

# Development of mobile applications in the field of patient health and care

Guide to creating a mobile application following the basic principles of ethics, that is secure, accessible, usable, interoperable and trustworthy

**Coordination:**

2021, TIC Salut Social Foundation.



The contents of this work are subject to a license from Attribution-NonCommercial-NoDerivates 4.0 International. The license can be viewed on the Creative Commons website.

**Edited:**

TIC Salut Social Foundation

**1st edition:**

Barcelona, June 2021.

The TIC Salut Social Foundation would like to thank in particular the “Fundació Grífols” –UVIC-UCC Bioethics Chair; the i2cat Foundation; the UPC (Polytechnic University of Catalonia); Pasiona, the CTTI (Centre for Telecommunications and Information Technologies) m4social, ONCE, Table of the Third Social Sector, iSalus, Àmbit-bst-CSV Experts and the Committee of Functional Experts of the TIC Salut Social Foundation, for their full support in drawing up this guide and contributing their extensive knowledge on the different subjects in which they have participated.



# Introduction

As part of the adoption of the mHealth<sup>1</sup> policies, the Government of Catalonia approved the mHealth.cat Mobility Plan at the beginning of 2015, and entrusted the TIC Salut Social Foundation with the task of making it operational. The Plan encourages institutions in the sector to facilitate access to patient health and care services through mobile technologies, and at the same time, stimulate the transformation of the different healthcare and social processes.

Entrusted with this task, the TIC Salut Social Foundation, a body under the auspices of the Ministry of Health of the Government of Catalonia, has worked on setting up the mHealth Office, with the following objectives:

- Promote mHealth in Catalonia as a lever of transformation for the health and social services systems.
- Help connect supply and demand in the ecosystem in order to identify key projects for mHealth.
- Ensure the development of services as well as standards and other support elements, either using internal resources or in collaboration with third parties.
- Evaluate and contribute to the generation of evidence and knowledge, anticipating challenges and trends in mHealth.
- Communicate the progress made and projects being carried out for citizens or professionals to use

In this context, in 2018 a first version of the document was published called “*Quick guide to recommendations for the development of health and social care apps*” offering a set of guidelines in the development of apps in the field of health. This new more detailed guide is the result of the expertise gained in the implementation of different initiatives and projects and from working as a network with other agents in the sector. It offers recommendations and advice for developing a mobile application in the in the field of health and social services taking into account the principles of ethics, user experience, interoperability, privacy, security, reliability, clinical evidence and trustworthiness and transparency.

---

<sup>1</sup> mHealth or Mobile Health: Defined as the branch of eHealth in which the practice of healthcare is supported by mobile devices such as smartphones and digital tablets.

# Contents

<b>1. Introduction</b>	<b>6</b>
1.1 Context and starting point	6
1.2 Objectives and content of the document	8
<b>2. The functional and technical aspects of the App</b>	<b>9</b>
2.1 Ethics and content	9
2.2 Data security and privacy	15
2.3 Communication and representation of information standards	22
2.4 Usability, accessibility and user experience	27
<b>3. Aspects related to the App development process</b>	<b>36</b>
3.1 Development life cycle	36
3.2 Development methodologies	38
3.3 Development types and technologies	42
<b>4. Aspects related to markets, publishing and classification</b>	<b>51</b>
<b>5. Aspects related to the applicable regulations and quality</b>	<b>60</b>
5.1 The mobile app as a medical device	60
5.2 TIC Salut Social Foundation quality seal for mobile applications (TICSS seal)	67
<b>6. Impact of 5G technology on Apps</b>	<b>70</b>

# 1. Introduction

## 1.1 Context and starting point

Mobile technologies applied to the health and social welfare sectors have become tools of daily use for a large majority of citizens in Catalonia and elsewhere. The increase in the use of these mobile technologies in the field of health has only grown in recent years. The Green Paper on Mobile Health (2014) already identified the potential of mHealth as “an emerging and rapidly developing field which has the potential to play a part in the transformation of healthcare and increase its quality and efficiency”. Meanwhile, the mHealth App Economics report (2017) reported that there were a total of 325,000 Health Apps available on official markets (AppStore and Google Play).

Nevertheless, despite the degree of implementation of mHealth and its potential, there are only a few initiatives that promote specific strategies for establishing mechanisms that can supply the health and social services system with guaranteed and quality solutions which at the same time add value, are usable and can be recommended to citizens by health and social services professionals. Establishing regulatory frameworks that allow this technology to be adapted in the initial phases of projects, providing guarantees to users and thus establishing true tools of value and trust are key aspects of success in this constantly evolving sector.

