuaris finals puguin resoldre els seus problemes des del primer moment. En cas contrari, el projecte necessitarà una sèrie de modificacions i retocs que no són compatibles amb un simple manteniment, i serà necessari desenvolupar noves versions amb el cost afegit i els problemes que comporta la posada en marxa. D'aquesta manera es disposarà de tota la informació de la implantació del sistema de manera centralitzada, ja sigui per a resoldre possibles problemes o per a avaluar la implantació *a posteriori*.

Encara que el responsable de projectes en entorns de programari lliure per a l'administració de xarxes i sistemes operatius, o d'un desenvolupament específic basat en la utilització de programari lliure, no necessita ser un expert en tota la tecnologia i aplicacions utilitzades, sí que és necessari que conegui tots els passos o fases del projecte. També haurà de conèixer tots els punts crítics en el desenvolupament del projecte, tenint clar quin ha de ser l'estat d'avenç i quins elements s'han de generar després de cada pas o fase, i si s'està en condicions de passar a la següent amb garanties d'èxit, o s'ha de revisar l'actual.

El responsable del projecte ha de dominar, amb una visió global, cada un dels elements que hi intervenen i posar en joc les habilitats per al següent:

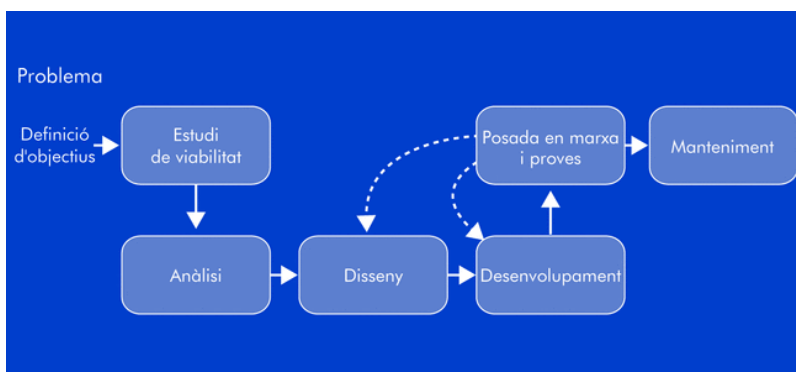
- 1) **Planificar.** S'han de planificar els recursos (físics i humans), i també la informació necessària en cada una de les fases del projecte. D'altra banda, és necessari que l'assignació de recursos es realitzi de manera racional segons la disponibilitat, el cost, l'especificitat i el temps.
- 2) **Organitzar.** Els aspectes d'organització són fonamentals en l'èxit del projecte i són responsabilitat de qui se n'encarrega. L'optimització de la utilització dels recursos i el cost dels serveis són dos dels elements que estan directament correlacionats amb l'organització i previsió (planificació). Dins d'aquesta organització, no s'han de descurar aspectes de coordinació entre

els projectes vinculats de l'empresa, la gestió i administració de la informació i s'han de tenir en compte factors humans i/o imponderables que puguin afectar-ne el bon desenvolupament.

- 3) **Verificar i validar.** El responsable del projecte ha d'estar en condicions de seguir el seu avenç i poder prendre decisions respecte a la seva qualitat, verificant si els resultats parcials obtinguts coincideixen amb el que s'ha planificat i validant-lo parcialment amb els requisits i objectius proposats.

L'objectiu d'aquest curs és definir les fases vitals que s'han de tenir en compte en un projecte i que el responsable haurà de seguir i supervisar durant la seva vida.

Figura 1. Cicle de vida d'un projecte



Com es mostra en la figura anterior, el cicle de vida del projecte està format per fases i interrelacions entre elles. Les interrelacions mostrades amb línies de punts depenen del grau de compliment dels objectius i no necessàriament són en tots els projectes. Hi pot haver altres interrelacions, el responsable del projecte haurà d'avaluar i prendre la decisió sobre si és necessari modificar el disseny o el desenvolupament sobre la base de la consecució de les metes proposades i el grau de compliment dels objectius establerts a l'inici del projecte. És evident que aquestes decisions són complexes, ja que poden excedir l'abast del projecte i poden generar costos addicionals no planificat *a priori*. El cap de projecte té la responsabilitat d'avaluar si aquest compleix o no els requisits del client i prendre les mesures oportunes.

A partir del problema, les fases d'un projecte es poden descriure de la manera següent:

- **Definició dels objectius i abast.** En molts projectes, els objectius són requisits del client i no es compta com una fase pròpiament dita, però en d'altres, a partir de l'especificació del problema, s'han de definir els objectius d'una manera clara i concisa. La definició dels objectius del projecte s'aconsegueix després d'una sèrie de reunions entre client (qui encarrega i utilitza el projecte) i cap de projecte (responsable del grup de treball que dissenya, desenvolupa i instal·la la solució informàtica en qüestió). A l'acceptació dels objectius del projecte s'hi arriba després de l'exposició

d'idees funcionals sense detalls tècnics (pluja d'idees) i amb el compromís per ambdues parts.

- **Estudi de viabilitat.** Dins aquesta fase s'analitzaran acuradament les possibles solucions als requisits del projecte per a la consecució dels objectius, quines solucions tècniques hi ha, i quina és la més adequada tenint en compte aspectes com la qualitat, la funcionalitat de la solució i el cost resultant d'aquesta.
- **Anàlisi.** Té per finalitat descriure detalladament el sistema que es vol realitzar, quins requisits funcionals ha de tenir i com es compliran els objectius establerts.
- **Disseny.** En aquesta fase es desenvolupa la idea des del punt de vista tecnològic, seleccionant quines eines s'utilitzaran i quina és l'arquitectura de l'aplicació.
- **Desenvolupament.** En aquesta fase, es realitzarà la programació de la solució adoptada, proposant diferents models de desenvolupament. És important que durant el desenvolupament es vagin fent proves de validació parcials per a la depuració dels possibles errors i vulnerabilitats de seguretat en la programació que es puguin crear en el desenvolupament de l'aplicació.
- **Posada en marxa i proves.** Durant aquesta fase s'instal·larà el sistema i es realitzaran proves d'integritat abans de passar-lo a producció. Aquesta fase pot generar que s'hagin de realitzar retocs en el desenvolupament o en el disseny per a solucionar problemes que hagin sorgit durant les proves. Un retrocés en el desenvolupament és acceptable en la majoria dels projectes, però s'ha de tenir en compte que durant la fase de disseny és complicat i que la decisió s'ha de meditar acuradament. Posteriorment, si es dóna el vistiplau, es passarà el sistema a producció amb la finalitat que els administradors i usuaris el puguin utilitzar.
- **Manteniment.** Amb la pràctica diària, els usuaris trobaran els errors que puguin sorgir de situacions particulars i s'analitzaran les possibles millores (versions evolutives) de l'aplicació. Durant aquesta fase o l'anterior, dependent del projecte, se solen incloure sessions de formació per als administradors i usuaris a fi de cedir gradualment responsabilitats des de l'equip del projecte cap als usuaris finals del sistema. Aquest aspecte és important per a evitar la dependència d'un equip respecte a l'altre i la sobrecàrrega de feina, sobretot si l'equip de projecte forma part de la mateixa institució o empresa.

Amb la definició anterior de projecte o sota altres noms o apartats, aquestes fases estaran presents en qualsevol projecte que involucri les tecnologies de la informació i la comunicació, independentment de la metodologia utilitzada per a la seva gestió (seqüencials, cascada, concurrents, etc.).

Pel que fa a aquest document guia, s'utilitzarà un cas pràctic per a aplicar els coneixements teòrics descrits. Es presentarà un exemple (basat en el cas pràctic) de cada apartat per a mostrar els aspectes concrets de com han de desenvolupar els estudiants el punt en qüestió (s'identificarà amb "Cas pràctic: [...]"). S'ha de tenir en compte que la solució proposada a aquest projecte no és l'única possible i que només té com a fi presentar una de les possibles alternatives i servir com a marc de referència per al seu desenvolupament. Durant l'etapa d'aprenentatge, és convenient treballar per analogia per a exercitar, d'aquesta manera, totes les fases proposades, aprendre a resoldre les dificultats que el projecte genera i seguir una metodologia adequada en la seva consecució durant aquest període de formació.

Es considera que aquest projecte serà desenvolupat per un equip format per un cap de projecte que es responsabilitzarà de totes les fases i un grup de treball (analistes, dissenyadors, programadors, etc.). La distribució de responsabilitats entre el cap de projecte i el seu grup dependrà del volum del grup de treball i la complexitat del projecte. Normalment, segons el projecte, el cap de projecte defineix el grup de treball i quines tasques o responsabilitats ha d'assumir cada un dels membres. En altres casos, el cap de projecte és un membre més del grup que a més de la direcció del projecte realitza altres tasques específiques.

Objectius

Hi ha una sèrie d'objectius que l'estudiant ha d'aconseguir al final del curs de *Projecte en administració de xarxes i sistemes operatius en entorns de programari lliure*, d'entre els quals es poden enumerar els següents:

- 1.** Assimilar els coneixements quant a la realització de projectes en entorns de programari lliure, que si bé es pot aplicar una metodologia general en el desenvolupament de projectes, presenten certes particularitats pel que fa a la tecnologia, la capacitat de decisió i la bondat de la solució.
- 2.** Adquirir experiència i coneixements a determinar quins són els punts vitals en un projecte d'administració de xarxes i sistemes operatius, adaptant a la situació particular les fases que s'han definit anteriorment.
- 3.** Analitzar, seleccionar i provar quines eines de programari lliure s'utilitzaran per a desenvolupar el projecte i quines es faran servir per a administrar les xarxes i els sistemes operatius, tenint en compte elements com l'entorn, la capacitació dels usuaris i l'autonomia que hauran d'haver adquirit quan el projecte hagi finalitzat.
- 4.** Aplicar al cas proposat la metodologia i els coneixements adquirits al llarg de les diferents assignatures, i també vincular tots aquests coneixements a un projecte que s'haurà de desenvolupar utilitzant únicament eines basades en programari lliure. Així mateix, el projecte (resultat de la feina de l'estudiant) s'haurà de basar en el mateix tipus d'eines.
- 5.** Generar tota la documentació i la informació necessària perquè, una vegada el projecte es trobi en producció (aplicat a l'empresa i amb usuaris d'aquesta), permeti resoldre les tasques bàsiques que hi estan vinculades com la instal·lació i configuració, supervisió, adaptació a l'entorn canviant d'una empresa o interpretació dels resultats. Aquesta informació també haurà d'incloure documentació, com ara manuals d'usuari perquè aquests puguin aprendre a treballar amb la solució proposada sense la intervenció de l'equip de projectes.

1. Estudi de viabilitat

La importància vital per a qualsevol projecte dins l'àmbit de l'administració de xarxes i sistemes operatius és l'estudi de viabilitat, ja que serà aquest, a partir d'unes necessitats plantejades, el que permetrà escollir la millor solució que inclogui les necessitats del client i si és o no factible el seu desenvolupament. És important, per tant, que l'estudiant **assumeixi** el rol del cap de projectes, i interpreti els requisits i necessitats de l'enunciat del projecte. En un cas real, el cap de projecte haurà de mantenir un diàleg en profunditat amb els responsables de l'empresa que encarreguen el projecte perquè li transmetin totes les necessitats, especificacions i requisits que haurà de complir la solució adoptada i evitar solucions incompletes o afegits a la solució inicial.

En l'estudi de viabilitat, s'ha de començar considerant tres aspectes bàsics en tota solució informàtica:

- L'estat actual del sistema i la seva configuració (maquinari o *hardware* i programari o *software*).
- Quins problemes s'han de solucionar i quins són els requisits de la solució, tant estructurals com funcionals, sense descurar les particularitats de qui treballarà (instal·larà i configurarà, mantindrà, utilitzarà, etc.) amb la solució i el seu grau de formació.
- Quines són les restriccions al sistema, al qual s'ha d'aplicar la solució, ja que en general el cap de projectes es trobarà amb unes restriccions *a priori* com "els usuaris estan acostumats a...", o aspectes tecnològics com "s'utilitza aquest maquinari, aquesta base de dades o la connexió és de X Mbps", etc.

L'estudi de viabilitat ha de considerar les diferents propostes per arribar a un acord amb el client, que serà qui prendrà la decisió final guiat pels seus assessors i pel cap de projectes. Per tant, aquest estudi, a més d'un resum executiu, haurà d'incloure els aspectes següents:

- **Econòmics:** un informe detallat preliminar, però amb un alt grau de detall sobre el cost de la solució proposada (o de cada una de les solucions si hi ha diferents alternatives viables). Aquesta anàlisi de costos ha de ser rigorosa i incloure totes les despeses previstes en la consecució del projecte, i s'ha d'evitar el costum d'incrementar els costos en un determinat percentatge pels imprevistos. En projectes en els quals hi ha costos molt difícils de valorar en el moment de l'anàlisi de viabilitat, es fixen els procediments mitjançant els quals es pugui fer front econòmicament al projecte sense haver-lo de redefinir en el moment de negociar el contracte.

- **Tècnics:** aquests aspectes permetran prendre una decisió precisa sobre quina de les propostes és la més adequada a les necessitats de cada cas, i que juntament amb l'anàlisi econòmica en permetran l'adequació. Es pot donar el cas que les solucions més adequades des del punt de vista tecnològic signifiquin canviar l'arquitectura del sistema i formar de nou a tot el personal, la qual cosa transforma aquesta opció en inviable. Davant aquestes situacions, s'haurà de prendre una solució de compromís.
- **Legals:** depenent del tipus d'instal·lació i dades que manegi l'empresa i ateses les normatives legals (protecció de dades, propietat intel·lectual, confidencialitat, seguretat industrial, etc.), l'estudi haurà d'incloure aspectes vinculats a aquests temes i de quina manera els aborda la solució proposada.
- **Aspectes de funcionament:** s'hauran d'incloure els mètodes i procediments operatius per a cada un dels diferents aspectes recollits en la solució proposada.

Quant a la selecció de la millor solució informàtica, podem diferenciar dos escenaris possibles:

- **Projecte final del curs.** Durant el desenvolupament del curs, l'estudiant haurà d'escollir la millor solució tenint en compte tant l'impacte en l'organització com en els seus diferents membres, el cost associat, no només econòmic, sinó en adaptació i esforç dels usuaris, etc., i els riscos associats que implica la posada en marxa de la solució proposada. Aquest últim aspecte s'ha de tenir especialment en compte, ja que serà un dels més importants a l'empresa: no hi ha d'haver pèrdua de la informació i s'ha de tenir en compte que s'han d'evitar o minimitzar (si són inevitables) les interrupcions del servei de l'empresa durant i després de la implantació de la solució.
- **Projecte en entorn real de l'empresa.** Si bé es poden aplicar els mateixos criteris del paràgraf anterior, s'han d'incloure els representants de l'empresa, als quals haurà d'explicar quina és la balança de beneficis-riscos de la solució adoptada perquè es prengui la decisió adequada conjuntament i la responsabilitat no recaigui únicament en el cap de projecte o almenys que la responsabilitat sigui delegada al cap de projectes amb el consentiment de l'empresa.

Els paràgrafs següents presenten el problema i detallen, per al cas de l'exemple proposat, com es tira endavant l'estudi de viabilitat.

1.1. Necessitats i requisits del client

Com s'ha esmentat anteriorment, aquesta és una fase molt important del projecte: interpretar bé les especificacions, requisits i necessitats del client permetrà dissenyar la solució adequada per a aquest problema. Si iniciem el projecte amb una idea equivocada del que vol realment el client, provocarà allargar molt més el projecte amb reunions posteriors i modificacions de la solució erròniament implementada.

