

GUIA BÀSICA DE RECOMANACIONS PEL DESENVOLUPAMENT D'APPS DE SALUT I ATENCIÓ SOCIAL

Com crear una aplicació
mòbil segura, accessible,
usable i interoperable.

Versió 1.0

Aquesta guia està destinada a totes aquelles persones que desitgin iniciar-se en l'entorn mHealth i vulguin tenir una visió general de quins són els aspectes que cal tenir en compte a l'hora de desenvolupar una aplicació mòbil de l'entorn salut i/o de benestar social.

GUIA BÀSICA DE RECOMANACIONS PEL DESENVOLUPAMENT D'APPS DE SALUT I ATENCIÓ SOCIAL





Avís legal:

Aquesta obra està subjecta a una llicència Reconeixement 3.0 de Creative Commons. Se'n permet la reproducció, la distribució, la comunicació pública i la transformació per generar una obra derivada, sense cap restricció sempre que se'n citi el titular dels drets (Generalitat de Catalunya, Fundació TIC Salut Social). La llicència completa es pot consultar a: <https://creativecommons.org/licenses/by/3.0/es/deed.ca>

D'aquesta edició:

© 2018, Fundació TIC Salut Social

Generalitat de Catalunya

Primera edició: Agost de 2018

Presentació

En el marc de l'adopció de les polítiques mHealth, el govern català va aprovar a principis del 2015 el Pla Estratègic de Mobilitat instant a les institucions del sector a apropar els serveis sanitaris i d'atenció a la persona a través de les tecnologies de la mobilitat, i al mateix temps, facilitar la transformació dels diferents processos assistencials i socials per aconseguir una atenció integrada i pròxima al pacient. En base aquest encàrrec i sota el paraigües de la CE, la Fundació TIC Salut Social, organisme del Departament de Salut de la Generalitat de Catalunya, ha treballat en la posada en marxa d'un model d'acreditació d'aplicacions mòbils del sector de la salut i dels serveis socials que permet certificar a les aplicacions a ser aptes per facilitar i millorar el seguiment de la salut dels ciutadans aportant garantia de qualitat i seguretat.

Fruit de l'expertesa generada, el treball en xarxa amb altres agents del sector i una experiència pilot enriquidora, neix aquesta Guia de recomanacions pel desenvolupament d'apps. La Fundació TIC Salut Social aposta per un procés d'acreditació obert i transparent a tot tipus d'iniciatives amb l'objectiu de donar confiança a la ciutadania i als professionals. En aquesta línia, la Guia recull una sèrie de recomanacions per acompanyar al desenvolupador en la programació d'aplicacions mòbils segures, accessibles, usables i interoperables.

Esperem que els passos i recomanacions que englobem a la guia siguin d'utilitat, tant per les administracions, empreses, universitats i start-ups que treballen per compartir el seu coneixement cap al ciutadà i que esdevingui una eina de consulta a l'abast de tothom.

Agraïments

Entitats col·laboradores

La Fundació TIC Salut Social vol agrair molt especialment a CSV Experts, iSalus, m4social, la Taula del Tercer Sector, Pasiona i la UPC (Universitat Politècnica de Catalunya) per la seva plena disposició a desenvolupar aquesta guia i per aportar el seu ampli coneixement en les diferents temàtiques en les que han participat.

Sumari

1.	<u>Introducció</u>	8
2.	<u>Àrea mHealth de la Fundació TIC Salut Social</u>	10
3.	<u>Tipus d'aplicacions</u>	12
4.	<u>Markets d'aplicacions</u>	14
5.	<u>El procés de desenvolupament d'aplicacions mòbils</u>	16
6.	<u>Llenguatges i entorns de desenvolupament</u>	20
7.	<u>Consells pel desenvolupament i la publicació d'aplicacions</u>	22
8.	<u>Seguretat i protecció de dades</u>	24
9.	<u>Accessibilitat</u>	28
10.	<u>Usabilitat</u>	29
11.	<u>Experiència d'usuari</u>	31
12.	<u>Estàndards de comunicació i representació d'informació</u>	32
13.	<u>Medical Device: L'aplicació mòbil com a producte sanitari</u>	36
14.	<u>Entitats participants</u>	40
15.	<u>Autors</u>	41

1. Introducció

L'ús de les tecnologies de la mobilitat en l'àmbit de la salut és un aspecte quotidià dels ciutadans a nivell mundial, ja que disposen d'un telèfon mòbil i fan ús d'aplicacions de tots els àmbits, inclosos el de la salut, el de benestar i el social. Actualment es poden trobar al mercat més de 325.000 Apps de salut i benestar, un aspecte que determina una demanda creixent i que fa necessari establir marcs reguladors que permetin adequar aquesta tecnologia en les fases inicials dels projectes, aportant garanties als usuaris i establir així veritables eines de valor i de confiança.

Durant aquests darrers anys, la Fundació TIC Salut Social, juntament amb el Departament de Salut i el Departament de Treball, Afers Socials i Famílies, ha treballat en el sector de les aplicacions mòbils, fent d'observatori de les noves tendències en mobilitat i definint un procés d'acreditació d'aplicacions mòbils del sector salut i d'atenció a la persona.

Objectius

En base a aquests model d'acreditació i a la participació activa de diferents entitats i col·laboradors externs, s'ha desenvolupat aquesta guia, que recull les recomanacions que caldria tenir en compte a l'hora de desenvolupar qualsevol aplicació mòbil destinada a l'atenció a persona, especialment dels entorns salut i social.



Parts de la guia

Aquesta guia aborda tant els aspectes tècnics com funcionals que qualsevol aplicació mòbil ha d'incorporar en les fases inicials del projecte de desenvolupament. Aquest contingut està estructurat en els següents apartats:

- **Introducció:** Presentació de l'objectiu i relació de les parts de la guia.
- **Tipus d'aplicacions:** Apps que existeixen amb les principals característiques de cadascuna d'elles i casos en què són més adequades.
- **Markets d'aplicacions:** Entorns de distribució d'aplicacions mòbils, juntament amb consells per tal de publicar-hi les aplicacions.
- **El procés de desenvolupament d'aplicacions mòbils:** En aquest apartat es descriuen les metodologies de treball més utilitzades actualment per desenvolupar aplicacions i els rols que han d'aparèixer dins l'equip d'un projecte software d'aplicacions mòbils.
- **Llenguatges i entorns de desenvolupament:** En aquest apartat es descriuen les diferents tecnologies per cada tipus d'aplicació i sistema operatiu. També es llisten els programaris de desenvolupament que es poden utilitzar en cada cas.
- **Consells per al desenvolupament i la publicació d'aplicacions:** En aquest apartat es llisten alguns consells i recomanacions a tenir en compte, des de la fase de planificació del projecte, fins a la publicació de l'aplicació.
- **Seguretat i protecció de dades:** En aquest apartat es llisten recomanacions de seguretat i protecció de dades que han d'incloure les aplicacions a desenvolupar.
- **Accessibilitat:** Mètodes i tècniques per garantir que una aplicació pugui ser utilitzada per el rang més ampli de persones possible.
- **Usabilitat:** Criteris a tenir en compte a l'hora de dissenyar i desenvolupar l'aplicació, per tal de garantir que és fàcil d'utilitzar.
- **Experiència d'usuari:** Relació de consells destinats a millorar la percepció de l'usuari en utilitzar una App.
- **Estàndards de comunicació i representació d'informació:** Presentació del concepte d'interoperabilitat i dels principals estàndards per assolir-la en l'entorn mòbil.
- **Medical Device: L'aplicació mòbil com un producte sanitari:** Relació de punts clau per ajudar a identificar si una App és un producte sanitari, així com els aspectes a tenir en compte en cas que ho sigui.
- **Entitats participants:** Relació d'entitats que han elaborat aquesta guia.
- **Autors:** Detall dels autors que han participat en la redacció del present document.

2. Àrea mHealth de la Fundació TIC Salut Social



L'àrea mHealth neix a partir del Pla mestre de mobilitat, un projecte impulsat pel Departament de Treball Afers Socials i Famílies i la Fundació TIC Salut Social amb la col·laboració de la Fundació Mobile World Capital Barcelona.

Aquesta àrea pretén establir un marc de referència comú que inclogui les estratègies de mobilitat d'aquests departaments per permetre una visió completa de la influència de la mobilitat en la prevenció i promoció de la salut i en l'atenció social, a fi d'aportar solucions i serveis a través de sistemes mòbils i amb l'objectiu principal de millorar i personalitzar la salut i el benestar de les persones.

En aquesta línia, mHealth aproxima els serveis sanitaris i socials a la ciutadania per mitjà de tecnologies de mobilitat com ara *smartphones*, tauletes o ordinadors portàtils, eines que permeten interactuar de manera àgil i personalitzada.

Una de les ofertes dels serveis que s'ofereixen és el procés d'acreditació d'aplicacions mòbils de l'entorn salut i social.



Procés d'acreditació d'aplicacions mòbils

La Fundació TIC Salut Social, ha creat un model d'acreditació d'aplicacions mòbils del sector de la salut i dels serveis socials que permet certificar a les aplicacions a ser aptes per facilitar i millorar el seguiment de la salut dels ciutadans aportant garantia de qualitat i seguretat.

Les aplicacions són sotmeses a un conjunt de criteris de domini públic, els quals poden ser consultats a la web de la Fundació. Promovent l'ús segur d'aplicacions de qualitat i donant a conèixer aquelles amb un alt valor afegit, es contribueix a millorar l'empoderament del ciutadà, fent-lo conscient i corresponsable de la seva pròpia salut.

El disseny del model d'acreditació s'ha basat en un consens transversal entre diferents professionals i agents del sector: experts de l'àmbit tecnològic, professionals sanitaris (metges, infermeres, psicòlegs i treballadors socials), professionals de la comunicació en salut, pacients experts i representants institucionals i ciutadans. Aquesta diversitat ha permès adoptar un model que engloba totes les visions i des de diferents àmbits, permetent obtenir un conjunt de criteris agrupats en quatre àrees d'expertesa: *usabilitat i experiència d'usuari; tecnologia i fiabilitat de l'aplicació; seguretat i confidencialitat de les*

dades; i funcionalitat i avaluació de continguts. Aquestes àrees d'expertesa són avaluades per separat, sent independents cadascuna d'elles i adaptables a les diferents demandes del sector. El Col·legi Oficial d'Infermeres i Infermers de Barcelona (COIB), el Col·legi Oficial de Metges de Barcelona (COMB), el Col·legi Oficial de Llicenciats en Educació Física i Ciències de l'Activitat Física i l'Esport de Catalunya (COPLE-FC), l'Associació d'Infermeria Familiar i Comunitària (AIFiCC), la Societat Catalana d'Especialistes en Psicologia Clínica (SCEPC), la Societat Catalana de Medicina Familiar i Comunitària (CAMFIC), són algunes de les entitats que col·laboren en l'avaluació d'aquestes àrees.

Les aplicacions que superin el procés d'acreditació rebran un segell de confiança i seran publicades al catàleg de la web de la Fundació, per tal de donar visibilitat i posar-les a disposició del sector perquè les puguin consultar i utilitzar.

La Fundació TIC Salut Social aposta per un procés d'acreditació obert i transparent a tot tipus d'iniciatives amb l'objectiu de donar confiança a la ciutadania i als professionals.

Tota la documentació relativa al procés d'acreditació, així com els criteris es poden consultar a la web de la Fundació TIC Salut Social.



3. Tipus d'aplicacions

Un dels principals elements que cal tenir en compte a l'hora de planificar el desenvolupament d'una App, és triar el tipus d'aplicació que es vol desenvolupar. Aquesta decisió té implicacions en molts aspectes com els coneixements necessaris del equip, el cost del projecte o el resultat final obtingut. A continuació es presenten els diferents tipus d'aplicacions existents i les característiques principals de cadascun.

Natives

Les aplicacions natives es desenvolupen específicament per a cada sistema operatiu amb un llenguatge de programació compatible. Aquest tipus d'Apps permeten explotar tots els recursos del dispositiu (com la càmera o el GPS), així com utilitzar les APIs pròpies del sistema operatiu.