In recent years, the TIC Salut Social Foundation has worked in the field of mobile applications, collaborating with different professional associations, universities, healthcare centres and companies, observing the new trends in mHealth and defining and launching a certification process for mobile solutions in the patient health and care sector. Based on this experience a series of questions have been drawn up that need to be asked before starting the process of creating an App in order to consider key aspects that will affect the course of the initiative.

### Questions to ask before creating a health App

The internet offers an almost infinite number of resources that talk about how to design a mobile application but in many cases these do not cover all the steps that should be taken into account when developing a mobile phone application for use in the health sector. Designing an App offering health advice and recommendations to patients or citizens who want to promote disease prevention and maintain a healthier lifestyle, must not only take into account the issues of operability and usability. It also depends on other factors, such as guaranteeing data privacy, taking into account whether data should be shared with a healthcare professional, or even considering whether the app makes sense from an ethical standpoint.



There are several stages involved in the lengthy process of creating a mobile health application (*from the moment the idea is generated, to how it will be implemented and published in the different application markets*), and it is important to know how to overcome each of them to achieve the final objective.

Below is a description of the key aspects for identifying all the relevant elements during the initial phases of the project in order to address them appropriately:



## 1.2 Objectives and content of the document

This guide offers a set of recommendations to help respond to the questions above. In this regard, it aims to identify and offer guidance on those aspects that directly affect the creation of effective, usable, safe applications that guarantee the rights and freedoms of the people who will use them.

In this regard, the guide addresses both the technical and functional aspects that any mobile application must incorporate in the initial phases of the development project. Specifically, it is structured as follows:

- First, it details the necessary functional and technical aspects, specifically:
  - Ethics and content,
  - Data security and privacy,
  - Communication and representation of information standards,
  - Usability, accessibility and user experience,
  - Tips for publishing and classification.
- Second, the document describes aspects related to the development of the solution, specifically:
  - The process of developing a mobile application, methodologies and technical equipment,
  - Types of development technologies.
- Third, it presents a series of recommendations regarding the publication of the App in the corresponding markets and its classification.
- Fourth, the specific regulations applicable to the certification of mobile applications in health are briefly mentioned:
  - The mobile application as a medical device,
  - TICSS quality seal.
- Finally, mention is made of the impact of 5G communications technology

## 2. The functional and technical aspects of the App

### 2.1 Ethics and content

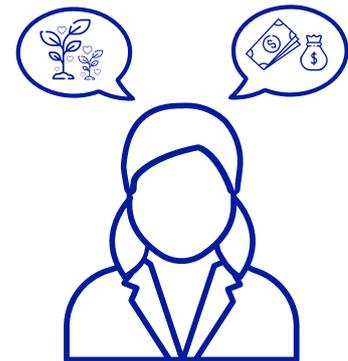
Reviewing whether the application informs the user of its benefits, whether health professionals have been involved in the design of the App, checking the scientific evidence, sources of funding, how often they are reviewed and whether the contents are updated, the sources of information, ethical aspects, whether it informs the user about any adverse risks, information on minors, or whether it offers a point of contact for queries, are aspects that should be considered when designing an App aimed at the health sector.



**Ethics and content**

### Ethical aspects Apps should take into account

It is a fact that new technologies have already digitally transformed many areas of people's lives and the field of health and social care is another example of this with the proliferation of tools and resources, initially designed to make life easier and cover the needs of the sector. However, our society is diverse, with a high percentage of people for whom access to technology is difficult for various reasons, and that the widespread implementation of these resources could lead to a discriminatory digital divide for many and yet another form of inequity and inequality. Very basic and essential accessibility criteria must be taken into account so as not to exclude certain vulnerable groups in the use of these Apps.



The development of digital resources (patient portal, mobile applications for data collection and management, etc.) must not lead to a regression in essential values that have so far sustained the professional-citizen relationship, nor effect the trust placed in health and social organizations. It is worth bearing in mind that in the social environment digitalization and the use of electronic tools is even less widespread, and therefore its awareness and use should be promoted among citizens.

The development and application of technology must always be based on the respect for human rights and fundamental ethical criteria such as respect for autonomy, proportionality and security in the use of data, transparency, but only as a complement and never a substitute for the essential relationship of proximity with the professional, to generate confidence and effective care. The digital user must be so out of conviction and never out of obligation or because they have no alternative.