Les necessitats poden ser plantejades per un client extern si treballem per a una empresa que es dedica a desenvolupar solucions d'aquestes característiques, és a dir, el projecte serà per a una altra organització. En aquest cas, és aconsellable dedicar un quant temps a analitzar les característiques de l'empresa, el model de negoci, els hàbits i costums, la tecnologia utilitzada fins a aquell moment, etc. També podem trobar-nos que és la mateixa empresa que disposa d'un departament de projectes amb un responsable i el projecte és per a un altre departament o unitat de la mateixa empresa. En aquest últim cas, el cap de projectes ja té un coneixement clar de com funciona l'empresa i les qüestions vinculades al projecte que s'han de realitzar.

Per a aquesta primera fase de l'estudi, és recomanable fer una anàlisi de dalt a baix (*top-down*) començant per una descripció general i anant aprofundint en detalls dels aspectes comentats en el paràgraf anterior que tinguin especial rellevància.

És important que el cap de projecte (l'estudiant en el nostre cas) obtingui una visió global del problema sense detenir-se en detalls i a continuació comenci a desglossar cada un dels aspectes esmentats anteriorment.

Com a cas pràctic s'utilitzarà una empresa que necessita actualitzar el seu sistema informàtic i un conjunt de serveis nous (vegeu les figures següent). L'estudiant ha de tenir en compte que aquest exemple s'utilitza únicament per a explicar de manera aplicada els conceptes teòrics i, per tant, els detalls que es donen són únicament com a ampliació d'aquest cas específic, però que són útils en la majoria de projectes d'aquestes característiques. Cada estudiant haurà de realitzar tots els passos per al seu projecte específic.

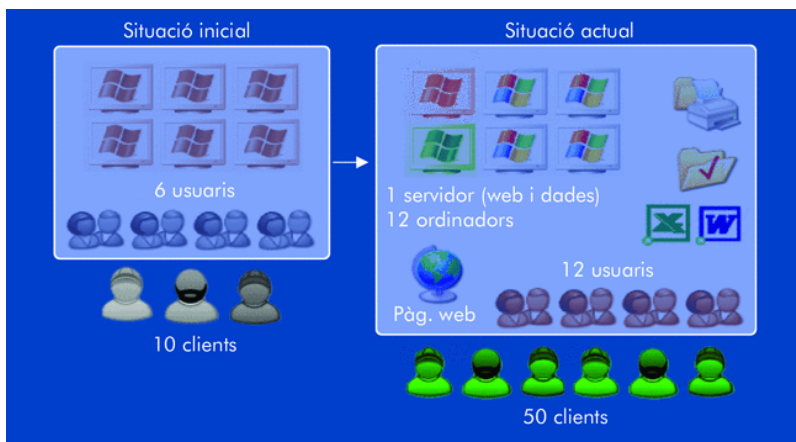
Cas pràctic: situació actual

Projecte d'adequació i renovació de la xarxa i els serveis informàtics per a l'empresa NtEum, SA

L'empresa NtEum es dedica a la prestació de serveis en l'àmbit editorial. El seu sistema informàtic està basat en solucions propietàries, i ha experimentat un creixement molt elevat en un període de temps curt i gens planificat. A mesura que s'ha necessitat un nou lloc informàtic, aquest s'ha creat sense una planificació prèvia de l'estructura general.

L'empresa ha passat d'un parc informàtic de sis ordinadors basats en tecnologia Windows[®]98, autogestionat, amb sis usuaris en ofimàtica, administració i serveis a clients externs (deu clients en aquest cas) a un sistema basat en la mateixa tecnologia però diferents versions de sistemes operatius (Windows[®]98, Windows[®]2000), amb dotze ordinadors més un servidor (pàgina web i dades internes), amb dotze usuaris (empleats de l'empresa) i cinquanta clients externs. Aquest canvi s'ha produït en sis anys per les necessitats de l'empresa sense planificació ni renovació tecnològica (només amb noves adquisicions de sistemes informàtics i personal).

Figura 2. Projecte en administració de xarxes i SO de NtEumactual



Cas pràctic: projecte

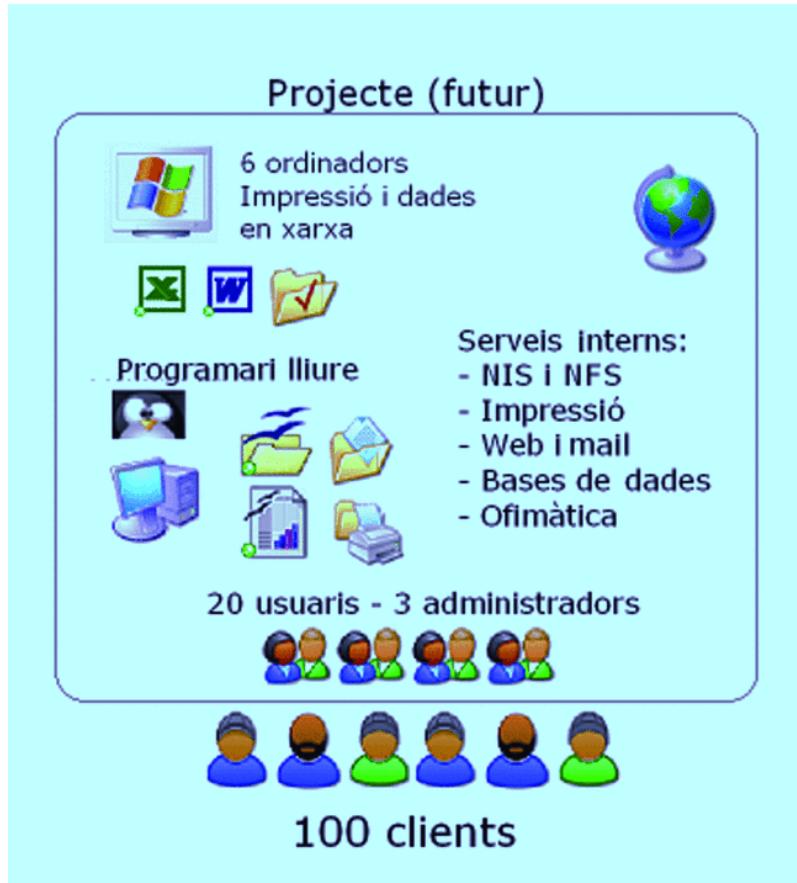
Projecte d'adequació i renovació de la xarxa i els serveis informàtics per a l'empresa NtEum, SA

L'empresa sol·licita un projecte de renovació, adaptació i creixement per a fer front a un conjunt de serveis no prestats en l'actualitat i a un previsible augment de negoci a cent clients externs. Es desitja integrar part del sistema informàtic actual, sis màquines d'usuaris amb dades compartides i impressió en xarxa, i que aquesta solució es basi en programari lliure.

El projecte ha de recollir la integració i/o substitució dels serveis actuals, el suport per als nous serveis i la formació dels usuaris i administradors del sistema amb l'ampliació a vint usuaris (administratius i comercials empleats de l'empresa) i tres tècnics (administradors i/o gestors del sistema informàtic propis de l'empresa).

La solució adoptada ha de permetre a l'empresa complir tota la legalitat vigent, tenir presència a Internet per mitjà d'una plataforma de contractació i gestió de serveis electrònics (e-services) i poder realitzar la transferència d'informació (productes i gestió) via canals de comunicació segurs entre empresa i client extern.

Figura 3. Passat i present de NtEum



El nou sistema ha d'integrar els sis ordinadors Windows® i ha de prestar els següents serveis interns basats en programari lliure: accés únic (NIS), dades compartides (NFS, i dades Windows®), seguretat (zona de servidors i DMZ), impressió en xarxa, servei de web i correu, base de dades (clients i personal), i ofimàtica. A més, per als clients de l'empresa ha de proporcionar els serveis següents: http, correu electrònic, servei d'impressió a les màquines de l'empresa, FTP, fòrums, comerç electrònic, base de dades. Tots els serveis de dades hauran d'incloure polítiques de seguretat d'informació (*backup* o còpies de seguretat) adequades a les seves característiques.

És important considerar que el canvi tecnològic haurà d'anar acompanyat de formació (si bé s'incorporarà suport tècnic com a empleats de l'empresa), ja que tota la infraestructura afectarà la feina de l'empresa i que aquesta feina s'ha de planificar acuradament.

1.2. Anàlisi de la situació actual

Després de la descripció de necessitats i requisits, és necessari fer una anàlisi dels sistemes de l'empresa i realitzar un diagnòstic tan precís com sigui possible fent una llista de tasques i activitats (*checklist*) per a no oblidar-se'n cap (per evident o mínima que sigui). Aquesta llista permetrà tenir present la integració del sistema actual amb el nou i evitar improvisacions posteriors quan el projecte estigui en proves o en producció real. Hi haurà un període de temps en què hauran de conviure ambdós sistemes, ja que hi haurà aplicacions que no es podran migrar de manera ràpida.

En aquest cas, s'inclouran: les característiques tècniques dels ordinadors (maquinari, sistema operatiu i programari) i les aplicacions i serveis que s'hauran d'integrar en el nou sistema. Així mateix, és important fixar quins usuaris par-

ticiparan en l'estudi de la situació actual per resoldre qüestions i prendre decisions de disseny de la nova infraestructura. En el nostre exemple pràctic, caldrà descriure cada un dels sistemes:

- Ordinadors: tipus, SO, aplicacions i serveis.
- Impressores: tipus i serveis.
- Xarxa: tipus, estructura i característiques de seguretat.
- Dades compartides: tipus de servei, característiques de seguretat.
- Estructura física: on es troben els equips i on s'allotjaran els nous sistemes / usuaris / personal tècnic.
- Per als punts que així ho justifiquin (per exemple, els que estiguin vinculats a estructures físiques o lògiques), es poden incloure diagrames per aclarir l'estructura dels serveis o la seva dependència del maquinari.

1.3. Definició de requisits del sistema

En la descripció del sistema (i tenint en consideració l'opinió dels seus usuaris, sobretot en la part que s'haurà d'integrar), s'hi han de recollir els requisits que ha de complir el projecte del qual s'estudiarà la viabilitat.

Amb els requisits es podran avaluar les possibles alternatives per a solucionar el problema proposat. És necessari tenir en compte que en aquesta descripció s'ha d'incloure una prioritat (p. ex., entre 1 i 10, essent 10 la màxima prioritat) a fi de tenir una valoració relativa de cada un. Així, en la fase de disseny de l'arquitectura se sabrà quants recursos se li han d'assignar. La manera més simple d'elaborar aquesta llista de requisits és senzillament amb una descripció, una breu explicació i la prioritat que se li assigna.

Cas pràctic

En el nostre cas s'haurà de fer una llista descriptiva dels serveis que es necessitarà instal·lar per a dimensionar l'arquitectura del sistema i decidir quin és el maquinari que s'ha de comprar i quina es pot aprofitar. En el cas de l'exemple presentat, els serveis detallats són els següents:

- Accés únic (10): servei de NIS i NFS perquè cada usuari pugui accedir a les dades independentment del terminal que es faci servir.
- Accés controlat a web i correu segur (10).
- Base de dades de l'empresa amb dades sobre clients i personal per a la facturació i nòmines (10) (separades).
- Sistema de seguretat per mitjà de tallafocs i DMZ (10).
- Impressió en xarxa (9).

- Servei d'accés a publicar dades en el sistema web de l'empresa (9).
- Gestió de còpies de seguretat i informació dels sistemes (*logs*) (8).

Per als clients de l'empresa s'han d'oferir els serveis següents:

- Portal web (8): informació de serveis i punt d'entrada a la compra de serveis per comerç electrònic.
- Impressió remota (7): els clients poden enviar les seves impressions d'alta qualitat als dispositius de l'empresa.
- Servei d'emmagatzematge d'arxius (7): els clients poden llogar un espai d'emmagatzematge per mantenir-hi els seus arxius particulars (FTP en forma segura –SFTP–).
- Fòrums de discussió, atenció al client i suport (PMF o FAQ en anglès) (5).
- Servei de base de dades als clients (4).

Els requisits legals han de ser coberts pel control de seguretat i la separació de dades amb identificació dels usuaris i seguiment de l'activitat a cada servidor. Quant a llicències de programari, s'han de basar (sempre que sigui possible) en solucions de programari lliure i han de ser el més restrictives possible. Els requisits econòmics s'han de fonamentar en l'equilibri entre el cost i el servei que s'ha de donar (és important preveure si els servidors necessiten sistemes de vint-i-quatre hores i set dies a la setmana de funcionament per definir el manteniment del maquinari i els sistemes d'alimentació interrompuda –SAI–). Els requisits operatius passen per la responsabilitat del personal tècnic de l'empresa per mantenir tots els serveis una vegada configurats i posats en marxa per l'equip de projectes.

1.4. Estudi d'alternatives de solució

Amb els requisits que planteja el nou projecte, s'han d'estudiar diferents solucions, sempre que sigui possible, que els compleixin tots. Per a això, és necessari incloure tota la informació recollida fins al moment i per a cada alternativa s'haurà d'especificar tant la seva definició funcional, com tècnica, els seus punts forts i els seus punts dèbils, i descriure el valor econòmic que comporta tant en costos directes per exemple, les llicències, com en costos derivats, per exemple les actualitzacions. Tampoc no s'ha de descurar la informació sobre els estàndards que compleix, la implantació en el mercat i el suport que ofereix.

Cas pràctic

En el nostre exemple pràctic, un dels requisits és que la solució es basi en programari lliure, però es poden analitzar diferents opcions dins aquestes variants:

Llicència GPL –*GNU Public License*– (mínim cost): servidors i màquines d'escriptori basats en distribucions com Fedora, Debian, etc.

- Parcialment en llicència GPL: servidors en Fedora, Debian o similars i escriptoris en opcions amb llicència GPL, com per exemple, Linspire, SuSE, Mandrake, etc.
- Llicència LGPL o altres de similars (més cost): servidors i escriptori basats en distribucions com Red Hat, Mandrake, SuSE, etc.





S'han d'analitzar acuradament aquestes opcions tenint en compte el cost derivat de les llicències que, en aquest projecte, a banda del maquinari, serà el cost més elevat juntament amb el de personal.

S'ha de recordar que la definició de codi obert és una especificació del tipus de codi i no és en si mateixa una llicència de programari. Una de les llicències d'aquest tipus més comunes és la GPL, amb la qual es pot copiar, modificar i distribuir el programari, però les modificacions s'han de fer públiques sota la mateixa llicència. La GPL es diferencia de la llicència LPGL, en el fet de permetre la barreja amb programari propietari. Hi ha una gran quantitat de programari lliure que inclou part propietària, com per exemple, MPL (*Mozilla public license*) i NPL (*Netscape public license*) en Mozilla, BSD en Apache, etc.

Per a comptabilitzar i facilitar el pas següent, és interessant confeccionar una taula que presenti les característiques possibles. Es mostra a tall d'exemple la següent figura, en la qual es comparen les diferents opcions de sistemes operatius. S'haurà de completar la informació sobre el preu de les distribucions actualitzades, i en l'última fila haurien d'incloure's les necessitats mínimes per a processador, memòria, disc i el preu resultant de la computadora, no inclosos en aquest exemple.