Les aplicacions natives permeten optimitzar millor el rendiment, fet que esdevé un aspecte clau en les aplicacions que requereixen una gran quantitat de recursos com els jocs 3D.

Aquesta mena d'Apps es distribueixen als *markets* oficials i es poden utilitzar sense connexió a internet, tot i que la majoria tenen funcionalitats que requereixen connectivitat per determinades funcionalitats o, directament, no es poden utilitzar sense connexió.

El cost de desenvolupar una aplicació nativa, en comparació amb les d'altre tipus, normalment és més elevat, ja que, si bé en cas d'usar com a back-end un servidor extern aquest sí serà compartit per les diferents Apps natives, el front-end s'haurà de desenvolupar des de zero (o gairebé) pels diferents sistemes operatius. Aquest fet també implica haver de mantenir les diverses aplicacions en fase de producció.

El desenvolupament d'aplicacions natives es recomanen en cas d'haver de desenvolupar Apps que requereixin d'un alt rendiment o consum de recursos hardware.

Web

Una aplicació web es aquella que s'utilitza accedint a una URL a través d'un navegador (com Chrome, IE o Safari). Aquest tipus d'Apps es poden executar en qualsevol sistema operatiu, tant en dispositius mòbils com en ordinadors de sobretaula, i requereixen disposar de connexió a internet. És important destacar que **les aplicacions web no permeten utilitzar els recursos del dispositiu, no cal que compleixin les condicions dels markets, ja que no s'hi publicaran, i l'usuari sempre disposa de l'última actualització de manera transparent.**

Degut a que són multi-plataforma, el desenvolupament de les aplicacions web sol ser més econòmic, ja que no se'n han de fer diferents versions o adaptacions per cada sistema operatiu del dispositiu. Malgrat això, sí que cal considerar la fragmentació entre els diversos navegadors (i versions de cadascun), així com les diferències de resolució de les pantalles o les capacitats hardware com la memòria o el processador (especialment importants en aplicacions pesades).

És recomanable desenvolupar aquests tipus d'Apps quan es vulgui tenir una única aplicació, inclús entre dispositius mòbils i PC. Caldrà, però, tenir en compte que pot ser necessari realitzar modificacions, en alguns casos força importants, a nivell d'interfície gràfica, per tal que l'aplicació s'adapti i sigui usable en les diferents plataformes. Un dels principals inconvenients és la limitació per usar els elements hardware del dispositiu o altres opcions com les notificaciones natives.

Híbrides

Una aplicació híbrida és un concepte a mig camí entre una aplicació nativa i una aplicació web. Estan construïdes en els mateixos llenguatges i estructura que les aplicacions web, però es distribueixen de la mateixa manera que les Apps natives. **Les aplicacions híbrides tenen els mateixos requeriments de connectivitat a internet que les natives, així com la mateixa capacitat per accedir als recursos hardware del dispositiu.**

Aquest tipus d'aplicacions generalment impliquen un menor cost, en comparació amb les natives, degut a que es realitza un únic desenvolupament (i per tant també un únic manteniment) que serveix pels diferents sistemes operatius. Es pot donar el cas que s'hagin de realitzar ajustos menors per problemes puntuals amb algun dels sistemes operatius, però l'esforç sempre serà molt menor al que implica desenvolupar diferents aplicacions natives.

Les aplicacions híbrides són recomanables per a desenvolupar eines de gestió o de visualització d'informació en les que es vulgui reduir el cost de desenvolupament i manteniment. Exemples adequats per Apps d'aquest tipus poden ser les que permeten demanar cita online, realitzar e-consultes o conformar repositoris de consells pràctics de salut.

Generades

Les aplicacions generades també es denominen *falses natives*, donat que consisteixen en escriure una App en un llenguatge de programació no natiu, com per exemple C#, i després generar les aplicacions natives, de manera automàtica, amb el llenguatge específic de cada plataforma. D'aquesta manera, s'aconsegueixen aplicacions similars a les natives, tot i que no arriben als mateixos nivells d'optimització. Aquesta aproximació permet reduir costos, ja que bona part de l'aplicació és comuna per totes les versions, tot i que és possible que alguns elements puntuals s'hagin de programar amb llenguatge natiu. És per això que es recomana tenir nocions dels llenguatges propis de cada sistema operatiu, i poder desenvolupar codi específic.

Les aplicacions generades són recomanables per controlar els costos de desenvolupament i manteniment, però cal valorar bé els avantatges i les limitacions de cada plataforma concreta que permet generar-les. Un exemple és el motor gràfic Unity, que permet crear aplicacions 3D compatibles amb PC, mòbils i consoles, amb un únic desenvolupament.

4. Markets d'aplicacions

Les botigues d'aplicacions, també conegudes amb l'anglicisme *markets*, són plataformes web que permeten vendre i/o distribuir aplicacions mòbils. En funció del sistema operatiu al qual s'adrecin, les Apps es publicaran a un market específic, essent Google Play Store per Android i l'App Store per iOS els més utilitzats. També existeixen altres botigues d'aplicacions com: Amazon App Store que és la principal alternativa a Google Play Store; F-Droid que té la restricció de que totes les aplicacions que conté són de codi lliure i es pot accedir al seu codi i/o contribuir-hi; GetJar que és una botiga multi-plataforma (Android, iOS, Windows Mobile); o Microsoft Store que és la botiga oficial de Microsoft.

En el present apartat es presentaran més en detall els markets Google Play Store i App Store, destacant alguns consells per a publicar-hi aplicacions.

Google Play Store



14

Google Play és la botiga oficial de la companyia Google i en la qual s'hi poden trobar llibres digitals, música, pel·lícules, notícies i publicacions, així com Apps i jocs. L'apartat específic per a les aplicacions mòbils s'anomena Google Play Store i conté més de 3,7 milions d'Apps organitzades en diferents categories, entre elles la de "Salut i fitness".

Per tal de publicar una aplicació al Google Play Store, és imprescindible complir les polítiques del **programa per a desenvolupadors**, i acceptar l'acord de distribució per a desenvolupadors. La companyia ofereix consells a seguir per a complir aquestes polítiques, relatius als continguts, la suplantació d'identitat, la propietat intel·lectual, la privadesa, la seguretat i l'engany, l'obtenció d'ingressos o la presència de publicitat. Aquests consells es poden consultar al següent enllaç: Centre de polítiques per a desenvolupadors.

A mode de resum, es destaquen els següents aspectes a tenir en compte a l'hora de publicar una aplicació al Google Play Store:

1. Fer un **pla de proves** exhaustiu que inclogui tant testos unitaris com d'integració, per minimitzar els possibles errors que puguin sorgir i que **l'experiència d'usuari es vegi alterada en la menor mesura possible**.
2. L'aplicació es pot publicar en mode *alpha* o *beta* per assegurar-ne la qualitat de abans de publicar-la a producció
3. Revisar i garantir que l'App **compleix totes les polítiques** del *market*.
4. Assegurar que la **descripció** de l'aplicació és correcta per a tots els públics, i que és **compreensible** i **concisa**.
5. Assegurar que es compti amb els **drets d'autor** de totes les imatges que s'utilitzen, incloses les que es mostren al *market*.
6. Si l'aplicació utilitza dades dels usuaris, assegurar que s'inclogui una **política de privacitat** que exposi les dades que es recullin i la seva finalitat.
7. Assegurar que l'aplicació i els anuncis que s'hi visualitzen **no inclouen continguts inadequats** (com contingut sexual explícit, d'incitació a l'odi, amenaces o assetjament). A més, si s'inclouen anuncis a l'aplicació aquests no poden ser ni molestos ni enganyosos.
8. Incloure tota la **informació bàsica (amb traduccions si es vol publicar a diferents**



dominis lingüístics) i rellevant sobre l'aplicació: una **descripció** precisa, **imatges** que en mostrin el funcionament, l'etiquetatge PEGI corresponent i, si escau, un vídeo de demostració.

9. Elecció de la icona que segueixi les guies de recomanació d'Android
10. Assegurar que l'**e-mail de contacte associat a l'App és el correcte** perquè Google pugui comunicar qualsevol problema amb l'aplicació.
11. **Signar l'aplicació en mode release.**
12. **Elecció adequada dels noms dels packages ja que són únics, permanents i no es poden reutilitzar en el futur.**
13. Es poden publicar diferents aplicacions (APKs) a la mateixa fitxa de l'aplicació per diferents configuracions i/o compatibilitat de dispositius.
14. Optimització de l'aplicació: deshabilitar els *logs* que no siguin necessaris, recursos que no es facin servir, etc.
15. Verificar que els permisos definits en el manifest són realment necessaris.
16. Verificar la versió mínima escollida.
17. **Corregir els errors** que es comuniquin o posar-se en contacte amb Google per comunicar qualsevol disconformitat o necessitat addicional d'informació.
18. Utilitzar eines com Crashlytics o Google Analytics per simplificar la gestió del manteniment correctiu i evolutiu i analitzar com s'està utilitzant l'aplicació.



App Store

L'App Store també és una plataforma de distribució digital però és propietat de la companyia Apple Inc. En aquest market s'hi poden comprar i/o descarregar més de 2,1 milions d'Apps, també organitzades per categories com la de "Salut i fitness".

Totes les aplicacions que es publiquen a l'App Store són revisades prèviament per un expert d'Apple, però els requisits que cal complir són força similars als indicats per a la publicació al Google Play Store. Malgrat això, cal tenir en compte les següents particularitats:

1. Per tal que els experts d'Apple puguin revisar l'aplicació, serà necessari crear un perfil a iTunes Connect i proporcionar tota la informació rellevant relativa a l'App, amb l'objectiu d'oferir una idea clara sobre la seva funcionalitat.
2. Les captures de pantalla que s'incorporin a iTunes Connect han de coincidir exactament amb les del producte final, sinó no es passarà la validació de seguretat.
3. No es pot incloure cap referència ni logotip de Google o d'altres plataformes com Windows o Amazon a les aplicacions que es vulguin publicar.
4. Les aplicacions "In House" (de distribució interna) no es poden desplegar en aquesta botiga. Aquest tipus d'aplicacions s'han de desplegar en el programa Enterprise d'Apple.
5. Utilitzar eines com Crashlytics o Google Analytics per fer el seguiment de la utilització de l'aplicació

Per tal de provar versions beta de les aplicacions, Apple posa a disposició dels desenvolupadors l'eina TestFlight. Aquesta App permet publicar la versió de prova d'una aplicació, i convidar fins a 10.000 usuaris per a que se la descarreguin i la provin, proporcionant el seu feedback. L'eina també permet realitzar validacions internes per part de l'equip de desenvolupament, i amb una limitació de fins a 30 dispositius.

Al següent enllaç es poden consultar les recomanacions i bones pràctiques que s'especifiquen al web de l'App Store, relatives a la revisió de les aplicacions, la interfície d'usuari o el màrqueting: <https://developer.apple.com/app-store/guidelines/>.

5. El procés de desenvolupament d'aplicacions mòbils

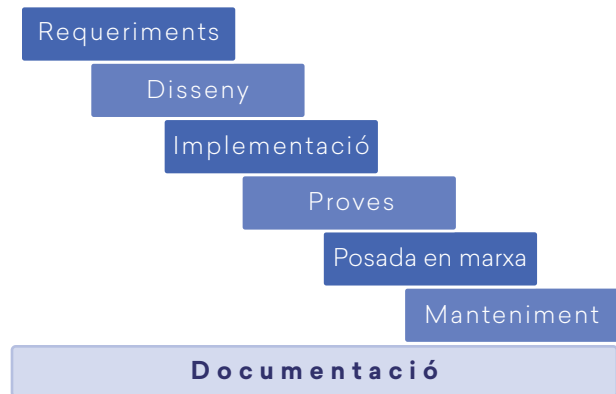
El procés o cicle de vida del desenvolupament software té característiques diverses depenent de l'àmbit, les funcionalitats o els riscos associats al tipus d'aplicació que cal desenvolupar. En funció d'aquestes variables, aspectes com la complexitat, la durada o el cost poden variar de manera molt significativa. En aquest apartat es presentaran les metodologies que es poden aplicar per a desenvolupar aplicacions, amb les corresponents fases i rols de cadascuna d'elles.