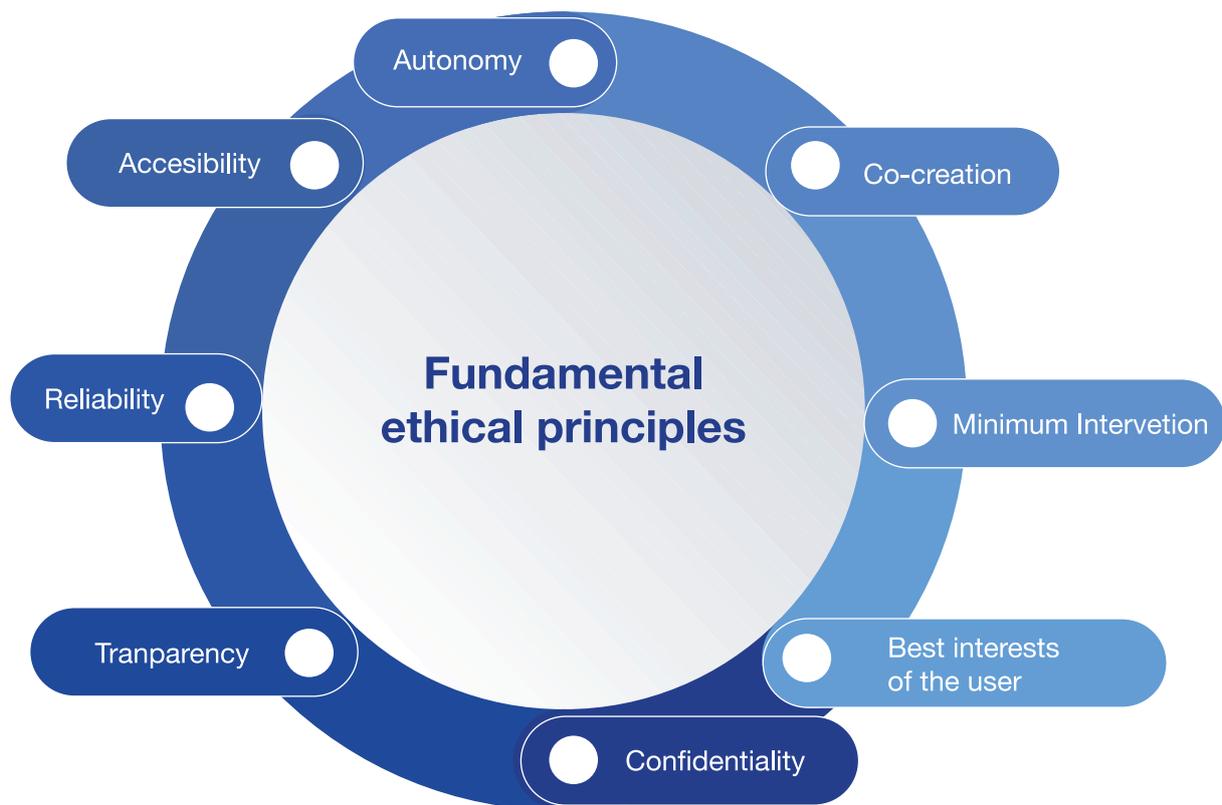
That said, the use of Apps in both the context of health and social care must be a mechanism that covers two alternative or complementary objectives:

- **facilitate and improve** the relationship and interoperability with the professional who supports the patient in whatever they need (this can include recommendations, facilitate changes in behaviour, habits or decisions, etc.)
- **organize and control** the information (with or without interaction with the professional and individual patient management).

In any case, the fundamental **ethical principles** that should govern these are:

- **Autonomy:** the use of any App must be based on the consent and acceptance of the individual, using means of verification that are secure, clear and transparent, both regarding the information needed to know what type of user data may be requested, returned or evaluated and how it will be processed and for what purposes, and in regarding the clear granting of consent.
- **Co-creation/participation:** the best way to ensure that the App is accessible and user-oriented is to take into account the opinion of the final recipients from the start during its design and planning.
- **Best interests of the user:** an App must be designed with the user's best interests in mind, depending on its purpose –in the health or social fields–, as a complement to the work of the associated professional and not to replace or eliminate more costly resources, which would negatively affect trust and security.
- **Proportionality and minimum intervention (minimization of the required data):** any App involves the management of personal data, identifiable or pseudo-anonymized, which is why it should work with the minimum data necessary for the purpose and service being provided.
- **Confidentiality as an expression of the right to personal privacy:** as already required by the current legal framework, the user must be guaranteed the security and traceability of the data managed by the APP, as an indispensable ethical imperative to avoid fraudulent use or harm to the patient.
- **Transparency:** the App must at all times inform the user who is managing the information, for what purposes, how they can be access it and how they can exercise their rights over their data.
- **Reliability:** any health or social App should be validated by an organization or entity that accredits its scientific reliability, even more so those that are considered “medical devices”, to ensure it generates trust among the public.