Cas pràctic

Figura 4. Comparació entre els diferents sistemes operatius

Sistema operatiu	 debian Woody	 Mandrake Versió Community 10	 redhat Versió 9	 Windows Professional
Recomanat per a	Servidors – escriptori	Escriptori – servidors	Servidors – escriptori	Escriptori – servidors
Tipus d'usuari	Tècnic	No tècnic	Tècnic	No tècnic
Cost aproximat	€	€	€	€
Temps instal·lació (min) (mateix PC aprox.)	27' (depenent elecció)	20' (depenent elecció)	40' (depenent elecció)	58'
Seguretat	Excel·lent	Excel·lent	Avançada	Regular (afectada per virus especialment)
Fiabilitat	Molt alta	Molt alta	Molt alta	Normal (incoent-hi tots els SPack)
Execució des de CD sense instal·lació (CD autònom)	No (directament)	Sí	No (directament)	No
Suport per a ofimàtica	Inclou OpenOffice (compatible amb MSOffice)	Inclou OpenOffice (compatible amb MSOffice)	No per defecte (es pot instal·lar OpenOffice cost 0 €)	No per defecte Opcions: OpenOffice 0 €, MSOffice STD 2003 399S
Permet execució i creació d'aplicacions MS Windows	Sí per mitjà de Wine – s'ha d'incloure –	Sí per mitjà de Wine – inclòs –	Sí per mitjà de Wine – s'ha d'incloure –	Natiu
Permet l'execució i creació d'aplicacions Linux	Natiu	Natiu	Natiu	Parcial per mitjà de programari lliure CygWin
Eines: adm. control de xarxa, desenvolupament programari, etc.	Sí	Sí	Sí	Parcial (algunes d'adm. i utilitats)
Base de dades, tallafoç, servidors, etc.	Sí	Sí	Sí	Tallafoç: sí (XP) Web: sí (XP, 2k)
Necessitats mínimes de maquinari	Processador Memòria Disc Preu	Processador Memòria Disc Preu	Processador Memòria Disc Preu	Processador Memòria Disc Preu

Relacionat amb aquest aspecte, i depenent del projecte, és interessant calcular el que es denomina *TCO* (cost total de la propietat d'un ordinador o *total cost of ownership*), que comporta les partides següents:

- Cost total del maquinari i programari
- Millores previstes en maquinari i programari (*updates*)
- Manteniment
- Suport tècnic

- Entrenament o aprenentatge

Es pot prendre com a base per a una estimació d'aquest cost 3-4 vegades el cost de compra del PC i pot aportar informació sobre la plataforma i/o arquitectura més adequada que s'ha de seleccionar. En el nostre cas, aquest càlcul el farem de manera separada per veure la influència dels diferents elements que defineixen la inversió total (aproximada) del projecte.

1.5. Valoració i elecció de les possibles solucions

Amb una llista de requisits i característiques tècniques de les possibles alternatives, s'ha d'escollir la solució més apropiada per al projecte. En aquesta fase es tindrà en consideració tota la informació recollida fins al moment (la descripció general, l'abast, la situació actual, etc.) i per a cada alternativa s'haurà d'especificar en què consisteix, tant des del punt de vista funcional com tècnic.

En el nostre cas, la taula anterior serà de vital importància per a la presa de decisions. Aquesta taula (aquí només feta com a exemple i amb dades figurades) s'ha d'ampliar perquè inclogui una llista de tots els serveis que són necessaris per a cada requisit i així pugui facilitar la selecció de la solució més adequada. Aquesta taula permetrà, a més, poder fer un informe executiu de cada opció.

Cas pràctic

Per exemple, per al punt 4 de les necessitats del cas pràctic proposat: "Sistema de seguretat per mitjà de tallafocs i DMZ (prioritat 10)"

- **Debian:** integrat i d'alta confiança i funcionalitat (*iptables*). Configuració molt tècnica. Grau d'adaptabilitat: 100%. Ubuntu es considera que es basa en Debian.
- **Mandrake:** integrat (Shorewall). Versatilitat limitada (és possible instal·lar *iptables*). Molt fàcil de configurar i controlar. CentOS es considera que es basa en RedHat.
- **RedHat:** anàleg a Debian. CentOS es considera que es basa en RedHat.
- **Windows®:** només en XP i 2000 Server amb SP2, Vista, 2003 Server i 2008 Server. Limitat i configuració acceptable.

És necessari tenir en compte que en l'anàlisi econòmica s'han de considerar les despeses directes i les indirectes de cada solució. A més d'estudiar la viabilitat econòmica de les diferents solucions, s'hauran d'analitzar els riscos associats a cada una d'elles.

Per a cada una de les alternatives existents, serà necessari descriure quines incerteses, problemes potencials, etc. hi ha. Per exemple, s'haurà de valorar com a risc: seguretat, continuïtat del SO, responsabilitats legals, cicle de millores estable (*updates*), etc. Per a cada un dels riscos potencials detectats, s'han d'incloure les alternatives possibles per a fer front a aquestes possibles amenaces i de quina manera poden afectar la capacitat de funcionament del sistema.

A més, com que tota la infraestructura ha de ser autònoma una vegada instal·lada i configurada, no s'han de descartar els costos associats a la formació dels usuaris i els tècnics de l'empresa. Hi ha projectes en els quals és possible contractar empreses dedicades al manteniment i actualització d'aquests serveis (*outsourcing*), els quals tindran també uns costos associats. En aquest últim cas, s'ha de redactar un contracte precís que detalli totes les opcions i prestacions que s'han de realitzar, i també les penalitzacions que es derivin de l'incompliment de les condicions.

Cas pràctic

Tenint en compte tots els elements de judici plantejats abans, la solució adoptada (una de les possibles) per al nostre sistema exemple seria:

- **Sistema operatiu:** CentOS 5.
- **Accés i administració:** NIS (accés a les dades independent de l'ordinador), Webmin, LinuxConf (administració i configuració).
- **Arxius i impressió:** NFS (compartició d'arxius), Samba (compartició d'arxius des de sistemes Windows), CUPS (servidor d'impressió).
- **Xarxa i seguretat:** DHCP (servei d'informació per a màquines), NAT (traducció d'adreces de xarxa), DNS (serveis d'informació de noms), Xinetd (servei de connexions segur), SSH (serveis de connexió interactius segurs), Wu-FTP, SFTP (servei de transferència de fitxers –normal i segur–), Iptables (tallafocs), Nmap, Nessus, Snort, Logcheck, Tripwire (eines de seguiment, seguretat i control), Gnupg (gestió de firma digital i xifratge).
- **Serveis web, correu i missatges (o news):** Apache-Tomcat (servei de http), Exim (servei de correu), IMAP (servei de correu electrònic a clients), LeafNode (servei de missatges).
- **Servei de base de dades:** PostgreSQL.

Amb tota aquesta informació es procedirà a redactar un informe, que inclourà un pressupost detallat dels costos del projecte (tant els costos directes com indirectes). L'informe haurà de ser acceptat i aprovat per totes dues parts (clients - grup de projectes). Depenent de la relació entre el client i el grup de projecte –és a dir, si és un client extern o si el projecte és per a la mateixa empresa–, es firmarà un annex al contracte que contingui els elements necessaris que indiquin quines són les obligacions i responsabilitats de cada una de les parts.

2. Anàlisi del sistema

Una vegada definida la solució que s'ha d'adoptar en l'estudi de viabilitat, és necessari especificar-la detalladament amb l'objectiu de preparar el seu disseny i la seva arquitectura. Aquesta especificació es realitza durant la fase d'anàlisi del sistema, en la qual és molt important la interacció amb els seus usuaris per a descartar la possibilitat d'omissió i/o errors que portin a dissenys no adequats.

2.1. Definició del sistema

Com es veurà en els paràgrafs següents, s'ha de descriure el sistema amb un alt grau de detall, especificant no només la funció de cada element, sinó també la seva comunicació amb la resta de components. També haurà de ser necessari especificar quin usuari serà el responsable de cada un d'aquests components i fins on arribarà la seva responsabilitat.

És important destacar que en aquest apartat i per al cas pràctic proposat, la descripció es farà tenint en compte la d'alt nivell realitzada en l'estudi de viabilitat de l'apartat anterior, però amb més profunditat. Es pot considerar que aquesta anàlisi ha de servir per a poder, més endavant, fer el disseny de la solució adoptada.

És per això que dins d'aquest apartat s'utilitzarà l'enumeració de requisits de l'estudi de viabilitat per a establir-ne les necessitats exactes i quin serà l'intercanvi d'informació, en el cas que n'hi hagués, entre els diferents components del sistema. S'ha de remarcar la importància d'aquest apartat, ja que una definició inadequada del sistema d'intercanvi d'informació pot donar lloc a un disseny que no prevegi totes les necessitats de l'empresa. Aquesta situació implicaria salts enrere en el projecte amb els habituals retards i costos afegits a la finalització del projecte.

2.2. Requisits exactes del projecte

- **Requisits legals.** Atès que l'empresa processarà dades relatives al client i als empleats, s'han de complir totes les normatives vigents en protecció de dades i accés a la informació. Per això, el disseny haurà de tenir en compte els mètodes d'identificació i control d'accés a la informació d'acord amb la llei i amb els mecanismes de certificació i control i xifratge adequats per a complir el que s'estableix legalment; aquest punt és molt important, no quant a l'aplicació, sinó al compliment de les normatives existents en l'àmbit de protecció de dades personals.
- **Requisits de propietat intel·lectual i llicències.** El projecte s'ha de basar en programari lliure i en llicències tan poc restrictives com sigui possible,

tenint en compte les inversions necessàries, tant en l'adquisició inicial com en el servei postvenda d'actualitzacions i manteniment.

- **Requisits d'accés únic.** Les dades de l'empresa se centralitzaran en un servei distribuït (intern) d'arxius mitjançant NFS. Aquesta opció haurà de tenir en compte els requisits de seguretat per a l'acceptació del servei de les màquines client i amb els criteris adequats quant als permisos de lectura-escritura per a cada màquina client. També les dades d'usuari de la xarxa hauran d'estar centralitzades per un servei d'informació distribuïda (NIS), de manera que cada usuari tingui a l'empresa un únic compte i una paraula clau per a l'accés als serveis des de totes les màquines. Atesa l'existència de màquines Windows, serà necessari la integració amb un servidor Samba per a l'accés unificat als arxius des d'aquestes màquines.
- **Requisits d'accés web, correu electrònic i grups de discussió.** Els serveis d'accés a pàgines d'informació de l'empresa hauran de preveure la possibilitat d'accés xifrat (https) mitjançant SSL, ja que l'accés a algunes parts de la informació haurà de ser confidencial per a alguns clients. Així mateix, haurà de preveure mòduls d'identificació per a l'accés personalitzat a aquesta informació. Els serveis de correu electrònic també hauran de ser xifrats per mitjà de SSL, tant si ho són mitjançant clients IMAP o POP (imaps i pops), com si ho són mitjançant correu web (https). Els serveis de correu electrònic hauran de tenir en compte la inclusió de recursos antivirus i anticorreu brossa, tant per als missatges d'entrada com per als de sortida. L'accés als grups de notícies podrà ser obert, però la inclusió de notícies s'haurà de controlar mínimament, per exemple, amb registre previ i amb moderadors per a cada tema, ja que aquest sistema serà utilitzat tant per a les preguntes més freqüents (PMF o FAQ, en anglès), com per a la informació de suport.
- **Requisits sobre la base de dades.** La base de dades de l'empresa haurà de ser només accessible a les màquines de la intranet, ja que contindrà informació sensible sobre clients, facturació i nòmines pròpies de l'empresa. Es podrà accedir a aquesta base de dades mitjançant peticions SQL des de serveis web com, per exemple, mitjançant pàgines web amb PHP o interactivament, però sempre amb paraula clau en tots dos casos.
- **Requisits del sistema de seguretat.** El sistema informàtic de l'empresa (servidors i màquines client) haurà de tenir un disseny basat en un tallafoc institucional amb una definició clara de zones desmilitaritzades (DMZ), amb adreces privades i traducció per mitjà de NAT. El sistema haurà de disposar de les eines de detecció i anàlisi adequades que permetin als administradors realitzar un control adequat del sistema informàtic. L'organització interna de la xarxa es basarà en un sistema de distribució d'adreces IP automàtic (DHCP) i un servidor de noms secundari intern (DNS). S'haurà de

tenir en compte la possibilitat que cada usuari tingui un certificat digital intern (GnuPG) gestionat per mitjà d'una unitat certificadora local (CA).

- **Requisits d'impressió en xarxa.** Des de cada màquina de l'empresa es podrà accedir a un servei d'impressió centralitzat mitjançant cues (CUPS), on cada màquina client haurà de ser identificada i el treball d'impressió realitzat haurà de ser comptabilitzat.
- **Requisits de servei d'accés per a publicar dades en el sistema web de l'empresa.** L'empresa disposarà d'un sistema d'accés a la informació via un servei de web, i els usuaris del sistema (empleats de l'empresa) podran publicar i/o actualitzar informació de l'empresa, i ells mateixos en seran responsables. Tot això s'haurà de fer amb mètodes d'identificació adequats. S'haurà d'implementar un control d'accés per a aquest servei i un registre de modificacions. Així mateix, s'haurà d'habilitar un sistema de registre d'activitat de servei de web a fi de tenir informació analítica per a l'empresa (des d'on s'accedeix i a quina informació) com des del punt de vista de l'anàlisi de seguretat.
- **Requisits de gestió de les còpies de seguretat i informació dels sistemes.** El sistema informàtic haurà de disposar d'un servei de còpies de seguretat de la informació planificada tenint en compte que aquestes hauran de complir unes condicions d'emmagatzematge físiques de seguretat quan les dades tinguin requisits de confidencialitat. Així mateix, s'han d'integrar totes les dades de seguretat (logs) en un sistema centralitzat perquè l'anàlisi sigui efectiva i real del sistema en conjunt, no només de màquines particulars.

Per als clients de l'empresa, s'oferiran els serveis següents:

- **Requisits del portal web.** El portal d'informació de l'empresa haurà d'admetre serveis complementaris com pot ser la compra de serveis oferts a l'empresa, com ara impressions remotes en dispositius d'alta qualitat o altres serveis amb valor afegit. Per a això, es tindran en compte les possibilitats de petició del servei, la identificació del client, el mode de lliurament i el pagament del servei. S'hauran d'establir models diferents si són clients registrats/habituals o clients nous o ocasionals, ja que els dos tipus de perfils haurien de sol·licitar informació diferent.
- **Requisits del servei d'emmagatzematge d'arxius.** L'empresa oferirà un servei d'emmagatzematge d'informació a clients registrats, per la qual s'haurà de muntar un servei de gestió d'espai (quotes) i accés de clients remots als seus espais de disc (conegut generalment com a FTP *-file transfer protocol-*, però que s'haurà d'implementar com a SFTP *-secure FTP-* per a evitar problemes de seguretat). Aquest servei haurà de ser fiable, per això

el sistema disposarà de còpies de seguretat gestionades per l'empresa de manera separada de les còpies de seguretat pròpies.

- **Requisits del servei de base de dades als clients.** Els clients podran disposar d'un servei de base de dades amb accés remot. L'empresa només proveirà un servei d'allotjament i accés segur a la informació de la base de dades del client, però no serà responsable de la creació de la base ni de les dades emmagatzemades. La seva responsabilitat serà només de servei i seguretat per a l'accés a aquesta base de dades.

D'altra banda, l'empresa s'encarregarà de qüestions d'organització, en concret del manteniment del sistema informàtic, tant del maquinari com del programari:

- **Requisits tecnològics, de manteniment i administració.** L'empresa haurà de contractar un pla de manteniment de maquinari o formar les persones adequades per prestar aquest servei. Atesa la magnitud del projecte, és crucial per al bon funcionament posterior a la implantació i integració de la formació dels administradors i dels usuaris finals que utilitzaran la solució. Aquesta formació es podrà realitzar durant els períodes de prova per a generar confiança i perquè acceptin la nova tecnologia sense preconceptes ni falses expectatives.
En el cas de l'exemple que estem seguint, quant a la base de les aplicacions, el sistema proposat es basarà en GNU/Linux CentOS 5 i altres aplicacions de programari lliure (CUPS, Apache, Postgres, Samba, etc.) i si fos necessari desenvolupar algun servei complementari, s'efectuarà sota el mateix esquema utilitzant llenguatges que els administradors i programadors de l'empresa coneguin (C, C++, PHP, Shell Script, etc.).
- **Requisits d'organització.** L'empresa haurà d'elaborar un organigrama de recursos humans per redefinir les tasques i les responsabilitats en el nou sistema. Aquesta reorganització tindrà per objectiu assignar les noves responsabilitats entre els empleats de l'empresa tenint en compte els seus nous serveis i productes. Així mateix, és convenient que l'empresa generi un llibre blanc que reculli no només la funcionalitat de l'empresa, els seus serveis i productes, sinó també els procediments i les responsabilitats de cada empleat des d'un punt de vista funcional. Aquesta informació la coneixeran tots els empleats i serà on quedaran anotades per escrit les responsabilitats de cada un dels integrants de la plantilla de l'empresa.
- **Requisits de seguretat.** L'empresa, a més, haurà de tenir un document de seguretat que descriu tant les responsabilitats dels usuaris que treballin amb informació sensible, com els procediments que s'han de seguir en cas que es descobreixin problemes relatius a la seguretat (intrusos, pèrdua de confiança o informació, vulnerabilitats del sistema, etc.) a fi d'incloure tots

els passos que han de seguir els responsables davant situacions d'aquest tipus.