Fases del procés de desenvolupament

El procés de desenvolupament es pot gestionar seguint diferents metodologies, l'elecció de les quals es realitzarà segons el tipus de projecte que s'està duent a terme. Malgrat això, existeixen algunes etapes bàsiques que són comunes a gairebé tots els processos de desenvolupament software, les quals es presenten a continuació:

- **Preses de requeriments:** S'identifiquen i defineixen inicialment els casos d'ús o funcionalitats que ha de cobrir l'App, els quals poden patir modificacions al llarg del procés de desenvolupament si es considera oportú. Les modificacions poden donar-se per diferents factors com canvis en les necessitats dels potencials usuaris, canvis legals o limitacions de recursos entre d'altres.
- **Disseny:** Es realitza un anàlisi funcional i tècnic per determinar quins requeriments es volen implementar i com es farà.
- **Implementació:** Etapa en la que es desenvolupen les funcionalitats, els elements gràfics, etc. especificats en el disseny.

- **Proves:** Es realitzen totes les proves necessàries per garantir que el software compleix amb les necessitats identificades i amb el disseny:
 - Les **proves unitàries** són aquelles que avaluen una funcionalitat de manera aïllada de la resta del sistema, com pot ser un mètode o una funció.
 - Les **proves d'integració** consisteixen en provar com funcionen conjuntament dos o més elements als que prèviament se'ls hauria d'haver aplicat les proves unitàries pertinents.
 - Les **proves d'estrès**, es centren en el rendiment del software quan aquest es porta al límit, per veure fins a quin punt segueix funcionant amb normalitat i què succeeix quan aquest límit es sobrepassa.
 - Les **proves de penetració**, són proves en les quals es simula que hi ha un atacant malintencionat que, mitjançant totes les tècniques al seu abast, tractarà de vulnerar la seguretat de l'aplicació per extreure'n dades, corrompre el seu funcionament, etc.
- **Posada en marxa:** El software es configura, es desplega i es distribueix posant-lo a l'abast dels usuaris per començar-lo a utilitzar.
- **Manteniment:** Procés en el qual es van aplicant mesures per corregir possibles errors de l'aplicació, així com millores per satisfer nous requeriments que poden aparèixer al llarg del temps.
- **Documentació:** Ha de ser una tasca transversal a realitzar durant tot el cicle de vida del producte (inclosa l'etapa de manteniment), pel qual s'han de documentar les funcionalitats i casos d'ús als quals donar resposta, el disseny intern del software, les modificacions que es realitzin al llarg del procés, etc.



Metodologies de desenvolupament

Com s'ha indicat anteriorment, existeixen diferents metodologies per al desenvolupament d'aplicacions, les quals es classifiquen en paradigmes en funció de les seves característiques principals. Dos dels paradigmes que més s'utilitzen actualment són:

- La **metodologia clàssica, en cascada**, en la qual les fases es succeeixen de manera lineal. Aquest enfocament és poc flexible, ja que un cop es finalitza una fase es passa a la següent, amb la idea de no tornar a abordar-la posteriorment. Les metodologies en cascada resulten adequades per a projectes en què els requeriments i l'abast són molt clars i quan la tecnologia que s'utilitzarà és molt coneguda per part de l'equip de desenvolupadors.
- Les **metodologies àgils** busquen oferir una gran flexibilitat, permetent adaptar el desenvolupament a qualsevol canvi que es requereixi. En aquestes aproximacions, la modificació dels requeriments és constant i habitual, i es percep com un mitjà per augmentar les possibilitats d'èxit d'un projecte i prioritzar el software que funciona. D'aquesta manera, no cal tenir tots els requeriments definits ni implantats per poder posar l'aplicació a l'abast dels usuaris reduint el time-to-market.

Metodologies Àgils

Les metodologies àgils es basen en els principis de l'anomenat manifest àgil, que es poden agrupar en els quatre punts següents:

- Prioritzar els individus i les seves interaccions per sobre dels processos i les seves eines.
- Prioritzar el software que funciona per sobre de la documentació exhaustiva.
- Col·laboració amb el client per sobre de la negociació contractual.
- La resposta al canvi per sobre del seguiment d'un pla prèviament definit i fixe.

És molt important tenir en compte que prioritzar no vol dir que s'hagi de descuidar els aspectes com la documentació o la negociació contractual, donat que, per exemple, si el contracte és massa ambigu o la documentació realitzada és incompleta és molt probable que s'acabin produint molts problemes durant el cicle de vida del software.

En un projecte àgil el cost també pot variar, ja que depèn dels requeriments que es van redefinint. Una altra de les característiques principals d'aquestes metodologies, és el paper que hi tenen els clients: cal establir-hi una relació de col·laboració i confiança per poder definir els requeriments i realitzar les validacions de les funcionalitats desenvolupades.

Les metodologies d'aquest tipus són adequades per projectes poc definits, força canviants i en les que el client estigui molt implicat i disposat a invertir-hi esforç.

Algunes de les metodologies àgils més conegudes són:

Scrum

És un marc que defineix etapes incrementals, flexible i que permet respondre a necessitats canviants. Resulta adequat per a projectes amb una certa complexitat en què no es disposa de tots els requeriments definits completament a l'inici, i en què aquests són susceptibles de ser modificats o ampliat. Pel que fa a la composició del equip, es recomana que aquest tingui un volum d'entre tres i deu persones.

Scrum treballa sobre iteracions que anomena **sprints**, al finalitzar dels quals cal haver desenvolupat un increment del producte que sigui potencialment lliurable. La durada d'un sprint usualment està compresa entre una i quatre setmanes.

En un equip Scrum es troben tres rols ben diferenciats:

- **Product Owner:** Fa la funció d'enllaç entre el client i l'equip, busca finançament i decideix quines característiques i funcionalitats són indispensables per tal que el producte pugui ser lliurat. Sobre les tasques no indispensables se les ordena assignant una prioritat.
- **Scrum master:** S'encarrega d'orientar a l'equip en els principis i conceptes de Scrum a més de facilitar-ne el procés.

- **Equip Desenvolupament:** Equip relativament petit multifuncional i autoorganitzat. Es considera que si l'equip fos molt gran es faria difícil assolir aquesta autogestió, ja que a més no hi ha jerarquia dins del equip. D'altra banda, si fos molt petit seria difícil que tingués tots els coneixements necessaris pel projecte. L'equip s'encarrega de negociar amb el product owner quins desenvolupaments es faran en cada sprint i de fer-los.

El cicle de vida Scrum s'inicia amb la creació d'una llista de requeriments ordenats per prioritat anomenada **backlog** del producte, per part del product owner. A continuació, el product owner negocia amb l'equip de desenvolupament quins requeriments es desenvoluparan en el proper sprint. Quan comença el sprint, l'equip de desenvolupament s'encarrega de gestionar-se per desenvolupar les tasques incloses al sprint. Diàriament es fa una reunió ràpida a peu dret entre l'equip de desenvolupament, el Scrum master i el product owner per aclarir dubtes, solucionar problemes, comentar temes, etc.

Un cop finalitzat el sprint es duu a terme la revisió d'aquest en el que l'equip presenta el desenvolupament fet. El cicle de la iteració finalitza amb la realització de la retrospectiva del sprint, en la qual es valoren possibles millores en base a l'experiència viscuda. Un cop finalitzat un sprint se'n inicia un altre, i el procés es repeteix fins a tenir desenvolupats tots els requeriments.

Programació Extrema (XP)

Aquesta metodologia té un doble objectiu, d'una banda millorar la qualitat del software, i de l'altre, la de les condicions del equip de desenvolupament. Per assolir aquesta doble fita, es defineixen els 5 valors bàsics següents:

- **Comunicació:** es considera el desenvolupament de software com un treball d'equip i per això es creu indispensable la comunicació entre els membres que el componen, la qual ha de ser, preferentment, cara a cara.
- **Simplicitat:** cal desenvolupar únicament el que és absolutament necessari i el disseny de la solució ha de ser el més simple possible, per facilitar el manteniment, el suport i la revisió del producte.

- **Feedback:** s'han d'usar les grans quantitats de feedback que es reben de diversos orígens per identificar les àrees de millora i revisar les practiques dutes a terme.
- **Coratge:** es considera important actuar amb valentia i no deixar-se portar per la por. Per exemple atrevir-se a acceptar que algun procés no es fa correctament i provar quelcom nou o detectar i reportar problemes organitzacionals.
- **Respecte:** és basic que els membres d'un equip es respectin entre si per poder treballar junts, possibilitant la comunicació i la col·laboració.

Kanban

Aquesta metodologia té per objectiu aconseguir un producte de qualitat i reduir al mínim els colls d'ampolla que es poden produir al llarg del projecte. Les regles bàsiques que segueix són:

- **Acceptar el canvi:** com és comú a les metodologies àgils, es vol motivar a les persones a veure el canvi com a quelcom positiu i encoratjar a modificar allò que no funciona bé o que es pot millorar.
- **Lideratge:** cal tenir iniciativa i gestionar correctament l'equip o la tasca assignada.
- **Respectar el procés actiu i les responsabilitats:** cada membre de l'equip ha de tenir clar quines són les seves funcions en cada moment del projecte, de manera que aquestes s'han d'haver definit prèviament.
- **Kanban no estableix com s'ha de fer una tasca,** sinó que ajuda a decidir si s'està fent de la millor manera o si es pot millorar alguna cosa.

Selecció d'una metodologia

Per tal de decidir quina metodologia s'ajusta més a les necessitat del projecte de desenvolupament concret que s'ha de dur a terme, és necessari tenir en compte factors com el tipus d'empresa, les característiques del projecte software concret o el volum i les característiques de l'equip que s'hi implicarà. Aquesta decisió s'ha de prendre específicament per a cada projecte, i es pot donar el cas en què, dins d'una mateixa empresa, s'utilitzin metodologies diferents en funció de cadascun.

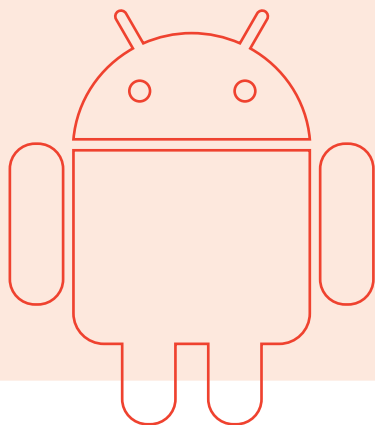
Les característiques pròpies de l'entorn mòbil fan que els requisits inicials de les aplicacions siguin canviants i que, en molts casos, resulti prioritari reduir al màxim el *time-to-market*, requerint un producte funcional el més aviat possible. És per aquests motius que les metodologies àgils van molt lligades als desenvolupaments d'aquest tipus d'aplicacions, ja que permeten adaptar el cicle de vida del software a aquest entorn altament canviant.

6. Llenguatges i entorns de desenvolupament

En funció del sistema operatiu al qual es vol destinar l'aplicació (principalment Android o iOS), i del tipus d'aplicació que cal desenvolupar (nativa, web, híbrida o generada), es poden utilitzar uns determinats llenguatges de programació o uns altres, els quals es presenten a continuació.

Aplicacions Android

El llenguatge de programació a utilitzar en el desenvolupament d'aplicacions Android és **Java**. Amb aquest llenguatge s'implementen les funcionalitats de l'aplicació, la gestió de l'emmagatzemament i l'accés a dades i la interacció amb el hardware (com la càmera o el GPS). Pel que fa a la interfície d'usuari en les **aplicacions natives**, aquesta es crea amb un assistent que ofereix l'entorn de desenvolupament, i que s'encarrega de generar el corresponent **XML** de disseny i configuració. A través de l'assistant (i/o d'aquest XML) es poden organitzar i personalitzar tots els elements de l'aplicació com la posició, l'aspecte o la distribució dels botons, formularis i imatges.