- **Accessibility:** as far as possible these Apps should aim to be accessible at a basic level for the public, with an intuitive and agile design, which facilitates and does not complicate its use, especially if it is aimed at vulnerable people whose safety and proper care depend on its correct use. This accessibility should also be taken into account for people with physical and intellectual disabilities, who are often excluded.



## The content of the App

The main reason for developing a health app is to offer content on a topic which is useful to a specific group of people- Whether in the form of recommendations, procedures, assisted registration, etc. All adapted to a mobile device that accompanies the user at all times and allows them to use sensors, reminders, Internet connection, etc. The content can therefore be considered the *raison d'être* of the application being developed. For this reason, it is of fundamental importance and must be endowed with the necessary quality and confidence that a Health App requires.



The involvement of healthcare professionals or a multidisciplinary team who specialize in the solution being developed and who create the content based on solid scientific evidence is key. Content is the *raison d'être* of the App, what adds value and makes it stand apart from other solutions.

With this in mind, the following considerations

- **Indicate the people responsible and the authors of the content, to which entity they belong and complement this with information on their background and specialization.**

The App should indicate which health professionals, public bodies or scientific societies have developed the content offered in it. In addition, providing complementary information such as their training or clinical specialty, imply transparency and generate trust in the user.

- **Indicate the sources of information based on scientific evidence used to prepare the content and the date of update.**

The content offered in the App must be based on scientific evidence which should be clearly indicated in the App so it can be consulted by users. Almost as important is indicating the date on which the content was created or updated. These references should be cited in accordance with one of the standards for the publication of bibliographic references (Vancouver, APA, AMA, etc.).

- **Ensure that the data systems used are of recognized reliability and validity.**

Data systems are any sets of data that interact with each other based on a series of rules or protocols. They may be clinical languages, measurement scales, vaccination schedules, normal value ranges, etc. It is essential that these are of recognized reliability and validity to offer the user an accurate and reliable solution. In addition to validity, it should be remembered that measurement systems can vary depending on the region and a good App will be adapted to offer a better understanding of its elements.

- **Provide a warning that the App and its contents are not intended to replace the services of a health professional and the risks that may arise from its use.**

This recommendation may seem obvious or irrelevant, but it is very important that the App indicates that the services offered are merely a guide, and that in no case can they replace the assessment or care of healthcare personnel. It is also important to indicate any potential health risks that may result from misuse of the Application or its services.

Regarding the way the content is displayed within the App, the following points should be taken into account:

- **Use comprehensible and simple language, with messages adapted to the profile of user in terms of style and language level.**

The language used should be plain, making it easy for any profile of user to understand the content. Technical jargon should be used only when strictly necessary. That is, when similar terms do not exist or when using non-technical terms could lead to confusion.

- **Complement the quality content by making support and contact mechanisms available to the user:**

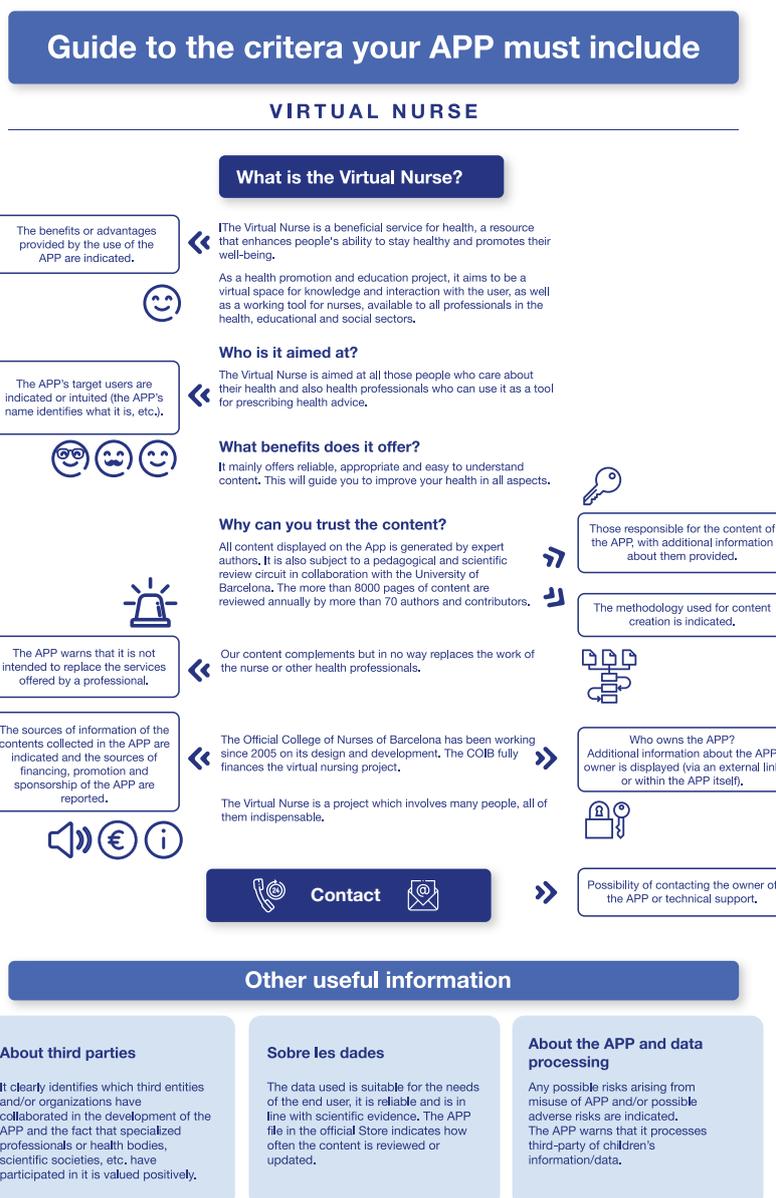
Often the person using the App will have doubts related to the content of the App, so it is very important to provide contact mechanisms. Whether by email, phone, chat, contact form, etc.