2.3. Establiment de requisits

L'objectiu d'aquesta fase serà complementar els requisits definits anteriorment incorporant la informació dels usuaris i directius de l'empresa. És aconsellable que la discussió de la primera fase (usuaris i directius) es realitzi segons els nivells de responsabilitat, ja que es tendeix a parlar amb els directius, de qüestions tècniques o de molt baix nivell, i amb usuaris tècnics, sobre decisions d'empresa. Per a iniciar aquesta tasca, és important dividir el sistema en subsistemes, la qual cosa en facilitarà l'anàlisi i permetrà aclarir els diferents aspectes sobre serveis i recursos que tindrà el sistema.

L'obtenció dels requisits estipulats pels usuaris (directius de l'empresa, usuaris, tècnics, etc.) serà el primer pas en l'establiment de condicions. Durant reunions de treball amb ells, s'han de prefixar i esbossar els criteris mínims, esbrinant quins són necessaris, quins són les expressions de desig i quins són indiferents per a uns o altres usuaris. És convenient una distribució dels requisits en àmbits per a centrar el debat i les discussions, com per exemple en els següents:

- **Funcionals.** Seguir els llaços de moviment, processament i utilització de la informació determinant els serveis necessaris per a això.
- **De prestacions.** Com s'hauran d'optimitzar els recursos i on es trobarà la informació, com s'hi accedirà i quins criteris s'utilitzaran per a fer-ho.
- **De seguretat.** Com afectarà el moviment de la informació a la seguretat d'aquesta i quins elements/serveis s'hauran de tenir en compte en la seva organització.
- **D'implantació.** Com s'organitzarà la informació, on s'emmagatzemarà i qui serà el seu responsable, tant per als serveis com per a l'atenció als clients externs de l'empresa. En determinats casos, és necessari també especificar com serà el transitori entre el sistema d'informació actual i el nou, i com es realitzarà aquest canvi.
- **De disponibilitat.** Com s'han de garantir els serveis i què s'ha d'esdevenir en cas que en falli un.

Els responsables o directius de l'empresa hauran d'establir els procediments d'accés, seguretat, confidencialitat, o si s'escau de negoci, de la informació que es controla i/o manipula a l'empresa.

Els responsables tècnics (els administradors dels sistemes informàtics) hauran d'establir els criteris i/o necessitats per oferir el servei adequat i discutir la prestació de nous serveis. També hauran d'opinar sobre qüestions com interfícies d'usuari, funcionalitat i distribució dels serveis, programació de nous recursos nous, etc. És important tenir en compte els requisits dels usuaris tècnics i aprofitar l'experiència en el sistema.

El pas següent serà l'especificació formal dels casos d'ús, on s'haurà d'incloure una descripció del problema, com els usuaris i responsables interactuaran amb el sistema, quines interfícies utilitzaran i quins procediments seguiran en cas d'errors, fallades o avaries en algun dels components del servei.

Cas pràctic

Exemple de cas d'ús: accés a un servei de suport

L'empresa determina que el sistema de suport del servei d'impressió remota es realitzarà mitjançant el portal de serveis integrats de l'empresa. El client (en el nostre cas extern) accedirà a aquest servei mitjançant una pàgina web en la qual s'haurà de registrar o, si ja ho està, introduir el seu identificador i clau, la qual cosa es validarà en el registre respectiu de la base de dades.

En cas que l'usuari no estigui registrat, es procedirà al registre i s'enviarà per correu electrònic l'identificador i clau d'accés del servei, generats de forma segura. Es notificarà al responsable del servei de suport aquesta nova petició.

Quan el client extern introdueixi la seva pregunta, el sistema enviarà una petició al responsable de servei i aquest l'assignarà a un tècnic després de verificar les qüestions que s'hi refereixen. El tècnic respondrà el client i quedarà registrat quan s'ha realitzat la petició i quin tipus de resposta s'ha donat al problema. Quedarà en mans del responsable del servei la qüestió de si aquesta consulta passarà com a PMF o quedarà com una consulta interna.

Mitjançant la consulta amb els directius de l'empresa, els administradors i els responsables de servei, s'ha de verificar si aquest és el procediment correcte, si tots estan d'acord quant a criteris de qualitat, servei i responsabilitat i si és així, s'haurà de fixar quins elements maquinari i programari seran necessaris per a complir aquests requisits.

S'haurà de tenir en compte que com més casos d'ús es generin, millor quedaran establertes (i a més, per escrit) les necessitats de l'empresa i menys probabilitats hi haurà de tornar enrere durant la implantació a causa de procediments no recollits o oblidats que ens provocarien un increment excessiu en els costos finals. També és necessari recordar que el cap de projectes ha d'escoltar totes les propostes, ser receptiu a les necessitats i intentar fer prevaler criteris de servei, negoci i empresa, de manera que l'usuari per si mateix descarti opcions personals o conveniències d'un departament.

2.4. Definició d'interfícies d'usuari

En aquesta fase de l'anàlisi s'especificarà com seran les interfícies de comunicació, no només la d'usuari-màquina, sinó també usuari-usuari, usuari-responsable i la d'usuari-client extern. En el caso d'usuari-màquina (considerant com a usuaris els empleats de l'empresa per a la qual es desenvolupa)

lupa el projecte), que és on haurà de prestar més atenció el cap de projecte, es realitzarà una especificació tenint en compte diferents perfils d'usuaris, quins tipus de permisos tindran, i de quines capacitats i flexibilitats disposaran, i també quin nivell jeràrquic ocupa cada un i com s'accedirà a informació de serveis vinculats o dependents.

Una vegada descrits completament els perfils d'usuari, s'identificarà quina classe d'interfícies tindran i com accedirà a la informació cada un d'ells.

Cas pràctic

Definició d'interfície d'usuari: interfície del responsable de l'actualització de la base de dades dels clients i empleats

- S'utilitzarà des d'un sistema Unix o Windows?
- És usuari tècnic o administratiu?
- La interfície haurà de ser en mode text o gràfica?
- Els registres de transaccions s'han d'enviar al responsable del servei?
- La validació de l'usuari es realitzarà mitjançant variables de sessió o mitjançant claus incorporades a l'accés de la base de dades?
- La memòria cau de la màquina d'accés ha de ser eliminada una vegada acabada la transacció?

Mitjançant taules de definicions d'aquest estil, es definiran les interfícies per a cada perfil tenint en compte que només ha de tenir els privilegis d'accés a la informació de la qual és responsable.

2.5. Especificació del pla de proves

L'últim aspecte que s'ha de preveure en la fase d'anàlisi és l'especificació del pla general de proves que servirà per a establir si el sistema compleix tots els requisits de les parts involucrades en el projecte (directius, administratius, usuaris, tècnics, etc.).

Normalment, s'acostuma a dividir el pla de proves en diferents nivells que solen ser útils per a desvincular serveis, departaments o unitats funcionals.

- **Proves unitàries del servei:** per exemple, connexió a la base de dades des de perfils administratius o de gestió econòmica.
- **Proves d'integració:** accés de clients a servidors en DMZ complint els criteris de seguretat i protecció de la informació estipulats.
- **Proves de sistema:** verificació de si el servei compleix tots els requisits estipulats per l'usuari/client.
- **Prova d'implantació:** com es realitzarà la transferència d'informació entre el sistema actual i el nou i com s'administraran els transitoris.

- **Proves d'acceptació:** com s'identificaran els usuaris davant el nou sistema i com accediran a la informació i acceptaran les seves noves responsabilitats.

Aquest conjunt de proves és summament crític, ja que permetrà validar el sistema en el seu conjunt tenint en compte des d'aspectes com la funcionalitat, fins a aspectes més subjectius, com la seguretat, el rendiment, la disponibilitat, etc.

Per a cada prova se n'haurà d'estipular l'horitzó, entenent per *horitzó* quins usuaris implica, quins serveis dóna, quins criteris d'acceptació ha de complir la prova per ser superada, quin serà l'entorn de proves (maquinari i programari), com es realitzaran les proves i en quines condicions, etc.

Si bé aquest pas de proves pot ser tediós i poc creatiu, garanteix que el disseny es realitzarà sobre les condicions adequades i que els resultats podran complir els requisits especificats. És per això que cal que el cap de projectes tingui especial cura en la definició de les proves, ja que aquestes seran la garantia, una vegada superades, que el sistema funciona d'acord amb els criteris preestablerts durant les primeres fases del projecte.

Cas pràctic

Especificació de les proves: prova d'integració entre el servei de web i el de base de dades

Permetrà conèixer si un client pot accedir a canviar informació personal desada a la base de dades de l'empresa mitjançant un accés segur per mitjà d'una pàgina web.

- El client només haurà de poder canviar la seva filiació, però no altra informació com el codi de client, informació econòmica, etc.
- L'accés haurà de ser segur.
- S'haurà de comprovar la comunicació entre el servidor web i el servidor de base de dades.
- S'haurà de verificar la impossibilitat de canvis no autoritzats d'informació.
- S'haurà de verificar la comunicació autoritzada entre el servidor web i el de base de dades.
- Es treballarà sobre dades reals i amb usuaris reals del sistema accedint des d'Internet i verificant el xifratge de les comunicacions.
- Es verificaran accessos i elements de seguretat tant dins la intranet (servidor-servidor), com externs (navegador del client - servidor web).

La prova es considerarà com a superada quan s'hagi pogut canviar només la informació desitjada i s'hagin verificat tots els aspectes de seguretat esmentats.

3. Disseny del sistema

L'objectiu de la fase de disseny d'un projecte d'aquestes característiques és obtenir els elements (models i especificacions) definits en l'etapa d'anàlisi (l'etapa anterior), però d'una manera concreta. Bàsicament, s'ha de tenir en compte que en aquesta etapa s'inicia el procés de selecció dels serveis i la seva distribució física, i s'analitza com s'executaran aquests serveis en una arquitectura maquinari. A més, s'hauran de determinar les especificacions de desenvolupament i integració, i també definir l'entorn de proves que cal fer per decidir el bon comportament del sistema i seleccionar quins criteris s'utilitzaran perquè aquestes siguin representatives del correcte funcionament del sistema.

Es poden resumir les tasques en aquesta fase:

- Definició de l'arquitectura del sistema: identificació dels components maquinari, la seva interconnexió, jerarquia del programari, seguretat i privilegis. És a dir, tot el que és necessari perquè el sistema pugui ser configurat i posat en marxa amb garanties.
- Especificacions, requisits de seguretat i estàndards que s'utilitzaran tant en aquesta fase, com en el desenvolupament del sistema.
- Identificació dels subsistemes, requisits d'integració, llicències i funcionalitat.
- Casos d'utilització especificats anteriorment, revisats per a obtenir les condicions de funcionament determinades en les fases anteriors.
- Serveis, components, interfícies, mètodes i classes que s'hauran d'instal·lar en la fase de desenvolupament.
- Assegurar la garantia de l'èxit de la implantació del sistema.

Com es pot observar, aquesta fase, en la qual es determinaran molts conceptes, serà essencial per al projecte i, encara que hi ha algunes tendències de desenvolupament i construcció iteratives, a fi d'obtenir resultats de manera immediata i identificar així errors en el disseny, és important recordar que totes les decisions quant a seguretat, especificacions o estàndards preses aquí facilitaran les tasques futures.

És important identificar aquí els serveis i garantir aspectes legals, ja que aquests poden afectar la funcionalitat i comprometre la utilització del sistema per qüestions no derivades de la seva execució, sinó d'elements que contravenen normes o aspectes jurídics. Aquestes qüestions poden impedir donar els ser-

veis amb legalitat i garantia, no només per a l'empresa, sinó també per als seus clients i treballadors. Cal tenir sempre present la normativa i lleis vigents a posteriori i en cada moment perquè no ens donin problemes més endavant.

3.1. Arquitectura

L'arquitectura del sistema és el primer pas per a identificar els elements maquinari i on s'executaran els serveis. L'objectiu és disposar d'un conjunt de documents i diagrames complets (que continguin tot el nivell de detall necessari i suficient), que siguin comprensibles per a personal no tècnic, com per exemple la direcció de l'empresa, i alhora, que es puguin utilitzar com a base per a aprofundir en el disseny del sistema.

3.2. Definició de nivells d'arquitectura

En el nostre cas utilitzarem la separació funcional i lògica de l'arquitectura, però hi ha altres esquemes per a definir-la:

- **Arquitectura funcional:** descripció a grans trets dels blocs del sistema, sense entrar en detalls i identificant les interrelacions entre ells. Serà útil per a comunicar i verificar amb la direcció de l'empresa i amb el personal no tècnic la visió global del sistema.
- **Arquitectura lògica:** detalla l'anterior i incorpora els detalls de la interacció de cada un dels subsistemes (blocs), que permetran a cada instal·lador i desenvolupador treballar preocupant-se només de la feina encarregada.

Cas pràctic

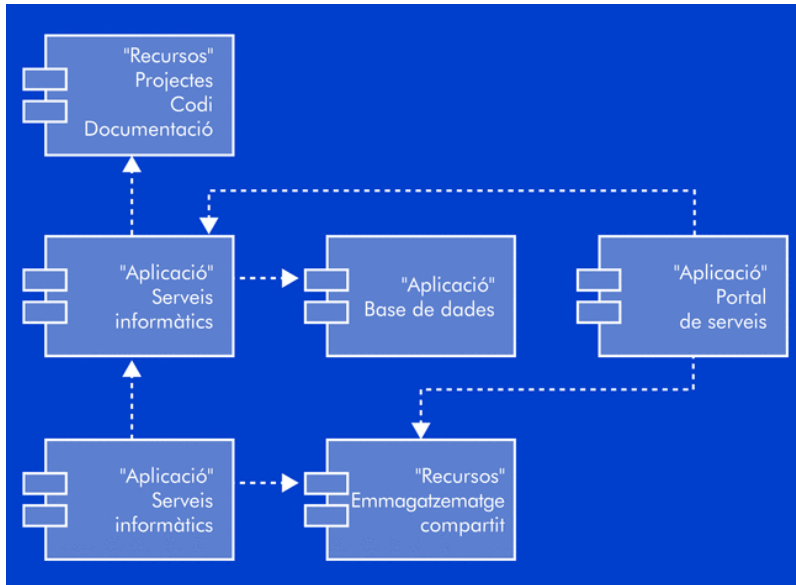
Cas pràctic de definició d'arquitectura: arquitectura global de l'empresa NtEum, SA

Es pot representar l'arquitectura del nou sistema amb notació UML (*unified modeling language*), definida per l'Object Management Group (OMG), i targetes CRC (*classe-responsabilitat-col·laborador* o *class-responsibility-collaborator*) de suport.

Nota

L'UML es pot consultar a <http://www.uml.org> i a l'organització OMG, Object Management Group, a <http://www.omg.org>.