En el cas de les **aplicacions híbrides**, s'utilitzen llenguatges web com **HTML** (o **HTML5**), **JavaScript** o **CSS**. L'aplicació resultant s'executa sobre un component anomenat *webview* que és com un navegador amb motor de Javascript (compatible amb Chrome). Existeixen multitud de plugins que permeten accedir a recursos de baix nivell en el hardware (p.ex sensors) i sempre hi ha l'opció d'implementar un plugin ad hoc. Aquests tipus d'aplicacions minimitzen el *time-to-market* de l'aplicació ja que amb un únic desenvolupament, sobre un alt nivell d'abstracció (HTML5), es poden generar productes per les principals plataformes. Per un altre costat, aquesta abstracció afegeix l'inconvenient que no pot ser utilitzat per aplicacions que precisen un alt rendiment o un ús exhaustiu del hardware (com jocs o realitat augmentada).

L'entorn de desenvolupament oficial per a aplicacions Android, tant natives com híbrides, és **Android Studio**. Aquest programari ofereix un editor de codi, les llibreries (SDK) d'Android i Java, un visor de l'aplicació en temps real, i una màquina virtual que permet executar l'App per provar-la. També es poden utilitzar altres entorns, com per exemple **Eclipse**, si prèviament s'instal·len els components específics per tal de poder compilar i executar Android.

També es poden desenvolupar aplicacions Android amb altres llenguatges com **Kotlin** o **GoLang** i/o C++, amb Android Native Development Kit (NDK) si es volen desenvolupar funcionalitats de baix nivell.



Aplicacions iOS

En el cas del sistema operatiu iOS, les apps es poden programar utilitzant **Objective-C** o **Swift**. Objective-C és el llenguatge que es va utilitzar inicialment en el desenvolupament d'aplicacions natives o híbrides per a iOS però al 2014, Apple va anunciar el llançament d'un nou llenguatge més fàcil, potent i intuïtiu, anomenat Swift. Malgrat això, actualment segueix essent possible desenvolupar apps utilitzant només Objective-C i, de fet, les principals llibreries per a iOS estan escrites en aquest llenguatge, incloent l'API específica Cocoa Touch. Swift també es basa en aquesta mateixa API, de manera que resulta fàcil migrar una aplicació de Objective-C a Swift i ambdós llenguatges es poden complementar utilitzant-se per al desenvolupament d'una mateixa app.

En les **aplicacions natives**, tant Objective-C com Swift s'utilitzen per al desenvolupament de les funcionalitats, la gestió de dades i l'accés a hardware, però també per implementar la capa de presentació (interfície d'usuari). En les **aplicacions híbrides** també s'utilitzen aquests llenguatges per implementar les funcionalitats, però la interfície d'usuari es desenvolupa amb els llenguatges web citats prèviament (HTML/HTML5, Javascript o CSS). Phone Gap o Apache Cordova són exemples de SDKs que permeten crear webapps multiplataforma i que generen productes per les diferents plataformes existents.

L'entorn de desenvolupament oficial per iOS és XCode (per al sistema operatiu MacOs) i ofereix un editor de codi amb control de canvis, així com eines de test i simulació.



Aplicacions generades

Per a desenvolupar aplicacions generades s'utilitzen entorns específics com **Xamarin Studio** o **Genexus**, que permeten dur a terme implementacions multiplataforma. En el desenvolupament s'utilitza un llenguatge únic en funció de l'eina utilitzada (per exemple **C#**, **.NET** o JavaScript), que serà compilat i transformat automàticament al llenguatge natiu desitjat. Com a resultat d'aquest procés, s'obtingran els executables específics per cada sistema operatiu suportat (com iOS o Android).

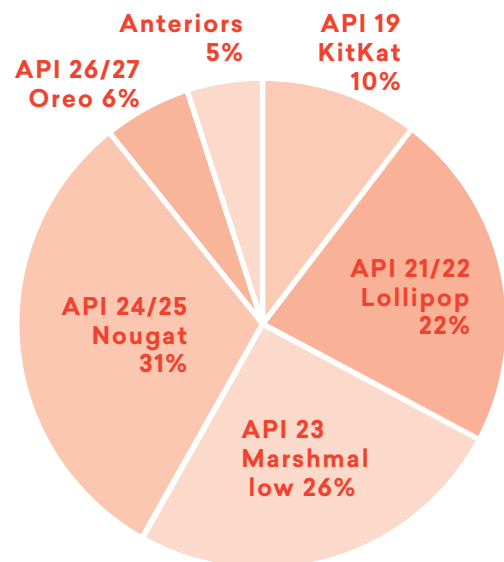
7. Consells pel desenvolupament i la publicació d'aplicacions

En aquest apartat es presenten una sèrie de recomanacions a tenir en compte des de la fase de planificació del projecte fins la publicació de l'aplicació, passant pel desenvolupament de la mateixa.

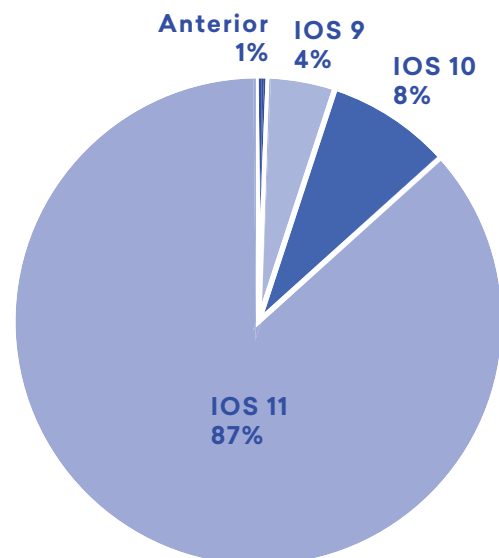
Planificació i disseny

Per una bona planificació, es recomana **tenir present el disseny** de la interfície d'usuari des del començament del projecte, ja que deixar-lo per més endavant pot suposar haver d'adaptar-hi les funcionalitats a posteriori. Per aquesta raó, cal mantenir una sincronització constant entre l'equip de desenvolupadors i el de disseny, i que treballin amb una visió conjunta des del principi.

Un altre factor important a considerar abans de començar a desenvolupar l'aplicació és **escollir la versió (API) en la què es programarà**. Optar per escollir la última versió pot suposar descartar un gran nombre d'usuaris que utilitzen versions inferiors per qüestions d'incompatibilitat. En canvi, escollir una versió molt inferior farà que no es puguin utilitzar les funcionalitats més noves, el que pot provocar que l'App quedi desfasada. En el cas d'Android s'haurà d'estudiar quina versió utilitzar per tal d'arribar al major nombre d'usuaris possible, però fent-ho amb una versió suficientment nova per gaudir de les últimes funcionalitats. Per iOS, n'hi ha prou amb programar dues versions anteriors a la darrera ja que, com es veu en el gràfic presentat a continuació, s'englobarà pràcticament tot el mercat:



Distribució de les versions d'Android al juny de 2018



Distribució de les versions d'iOS al juny de 2018

Desenvolupament

És recomanable utilitzar les **guies oficials de desenvolupadors d'Android i d'iOS** per tal de descobrir les funcionalitats que presenta cada sistema operatiu i resoldre els dubtes de programació que poden sorgir durant el desenvolupament. A més, també pot resultar útil unir-se a una **comunitat de desenvolupadors**, com per exemple **Stack Overflow**, on es podran compartir dubtes amb altres programadors i consultar els que ja han estat resolts amb anterioritat. Addicionalment es recomana que s'incorporin els següents aspectes en el model de desenvolupament:

- Utilització d'un sistema de control de versions (per exemple basats en GIT).
- Generació de productes de manera automatitzada i en un sol pas utilitzant sistemes d'integració continua.
- Fer servir una base de dades d'incidències o *bugs*.
- Fer *testing* (incloent d'usabilitat). Això implica tenir persones formades en aquest àmbit.
- Utilitzar eines d'anàlisi de la qualitat del software, per exemple Android Lint o SonarQube, per identificar problemes o vulnerabilitats.

Publicació

Una aplicació pot contenir **errors (bugs) que poden passar desapercibuts**, ja que es tracta d'errors que només es donen en certs dispositius o en versions específiques d'aquests. Com que és molt difícil provar l'aplicació en tots els dispositius i versions del mercat, es recomana fer una publicació inicial a mode de pilot, en **una regió delimitada**. Per exemple, si l'objectiu és publicar una aplicació a nivell internacional, és aconsellable que durant uns mesos es publiqui només al propi país, per assegurar que l'aplicació no conté errors. D'aquesta manera, s'evitarà que l'aplicació rebi valoracions negatives que podrien fer baixar el seu posicionament als *Markets* i perdi usuaris. Un cop corregits tots els errors, es pot passar a publicar l'aplicació en una regió més gran perquè els usuaris finals puguin valorar l'aplicació sense *bugs*.

D'altra banda, també cal **assegurar que les traduccions de l'aplicació són correctes** en totes les llengües, ja que una mala traducció podria proporcionar informació incorrecte i suposar comentaris negatius que també farien baixar el posicionament de l'aplicació.

Com donar visibilitat a l'App

Tot seguit es proporcionen una sèrie de consells perquè l'aplicació sigui destacada en els *markets* i tingui el major nombre de descàrregues possible:

1. **La icona** de l'aplicació és el primer que els usuaris veuran. Ha de destacar, ser **agradable** i estar **relacionada** amb la cerca.
2. **Les imatges** que es mostren al *market* han de **mostrar la principal funcionalitat** de l'aplicació, ser comprensibles en l'idioma de l'usuari i **estar ordenades** per prioritat.
3. Es pot aprofitar el **vídeo per mostrar les funcionalitats clau** de l'aplicació, i acabar de convèncer l'usuari que se satisfaran les seves necessitats.
4. Cal **escollir bé les paraules** clau de l'aplicació per aparèixer en les primeres posicions de les cerques.
 - a. **Investigar la competència** i assegurar que la descripció **destaca funcionalitats que diferencien l'App** de la resta.
 - b. L'App Store permet posar keywords (paraules clau) que poden ajudar a trobar l'aplicació. És important **escollir les més adequades**.
5. Si es publica l'aplicació en varies regions, cal assegurar que està **ben traduïda als idiomes principals** de cada localització.
6. **Les ressenyes i valoracions són importants**. Si inclouen paraules clau de l'aplicació i si són positives, faran pujar l'aplicació en les cerques en el *Market*.
7. També es poden utilitzar **eines de posicionament** que ajudaran a guanyar descàrregues:
 - a) **Google Play Developer Console** és una eina gratuïta que proporciona Google i permet definir i optimitzar de manera fàcil i intuïtiva tots els elements de la fitxa de l'App, així com veure'n les estadístiques.
 - b) **App Annie** té característiques gratuïtes i de pagament. Permet seguir els rankings d'aplicacions per país i estimar els ingressos que tindrà l'App.
 - c) **The Tool** és una eina de pagament que permet monitoritzar rankings de keywords, les valoracions dels usuaris i gestionar la publicitat.
 - d) **Sensor Tower** és una eina de pagament que ajuda a pujar el posicionament de l'aplicació en les cerques, gràcies a l'estudi de les keywords.



8. Seguretat i protecció de dades

24

La seguretat al món del software sovint passa erròniament a un segon pla, prioritzant el nombre de funcionalitats o la facilitat d'ús. La seguretat de les Apps està relacionada amb la seguretat de les dades que aquestes gestionen, i per tant, és clau que es faci tot el possible per garantir tant la seva seguretat com la privacitat de les dades. També és important entendre que la seguretat no s'ha de tenir en compte únicament durant el desenvolupament del software, sinó que es un procés continu de millora que s'haurà de dur a terme durant tota la vida de l'App.