- - **Include self-help items such as video tutorials, guides, or frequently asked questions to make it easier to use.**

When a user first accesses an App, they want to find things easily and quickly, and in many cases having visual elements and video tutorials that help them interpret the tasks they can carry out with the application, generates a more positive perception and a better predisposition to use the App, since the user is able to better understand all the functionality it offers.



Below is an infographic that offers an example of how this content has been applied in an existing application. The infographic shows how the information has been structured and which sections should be included in all mobile health applications so that the public can understand the type of application they have in their hands.



## 2.2 Data security and privacy

In today's environment where personal data requires special protection, Apps must guarantee the existence of mechanisms that preserve the **privacy** of user-generated data and **confidentiality** in the transmission of this information. Also, ensuring the proper storage of information and establishing **encryption mechanisms** when registering access passwords are aspects that offer transparency for the user and are of required compliance for any App.



**Security**

### Data security and protection

Security in the software world, where the priority seems to be the number of features or ease of use, is often mistakenly given secondary importance. The security of an App is related to the security of the data that it processes, and therefore, everything possible must be done to ensure both the security and the privacy of the data. It is also important to understand that security should not only be considered during the development of the software, but should be a continuous process of improvement throughout the life of the App.

Here are some important things to keep in mind to ensure that the App is secure:

- Identify the **sensitive data** that will be processed and identify the **processing** that will be carried out, and apply, for each type of processing, the **appropriate security guarantees**.
- Assess the security aspects right from the **early stages of the project**. Having a **security expert**, either internal or external, who can assess the security needs of the project is highly recommended.
- Use alternative user authentication systems such as OAuth **2.0** to facilitate access with credentials from other services such as Google or Facebook. If own credentials are to be used, they must be required to fulfil **minimum length and complexity requirements**, and this **data must be encrypted** and essential and stored on the server. The option of validation using biometric elements such as fingerprints can also be considered.
- Encrypt the sending of information with secure protocols such as TLS if the application communicates with external systems. All secure communications must also be with servers that are identified by **valid digital certificates**.
- Ensure that data is **encrypted**, if sensitive data is stored on the device.
- Ensure that the application requests on the phone **only those permissions that are essential**. In this way, if the application is compromised, access to the device is limited. Moreover, some users, aware of the existing security risks, will not be willing to grant certain permissions to the application if they do not consider them really necessary.

- **Verify all data entry paths** and check that only what is expected can be entered, including the type of data and length. In client-server applications, in terms of security, special emphasis should be placed on server validations. Validations on the client improve the user experience, but are easily circumvented, so they need to be replicated on the server.
- **Prevent attacks;** if the application works with a database, it is susceptible to an **SQL injection** attack. It is one of the most common types of attack and can involve the loss, corruption or theft of data. This attack consists of entering SQL code in the data entry parameters, such as a login password field, so that the query to the database is modified allowing the attacker to carry out whatever action they wish. To avoid this, it is important to validate all text strings used in actions against the database, so that SQL code cannot be included in them.
- Perform **penetration tests** (Pentesting) to verify that the App is not susceptible to known types of attacks. To carry out comprehensive Pentesting you can follow different guides or frameworks, such as the one provided by **OWASP<sup>2</sup>** (Open Web Application Security Project), a global non-profit organization focused on improving software security.
- Check that all applicable data protection laws are complied with. The European Data Protection Regulation (**GDPR<sup>3</sup>**), fully applicable since May 2018, has meant a significant change in the duties and obligations of personal data controllers and processors. This Regulation is further complemented with the approval of Organic Law 3/2018, of 5 December, on the protection of personal data and the guarantee of digital rights (LOPDGDD). These laws, which can be consulted on the website of the AEPD<sup>4</sup>, not only regulate technical aspects related to data protection, but also include aspects relating to the rights of users that must be guaranteed, or to the consent that must be requested to carry out the processing of personal data.