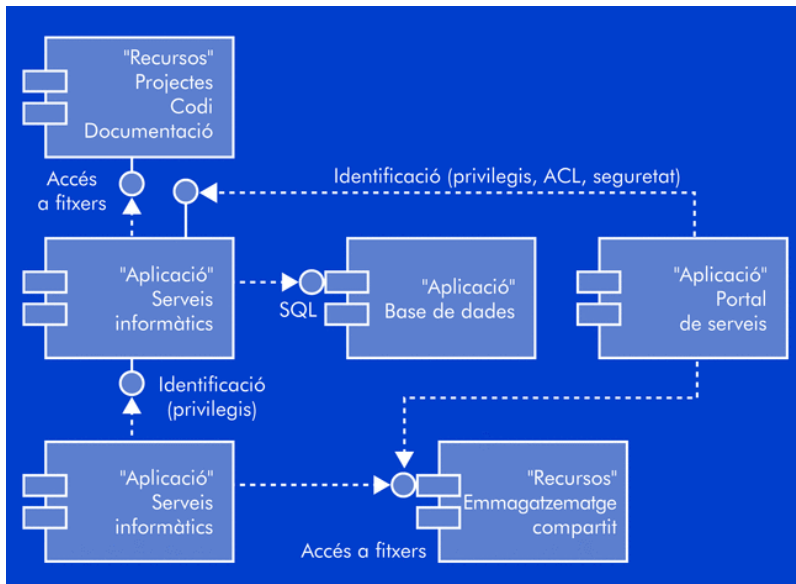
Figura 5. Arquitectura de components



En el diagrama anterior es poden observar a tall d'exemple els components del sistema (una distribució d'ells) i els connectors que els uneixen. Aquests indiquen que algun tipus de comunicació es produeix entre ells i és recomanable en aquest nivell barrejar components tècnics i no tècnics (per exemple, de negoci), els quals hauran d'estar degudament identificats amb un estereotip com <<recursos>> i <<aplicació>>, o d'altres.

Amb aquest diagrama i el consentiment del client, es passarà a aprofundir sobre les interfícies dels components per a obtenir així l'arquitectura lògica del sistema estenent el diagrama anterior i detallant els processos de comunicació.

Figura 6. Arquitectura lògica



Com a suport a la generació del diagrama anterior o per a validar les interfícies, es poden utilitzar targetes CRC com:

Taula 1. Exemple de targetes CRC per al portal de serveis

Portal de serveis	
Ofereix els serveis als clients de l'empresa <ul style="list-style-type: none"> • Web, correu i missatges • Impressió remota • SFTP • Comerç electrònic • Base de dades 	Identificació i fitxers

Ha de figurar en primer lloc el nom del component. En la columna esquerra s'haurà de reflectir tot el que el component fa (serveis que presta en el nostre cas) o també les dades que té. S'inclourà tot el que és de la seva responsabilitat i la informació que ha de mantenir. A la part dreta s'incorporaran els elements amb els quals es relaciona per a portar a terme els serveis indicats a l'esquerra.

Aquestes targetes s'utilitzen generalment en les últimes tecnologies de disseny i planificació del programari i permeten "definir" diagrames de components dinàmicament, utilitzant un espai (per exemple, una taula) per a col·locar les targetes segons el seu grau de comunicació (a prop o lluny del recurs/aplicació), per a aconseguir així una visió global de l'arquitectura lògica durant una sessió de planificació.

3.3. Especificació d'estàndards, normes de disseny i construcció

És important que totes les persones vinculades al projecte (tant les que el desenvoluparan com les que l'implantaran i el mantindran) segueixin unes normes de generació de documentació. S'ha de triar no solament la manera en què es realitzaran els diagrames, amb quines eines i com s'especificaran, sinó també com es generarà tota la documentació. És recomanable utilitzar eines ben conegudes i provades que facin servir formats estàndards, amb això ens assegurem que anys després la informació segueixi accessible en tot moment. Quant al llenguatge, detalls i notació, es recomana utilitzar també estàndards predefinits o en cas d'adoptar-ne un d'intern, és preceptible que el coneguin (o que el seu aprenentatge sigui al més simple possible) totes les persones vinculades d'una o una altra manera al projecte, ja que no solament facilitarà la comunicació amb els diferents grups involucrats, sinó que permetrà la seva reusabilitat i comprensió per part de persones externes o acabades d'incorporar a l'equip de treballadors.

S'han de definir, entre altres coses:

- Eina, format i plantilla dels documents de disseny.
- Notació utilitzada en els diagrames de disseny.
- Recomanacions d'estil, detalls, idioma i format de la documentació tècnica, i també les eines utilitzades per a generar-la i/o modificar-la.

Cas pràctic

Definició de les normes i notacions del projecte d'adequació i renovació dels serveis informàtics de NtEum, SA

És recomanable que els documents creats a partir d'aquest punt, i que seran motiu de revisió per part de diferents persones amb diferents graus de coneixements tècnics, comparteixin característiques i uniformitat, i també un format adequat. En el nostre cas particular d'exemple s'optarà pel següent:

- **Documents de disseny:** aquests documents els han de poder consultar tant el personal tècnic, com altres membres de l'empresa sense formació d'aquestes característiques. S'acorda treballar en format ODT, que és l'estàndard d'Open Office per als documents susceptibles de patir modificacions amb control de correccions i basat en una plantilla predefinida que contingui: el títol del document, el responsable del document, la llista d'autors que hi han intervingut i la data en la qual ho han fet, i el resum executiu dels canvis introduïts (és a dir, el control de versions) indicant el canvi, la data i els autors. Així mateix, s'utilitzarà cada document i es generarà simultàniament en el format PDF per a la seva comunicació i publicació a persones vinculades al projecte, però sense capacitat d'introduir canvis.
- **Diagrames de disseny:** per a aquests s'utilitzarà la notació UML.
- **Documentació tècnica:** aquesta serà la que probablement haurà de passar més revisions i controls i tindrà enllaços a les eines utilitzades, interfícies de programació (API), configuració i exemples, etc., per la qual cosa es recomana utilitzar un format tan flexible com sigui possible i integrable a les eines utilitzades. Per les característiques del projecte, és recomanable utilitzar XML per a aquest tipus de documentació aprofitant estàndards ja establerts com poden ser DocBook (<http://www.docbook.org> o també <http://www.oasis-open.org/docbook/>) a fi de facilitar la interacció i comprensió. Amb aquest estàndard es podrà:
 - Jerarquitzar un document i estructurar-lo en diferents fitxers, els quals es podran revisar independentment.
 - Fàcil inclusió de referències a altra informació com arxius, URL, figures, etc.
 - Generació simple d'altres formats per a la seva visualització com PDF i HTML.
 - Com que es treballa amb XML, els fitxers són independents de l'editor utilitzat.
 - Permetrà obtenir documentació inclosa en el codi font de les aplicacions i incloure-la automàticament en la major part de situacions.

És important destacar que aquest tipus d'eines són d'última generació i pot ser que algunes de les persones involucrades en el seu desenvolupament, implantació o manteniment no n'estiguin al corrent. Tanmateix, s'haurà de tenir en compte aquesta situació i prendre les mesures correctores necessàries, ja que la seva utilització redundarà en una millor accessibilitat a la informació i millorarà el cicle de vida de manteniment del sistema. I atès que tota la informació la tindrem disponible, serà molt fàcil la seva actualització i revisió.

3.4. Identificació de subsistemes

Per a reduir la complexitat de dissenyar-ho tot a un nivell de detall determinat, es pot dividir el sistema en seccions o subsistemes agrupats per funcionalitat (més efectiu des del punt de vista tècnic) o per organització (més comprensible des del punt de vista no tècnic); d'aquesta manera es facilita la seva comprensió, revisió i utilització.

Per a realitzar una divisió funcional, és important tenir en compte:

- Funcionalitat comuna o relacionada per interacció i/o execució.

- Accés a recursos compartits o validats per la mateixa identificació (dades o sistemes).
- Integració en una interfície d'usuari única malgrat que els serveis puguin estar físicament distribuïts.
- Millores en la utilització de la informació, sobretot quan aquesta ha de tenir nivells elevats de seguretat o ser de gran volum.

Cas pràctic

Identificació de subsistemes

En el nostre cas particular, analitzant segons la funcionalitat, s'obté:

- Subsistema serveis informàtics (integrador model 3-tier respecte a la base de dades i mòdul de seguretat)
- Subsistema portal de serveis
- Subsistema serveis interns
- Subsistema de base de dades
- Subsistema d'emmagatzematge compartit
- Subsistema d'accés a la documentació

Aplicant un criteri uniforme, seria desitjable definir una única interfície integrada per a obtenir així un únic subsistema (no aplicable en aquest cas) o sota un criteri d'optimització de recursos, intentar reduir els subsistemes al mínim possible, però sense perdre definició o separació de funcions en el sistema real.

Amb això es disposa d'una visió (mapa) de l'arquitectura del sistema global, que una vegada verificat i validat, permetrà aprofundir en cada un dels seus components per mitjà dels casos d'ús del sistema.

3.5. Casos d'ús reals

Una vegada identificats els subsistemes, és el moment d'analitzar els casos d'ús establerts en la fase d'anàlisi i determinar les operacions que s'han de desenvolupar de cada un d'ells.

A partir dels escenaris escollits en la fase d'anàlisi, es determinaran quins subsistemes estan inclosos en cada un d'ells i es dissenyarà el seu funcionament tenint en compte:

- L'àmbit en el qual s'apliquen considerant no només aspectes tecnològics.
- Les situacions particulars de cada cas d'ús (excepcions).
- Detalls vinculats a la implementació identificats en aquesta fase. En aquest nivell de disseny s'han de tenir en compte tots els aspectes vinculats a la seguretat global del sistema.
- Restriccions aplicables a la interfície.
- Nous requisits o funcionalitats necessàries i no previstes en l'anàlisi inicial.

Si el sistema que es desenvolupa inclou bases de dades, com és el nostre cas d'estudi, s'haurà de definir la seva estructura (taules, mètodes, sentències SQL, validació, etc.).

És important durant aquesta fase revisar tots els requisits del sistema establint qui els servirà (incloent-hi aspectes de seguretat, interacció i vinculació). Conseqüentment, també s'obindrà el disseny de totes les proves que asseguraran el correcte funcionament del sistema durant el desenvolupament i les condicions sota les quals s'implantarà.

3.6. Revisió de casos d'ús per subsistema

Per a cada cas d'ús s'ha de definir el següent:

- Subsistemes i actors que hi intervenen.
- Informació que intercanvien.
- Quin perfil d'usuari està habilitat en aquest cas i quina funcionalitat està habilitada.

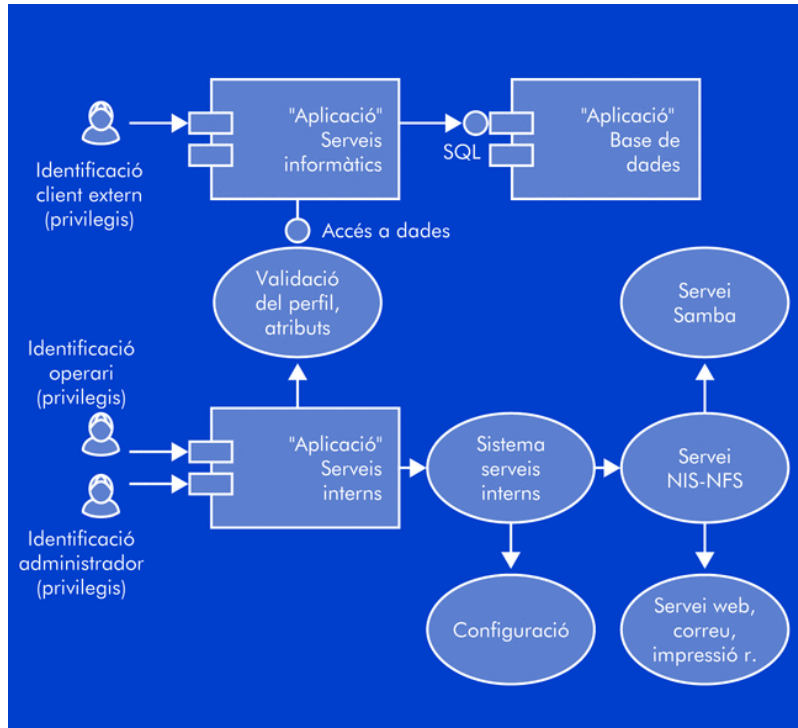
La definició de l'intercanvi d'informació permetrà verificar i detallar les interfícies de cada subsistema tenint en compte tots els casos d'ús en què intervenen i anar completant, així, la definició de subsistemes realitzada en fases anteriors.

Cas pràctic

Diagrama UML de revisió de casos d'ús amb components implicats del subsistema serveis interns

A continuació es desenvoluparà la definició de subsistemes realitzada amb la revisió de casos d'ús implicats en serveis interns.

Figura 7.



Com a complement d'aquest esquema, s'han de definir els missatges que hi intervenen, els fitxers als quals es té accés per a lectura i escriptura i sota quins perfils s'esdevenen aquests canvis, a fi de detallar cada una de les el·lipses del diagrama amb cada una de les fletxes, és a dir, cada element amb els seus atributs, mètodes, etc.

3.7. Especificacions de desenvolupament i proves

A partir d'aquest punt, s'estarà en condicions d'establir les característiques del sistema en els aspectes següents:

- Tecnologia: maquinari, programari i comunicacions.
- Serveis i eines de desenvolupament: identificació dels serveis i quin programari els prestarà, adequació d'aquests a l'entorn tecnològic i eines que permetran donar la funcionalitat requerida.
- Eines de documentació.
- Seguretat i protecció: nivells, separació i control.
- Restriccions tècniques.

Amb tot això s'ha de definir el conjunt de proves necessàries que s'han de realitzar una vegada implantat el sistema, les quals s'han de definir com a proves unitàries, amb el mínim nivell de dependència possible per a permetre un desenvolupament, implantació i prova per components. Per exemple, en

la prova de l'NFS, no serà necessari provar-ho amb tot el sistema en primer nivell, es podran fer proves de muntar recursos locals i amb només una màquina client per als diferents perfils d'ús. Serà necessari també definir una prova global (d'integració) per a provar més endavant el sistema d'NFS en tota l'arquitectura, però amb la certesa que el servidor funciona i està parcialment validat.

L'especificació de les proves unitàries es pot dividir en:

- **Proves de caixa negra:** es considera el component o servei des del punt de vista funcional, analitzant les seves entrades i sortides i comparant les seves respostes amb els resultats esperats.
- **Proves de caixa blanca:** es considera el component o servei com una estructura amb una seqüència lògica d'esdeveniments i es comprova la validesa d'aquesta, els missatges/esdeveniments incorrectes, etc.

Normalment, s'utilitzarà una combinació dels dos tipus de proves adequades per a cada servei/component. És recomanable invertir el procés tradicional de realitzar les proves dels components una vegada implementats per proves parcials (deixant per al final les proves d'integració). D'aquesta manera es poden provar els serveis i validar (parcialment) la seva funcionalitat.

En un projecte d'aquestes característiques, és important considerar que el desenvolupament d'una única prova pot ser molt complex i tenir un cost elevat a causa de la quantitat de requisits, perfils i restriccions que es manifesten. A més, una única prova pot tenir l'agreuja de deixar interrelacions o elements sense provar (atesa la seva complexitat) o introduir-hi efectes col·laterals, que després poden afectar els resultats obtinguts.

És important considerar també que l'apartat de proves amb freqüència provoca retards en el projecte. Aquest enfocament de proves unitàries (parcials) disminueix aquest efecte i optimitza el desenvolupament, ja que les proves s'enfoquen a obtenir resultats (satisfactoris) del subsistema desenvolupat o implantat.