A continuació es destaquen aspectes rellevants a tenir en compte per tal de garantir que l'App a desenvolupar és segura:

- Identificar les **dades sensibles** que es tractaran i determinar el **tractament** que se'n farà, així com aplicar, per cada tipus, les **garanties de seguretat adequades**.
- S'han de valorar els aspectes de seguretat des de les **primeres fases del projecte**. Es recomana comptar amb un **expert en seguretat**, ja sigui intern o extern, que valori les necessitats de seguretat del projecte.
- Utilitzar sistemes alternatius d'autenticació d'usuaris com **Oauth** per facilitar l'accés amb credencials d'altres serveis com Google o Facebook. En cas que s'utilitzin credencials pròpies, cal obligar a que aquestes tinguin uns **requisits mínims de longitud i complexitat**, i és imprescindible guardar aquestes **dades encriptades** al servidor. També es pot valorar la possibilitat de validació utilitzant elements biomètrics com empremtes digitals.
- Si l'aplicació es comunica amb sistemes externs, com per exemple serveis web, és molt important que tota la informació enviada estigui xifrada usant protocols de transport segurs com **TLS**. També cal que totes les comunicacions segures siguin amb servidors que estiguin identificats per **certificats digitals vàlids**.
- Si s'emmagatzemen dades sensibles al dispositiu, cal garantir que aquestes estan **encriptades**.
- L'aplicació ha de sol·licitar sobre el terminal **només aquells permisos que són imprescindibles**, d'aquesta manera en cas que l'aplicació sigui compromesa es limita l'accés al dispositiu. A més, alguns usuaris, conscients dels riscos de seguretat existents, no estaran disposats a donar determinats permisos a l'aplicació si no els consideren realment necessaris.
- Cal **verificar** totes les **vies d'entrada d'informació** i validar que només es pugui introduir allò que s'espera, incloent el tipus de dades introduïdes i la seva longitud dels diferents camps. En aplicacions client – servidor, quan es tracta de seguretat, cal fer un èmfasi especial en les validacions en el servidor. Les validacions en el client milloren l'experiència de l'usuari, però són fàcilment eludibles, de manera que cal replicar-les al servidor.
- Si l'aplicació treballa amb una base de dades és susceptible de patir un atac de **SQL injection**, que és un dels tipus d'atacs més comuns i que pot implicar la pèrdua, corrupció o robatori de la totalitat de les dades. Aquest atac consisteix en introduir codi SQL en paràmetres d'entrada de dades, com pot ser un camp per posar el password en un login, de manera que es modifica la consulta realitzada a la base de dades podent realitzar les accions que vulgui l'atacant. Per evitar això és important validar totes les cadenes de text que s'usen en accions contra la base de dades, impedit la introducció de codi SQL en aquestes.
- En les fases de test de l'App, no només realitzar **proves de les funcionalitats**, sinó també realitzar **tests de penetració** (pentesting) per verificar que no és vulnerable a les tipologies d'atacs coneguts. Per realitzar un pentesting complet es poden seguir diferents guies o frameworks, com per exemple el realitzat per la **OWASP** (Open Web Application Security Project), organització mundial sense ànim de lucre centrada en millorar la seguretat del software.
- Verificar que es compleix amb tot el que estableixen les lleis aplicables en matèria de protecció de dades. A nivell català, apliquen la llei espanyola LOPD (Ley Orgánica de Protección de Datos) i el reglament europeu **RGPD** (Reglamento General de Protección de Datos), que es poden consultar al web de l'**AEPD** (Agencia Española de Protección de Datos). Aquestes lleis no només regulen aspectes tècnics relatius a la protecció de les dades, sinó que també inclouen aspectes relatius als drets dels usuaris que s'han de garantir, o als consentiments que cal demanar per realitzar el tractament de dades personals.

Reglament General de Protecció de Dades

Aquest reglament aplica a totes les empreses que tractin dades de ciutadans europeus i regula la seguretat de les dades personals. A continuació se'n presenten els aspectes principals a mode de resum.

Dades personals

El RGPD defineix la dada personal com a qualsevol informació relacionada amb una persona identificable, que pot ser identificada directa o indirectament per referència d'un identificador. Com a exemples de dades personals es considera el nom, l'adreça, la localització GPS, la informació sanitària o els ingressos monetaris d'una persona.

Dins de les dades personals, el RGPD diferencia aquelles que es consideren sensibles i per tant requereixen les màximes mesures de seguretat. Aquest tipus de dades inclouen les relatives a:

- Opinions polítiques.
- Afiliació sindical.
- Conviccions religioses.
- Conviccions filosòfiques.
- Origen racial o ètnic.
- Dades relatives a la salut.
- Vida sexual.
- Dades genètiques.
- Dades biomètriques.
- Orientació sexual.

El RGPD en termes generals prohibeix el tractament d'aquestes dades excepte en certs casos, com pot ser tenir-ne el consentiment explícit de tractament.

Processament de dades

El processament de dades (o **tractament**) cobreix un ampli rang d'operacions sobre dades personals, ja siguin manuals o automàtiques, incloent la **recollida, gravació, organització, estructuració, emmagatzemament, adaptació o alteració, recuperació, consulta, ús, transmissió o disseminació**. Alguns exemples considerats processament de dades són: enviar e-mails promocionals, penjar fotos d'una persona a un web; emmagatzemar IPs o MACs dels usuaris; consulta d'una base de dades de contactes.

Drets del subjecte de les dades

Quan es realitza un processament de les dades d'un subjecte cal garantir els següents aspectes clau:

- **Consentiment clar i explícit:** és necessari comptar amb un consentiment explícit per tractar dades personals. El formulari de consentiment no pot ser ambigu i ha d'usar un llenguatge planer i comprensible. Cal, per exemple, posar un check per consentir cadascun dels tractaments que es faran de les dades i no serveix incloure un "Accepto termes i condicions d'ús" genèric. Els consentiments previs al reglament que no compleixin aquest punt **s'han de tornar a sol·licitar**. En el cas dels menors de 16 anys, el consentiment l'ha de proporcionar un tutor legal, i aquesta edat pot ser reduïda localment en alguns Estats fins a 13 anys.
- **Dret d'accés:** el subjecte ha de tenir accés a totes les seves dades en un format amigable i comprensible, conjuntament amb informació addicional detallada a l'article 15 del RGPD (com els propòsits del processament).
- **Dret de rectificació:** Els subjectes han de poder modificar totes les dades del seu perfil incloses les que s'han obtingut a través d'altres fonts.
- **Dret a l'oblit:** Cal garantir que s'eliminen físicament totes les dades d'una persona si aquesta així ho demana, i deixar de transmetre-les a tercers, sense demora injustificada. Aquest dret s'aplicarà sempre que no en comprometi d'altres com la llibertat d'expressió.

- **Dret a restricció del processament de dades:** En aquest cas s'eliminen totes les dades del subjecte de manera lògica, fent que no siguin visibles, de cap manera, des de l'aplicació.
- **Dret a la portabilitat de dades:** cal poder exportar totes les dades en un format estructurat
- **Notificar les activitats de processament a les autoritats de protecció de dades locals** de cada país o regió.
- **Minimització de dades:** Només sol·licitar aquelles dades que són estrictament necessàries pel tractament concret que se'n fa.
- **Mantenir les dades només el temps necessari:** Quan ja no siguin necessàries les dades pel propòsit pel qual s'han recollit, cal eliminar-les, tot i que s'ha de tenir en compte si cal mantenir-les per altres motius legals.
- **Cada tipus de tractament requereix un consentiment independent:** Cada tipus de tractament que es realitzi de les dades, o si es decideix posteriorment tractar les dades per un nou propòsit, requereix d'un consentiment específic.
- **Notificar als usuaris si les seves dades són transferides fora de la unió europea.**
- **Notificació d'intrusions:** Cal notificar als afectats i a l'autoritat supervisora les intrusions que puguin afectar als drets o llibertats dels individus. La notificació s'ha de realitzar el més aviat possible, desitjablement abans que passin **72 hores**.

Mesures relatives a les dades personals

El reglament marca que per les dades personals cal aplicar mesures apropiades, tenint en compte l'estat del art, els costos d'implementació, la naturalesa, l'abast, el context i objectiu del processament, així com els riscos i la severitat d'aquests pels drets i les llibertats de les persones.

Pel que fa a les dades personals sensibles, cal aplicar una sèrie de mesures que van més enllà i que es resumeixen a continuació:

- Disposar d'un registre actualitzat de tots els tractaments que es realitzin a l'entitat, documentant-los i aportant un conjunt d'informació específica.
- Si es realitza un tractament de dades personals de les categories especials (dades sensibles) a gran escala, cal realitzar també una avaluació de l'impacte. Aquesta avaluació ha de descriure els processaments realitzats i la seva proporcionalitat respecte als objectius. També ha de detallar tots els riscos respecte els drets i les llibertats dels subjectes relacionats amb aquests tractaments, així com les mesures preses per minimitzar aquests riscos.
- També cal prendre mesures de seguretat pertinents, com:
 - Disposar d'un llistat de persones autoritzades a tractar les dades.
 - Disposar d'un registre d'accessos amb data i persona que ha accedit.
 - Xifrar les dades en les comunicacions i en l'emmagatzemament.
 - Utilitzar pseudoanonimització de les dades si aquestes s'usen per exemple en entorns de test o per fer recerca.
 - No incloure dades personals als fitxer de log.
 - Si l'aplicació interacciona amb tercers per compartir dades, cal garantir que aquests també compleixen la normativa.

9. Accessibilitat

L'accessibilitat en l'àmbit digital consisteix en vetllar perquè tota la informació disponible, tant a la xarxa com a les aplicacions, així com el propi ús dels dispositius tecnològics, estiguin a l'abast de totes les persones, independentment de les seves condicions, característiques o capacitats. Aquest accés s'ha de poder realitzar utilitzant o no sistemes de suport, com els lectors de pantalla **VoiceOver** per a iOS o **Talk-Back** per Android. Els sistemes operatius com iOS o Android actualment ja ofereixen moltes opcions d'accessibilitat per a millorar l'experiència de l'usuari, i és important garantir que l'App a desenvolupar n'és compatible.

Sovint s'identifica l'adaptació dels programes, webs i Apps perquè siguin accessibles com un procés car i complex. Aquesta complexitat, i el cost que se'n deriva, queden extremadament minimitzats o completament anul·lats si des del primer moment es tenen en compte les directrius bàsiques d'accessibilitat. D'aquesta manera, l'element principal a tenir en compte en matèria d'accessibilitat és comprendre que s'ha de contemplar ja en el moment d'elaborar els wireframes i en l'anàlisi funcional de l'App.

L'organisme internacional encarregat de promoure l'accessibilitat a Internet és el **W3C** (*World Wide Web Consortium*), en el seu grup de treball **WAI** (*Web Accessibility Initiative*). Aquesta organització publica una sèrie d'estàndards d'accessibilitat a tenir en compte i que queden recollits a les guies WCAG (**Web Content Accessibility Guidelines**). Hi ha eines que permeten comprovar el nivell d'acompliment d'aquests estàndards, així com testejar Apps amb diferents tipus de filtres per a donar resposta a totes les necessitats de visualització. També n'existeixen d'altres per a testejar la qualitat de l'accessibilitat de les aplicacions i que ofereixen propostes de canvi. A l'hora de desenvolupar una App, és important considerar els complements que estan específicament dissenyats per a ajudar a utilitzar les aplicacions, com els lectors de pantalla, les eines de dictat, o els que permeten augmentar la mida del text.

L'octubre de 2017, la Fundació TIC Salut i Social va col·laborar en l'edició de la guia "**Accessibilitat Digital: Les TIC per a tothom**", elaborada per la Taula d'entitats del Tercer Sector Social de Catalunya a través del seu projecte **m4Social**. En aquesta guia s'hi poden trobar bones pràctiques i recomanacions per a elaborar webs, Apps i continguts accessibles, així com un llistat de referències a eines i complements existents.

10. Usabilitat



Es defineix usabilitat com la capacitat d'una persona per a utilitzar qualsevol objecte o eina artificial amb un objectiu concret. En aquest àmbit, l'objecte o eina és una App, que haurà de seguir uns criteris determinats per tal que l'usuari la pugui utilitzar de manera còmode i intuïtiva. En el present apartat es destaquen una sèrie de recomanacions i consideracions adreçades a garantir que l'App a desenvolupar sigui usable, i relatives a la navegació, la llegibilitat i la coherència, el contingut i la previsió de la mateixa.

Navegació

L'aplicació hauria de proporcionar a l'usuari una manera **intuïtiva, ràpida i senzilla** d'interactuar-hi, complint sempre que sigui possible la **regla dels 3 clics**. Amb aquesta aproximació, es garanteix que l'accés a tota la informació es pot realitzar en un màxim de 3 passos. També resulta important **reduir el procés d'aprenentatge** de l'usuari al mínim, i oferir **recursos d'ajuda** com un tutorial que el guiï pas a pas en la utilització de les funcionalitats de l'App. Determinar si serà possible visualitzar la interfície tant en horitzontal com en vertical implicarà uns requeriments específics de desenvolupament, pel que serà necessari tenir-ho en compte en el disseny de la mateixa.