---

<sup>2</sup> OWASP: [https://wiki.owasp.org/index.php/OWASP\\_Unified\\_Pentesting\\_Framework#tab=Main](https://wiki.owasp.org/index.php/OWASP_Unified_Pentesting_Framework#tab=Main)

<sup>3</sup> For more information on the GDPR, go to the following website: [https://ec.europa.eu/justice/smedataprotect/index\\_en.htm](https://ec.europa.eu/justice/smedataprotect/index_en.htm).

<sup>4</sup> Spanish Data Protection Agency: <https://www.aepd.es/es>

## General Data Protection Regulation

This regulation governs the processing of personal data, the use made, and the free movement of such data. The GDPR applies to all companies, organizations, bodies and institutions that process data of European citizens. The most important aspects are summarized below.

### Personal data

The GDPR defines personal data as any information relating to an identifiable person who can be identified, directly or indirectly, in particular by reference to an identifier. Examples of personal data include the name, address, GPS location, health information or monetary income of a person.

Within personal data, the GDPR differentiates between that which is considered **sensitive data**, requiring **special protection** measures. This data is regulated by article 9 of the GDPR, which, from the outset, establishes a general prohibition of processing, notwithstanding exceptions. The special categories of data **include** those that reveal:

- Political opinions.
- Trade union membership.
- Religious beliefs.
- Philosophical beliefs.
- Racial or ethnic origin.
- Data concerning health.
- Sex life.
- Genetic data.
- Biometric data.
- Sexual orientation.



The **general prohibition on processing** this data does not apply if any of the circumstances of article 9.2 of the GDPR applies, such as having explicit consent.

Another important aspect to take into account is identifying who is the **data controller** and who is the data processor. Where:

- The data controller is the natural or legal **person**, public **authority**, agency or other **body** which, alone or jointly with others, determines the purposes and means of the processing of personal data.
- The data processor is the natural or legal person, public authority, agency or other body which **processes personal data on behalf of the controller**.

## Data processing

Data processing covers a wide range of operations performed on personal data, whether or not by automated means, such as collection, recording, organization, structuring, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission or dissemination. Examples of data processing include: sending promotional emails, posting photos of a person on a website, storing IPs or MACs of users or managing a patient database.

## Rights of the data subject

One of the key aspects of the data protection regulation is the granting to all natural persons of different rights and powers in relation to the processing of their personal data, starting with the **right to information** and continuing with the **right of access, rectification, erasure, restriction, portability, objection and non-automation**. Therefore, when processing a data subject's data, the following key aspects must be guaranteed:

- **Clear and explicit consent:** explicit consent is needed to process personal data. The consent form cannot be ambiguous and must use plain and understandable language. For example, a box asking for consent for each form of data processing should be included rather a generic one stating "I accept the terms and conditions of use". Consent given prior to the regulation that does not comply with this point **must be requested again**. In the case of children under the age of 16, consent must be provided a legal guardian. This age may as low as 13 years of age in some states.
- **Right to information:** The right to information is part of the essential core of the right to the protection of personal data and is related to the principle of transparency, which the GDPR has reinforced.

*Information will be provided in a concise, transparent, intelligible and easily accessible form, using clear and plain language, in particular for any information addressed specifically to a child. This information may be provided in combination with standardized icons in order to give in an easily visible, intelligible and clearly legible manner a meaningful overview of the intended processing.*

- **Right of access:** the data subject must have access to all their data in a user-friendly and understandable format, together with the additional information detailed in article 15 of the GDPR (such as the purposes of the processing). The right of access is one more element of the **principle of transparency** pursued by data protection legislation, ensuring that data controllers are transparent with regard to the data processing they carry out.
- **Right to rectification:** According to Article 16 of the GDPR, data subjects must be able to request the modification of false or inaccurate data concerning him or her, including by means of providing a supplementary statement. The data subject must obtain the rectification of their data within a **period of 1 month**.