Cas pràctic

Especificacions de desenvolupament

En el nostre cas pràctic, tenint en compte els serveis interns, les especificacions de desenvolupament serien:

- Servei NIS: integració de tots els usuaris (excepte el *root*) per mitjà d'ordinadors basats en GNU/Linux. Els ordinadors basats en Windows tindran usuaris locals amb accés al sistema Samba validat en un sistema GNU/Linux.
Documentació:
http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/html_single/NIS-HOWTO.html
- Servei NFS: nivell 3 amb exportació només per a les màquines del sistema local sense inclusió de convalidació de l'usuari *root* en el sistema importat.
Documentació:

http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/html_single/NFS-HOWTO.html
<http://www.nfsv4.org>

- Servei de web: servidor Apache-Tomcat i autenticació i seguretat (SSL). Configuració segura de directoris, seguiment d'enllaços i accés per servidors virtuals.
 Documentació: <http://apache.org/>
<http://www.php.net/http://www.postgresql.org/>
- Servei de xarxa: gestió d'IP mitjançant DHCP, intranet basada en adreces privades amb accés exterior per mitjà de NAT, servei de DNS local, serveis segurs per mitjà de Xinetd, SSH, i Iptables.
 Documentació:
<http://tldp.org/HOWTO/HOWTO-INDEX/howtos.html>
- Servei de monitorització: basat essencialment entre Nmap, Nessus, Snort, Logcheck i Tripwire.
 Documentació: <http://www.insecure.org/nmap/>
<http://www.nessus.org/>
<http://www.snort.org/>
<http://sourceforge.net/projects/logcheck/>
<http://www.tripwire.org/>
- Servei de configuració: basat en Webmin i Linuxconf.
 Documentació: <http://www.webmin.com/>
<http://sourceforge.net/projects/linuxconf/>
- Servei d'impressió remota: per mitjà de servei d'impressió CUPS amb validació d'usuari. Control local de recursos i comptabilitat d'utilització per mitjà de la base de dades.
 Documentació: <http://www.cups.org>
- Servei de correu electrònic: servei basat en Exim amb servidor segur d'IMAP. Control de quotes de correu i integració amb un servei de correu web basat en Squirrel.
 Documentació: <http://www.exim.org>
<http://www.washington.edu/imap/>
<http://www.squirrelmail.org/>
- Servei d'ofimàtica: accés a recursos compartits basats en OpenOffice per a edició de text, fulls de càlcul, dibuix o presentacions.
 Documentació: <http://www.openoffice.org/>
- Instal·lació d'aplicacions del client: en aquest cas, s'instal·laran les aplicacions existents del client (comptabilitat, gestió i control) validant la seva funcionalitat.

S'hauran de desenvolupar les interfícies i les pàgines web que permetin accedir als recursos integrant-les totes sota un aspecte corporatiu i unificat de l'empresa. La gestió i control s'hauran de sintonitzar per als administradors futurs reduint la complexitat (per exemple, instal·lant eines de configuració i/o gestió gràfiques per a les tasques més rutinàries).

Per a la codificació de pàgines web, s'utilitzarà PHP amb validació local per JavaScript en el cas dels formularis, i en la banda del servidor, si és necessari, es farà servir Perl (www.perl.org) com a llenguatge de suport per a les tasques més complexes.

A continuació s'hauran d'enumerar les proves unitàries, extreptes de les funcionalitats i interfícies del sistema:

- Connexió, desconnexió dels usuaris segons els serveis/perfils.
- Configuració automàtica de la xarxa (DHCP), control d'accés a Internet per NAT i verificació d'accés a serveis des de Xinetd. tots els serveis abans esmentats.
- Control d'integritat i seguretat de les dades intercanviades entre cada servei.
- Control de configuració dels serveis (WebAdmin).
- Accés al sistema de compartició de fitxers sobre màquines Windows amb criteris de seguretat i control.
- Accés al sistema d'impressió (CUPS) d'acord amb els criteris de seguretat i control d'accés.

Per a cada una d'aquestes proves, s'han de definir els paràmetres o informació d'entrada i els seus possibles resultats o informació de sortida. Això permetrà al grup de desenvolupament programar cada una de les proves particulars triades dins el marc de treball especificat.

3.8. Requisits d'implantació

Els requisits d'implantació seran els que s'hauran de complir per a cada component o subsistema quan es treballi en l'entorn real conjuntament amb la resta de subsistemes. No solament es considerarà l'entorn tecnològic, sinó que també es tindran en compte tots els tipus d'usuaris que pot tenir el subsistema. Això permetrà definir el pla de formació adequat als diferents nivells perquè els usuaris finals del subsistema puguin interactuar i obtenir les prestacions per a les quals va ser definit el servei.

Des del punt de vista tecnològic, s'han de determinar les condicions de l'entorn en el qual s'implantarà el nou subsistema, i les condicions de funcionament per a determinar que aquest subsistema funcionarà a ple rendiment sense esgotar els recursos, amb les condicions de seguretat adequades, i sense afectar els altres subsistemes amb els quals interrelaciona.

El document que reculli els requisits d'implantació haurà d'incloure a més els punts següents:

- Gestió de la documentació: qui i com s'hi tindrà accés, i sota quines condicions.
- Necessitats de migració de serveis/usuaris.
- Formació dels usuaris, administradors i clients per a les noves eines/serveis.
- Necessitats del maquinari i programari bàsic.
- Necessitats de comunicació.
- Nivells de seguretat i control d'implantació.
- Recomanacions en casos de contingència, emergència i procediments de recuperació.
- Recomanacions d'integració global.
- Recomanacions d'anàlisi d'utilització de recursos i possibilitats d'ampliació.

Aquest document es tindrà en compte en la fase d'implantació i hi poden figurar altres qüestions com procediments de seguretat addicionals, proves locals i des d'Internet, procediments de bloqueig o també qüestions vinculades a intents d'accés no permesos a dades i/o recursos.

Cas pràctic

Requisits d'implantació per al sistema de serveis interns

La implantació s'haurà de realitzar sota dos punts de vista: els usuaris del subsistema (tots els tipus existents) i el tecnològic i els seus recursos.

Des del punt de vista de l'usuari:

- Definició del responsable de seguretat i administració del subsistema de serveis interns. Aquesta figura tindrà com a responsabilitats:
 - Conèixer el funcionament del subsistema tenint accés a tota la documentació.
 - Definir les polítiques tant d'altres, baixes, com de bloqueig d'usuaris, recursos i serveis.
 - Comprovar i monitoritzar el correcte funcionament, i també verificar les polítiques de seguretat amb un pla prèviament especificat i consensuat.
 - Analitzar els riscos i tenir plans de contingència per a cada un d'ells, i també les actuacions que poden dur a terme els equips de treball.
- Atesa la vital importància d'aquest subsistema per a l'empresa, és important disposar d'un grup de treball per a cobrir tots els torns i definir si els serveis quedaran en funcionament permanent o si només s'hi podrà accedir en franges horàries predeterminades. només s'hi podrà accedir en franges horàries predeterminades.
- Per a la resta dels usuaris, es descriuran les tasques d'acord amb cada perfil i es determinarà què és el que poden fer i a on poden accedir per a realitzar l'activitat diària que habilita el seu perfil. A més, s'han d'especificar els canals de comunicació per a l'enviament d'incidències i com es fa la comunicació d'errors o les peticions d'activitats i accions necessàries no previstes en el seu perfil actual.

Des del punt de vista tecnològic, la implantació del subsistema serveis interns tindrà un gran impacte sobre la forma de treball de l'empresa, tant pel que fa a recursos, com a adaptació i formació del personal existent. Des del primer moment és important tenir en compte aquesta situació i aprofitar-ho per a vèncer les reticències inicials dels usuaris, amb la finalitat que adquireixin confiança amb el nou sistema. Es recomana organitzar sessions informatives preparatòries per a evitar el recel amb el qual una persona s'adapta a un nou entorn tecnològic i fer que canviïn els seus hàbits en forma de col·laboració i no d'imposició.

S'haurà de preveure una sèrie de qüestions com a requisits tecnològics d'implantació que milloraran els aspectes esmentats:

- El maquinari ha de funcionar de manera transparent per als usuaris locals i amb bons índexs de prestacions millorant el rendiment actual almenys en un factor de 2 (recomanable 5).
- Els usuaris han d'estar donats d'alta amb anterioritat perquè puguin provar els nous sistemes i anar-s'hi habituant (sempre que sigui possible per qüestions d'espai o reutilització del maquinari).
- Els comptes de correu han d'estar migrats i durant un quant temps mantenir la seva duplicitat.
- Els arxius s'han de muntar de manera que l'usuari ubiqui els seus arxius de manera anàloga al sistema anterior.
- Les pàgines web han de ser totalment funcionals i operatives, i també les interfícies d'usuaris, als serveis i les aplicacions utilitzades pels usuaris de l'empresa.
- Ha d'estar provada i verificada la interrelació amb els altres subsistemes des del punt de vista de les comunicacions i la coherència de les dades.

En aquest apartat s'hauran d'incloure totes les qüestions relatives a la implantació del subsistema en qüestió per a evitar situacions que després en les proves d'integració pu-

guin generar retards o modificacions per falta de compatibilitat en la interacció dels subsistemes. És important tenir en compte una seqüenciació d'implantació, ja que s'haurà de disposar d'altres subsistemes implantats anteriorment perquè l'actual pugui funcionar totalment.

4. Desenvolupament

L'objectiu principal d'aquesta fase és la construcció ordenada del sistema del qual s'ha avaluat la viabilitat anteriorment. En les metodologies tradicionals, el desenvolupament es produeix quan s'han superat satisfactòriament les fases anteriors, però mètodes més recents aconsellen iniciar aquesta fase al més aviat possible, ja que això permet més agilitat en el cicle de vida del projecte en el sentit que permet disposar de prototips per a ser avaluats abans que amb la metodologia clàssica. Alguns autors denominen aquesta metodologia *evolutiva* i és aconsellable, si el cap de projecte ho considera convenient, començar-la de manera concurrent amb la fase d'anàlisi.

No obstant això, sempre és necessari adaptar les metodologies a les necessitats del projecte, utilitzant submodels (d'entre altres de més extensos) que s'adaptin millor a la realitat que el que es desenvolupa.

Generalment, és molt productiu que part dels desenvolupadors s'involucrin en el disseny participant, per exemple, en el desenvolupament d'un subsistema i en el disseny d'un altre. Això permetrà una alta motivació i una gran qualitat en la solució desenvolupada, i que siguin capaços de detectar les necessitats d'implantació o errors de disseny que altrament s'haurien postposat fins a la primera revisió, amb els possibles problemes afegits que això implica (retards, modificació d'altres subsistemes interconnectats, etc.).

S'ha de prestar molta atenció perquè aquesta dualitat dissenyador-desenvolupador sigui analitzada pel cap de projectes detalladament, ja que és necessari que les persones tinguin disponibilitat i puguin aportar els seus coneixements en tots dos sentits. Molts fracassos en grans projectes provenen del fet que tots els membres de l'equip implicats són en totes les fases, la qual cosa provoca vicis en el raonament, disseny o desenvolupament. L'ambient de treball basat en la crítica constructiva és el més adequat perquè un projecte tiri endavant en un grup organitzat de treball.

Independentment d'aquestes qüestions, hi ha un conjunt d'aspectes que concerneixen el desenvolupament i que s'han de definir a l'inici d'aquesta fase. S'ha de planificar l'inici i el final del desenvolupament sincronitzant les diferents activitats del projecte que s'hauran d'implantar al final de la fase (incloent-hi no solament el mateix desenvolupament, sinó també la instal·lació, les proves, la documentació, els plans de formació, etc.).

4.1. Planificació de les activitats d'integració del sistema

En arribar a aquest punt, es disposa d'informació sobre què necessita ser desenvolupat, quins components programari i maquinari s'utilitzaran i com s'integraran en el sistema actual, quines eines i/o serveis es faran servir, en quin entorn, etc.

Les activitats de desenvolupament que permetran arribar a l'objectiu plantejat seran:

- Concretar les versions del programari o components programari (inclosos llibreries, mòduls, classes, etc.) que s'utilitzaran.
- Estudiar aquests components i realitzar la seva implantació en l'entorn de desenvolupament.
- Desenvolupar les proves unitàries.
- Desenvolupar, instal·lar i configurar els components necessaris.
- Realitzar la documentació.
- Planificar la formació als usuaris del sistema.
- Realitzar les proves d'integració.
- Validar i aprovar el sistema.

L'objectiu final d'aquesta fase és l'aprovació del sistema perquè es pugui implantar. En molts casos (com l'exemple tractat aquí) el desenvolupament a vegades es realitza sobre el mateix maquinari que després es posarà en producció en la fase d'implantació. Aquesta forma de treball és adequada quan són projectes nous que es realitzen dins l'empresa en la qual s'implantarà i es pot disposar d'aquest equipament anteriorment.

Les activitats s'han de planificar per a seguir un seqüència de passos que comportin la finalització del projecte d'una manera ordenada fins a la seva aprovació (fase d'implantació, apartat d'acceptació del sistema). L'ordre de les activitats no necessàriament ha de ser seqüencial, i el seu grau de simultaneïtat estarà fixat pel nombre de desenvolupadors, el seu perfil i la diversitat de recursos de què es disposi. S'ha de tenir en compte que hi ha activitats que són perfectament concurrents. Un exemple podria ser el desenvolupament d'un subsistema i la seva documentació, mentre que hi ha un altre tipus de situacions en les quals és millor una seqüencialització de les tasques, ja que hi ha

relacions de precedència entre elles. Aquestes relacions poden ser complexes i l'experiència del cap de projecte permetrà trobar el camí òptim per a la seva resolució.

Cas pràctic

Planificació d'adequació i renovació de la xarxa i serveis informàtics per a l'empresa NtEum, SA

Un mètode adequat per a fer aquest tipus de planificació són els diagrames de Gantt. Normalment, es procedeix incloent una durada estimada de les tasques i la vinculació entre elles, marcant les diferents fites i posant de manifest la sincronització i la seqüenciació entre les diferents activitats.

Aquest diagrama pot ser revisat a mesura que avanci el projecte per a incloure les diferències amb la realitat fent-ne versions, les quals serviran, una vegada hagi finalitzat el projecte, per a analitzar les problemàtiques que han impedit complir la primera estimació de temps i extreure'n experiència per a futurs projectes.

A continuació es mostra un exemple de tres gràfics obtinguts amb una eina de planificació de projectes: el diagrama de Gantt, el diagrama PERT (diagrama orientat al termini d'execució o *program evaluation and review technique*) i el calendari.