Pel que fa a l'àrea de clic, no s'han de fer botons inferiors a 9 mm d'amplada i alçada (l'equivalent a uns 44px), ja que serien massa petits en relació amb les proporcions dels dits dels usuaris. És igualment important respectar la separació entre els botons per evitar interaccions no desitjades, com pressionar-ne dos a la vegada, així com assegurar que cada element té l'aparença d'allò que representa. D'aquesta manera, tot botó, enllaç o zona clicable ha de ser **identificable** per part de l'usuari, ja sigui per la forma que espera que tingui o per un color que sigui coherent amb el codi establert dins de l'univers de la plataforma.

Llegibilitat i coherència

El **disseny** de la interfície ha de ser el més **senzill i pràctic** possible, sense elements que no aportin res, fet que no està renyit amb presentar un disseny atractiu. És més fàcil per a l'usuari recordar i interactuar amb una estructura senzilla, que amb una de complexa que presenti moltes opcions i informació que pot generar estrès, soroll a la pantalla i, fins i tot, una presa de decisions errònia. No es pot perdre de vista que totes les accions i elements interactius han de ser fàcils de trobar.

L'ús del color és un factor important per ajudar a jerarquitzar la informació i endreçar el contingut dins de les aplicacions. És recomanable fer servir **la regla de "60-30-10"**, on: el 60% és pel color principal (com per exemple un color de fons que ajudi a la lectura); el 30% és per a un color secundari que ajudi a complementar i donar contrast a la part visual; i finalment, un 10% es reserva als colors d'accent (links, botons, elements importants, etc.). Els colors haurien de formar part d'una gama cromàtica que proporcioni alt contrast i defineixi clarament els elements principals. Es poden emprar recursos cromàtics que segueixin el codi de colors universal, com poden ser el vermell per a l'error o la cancel·lació, i el verd per denotar correctesa o confirmació.

Resulta igualment important tenir una llegibilitat alta perquè els missatges es puguin transmetre fàcilment i de manera clara. És recomanable fer servir les tipografies del propi sistema operatiu, ja que aquestes estan especialment dissenyades per aquest propòsit. A més a més, és aconsellable fer servir entre **1 i 3 tipus de tipografia** amb un **màxim de 3 mides** o estils diferents, així com aplicar **diferents estils i jerarquies** per ajudar a endreçar la informació (com l'ús de títols, subtítols o camps de text).

Si s'ha de fer servir iconografia per representar les funcions habituals d'una App, el més recomanable és no fer un disseny nou, sinó **imitar el disseny estàndard** que fan servir pel mateix altres aplicacions que l'usuari segurament ja coneix. Aquestes icones haurien de tenir una mateixa coherència gràfica i funcional, amb l'objectiu que l'usuari les pugui identificar i relacionar fàcilment.

Pel que fa al disseny, és especialment rellevant **tenir en compte el sistema operatiu** pel qual s'està desenvolupant l'aplicació, ja que alguns criteris d'usabilitat s'hi han adaptat específicament. En aquest sentit, si el sistema operatiu disposa de botons per tal de navegar, per exemple, cal que l'App hi sigui compatible. Per exemple, a Android sempre ha de funcionar el botó endarrere del SO, encara que ja es tingui un botó específic per dur a terme aquesta funció dins de l'aplicació.

Contingut

El contingut d'una App es pot estructurar de diferents maneres. En funció de quin sigui el seu objectiu final, la interacció amb l'usuari pot ser:

- **Jeràrquica:** existeix un pantalla principal (o índex) des d'on s'accedeix a la resta de pàgines, les quals també en poden enllaçar d'altres (a mode de subpàgines), creant una jerarquia. En aquesta mena d'organització ha d'existir un menú que permeti navegar-hi.
- **Lineal:** des d'una pantalla es pot accedir a la següent i a l'anterior, com si fossin les pàgines d'un llibre. No és recomanable si el nombre de pàgines és molt elevat, però podria servir per exemple per un tutorial.
- **Lineal amb jerarquia:** és una estructura híbrida, en què l'organització és jeràrquica, però que permet navegar de manera lineal per pàgines que estan al mateix nivell.
- **Xarxa:** no hi ha cap ordre aparent. No és una estructura recomanada quan existeix un gran volum de pàgines perquè desorienta a l'usuari.

En qualsevol d'aquests casos el contingut ha de tenir **descripcions concises, precises i clares**, evitant la redundància. Aquest criteri és especialment important per a la visualització de continguts en dispositius mòbils, degut a l'espai limitat que tenen respecte, per exemple, els ordinadors de sobretaula.

Previsió

És recomanable generar les especificacions que detallin totes les funcionalitats que ha d'incloure l'App per tal de satisfer les necessitats de l'usuari, des del punt de vista funcional (i no només tècnic). Aquestes especificacions funcionals han d'incloure els següents elements:

- Detallar les situacions crítiques que puguin implicar les funcionalitats de l'App.
- Descriure les funcions que es volen implementar.
- Evitar detalls tècnics, excepte si és estrictament necessari.
- Consistir en descripcions concises, precises i clares, evitant la redundància.

Per tal de minimitzar els possibles errors d'usuari, és imprescindible mirar de detectar-los amb anterioritat i oferir un sistema de prevenció per evitar-los. Un exemple és l'auto-completat del buscador Google, que realitza una proposta de cerca corregida. També cal tenir en compte casos en què l'usuari pot cometre errors propis, i no de sistema, on ha de poder trobar sempre de manera ràpida l'opció que desfà l'acció que acaba de realitzar.

També és important que una App pugui ser usada per tot tipus d'usuaris. Si aquesta té funcions avançades, no s'hauria d'obligar a l'usuari inexpert a fer-les servir. Seguint amb l'exemple, el cercador Google incorpora operadors per tal de filtrar millor les cerques, però no és necessari que l'usuari els conegui per utilitzar-lo.

L'aplicació no hauria de necessitar documentació per tal de que l'usuari sigui capaç d'utilitzar-la, ja que un dels conceptes bàsics d'usabilitat és que aquesta sigui intuïtiva. Tot i així, és necessari donar suport a través del web de l'aplicació o l'aplicació mateixa, mitjançant una secció de preguntes freqüents o icones d'interrogació al costat de certes funcions. D'aquesta manera, és recomanable disposar d'una documentació de les funcionalitats de l'App i de com s'utilitzen.

11. Experiència d'usuari / UX

L'experiència d'usuari és el conjunt d'elements relacionats amb la interacció de l'usuari, que fan que aquest tingui una percepció positiva o negativa d'una aplicació. No és un factor que depengui només de la usabilitat, l'accessibilitat o el disseny visual, sinó que també incorpora aspectes relatius a les emocions, les preferències o la sensació de fiabilitat que té del producte.

A continuació es destaquen aspectes que poden influir en la percepció final que l'usuari tindrà de l'aplicació:

Tenir en compte el primer ús: S'ha de definir la informació que es mostrarà quan l'usuari encara no hagi introduït dades. Normalment no és bona idea deixar les parts de les pantalles simplement "buides", sense informació prèvia o indicacions relacionades. Alguns exemples en serien la visualització de la pestanya "preferits" quan l'usuari encara no n'ha definit, l'aparença del cistell de compra quan està buit o l'absència de resultats de cerca. En aquests casos, es pot solucionar amb missatges informatius com: "La llista de preferits és buida" o "No s'han afegit productes al cistell", però resulta més útil aprofitar també l'espai per ensenyar a l'usuari com omplir-ho.

Missatges d'error: S'han de preveure el màxim d'errors que es puguin produir a l'aplicació i no mostrar-ne descripcions incomprensibles per a l'usuari. En aquest sentit, cal identificar elements que poden desencadenar aquests errors, com els camps d'informat obligatori, les restriccions de tipus de dades acceptades o la mida màxima dels fitxers; i implementar sistemes de prevenció dels mateixos. En el cas que es produeixi un error a l'aplicació, cal informar-ne a l'usuari, tot indicant com el pot solucionar. L'usuari ha de saber en tot moment quin és l'estat del sistema, per evitar que es senti confós o perdut.

Donar l'opció de desfer canvis. Tant si s'elimina un element com si es modifica, sempre s'ha de tenir en compte que l'usuari es pot equivocar i/o pot voler desfer els canvis. Així doncs, és una bona opció incorporar un botó de "desfer" i/o una paperera on van a parar els elements eliminats i des de la qual es puguin restaurar. Informar a l'usuari que té l'opció de desfer les accions que duu a terme fa que es pugui mostrar més confiat a l'hora d'utilitzar l'App i que no tingui por a explorar-la.

Incorporar missatges de confirmació. Abans de dur a terme accions permanents, és millor obrir un missatge per confirmar que l'usuari vol fer el que realment està indicant. Sovint es mostra un simple botó "Acceptar" o "Guardar" del qual no se'n coneixen les conseqüències que comportarà. Així doncs, recordar a l'usuari que l'acció que està duent a terme és permanent, és una bona opció per tal que es senti més còmode amb l'ús de l'App.

Fer els botons i menús familiars. Dissenyar-los de manera que siguin intuïtius i coneguts per l'usuari. Per exemple, utilitzar les mateixes icones que fan servir les aplicacions més conegudes o les del mateix àmbit és una bona opció. De la mateixa manera, fer servir una interfície gràfica semblant a la majoria d'aplicacions o seguir un patró dins de l'aplicació, faran que l'usuari utilitzi l'eina més fàcil i intuïtivament.

Gestionar els temps d'espera. La no-acció és un dels elements que més molesta als usuaris, ja que no els resulta agradable haver d'esperar per fer allò que han indicat. Cal notificar d'alguna manera que l'aplicació està realitzant accions que poden trigar un cert temps (per evitar que l'usuari arribi a la conclusió que l'App s'ha penjat, per exemple). El més habitual és utilitzar un element que mostri el progrés en forma de barra, rellotge o rodeta de càrrega, però també es poden amenitzar aquestes estones perdudes amb continguts més atractius com acudits, consells d'utilització o curiositats de l'App.

12. Estàndards de comunicació i representació d'informació

En l'entorn mHealth i mSocial coexisteixen aplicacions que permeten recollir diferents tipus d'informació, de manera que les dades generades pels ciutadans queden repartides en un conjunt d'Apps sense que es puguin intercanviar entre elles. Cada aplicació utilitza el seu format intern d'emmagatzemament i representació, així com interfícies pròpies de comunicació amb d'altres sistemes (normalment, limitada a plataformes propietàries de la mateixa companyia). En aquest escenari, un ciutadà pot estar utilitzant dues aplicacions que permeten recollir la mateixa dada (com el pes), sense que els valors d'una i altra es puguin intercanviar, comparar ni explotar globalment. Per tal de superar aquestes limitacions, és necessari posar-se d'acord en com s'ha de produir l'intercanvi. Els estàndards representen aquests acords, que es produeixen també a nivell internacional, i que permeten assolir la interoperabilitat entre sistemes, dispositius, aplicacions, etc.

Al sector Salut en general, i al SISCAT (Sistema sanitari integral d'utilització pública de Catalunya) en particular, ja fa anys que s'està treballant la interoperabilitat de sistemes, dispositius i serveis, com a resposta a la necessitat de compartir informació entre centres, fins i tot de diferents nivells assistencials. L'assoliment d'aquest intercanvi implica un treball exhaustiu d'estandardització, que en el cas del SISCAT, s'ha liderat des de l'**OFSTI** (Oficina d'Estàndards i Interoperabilitat) de la Fundació TIC Salut Social, i que s'està reproduint en d'altres àmbits com el social o el de la mobilitat.

Definició del concepte d'interoperabilitat

La interoperabilitat és la capacitat de compartir informació entre components (com sistemes o dispositius) sense que es perdi el seu significat. Aquesta comunicació ha de garantir l'intercanvi coherent de les dades entre departaments, organitzacions, nivells assistencials o regions, com països o continents. L'objectiu principal és proporcionar als professionals tota la informació rellevant dels seus pacients per assegurar que el procés de presa de decisions es produeix d'una manera segura, eficient i eficaç. La interoperabilitat garanteix l'accés a la informació independentment del lloc en què s'hagi registrat, afavorint-ne el reaprofitament, minimitzant punts cecs i assegurant el continu assistencial.