- **Right to erasure or right to be forgotten:** As indicated in Article 17 of the GDPR, the controller must guarantee the erasure of the data subject's personal data and stop transmitting it to third parties when requested and without undue delay. This right will apply as long as it does not compromise others such as freedom of expression.
- **Right to restriction of processing:** This right allows the data subject to request that their personal data not be used for certain processing. In this case, all the data is deleted in a logical manner so it is not seen in any way from the application. (Art. 18 of the GDPR).
- **Right to data portability:** The data subject has the right to receive the personal data that they had provided to a controller, in a structured, commonly used and machine-readable format, and transmit them to another controller. Therefore, it is necessary to be able to export all the data in a structured format, such as JSON or XML, although others such as CSV can also be used. (Art. 20 of the GDPR).
- **Right to object:** According to Art. 21 of the GDPR, the data subject can object at any time, on grounds relating to his or her particular situation, to certain processing of personal data concerning him or her.
- **Right not to be subject to automated individual decision-making, including profiling:** This refers to the fact that the data subject should have the right not to be subject to a decision based solely on automated processing, that is, without human intervention, including profiling, which produces legal effects concerning him or her. (Art. 22 of the GDPR).
- **Notifying the processing activities to the local data protection authorities** of each country or region.
- **Data minimization:** Only request the data that is strictly necessary in relation to the purposes for which it is processed.
- **Keeping the data only as long as necessary:** When the data is no longer necessary for the purpose for which it was collected, it must be deleted. There are exceptions if the data must be maintained for other legal reasons.
- **Each type of processing requires separate consent:** Specific consent is required for each type of processing that is carried out on the data, or if the data is processed subsequently for a new purpose.
- **Notifying users if their data is transferred outside the European Union.**
- **Notification of breaches:** Data subjects and the supervisory authority must be notified of any breaches that may affect the rights or freedoms of individuals. Notification should preferably be given within **72 hours**.

## Measures relating to personal data

The regulation establishes that the appropriate measures must be applied to personal data, taking into account the state of the art, the cost of implementation and the nature, scope, context and purposes of processing as well as the risks of varying likelihood and severity for rights and freedoms of natural persons.

For sensitive data, even stricter measures must be applied and these are summarized below:

- Maintain **updated records of all the processing carried out** within the organization, documenting them with specific information.
- If special categories of personal data (sensitive data) are processed on a large scale, an **impact assessment** must also be carried out. This assessment must describe the processes carried out and their proportionality with respect to the purposes. It must also detail all the risks regarding the rights and freedoms of the data subjects related to this processing, as well as the measures taken to minimize these risks.
- Take relevant security measures, such as:
  - Keeping a list of people authorized to process the data.
  - Keeping a record of access with dates and identification of the person who accessed the data.
  - Encrypting data when it is communicated and stored.
  - Pseudonymizing data if it are used, for example, in test environments or to perform research.
  - Not including personal data in the log file.
  - If the application interacts with third parties to share data, ensuring that these comply with the regulations.

Within this data protection framework, it is also necessary comply with the accountability principle and the **principles of data protection by design by default**. In other words, data protection in the design implies taking into account all the obligations and requirements imposed by the data protection regulations from the moment a new processing is planned. First, establishing **the data protection requirements in data-oriented design**:

- Minimize and limit the amount of personal data collected and processed.
- Hide and protect personal data, through pseudonymization, encryption and aggregation personal data.
- Separate the processing and storage to avoid the possibility of creating complete profiles of a person (for example, dividing databases).
- Aggregate the data collected and processed to protect the rights of the data subject, to ensure it is not detailed.
- Guarantee data protection by default. All parameters should be configured, by default, with the simplest privacy settings. That is, it must be the user who decides whether to change any parameter that involves less privacy.

And secondly, listing the **design requirements, from the perspective of processing:**

- Configure the software to inform the user about how it works and how personal data is processed.
- The application must allow the user to control their own personal data, enabling them to easily exercise their right of access, rectification or erasure of the data.
- Information on compliance with data protection regulations should be readily available (for example, by publishing the data protection policy).
- Compliance with data protection regulations must be documented with proof (security audit reports, vulnerability scan, etc.).

The protection of data by default consists of offering the maximum guarantees of privacy by default.

### Data Protection Impact Assessment (DPIA)

As indicated in the regulation, if sensitive data is processed on a large scale (which is what mobile applications do), an impact assessment must be carried out in order to identify the risks regarding the rights and freedoms of the natural persons related to the processing of the data, as well as the preventive measures to minimize these risks.