Figura 9. Diagrama de Gantt

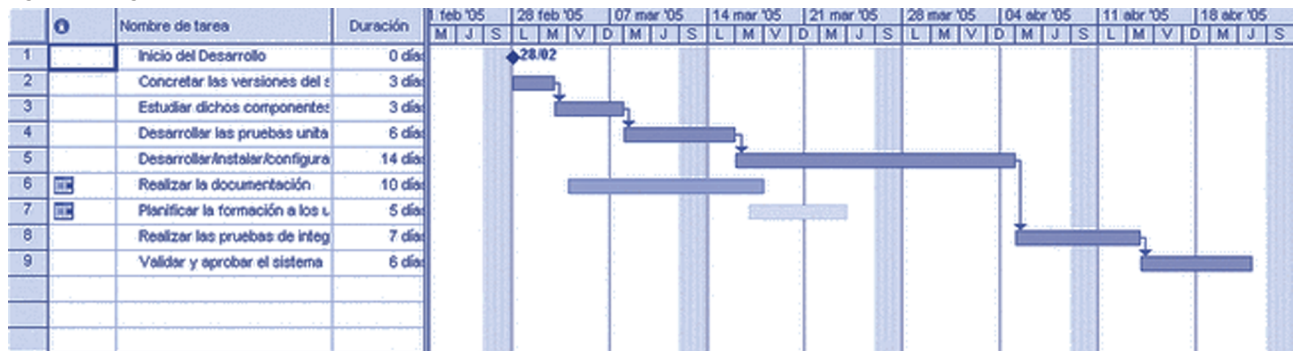
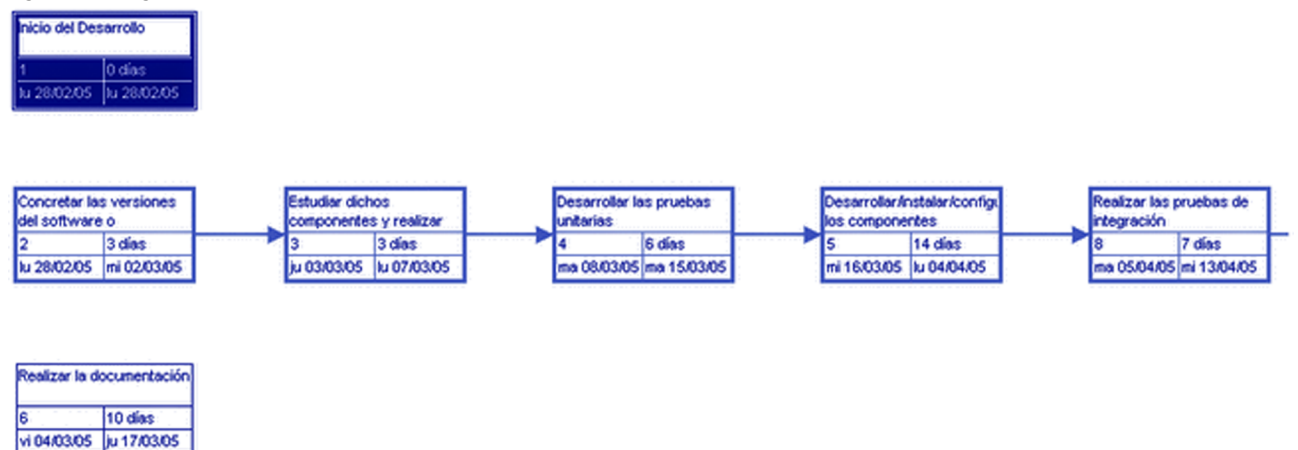


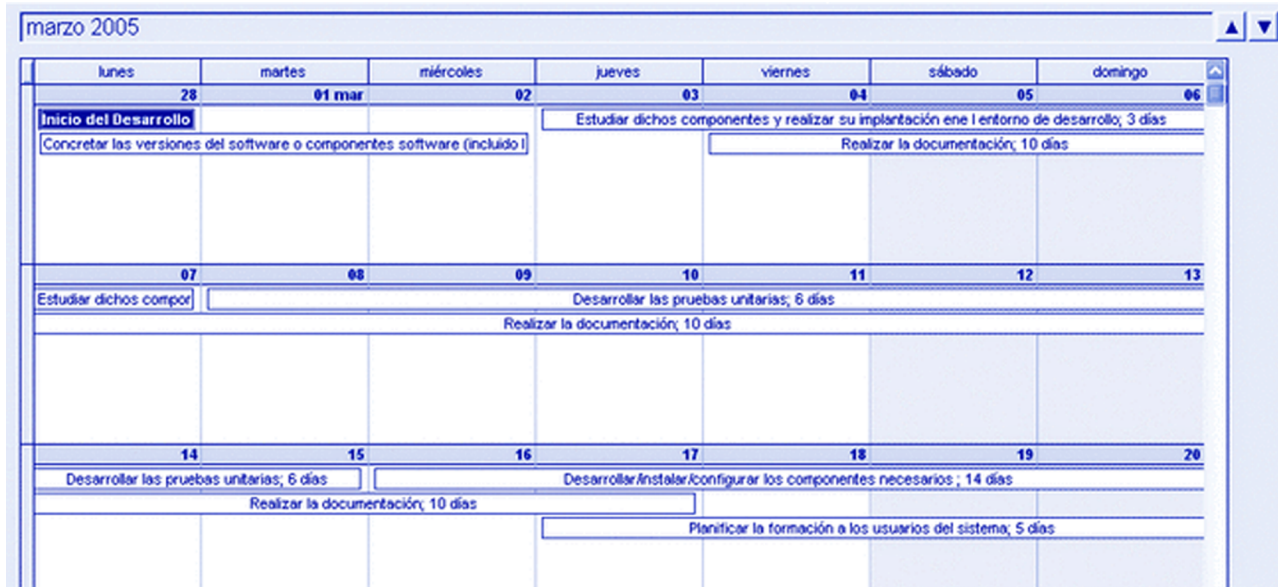
Figura 10. Diagrama de PERT



En aquests gràfics també es pot incorporar el seguiment dels recursos, de manera que es pugui controlar que no hi ha sobreassignacions o problemes derivats de la planificació sobre els recursos (humans i físics).

Aquests diagrames permetran a cada un dels actors veure la seqüència de fites i prendre les decisions oportunes per a aconseguir l'objectiu final del projecte. Aquests gràfics es poden complementar amb un document (plantilla) que en reflecteixi l'estat d'una manera periòdica:

Figura 11. Calendari



Taula 2

Adequació i renovació de la xarxa i serveis informàtics per a l'empresa NtEum, SA	
Data del report	dd/mm/aaaa
Data de finalització	dd/mm/aaaa estimada original dd/mm/aaaa estimada actual
Canvi des de l'últim report	+/- dd dies
Elements pendents de desenvolupament	<i>n</i> defectes, <i>m</i> funcionalitats
Elements pendents d'aprovació	0 defectes, <i>j</i> funcionalitats
Elements tancats	<i>l</i> funcionalitats
Recursos usats durant aquest període	Persona1: <i>j</i> hores Persona2: <i>i</i> hores Persona3: <i>k</i> hores
Resum de l'estat	El projecte avança segons el que s'ha planificat
Documents relacionats	Pla de projecte Disseny de mòduls
Estat detallat	

El desenvolupament d'aquesta setmana s'ha concentrat en la tasca X...
S'ha realitzat aproximadament el 30% del projecte i hi ha un retard de dos dies respecte a la data prevista...

Adequació i renovació de la xarxa i serveis informàtics per a l'empresa NtEum, SA	
Control de riscos	Problemes amb les versions de la llibreria XX utilitzada en la identificació dels usuaris Verificar subsistemes
Activitats programades	Activitat... Esdeveniment... Accions...
Actualització de la planificació	Es pot realitzar una còpia del diagrama de Gantt de la setmana

El cap de projecte haurà d'emplenar per a cada taula els valors relacionats amb les dates (dd/mm/aa) i els valors particulars de cada fila de la taula (*i, j, k, n, m, X, XX*, etc.).

4.2. Com es tria la llicència més adequada

Tant si es tracta d'un projecte intern com d'un projecte amb fins comercials, s'ha d'escollir la llicència dels components i mòduls que s'utilitzaran o desenvoluparan a l'inici del procés. Més encara quan s'integren components programari o serveis de diferents desenvolupadors que poden tenir diferents tipus de llicències, ja que aquesta pot condicionar la llicència final del projecte a desenvolupar.

La llicència escollida tindrà repercussions sobre els fitxers de codi font del nostre programari, ja que han de fer menció de les parts que són propietat d'altres desenvolupadors. S'haurà d'esmentar en la documentació la llicència escollida tenint en compte que en projectes open-source generalment s'han de publicar totes les llicències i s'han de mantenir les llicències de les parts que es fan servir de terceres persones.

Si el projecte es comercialitza sota una llicència pròpia, s'ha d'assegurar que els components, les eines i el codi resultant ho permeten perquè això no impliqui problemes legals per al client final. Aquest haurà de ser informat dels drets i deures respecte a la llicència del producte que acaba d'adquirir. La llicència també afecta els plans de manteniment i suport del producte i s'han de tenir en compte a l'hora d'escollir-la.

El cap del projecte i el director responsable s'han de concentrar en les incompatibilitats dels diferents models de llicència existent i han d'especificar l'escenari concret en el qual es desenvoluparà el seu programari identificant les possibles alternatives: codi propietari i llicència comercial, codi parcialment propietari i llicència comercial (implica tenir llicències d'explotació), codi lliure i llicència

comercial, codi lliure i llicència lliure. S'hauran de resoldre, a més, preguntes relacionades amb els drets d'autor, la garantia i el suport llegint detingudament les especificacions de cada llicència en cada cas.

Cas pràctic

Elecció de les llicències de desenvolupament

En l'exemple que estem descrivint serà un desenvolupament basat en codi obert d'utilització interna, per la qual cosa la llicència no tindrà efectes sobre el model de negoci de la companyia. Això, tanmateix, no evita que no s'hagin de posar les referències adequades en el codi desenvolupat ni en la documentació. Les possibles alternatives són les següents:

- Llicència propietària: si no es distribuirà el sistema, es poden desenvolupar sota llicència propietària.
- Llicència BSD: permet mantenir els drets d'autor (*copyright*) sobre el desenvolupament i és coherent amb les llicències de la resta dels components (les quals es basen en BSD i GPL). Aquesta llicència no obliga a distribuir el codi font final i permet al destinatari d'aquest programari la seva utilització, còpia, modificació, redistribució i venda. Alhora també permet la integració amb un altre producte de llicència propietària.
- Llicència GPL: permet mantenir els drets d'autor sobre el desenvolupament i és coherent amb la resta de llicències del nostre projecte (BSD i GPL). Però a diferència de la BSD, obliga a distribuir el codi font i impedeix la seva futura comercialització sota llicència propietària.
- Llicència LGPL: és pràcticament igual que la GPL, però permet que programari amb aquesta llicència sigui integrat en programari propietari. Un exemple clàssic és la llibreria C de Linux (amb llicència LGPL).
- Altres llicències per a codi obert: Apache (basada en BSD), Mozilla (MPL i NPL de Netscape).

Ateses les característiques del nostre projecte, ens decantarem per una llicència LGPL, ja que el projecte és més d'integració que de desenvolupament i satisfà les expectatives d'utilització del client.

4.3. Entorn de desenvolupament

L'objectiu d'aquesta tasca és traslladar el programari desenvolupat o integrat a l'equip de producció real, on haurà de prestar el servei. Aquesta tasca implicarà més o menys dificultats si l'equip de desenvolupament difereix de l'equip d'implantació. Aquest punt s'haurà de tenir en compte en el moment de realitzar les proves, ja que es poden tenir resultats diferents si s'ha implementat el programari en un sistema molt diferent del sistema en què després ha de funcionar.

Cas pràctic

En el nostre cas particular, com que es tracta d'un sistema nou, els equips de desenvolupament i implantació coincideixen, per la qual cosa, una vegada finalitzat el desenvolupament, la posada en producció serà gairebé immediata.

En un sentit ampli i detallant les activitats descrites en la planificació del desenvolupament, s'ha de tenir en compte:

- Preparació de l'entorn de generació i desenvolupament

- Generació del codi dels components
- Instal·lació dels serveis especificats
- Execució de proves unitàries
- Execució de proves d'integració

Seguint les recomanacions de les fases d'anàlisi i disseny, el seguiment i l'assistència al desenvolupament del codi facilitarà al grup de treball la seva consecució amb èxit i en els temps previstos. S'ha de tenir en compte que les eines tipus RAD (*rapid application design*) faciliten enormement el treball dels desenvolupadors, però que es necessita un període d'aprenentatge perquè la seva utilització sigui eficient.

Si els desenvolupadors han participat en el disseny, el temps d'estudi i distribució de les tasques serà fàcil i eficient, ja que es reduirà el temps necessari de preparació i s'imprimirà més ritme al desenvolupament del projecte.

En cas contrari, si les fases de disseny i desenvolupament han estat desvinculades entre si, fins que no finalitzi l'etapa de disseny no podrà començar la de desenvolupament i es necessitarà un quant temps extra perquè els desenvolupadors analitzin els requisits i en realitzin un estudi. Si s'encavalquen parcialment les fases de disseny i desenvolupament, es pot caure en situacions no desitjades, per exemple, interrompent el desenvolupament d'un subsistema perquè d'un altre, amb el qual està vinculat, encara no s'ha aprovat el disseny. A més, aquesta tàctica presenta un risc afegit ateses les revisions constants que s'hauran de fer del codi desenvolupat quan s'hagi d'integrar globalment.

Els resultats de les proves unitàries dissenyades i desenvolupades anteriorment són l'indicador més rellevant del ritme del projecte. Aquests permetran conèixer en tot moment si la planificació és l'adequada o si es desvia de les previsions inicials reflectides en la planificació acordada.

4.4. Documentació

Una vegada arribats a aquesta etapa, és necessària l'elaboració de la documentació d'usuari, tant d'administració com d'explotació. Tenint en compte les decisions de fases anteriors respecte al format, eines de generació i disponibilitat, s'ha de desenvolupar la seva estructura i contingut.

És important esmentar que aquesta tasca s'ha de realitzar segons la planificació realitzada en les primeres fases del projecte (diagrama de Gantt) i que no necessàriament ha d'anar al final del desenvolupament, si bé és cert que s'haurà

de completar al final d'aquesta etapa (en algunes empreses la documentació s'emplena després d'haver realitzat un prototip sintètic del codi però abans d'escriure el codi final).

És important que aquest tipus de recurs, que després serà utilitzat per diferents usuaris de perfils molt variats, sigui consensuat per a trobar l'estil i profunditat adequats al lector. Sol passar que la documentació no tècnica i escrita per personal tècnic solament és intel·ligible per a personal amb formació tècnica, per la qual cosa, el cap de projecte haurà de seguir de prop el procés de documentació perquè aquest procés tingui tots els requisits de qualitat necessaris.

Cas pràctic

En aquest projecte s'utilitzaran els formats adequats (DocBook) i ateses les característiques del projecte, la major part de la documentació serà la corresponent als serveis integrats que ja tenen una documentació adequada. Mitjançant el format triat, es procedirà a integrar tota aquesta documentació per a donar-li un aspecte uniforme i coherent. Pel que fa a la documentació tècnica, s'haurà de complementar amb les estructures funcionals i de relació del codi desenvolupat, i també amb els diagrames adequats al tipus de projecte en el qual es treballa (classes, mètodes, objectes, esdeveniments, missatges, etc.).

En el cas d'integració de documentació externa, s'ha d'esmentar la font d'on s'ha extret i la versió per a evitar diferències o situacions de discrepància que afectin o bé el programari o bé la documentació.

5. Implantació

La implantació és la fase en la qual es traspasa el programa en desenvolupament a producció, amb usuaris i connexió reals en l'entorn que operarà. Aquesta fase generalment és crítica en la vida del projecte i és una etapa de nerviosisme i d'alta pressió psicològica. Per tant, és important mantenir la calma durant aquest període i planificar acuradament el canvi.

Així mateix, es pot destacar que hi haurà una certa reticència al canvi per part dels usuaris que tinguin deficiències en la formació. Per tant, caldrà que el cap de projectes l'haurà de preparar amb els caps d'àrea fent sessions informatives i contestant totes les preguntes dels usuaris per anar preparant el camí cap al canvi.

En els casos en què el sistema se substitueixi per un altre de similar, el grup de desenvolupament es trobarà amb un ambient receptiu o una mica indiferent segons en quina mesura el nou sistema solucioni els problemes. Per contra, si el canvi (com és el nostre cas) és radical, ja que modifica la forma de funcionament del sistema en la seva totalitat, la llista d'incidències serà diferent i més llarga, ja que els usuaris no sabran com fer les tasques que fins llavors venien fent de manera rutinària.

En tot cas, és en aquesta fase en la qual s'ha d'implicar els usuaris participants en els casos d'ús analitzats del sistema i ensenyar-los mitjançant la formació adequada les seves noves responsabilitats o tasques.

També és important establir els criteris de qualitat de servei sota els quals el sistema ha de funcionar, és a dir, els serveis mínims sota els quals el sistema es considera totalment funcional, la qual cosa s'haurà de redactar d'acord amb les especificacions funcionals inicials. Òbviament, després de la implantació (o en un breu període posterior acordat per endavant) el sistema implantat els haurà de complir.

És important tenir en compte si la implantació és total (nous serveis, nova funcionalitat) o si el sistema s'ha d'integrar amb un altre de ja existent i treballar conjuntament amb ell. En el primer cas no hi haurà massa problemes en la seva instal·lació i posada en funcionament, mentre que en el segon, s'ha de garantir que la implantació no afectarà el funcionament dels serveis que hi ha, els quals hauran de continuar funcionant a ple rendiment, l'ampliació o modificació de parts del sistema informàtic no ha d'alterar el funcionament de la resta del sistema. Per exemple, podríem canviar tot el que es refereix a la xarxa però això no ha d'afectar el departament de comptabilitat, que hauria de poder realitzar les seves tasques de la mateixa manera que abans.