Aquesta capacitat, no és binària (en termes d'assolir-se o no), sinó que té diferents capes, nivells o dimensions, les quals es presenten a continuació:

- **Interoperabilitat tècnica:** En aquest nivell es troben les tecnologies i protocols que permeten establir comunicació entre els components.
- **Interoperabilitat funcional o sintàctica:** Afegeix intercanvi d'informació a la comunicació. Els estàndards d'aquest nivell defineixen l'estructura i format de la informació a intercanviar.
- **Interoperabilitat semàntica:** És el nivell d'interoperabilitat que té per objectiu garantir que la informació que s'intercanvia sense que es perdi el seu significat i que es pugui utilitzar al component receptor, com si s'hi hagués generat.
- **Interoperabilitat legal:** Permet garantir que es compleix la legislació vigent en cada agent implicat en l'intercanvi. Aquesta capa és especialment important en projectes transfronterers on l'intercanvi es produeix entre països o regions amb marcs legislatius diferents.
- **Interoperabilitat organitzacional:** Afegeix la capa de procés, de manera que l'intercanvi i ús de la informació estigui alineada amb els fluxos de treball de les institucions involucrades.

Cadascuna d'aquestes dimensions es pot garantir amb l'ús d'estàndards explícitament dissenyats per a aquest propòsit. En aquest apartat es farà èmfasi en les capes sintàctica i semàntica d'interoperabilitat, presentant els principals estàndards que es poden utilitzar per garantir-les en l'entorn mòbil.

Interoperabilitat sintàctica i l'estàndard FHIR

FHIR (*Fast Healthcare Interoperability Resources*) és un estàndard d'intercanvi d'informació desenvolupat i publicat per l'organització internacional **HL7**. Actualment es troba en estat d'esborrany, concretament en la fase STU3 (*Standard for Trial Use*), una versió quasi definitiva però que ja compta amb múltiples experiències d'implementació.



L'estàndard FHIR suposa un canvi de concepte respecte els estàndards anteriors d'HL7 com CDA R2 (*Clinical Document Architecture Release 2*) o la missatgeria 2.x, ja que s'ha dissenyat específicament per facilitar-ne l'aprenentatge per part dels desenvolupadors, la implementació, l'adaptació i el tractament. Les peces a intercanviar (anomenades recursos) també s'han simplificat, de manera que permeten enviar només la informació estrictament necessària en un format més àgil i adequat també per a l'entorn mòbil. Aquestes característiques han fet que FHIR s'assenyali com a l'estàndard de referència per a l'intercanvi d'informació a l'entorn mHealth.

FHIR utilitza una arquitectura **REST** (*Representational State Transfer*) per tal de dur a terme l'intercanvi de dades, amb estàndards estructurals com **XML** o **JSON**. La seva implementació permet tenir un servidor on emmagatzemar les dades, i una sèrie de serveis estàndard per a enviar-les i recuperar-les. Per facilitar la implementació de FHIR i el desenvolupament d'aplicacions que l'adopten, **s'han creat diverses APIs**, les més conegudes de les quals són **HAPI** (per al llenguatge Java) i **Firely** (per a .Net). Aquestes APIs també permeten la creació d'un servidor basat en FHIR, preparat específicament per emmagatzemar-ne els recursos.

La seguretat i el control d'accés a les dades generades és essencial per tal que l'aplicació **compleixi amb les normatives corresponents** i proporcioni un servei segur i fiable als usuaris. Les especificacions **SMART on FHIR** permeten cobrir aquestes necessitats, habilitant una integració segura amb els EHR (*Electronical Health Records*), portals, etc. i utilitza estàndards com **OAuth2** pels permisos o **OpenID** per a l'inici de sessió.

Com s'ha indicat anteriorment, els components bàsics de l'estàndard FHIR s'anomenen recursos i representen la unitat bàsica d'informació que es pot compartir. Cadascun d'aquests components permet representar una entitat concreta (com Pacient, Problema de salut o Observació) i té associades una sèrie de variables. A continuació es presenta una part de la definició del recurs Pacient, a mode d'exemple:

Structure

Name	Flags	Card.	Type
Patient			DomainResource
identifier	Σ	0..*	Identifier
active	? Σ	0..1	boolean
name	Σ	0..*	HumanName
telecom	Σ	0..*	ContactPoint
gender	Σ	0..1	code
birthDate	Σ	0..1	date
deceased[x]	? Σ	0..1	
deceasedBoolean			boolean
deceasedDateTime			dateTime
address	Σ	0..*	Address
maritalStatus		0..1	CodeableConcept
multipleBirth[x]		0..1	
multipleBirthBoolean			boolean
multipleBirthInteger			integer

Per a cadascuna de les variables que formen el recurs, s'especifica la seva cardinalitat (el nombre mínim i màxim de vegades que pot aparèixer dins el recurs) i el seu tipus. FHIR conté molts tipus diferents de variables, que poden anar des d'un nombre enter fins a informació codificada utilitzant vocabularis controlats com SNOMED CT. El llistat de recursos disponibles, així com la seva definició, es poden trobar al següent enllaç: Resource Index.

Interoperabilitat semàntica i l'estàndard SNOMED CT

Pel que fa a la capa semàntica d'interoperabilitat, els estàndards que permeten assolir-la tenen per objectiu que la informació intercanviada no perdi el seu significat, de manera que la normalitzen utilitzant vocabularis controlats. Aquesta estandardització permet representar els conceptes per tal que es puguin intercanviar, comparar i explotar, encara que procedeixin de diferents fonts.

Existeixen diferents vocabularis controlats per a normalitzar informació, i és important saber identificar quin és el seu propòsit per utilitzar-los correctament. Les anomenades **classificacions** com la CIM-10-MC/SCP (diagnòstics i procediments), CIAP-2 (atenció primària), ATC (principis actius) o NANDA (infermeria), s'han dissenyat per a agrupar i explotar la informació, de manera que es desaconsella el seu ús per al registre i la representació amigables i acurats. Per contra, les **terminologies** com LOINC (laboratori) o SNOMED CT estan específicament dissenyades per registrar la informació amb el màxim nivell de detall, i garantir-ne la interoperabilitat. En l'àmbit de la interoperabilitat semàntica, és habitual (i necessari) realitzar mapejos (equivalències) entre vocabularis per poder-ne combinar els avantatges i utilitzar-los cadascun per al seu propòsit concret.

La terminologia clínica SNOMED CT (Systematized Nomenclature of Medicine Clinical Terms) és un estàndard semàntic **internacional i multilingüe** que permet normalitzar conceptes principalment de la Salut, però que ja s'està començant a aplicar a d'altres àmbits com el Social o l'mHealth. El fet que el model conceptual d'aquest vocabulari estigui dissenyat per a oferir una visió transversal del ciutadà, fa que contingui conceptes que no són estrictament clínics. D'aquesta manera, SNOMED CT es pot utilitzar com a una terminologia "pont" o "d'enllaç" entre els diferents tipus de professionals.

SNOMED International (IHTSDO) és l'entitat sense ànim de lucre que té la propietat de SNOMED CT, així com la responsabilitat de distribució i actualització a nivell mundial. Al SISCAT, és l'OFSTI qui en té la responsabilitat de distribució i adopció a les necessitats del territori. Aquesta terminologia es va crear al 1965 per part del Col·legi d'Anatomopatòlegs Americà i actualment ja compta amb quasi mig milió de conceptes de diferents especialitats. En concret, SNOMED CT presenta els següents 19 eixos d'alt nivell, també anomenats jerarquies:


- ▼ SNOMED CT Concept
 - ▶ ambiente o localización geográfica (medio ambiente / localización)
 - ▶ calificador (calificador)
 - ▶ componente del modelo de SNOMED CT (metadato)
 - ▶ concepto especial (concepto especial)
 - ▶ contexto social (contexto social)
 - ▶ elemento de registro (elemento de registro)
 - ▶ entidad observable (entidad observable)
 - ▶ espécimen (especimen)
 - ▶ estadificaciones y escalas (escala de estadificación)
 - ▶ estructura corporal (estructura corporal)
 - ▶ evento (evento)
 - ▶ fuerza física (fuerza física)
 - ▶ hallazgo clínico (hallazgo)
 - ▶ objeto físico (objeto físico)
 - ▶ organismo (organismo)
 - ▶ procedimiento (procedimiento)
 - ▶ producto biológico/farmacéutico (producto)
 - ▶ situación con contexto explícito (situación)
 - ▶ sustancia (sustancia)


Les idees a SNOMED CT es representen a través de **conceptes** organitzats jeràrquicament, que poden tenir diferents **descripcions** (amb sinònims) i que estan connectats entre ells a través de **relacions**. Aquest vocabulari està disponible en anglès i castellà, i des de l'**OFSTI** ja s'està treballant amb el Departament de Planificació Lingüística del Departament de Salut de la Generalitat de Catalunya i el Termcat per anar-lo traduint a català (actualment ja se'n han traduït prop de 2400 descripcions).


SNOMED CT contempla diferents mecanismes per a facilitar-ne la seva adopció, entre els quals es destaquen els **subconjunts** i les **extensions**. Els primers estan destinats a agrupar components (conceptes, descripcions o relacions) amb un propòsit determinat i, per tant, permeten utilitzar només aquells elements que realment són necessaris. Les extensions, en canvi, es poden veure com a versions locals de la terminologia i aporten la flexibilitat per donar resposta a les necessitats reals. Dins d'una extensió, es poden crear conceptes, descripcions, subconjunts, etc. seguint les directrius de SNOMED International i sense malmetre la interoperabilitat dels continguts. Des de l'OFSTI es manté l'extensió catalana de SNOMED CT, amb quasi 3.000 conceptes que no es troben al vocabulari, i que es distribueix entre una i dues vegades l'any, en funció de la demanda.

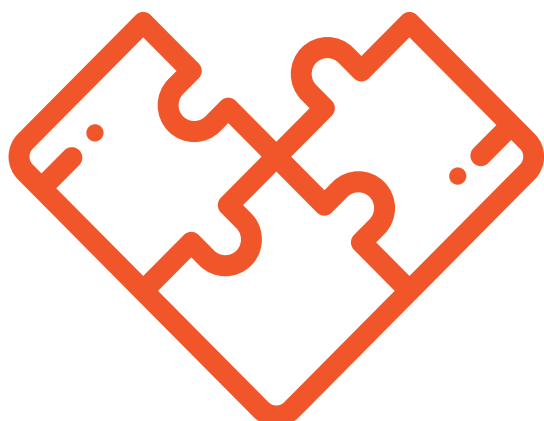
Per a facilitar l'ús de SNOMED CT en l'entorn mòbil, s'ha creat un subconjunt de variables de mobilitat mHealth i mSocial, que compta amb més de 50 conceptes, procedents de 6 Apps.

A continuació es presenten alguns de conceptes que formen part del subconjunt, a mode d'exemple:

● **práctica de natación (calificador)** ☆ 
SCTID: 20461001
20461001 | práctica de natación (calificador) |
en Swimming (qualifier value)
en Swimming
es práctica de natación
es práctica de natación (calificador)

● **distancia a pie (entidad observable)** ☆ 
SCTID: 165263003
165263003 | distancia a pie (entidad observable) |
en Walking distance
en Walking distance (observable entity)
es distancia a pie
es distancia a pie (entidad observable)

● **estado de curación de herida (entidad observable)** ☆ 
SCTID: 406214003
406214003 | estado de curación de herida (entidad observable) |
en Wound healing status (observable entity)
en Wound healing status
es estado de curación de herida (entidad observable)
es estado de curación de herida



Des de l'**Oficina mHealth.cat** i l'**OFSTI** oferim suport per normalitzar les variables de les aplicacions mòbils, fent que aquestes siguin interoperables a nivell internacional.

13. *Medical Device:* L'aplicació mòbil com a producte sanitari

L'ús creixent de les Apps de salut, ha implicat que el debat sobre quan s'han de considerar producte sanitari hagi anat en augment. Identificar en la fase de disseny si l'aplicació mòbil que s'està desenvolupant és o no un producte sanitari és fonamental, donat que, en el cas ho sigui, caldrà complir la legislació aplicable, la qual té per objectiu principal garantir la seguretat del pacient.

A nivell legislatiu, el producte sanitari està actualment regulat per la Directiva 93/42/CEE, transposada al Reial Decret 1591/2009. També cal tenir en compte que el producte sanitari per a diagnòstic in vitro està actualment regulat per la Directiva 98/79/CE, transposada als Reial Decret 1662/2000 i Reial Decret 1193/2012.

En aquest apartat es facilitaran una sèrie de punts clau que permetran ajudar a identificar si una App és o no un Medical Device (també conegut com a producte sanitari).

Definició de producte sanitari

La Directiva 93/42/CEE defineix un producte sanitari com:

“Qualsevol instrument, dispositiu, equip, programari informàtic, material o altra article, utilitzat sol o en combinació, inclosos els programes informàtics destinats pel seu fabricant a finalitats específiques de diagnòstic i/o teràpia i que intervinguin en el seu bon funcionament, destinat pel fabricant a ser utilitzat en éssers humans amb fins de:

- *diagnòstic, prevenció, control, tractament o alleugeriment d'una malaltia,*
- *diagnòstic, control, tractament, alleugeriment o compensació d'una lesió o d'una deficiència,*
- *investigació, substitució o modificació de l'anatomia o d'un procés fisiològic,*
- *regulació de la concepció, i que no exercixin l'acció principal que es desitgi obtenir a l'interior o a la superfície del cos humà per mitjans farmacològics, immunològics ni metabòlics, però la funció de la qual puguin contribuir dits medis”.*

A més, la Directiva 98/79/CE defineix un producte sanitari per a diagnòstic in vitro com:

“Qualsevol producte sanitari consistent en un reactiu, producte reactiu, calibrador, material de control, estoig de material i materials, instrument, aparell, equip o sistema, utilitzat sol o en associació amb altres, destinat pel fabricant a ser utilitzat in vitro per a l'estudi de mostres procedents del cos humà, incloses les donacions de sang i teixits, sol o principalment amb el fi de proporcionar informació:

- *relativa a un estat fisiològic o patològic, o*
- *relativa a una anomalia congènita, o*
- *per a determinar la seguretat i compatibilitat amb receptors potencials, o*
- *per a supervisar les mesures terapèutiques”.*

La Unió Europea va publicar al 25 de maig del 2017 els Reglaments Europeus, EU 2017/745 per a producte sanitari i EU 2017/746 per a producte sanitari per a diagnòstic in vitro, que, passat el període establert d'adaptació de 3 i 5 anys respectivament, seran aplicables a tots els països de la Unió Europea.

Com identificar si una App és un producte sanitari

A continuació es presenten els criteris de decisió i els requisits reguladors relatius als productes sanitaris.

Criteris de decisió

Les aplicacions mòbils en salut que es consideren producte sanitari sovint s'anomenen Apps mèdiques (*Mobile Medical App*), mentre que la resta són considerades Apps de benestar (*Wellness App*). La finalitat de l'aplicació (ús previst o *intended use*) és cabdal per a determinar si és o no producte sanitari: en cas que sigui **diagnosticar, donar suport al diagnòstic o a les decisions clíniques, fer càlculs per a determinar el diagnòstic o el tractament, o utilitzar-se per a qualsevol propòsit mèdic**, l'aplicació podria ser considerada un producte sanitari. Per aquest motiu, és necessari disposar d'una acurada i completa definició de la finalitat prevista de l'aplicació, la qual permetrà concloure si és o no producte sanitari.

38

Les aplicacions que són producte sanitari han de complir la definició de producte sanitari (segons Directives 93/42/CEE, 98/79/CE o segons nous reglaments EU 2017/745 o EU 2017/746). Tanmateix poden actuar com a accessori d'un producte sanitari o bé, convertir directament la plataforma mòbil en un producte sanitari. Per contra, en cas que l'aplicació només **emmagatzemi, arxivi, transmeti, faci recerca simple de dades, o presenti resultats de dades sense alterar-les ni manipular-les**, no serà considerada un producte sanitari.

A continuació es llisten algunes preguntes que ajuden a decidir, juntament amb els criteris exposats anteriorment, si l'aplicació és o no producte sanitari:

- L'aplicació està destinada a interpretar (o facilitar la interpretació) de dades modificant o representant informació individual relacionada amb la salut?

- L'aplicació interpreta o altera dades? O tant sols gestiona informació estàtica?
- L'aplicació calcula dosis a prendre/injectar?
- L'aplicació indica condició mèdica, malaltia o percentatge individual de risc de patir una malaltia?

En qualsevol cas, una mateixa aplicació podrà ser, o no, producte sanitari en funció de la finalitat prevista. Per exemple: una aplicació per a monitoritzar el ritme cardíac, en el cas que estigui indicada exclusivament per a ús esportiu, no serà un producte sanitari; en canvi, la mateixa aplicació destinada a un ús mèdic sí que podrà ser un producte sanitari.

La guia [MEDDEV 2.1/6 \(Qualification and Classification of stand alone software, July 2016\)](#), mostra un diagrama de decisió que aporta orientació sobre els passos necessaris a seguir per a decidir si un stand alone software és o no producte sanitari. Tal i com indica la mateixa guia, en aquest context s'entén com a stand alone software el programari que no està incorporat en un producte sanitari.

La guia "[Guidance: Medical Device stand-alone software including Apps \(including IVDMDs \(In Vitro Diagnostic Medical Devices\)\)](#)" publicada per la [MHRA](#) ofereix informació detallada per ajudar a decidir si una App és o no producte sanitari. Aquesta mateixa guia també clarifica quan es considera que una aplicació té un propòsit mèdic, aspecte que la converteix directament en producte sanitari.

Requisits reguladors per a producte sanitari

Una vegada s'ha identificat que l'aplicació és un producte sanitari, cal determinar a **quina classe pertany** d'entre les indicades a l'Annex IX de la Directiva 93/42/CEE. Aquesta classificació es basa en el risc potencial que el producte representa pel pacient i/o usuari, i determinarà els passos que cal seguir per a avaluar-ne la conformitat. Els productes sanitaris es poden classificar en:

- Classe I → risc baix
- Classe IIa → risc moderat
- Classe IIb → risc sever
- Classe III → risc alt

És important destacar que pel producte sanitari per diagnòstic in vitro no aplica aquesta classificació, sinó que l'annex II de la Directiva 93/42/CE defineix un llistat de productes que requereixen una ruta d'avaluació de la conformitat específica. El Reglament (EU 2017/746) indicat anteriorment introdueix canvis en aquest punt, establint una completa classificació també per als productes sanitaris per diagnòstic in vitro.

Qualsevol producte sanitari ha de disposar de la **declaració CE de conformitat** abans de comercialitzar-se. Això implica que el fabricant (o representant autoritzat en cas que el fabricant sigui extern a la Unió Europea) assegura i declara que el producte compleix la legislació vigent corresponent. Una vegada assegurada la conformitat, és necessari col·locar la marca CE al producte sanitari.

A grans trets, per a poder comercialitzar a Europa una App considerada producte sanitari, el fabricant ha de:

- Desenvolupar un expedient tècnic cobrint tots els requisits de la Directiva pertinent,
- disposar d'un sistema de gestió de la qualitat,
- d'acord amb l'article 100 de la Llei 14/1986 general de sanitat de 25 d'abril, les persones físiques o jurídiques que es dediquin a la fabri-

cació, importació, agrupació o esterilització de productes sanitaris, i les instal·lacions en què es portin a terme aquestes activitats, requereixen llicència prèvia de funcionament, atorgada per l'**AEMPS** (Agència Espanyola de Medicaments i Productes Sanitaris).

Els aspectes clau a tenir en compte quan es parla d'una aplicació mòbil com a producte sanitari són:

- Estàndards de documentació tècnica.
- Estàndards d'avaluació/evidència clínica.
- Possible necessitat d'intervenció, en funció de la classificació del producte, d'un organisme (organisme notificat) que declara conformitat del producte addicionalment a la declaració del fabricant.
- Estàndard en l'etiquetat del producte.
- Alt nivell de traçabilitat del producte tant durant el desenvolupament com durant les diferents etapes de fabricació (cicle de vida del producte).
- Control rigorós dels canvis sobre el producte (versions del software).
- Tractament d'incidències i/o queixes.
- Vigilància i seguiment post-comercialització.

A més, també cal tenir en compte que els nous Reglaments citats introdueixen requeriments addicionals, com la millora de la transparència mitjançant la creació de la EUDAMED (*European Database on Medical Devices*). Aquesta base de dades està actualment en desenvolupament i s'hi registraran tots els productes sanitaris a nivell europeu.

14. Entitats participants

A continuació es presenta la relació d'entitats que han elaborat aquesta guia:

- **CSV Experts:**

L'empresa CSV Experts ha desenvolupat els apartats corresponents a la regulació de productes sanitaris en entorns mòbils. En concret ha desenvolupat el següent punt:

[13. Medical Device: L'aplicació mòbil com a producte sanitari](#)

csvexperts

Always Meeting the Standards

- **M4Social, taula del tercer sector:**

L'empresa m4Social juntament amb la Taula del Tercer Sector han desenvolupat la part corresponent al desenvolupament d'apps accessibles, en concret el punt:

[9. Accessibilitat](#)

m4Social



Taula d'entitats del Tercer Sector Social de Catalunya



- **Fundació TIC Salut Social:**

La Fundació ha coordinat l'edició del contingut amb els diferents col·laboradors, ha proporcionat una primera versió de cada apartat a mode d'esborrany i ha revisat el text final. Des de la Fundació també s'han desenvolupat els següents punts:

[11. Experiència d'usuari](#)

[12. Estàndards de comunicació i representació d'informació](#)



TIC | Salut Social

- **PASIONA:**

L'empresa tecnològica Pasiona, ha realitzat la part corresponent a usabilitat en entorns mòbils en salut. En concret el punt:

[10. Usabilitat](#)

pasiona

- **iSalus:**

L'empresa iSalus ha treballat tota la part corresponent al desenvolupament d'aplicacions mòbils i seguretat de les dades. En concret han desenvolupat els següents punts:

[3. Tipus d'aplicacions](#)

[5. El procés de desenvolupament d'aplicacions mòbils](#)

[8. Seguretat i protecció de dades](#)



- **UPC:**

La Universitat Politècnica de Catalunya ha participat en el desenvolupament corresponent als llenguatges i entorns de desenvolupament d'aplicacions mòbils així com en donar una visió dels diferents mercats actuals. En concret ha desenvolupat els següents punts:

[4. Markets d'aplicacions](#)

[7. Consells pel desenvolupament i la publicació d'aplicacions](#)

[6. Llenguatges i entorns de desenvolupament](#)



15. Autors

A continuació es presenta la relació d'autors que han participat en l'elaboració d'aquesta guia:

- **CSV Experts:**
 - Josep Hortigüela, Consultor sènior i expert en productes sanitaris.
- **Fundació TIC Salut Social:**
 - Carme Pratdepàdua, Responsable de l'Oficina mHealth.cat.
 - Ariadna Rius, Responsable de l'Oficina d'Estàndards i Interoperabilitat.
 - Miquel Martí, tècnic en interoperabilitat de l'Oficina d'Estàndards i Interoperabilitat.
 - Pau Garcia, tècnic en interoperabilitat de l'Oficina d'Estàndards i Interoperabilitat.
- **iSalus:**
 - Marc Gòrriz, Responsable de projectes i desenvolupaments.
 - Dylan Rivera, Full-stack developer.
- **M4Social, taula del tercer sector:**
 - Jordi Serratosa, Coordinador m4social – Taula d'entitats del Tercer Sector.
- **PASIONA:**
 - Clara Planet, Responsable de UX/UI Solucions de Disseny.
- **UPC:**
 - Toni Oller, Professor Col·laborador a la Universitat Politècnica de Catalunya, Departament d'Enginyeria Telemàtica.
 - Jesus Alcober, Professor Titular a la Universitat Politècnica de Catalunya, Departament d'Enginyeria Telemàtica.