Cas pràctic

Planificació de la implantació del subsistema serveis interns

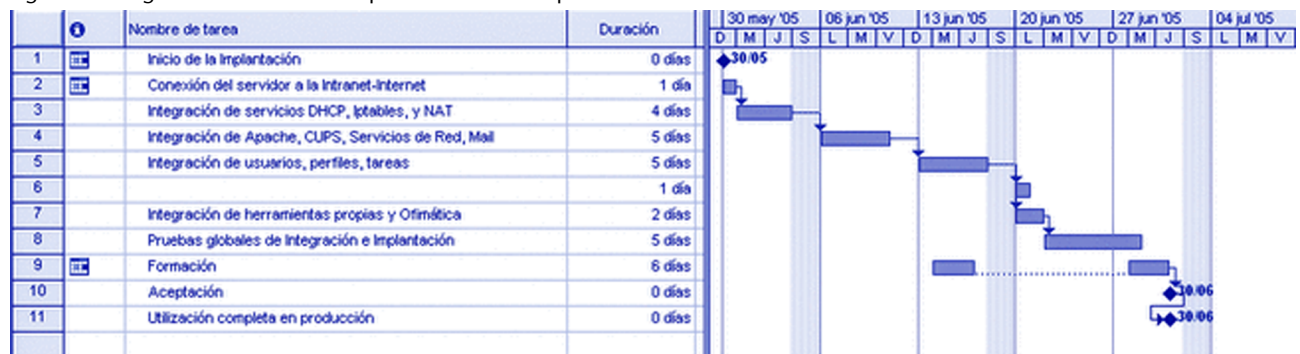
El cap de projecte, en aquest cas, haurà de dissenyar un calendari d'implantació que serà consensuat tant amb els desenvolupadors com amb els usuaris de casos d'ús. Per a això es poden utilitzar diagrames i eines com els que s'han fet servir en altres fases del projecte (per exemple, en la fase de desenvolupament).

Com s'ha esmentat anteriorment, l'etapa d'implantació requereix la dedicació total per part dels equips de treball, usuaris del mateix subsistema i d'altres subsistemes amb els quals aquest interrelaciona. En aquests casos, és important disposar (si la mida del projecte ho justifica) d'eines de treball col·laboratiu (Groupware) de comunicació i registre d'incidències. També es poden utilitzar mètodes més tradicionals com fulls d'incidències o registres per a anar centralitzant tots els problemes i/o incidències que es vagin produint durant la posada en marxa. D'aquesta manera, es disposarà de tota la informació de la implantació del sistema de manera centralitzada, ja sigui per a resoldre possibles problemes o per a avaluar la implantació *a posteriori*.

Independentment del mètode utilitzat, hi haurà d'haver un equip de persones, generalment del departament de disseny i desenvolupament de suport que doni resposta a aquestes incidències i solucioni els problemes que sorgeixin durant l'execució real del sistema implantat.

Un exemple d'un diagrama de Gantt amb la planificació de la implantació podria ser el següent:

Figura 12. Diagrama de Gantt amb la planificació de la implantació



5.1. Formació

La comunicació és un element d'importància vital en l'acceptació del projecte i en la utilització correcta per part dels usuaris; per aquesta raó, el cap de projecte haurà d'organitzar sessions informatives (o delegar en persones del seu equip amb la formació adequada per a fer-ho) amb els futurs usuaris al llarg del projecte. Una vegada que els usuaris coneguin els objectius del projecte, el seu disseny i en alguns casos el seu desenvolupament, aquestes persones hauran de rebre formació perquè puguin treballar eficientment amb el sistema desenvolupat i en la qual també se'ls expliqui la seva potencialitat, les seves obligacions i deures respecte al nou sistema.

Si bé en si mateixa la formació no sol formar part dels projectes en l'àmbit de les tecnologies de la informació i la comunicació (TIC), és important considerar-la com un element determinant en l'acceptació del producte final.

En aquest cas, per a evitar tòpics molt coneguts, s'han de determinar diferents perfils d'usuari que necessitaran ser formats sobre els diversos aspectes del sistema que s'implanta. Segons les obligacions i responsabilitats, la formació pot anar des de la utilització de metodologies no presencials fins a la formació presencial amb avaluació i anàlisi de casos pràctics.

Per a realitzar els plans de formació i desenvolupar el seu contingut, s'haurà de començar pels estudis de casos d'ús (utilitzats en les fases d'anàlisi i disseny) i ampliar-los segons les necessitats de l'empresa i les característiques del projecte.

5.2. Implantació del sistema, proves i nivell de servei

Les proves d'implantació són aquelles que es realitzen sobre el sistema funcional final. Generalment, difereixen de les unitàries i de les d'integració, que es realitzen no sobre les màquines de desenvolupament, sinó sobre les del client. En el nostre cas particular, com que serà equipament nou, les proves d'integració es realitzaran en el mateix maquinari on s'han fet les proves restants. No obstant això, aquesta circumstància es donarà quan es puguin determinar els errors de connectivitat, seguretat, etc., és a dir, quan el sistema treballi al "món real" i no al departament de desenvolupament.

És important assegurar la disponibilitat dels recursos actuals per a minimitzar l'impacte d'"aturar" l'empresa o serveis d'aquesta durant el temps més curt possible. Normalment aquests canvis i proves es realitzen en dies no laborables per a evitar molèsties als clients, però implica que totes les persones vinculades han de ser-hi presents com si es tractés d'un dia normal. En el cas d'empreses que hagin de mantenir actius tots els serveis en règim 24 x 7, evidentment s'haurà de fer el canvi directament en producció i s'haurà d'extremar la vigilància en els casos d'errors.

Cas pràctic

En resum, en el nostre cas i en aquesta activitat s'ha de fer el següent:

- Integrar tots els subsistemes en l'entorn real, modificant les IP, NAT i serveis de comunicació per a adequar-los a l'entorn de treball (no al de desenvolupament).
- Integrar tot el maquinari i programari existent amb els nous subsistemes i reencaminar les peticions de dades i serveis cap als nous subsistemes.
- Verificar la integració d'usuaris, serveis i subsistemes pel que fa a seguretat i autenticació (s'ha de recordar que això ja s'ha realitzat en proves unitàries, però no amb tot el maquinari i usuaris de l'empresa).
- Integrar els nous serveis (per exemple, web, correu electrònic) amb el sistema real sense que hi hagi interferències amb els existents. Això dependrà de la política de l'empresa si el canvi es fa d'una manera abrupta (reemplaçament directe de serveis) o bé de forma gradual (coexistència de serveis durant un temps prudencial).
- Traspàs de les dades al nou gestor de dades (o actualització si ja s'han fet traspàsos parcials) amb validació de les dades i verificació de les polítiques de seguretat.
- Configuració dels serveis externs (adreces, IP, etc.) perquè apuntin al nou sistema.

Una vegada comprovada la instal·lació correcta de tots els subsistemes, de les polítiques de seguretat, de l'accessibilitat i de la connectivitat, es llançaran els processos automàtics tenint en compte que s'haurà de realitzar una monitorització periòdica per a veure la seva evolució.

Després de les proves d'implantació dels primers resultats estables i d'haver corregit les deficiències i errors que s'han produït en els subsistemes, s'haurà d'avaluar el comportament d'aquests per a obtenir mesures de prestacions o eficiència que permetran, comparant-les amb els requisits inicials, establir criteris de compliment o no del sistema desenvolupat, és a dir, el nivell de servei.

En cas de no arribar a uns resultats mínims, s'hauran d'efectuar els canvis necessaris en el desenvolupament o en el disseny dels subsistemes que afectin aquests resultats. Aquesta tasca normalment és complicada, ja que s'ha de fer marxa enrere (encara pitjor si s'ha de tornar al disseny) i implicarà retards en el ple funcionament de l'aplicació, encara pitjor si s'ha de tornar a retocar el disseny inicial.

En aquest cas, el cap de projectes, juntament amb els directius de l'empresa, en primer lloc, ha de prendre una determinació sobre si el sistema segueix en funcionament i si es treballa en paral·lel per a solucionar aquests problemes o es treu "de línia" el sistema nou i s'ajorna la seva integració fins que els problemes es tinguin resolts. En segon lloc, el cap de projectes haurà d'analitzar on s'han produït els problemes i en quina fase s'han comès els errors perquè aquesta situació no es repeteixi en un futur (és una bona pràctica deixar constància documental sobre els errors o desviacions per a aquest fi). Aquest tipus de retard o problemes genera normalment pèrdues econòmiques a l'empresa, per la qual cosa, si l'equip de projecte és extern, se li demanaran responsabilitats –econòmiques, legals, etc. Mentre que si és intern, s'haurà d'analitzar on s'han donat els problemes per a depurar responsabilitats.

Cada cas és particular, però generalment s'ha de donar una prioritat a cada un dels problemes i errors perquè es puguin analitzar contextualitzats. Això permetrà tenir informació d'on és el problema i quant de temps es trigarà a resoldre'l. Per exemple, un cas d'autenticació generarà una sèrie de problemes derivats, que no ho són pas, sinó causa del primer, el qual pot ser simplement perquè l'usuari que es connecta amb un servei NIS no ha estat donat d'alta correctament en el servidor o perquè el mateix usuari ha canviat la paraula clau d'accés i encara no s'han refrescat les memòries cau de les màquines client. Aquesta situació no significa que el subsistema no funcioni, sinó que s'ha de determinar la causa i trobar quin és l'error inicial i quins són els errors correlacionats (és a dir, que depenen d'un altre).

Generalment, el cap de projectes amb el seu equip haurà d'aplicar una sèrie de mesures com utilització dels recursos, temps de resposta, temps de retorn, amplada de banda, etc., que permetin identificar el nivell de servei que s'obté del sistema per a realitzar previsions de funcionament quan el sistema es trobi a plena càrrega.

5.3. Acceptació del sistema

Aquesta activitat consisteix a presentar als responsables o a la direcció de l'empresa tota la documentació relativa a la implantació del projecte, incloent-hi els resultats des de les proves unitàries, d'integració i d'implantació fins als nivells de servei obtinguts per a la seva aprovació.

Aquesta activitat és molt important per a l'empresa si l'equip de projecte és extern perquè implica que s'accepta el projecte tal com està, tenint en compte (generalment regit per un contracte) que passarà amb les incidències que puguin sorgir. Això generalment està regit pel contracte que inicialment es firma entre les dues parts i hauria de quedar especificat des d'un principi. Una vegada acceptat el sistema, l'equip de projecte podrà considerar que el projecte ha finalitzat i que les modificacions o afegits posteriors generalment aniran a

banda del cost econòmic de l'anterior (no serà així si són deficiències o faltes en el desenvolupament inicial que no s'han detectat). La situació és molt més senzilla des d'aquest punt de vista si l'equip de projecte és intern a l'empresa, encara que s'han d'estipular molt bé els canals de comunicació per a la correcció d'errors o problemes, ja que si no l'equip de projecte sempre viurà pendent (per diverses peticions dels usuaris) del sistema desenvolupat.

5.4. Manteniment

Aquesta fase s'ha de planificar durant la resta de les etapes del projecte, atès que una gran part de la informació inclosa en la documentació del projecte ha d'anar orientada a mantenir-lo.

Hi ha diferents opcions quant al manteniment segons el tipus d'instal·lació de què es disposi –manteniment del maquinari, de les comunicacions, de la infraestructura (aire condicionat, per exemple), del programari, etc.–, les quals s'integraran dins la política de l'empresa sobre això. Es donen situacions particulars, com per exemple, el manteniment del maquinari el realitza la mateixa empresa o és un servei extern?

En el nostre cas considerem que tot el programari implantat és codi obert (*open-source*), per la qual cosa, es pot optar pel manteniment intern (amb l'equip adequat) o per subcontractar una empresa externa (per al programari integrat propietari se seguirà amb la política anterior definida per l'empresa). En el primer cas passa per contractar un tècnic de suport especialitzat o formar personal intern per a aquesta feina, ja que es disposa de tot el codi i la documentació necessaris per a això. La formació podria ser a càrrec de l'equip de projecte o obtenir-se en el mercat com a formació estàndard (si bé és recomanable per la seva eficiència el primer cas).

En cas que s'opti per una empresa externa de manteniment, serà important que la direcció de l'empresa, juntament amb l'equip de projecte, redactin un plec de condicions de manteniment perquè no hi hagi problemes en l'externalització d'aquest servei i que es realitzi d'acord amb els criteris dels qui han desenvolupat i implantat el projecte.

Resum

Durant el desenvolupament d'aquest material s'han descrit detalladament totes les fases de què consta un projecte tenint en compte que, d'acord amb la seva tipologia, algunes poden ser diferents o molt curtes, però que en la seva gran majoria apareixen en tots els projectes de l'àmbit de les TIC.

Aquestes fases, depenent de la metodologia emprada (més clàssiques o més dinàmiques –per exemple, les evolutives–), poden estar més o menys presents, fet que s'haurà de tenir en compte a l'hora d'escollir la metodologia de treball, però, generalment, les que sempre apareixen en projectes d'aquestes característiques són:

- **Estudi de viabilitat:** en aquesta fase es considera si el projecte es pot realitzar, tenint en compte les circumstàncies internes i externes, les diferents solucions possibles i els recursos de què es disposa.
- **Anàlisi:** s'analitzen les necessitats que es vol satisfer amb el nou projecte, s'ajusten els objectius finals i se centra la solució tecnològica. També en aquesta fase es defineixen les interfícies entre els diferents subsistemes que formaran el projecte i les d'usuari que permetran interactuar amb el sistema.
- **Disseny:** es realitza en aquesta fase el disseny tecnològic de la solució escollida, proposant una arquitectura global i analitzant i estudiant tots els casos d'ús (o els més representatius) existents.
- **Desenvolupament:** es construeix en aquesta fase la solució proposada tenint en compte l'entorn utilitzat, s'escullen les llicències, es genera la documentació i s'executen les proves d'acord amb el tipus de projecte i metodologia utilitzada.
- **Implantació:** es traspasa de l'entorn de desenvolupament al sistema real i es realitzen totes les proves i mesures de nivells de prestacions que conduiran a l'acceptació definitiva del projecte. Es defineix també el pla de manteniment i es prenen les decisions adequades perquè el projecte funcioni correctament durant la resta de la seva vida.

En el cas particular d'un projecte basat en programari lliure, s'ha de considerar en cada una de les fases anteriors aspectes diferencials amb altres projectes basats en programari propietari com poden ser l'adequació d'una solució existent als nostres propòsits, la viabilitat, l'arquitectura, les llicències, etc., que

normalment no apareixen en l'altre cas (per exemple, si es desenvolupa un mateix projecte amb tecnologia .NET, les llicències i l'arquitectura són úniques i prefixades per la tecnologia).

Bibliografia

AgileAlliance y Extreme Programming: herramientas para el desarrollo de software.

<http://www.agilealliance.org/>

<http://www.extremeprogramming.org/>

Dawson, Christian W.; Martín, Gregorio (2002). *El proyecto de Fin de Carrera en Ingeniería Informática. Una Guía para el estudiante*. Prentice Hall.

Dia: Herramienta para la creación de diagramas. <http://www.gnome.org/projects/dia/>

DocBook: Herramientas y metodología para la generación de documentación

<http://www.docbook.org> o también <http://www.oasis-open.org/docbook/>

Llicència pública general de GNU. <http://www.gnu.org/licenses/>

Métrica 3: Consejo Superior de Informática y para el impulso de la Administración Electrónica. Ministerio de Administraciones Públicas.

<http://www.csi.map.es/csi/metrica3>.

Newkirk, James; Martín, Robert (2001). *La programación extrema en la práctica*. Addison Wesley.

Planner: Herramienta de planificación de proyectos.

<http://www.simpleprojectmanagement.com/planner/home.html>

ReadySet: Herramientas para la generación y mantenimiento de documentación basado en plantillas.

<http://readyset.tigris.org/>

The Linux Documentation Project. <http://tldp.org/>

The Object Management Group – UML. <http://www.omg.org/uml/>

Annex

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in

performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered

work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of

the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license

document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```



```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <<http://www.gnu.org/philosophy/why-not-lgpl.html>>.