The Office of the Data Protection Officer of the TIC Salut Social Foundation has made the **DPIA tool**<sup>5</sup> openly available, developed and based on the model of the Catalan Data Protection Authority (APDCAT)<sup>6</sup>. The DPIA has been adapted to the specific needs of the health field to assess the processing of personal data in research and innovation processes. This tool enables the self-assessment of risks in the processing of personal data and their mitigation through simple and understandable language.

The tool also provides examples, definitions to help identify the actors involved in the process, describe the types of processing and measure the risks that exist to establish an action plan and measures to avoid them.

This tool offers guidance to the data controller and the agents involved in decision-making through automated proposals and allows a detailed analysis of the life cycle of the personal data involved in all stages of the project.

---

<sup>5</sup> DPIA tool. [https://ticsalutsocial.cat/wp-content/uploads/2020/12/aipd\\_creacio-metod\\_221220-2.pdf](https://ticsalutsocial.cat/wp-content/uploads/2020/12/aipd_creacio-metod_221220-2.pdf)

<sup>6</sup> APDCAT impact assessment. [https://apdcat.gencat.cat/ca/drets\\_i\\_obligacions/responsables/obligacions/avaluacio-impacte-relativa-proteccio-dades/](https://apdcat.gencat.cat/ca/drets_i_obligacions/responsables/obligacions/avaluacio-impacte-relativa-proteccio-dades/)

## 2.3 Communication and representation of information standards

There are many different mHealth-related Apps that allow different types of information to be collected, in different formats and from different devices. This means there is a huge amount of data generated by citizens spread across a series of Apps that cannot be exchanged between them. Each application uses its own internal storage and representation format, as well as its own communication interfaces with other systems (usually limited to platforms owned by the same company). In this scenario, a citizen may be using two applications that allow the same data to be collected (e.g. weight), without the applications being able to exchange, compare or globally exploit the data. To overcome these limitations, a consensus is required on how the exchange should be performed. This type of consensus is embodied in standards, both nationally and internationally, that achieve interoperability between systems, devices, applications, etc.

In the health sector in general, and in the Integrated Public Use Healthcare System of Catalonia (hereinafter, SICSCAT) in particular, work has been going on for years on the interoperability of systems, devices and services, in response to the need to share information between centres, even at different levels of care.

### Interoperability

Interoperability is the ability to **share information** between components (such as systems or devices) **without losing its meaning**. This communication must ensure the consistent exchange of data between departments, organizations, levels of care or different geographical locations. The main goal is to provide professionals with all relevant information about their patients to ensure that the decision-making process takes place in a safe, efficient and effective manner. Interoperability ensures access to information regardless of where it has been registered, encouraging its re-use and ensuring the continuum of care.

This capacity is represented in different dimensions that are described below:<sup>7</sup>

- **Technical interoperability** This refers to the technologies and protocols that allow communication to be established between the components.
- **Functional or syntactic interoperability:** Adds information exchange to communication. Standards at this level define the structure and format of the information to be exchanged.
- **Semantic interoperability:** This ensures that the information is exchanged without losing its meaning and that it can be used by the receiving component as if it had generated it itself.
- **Legal interoperability:** This that the legislation in force for each agent involved in the exchange is complied with. This layer is especially important in cross-border projects where exchange takes place between countries or regions with different legislative frameworks.
- **Organizational interoperability:** Adds the process layer, so that the exchange and use of information is aligned with the workflows of the institutions involved.

---

<sup>7</sup> <https://op.europa.eu/en/publication-detail/-/publication/1c578b32-4c82-11ea-b8b7-01aa75ed71a1/language-en>

Each of these dimensions can be safeguarded through the use of standards explicitly designed for each of these purposes. This section emphasizes the syntactic and semantic dimensions, describing the main standards that apply in the mHealth environment.

## Interoperabilitat sintàctica i l'estàndard FHIR

The Fast Healthcare Interoperability Resources (hereinafter HIR<sup>8</sup>), is developed and published by the Health Level Seven International standards organization (hereinafter HL7). It is currently in Release 4 (v4.0.1), the first version with regulatory content<sup>9</sup> in addition to content in STU (Standard for Trial Use).<sup>10</sup>



The FHIR standard represents a change of concept with respect to previous HL7 standards such as CDA R2 (Clinical Document Architecture Release 2) or messaging 2.x, as it has been specifically designed to facilitate learning by developers, implementation, adaptation and processing. The items to be exchanged (called **resources**) have also been simplified, so that they allow only the strictly necessary information to be sent in a more agile format that is also suitable for the mobile environment. These characteristics have led to FHIR being designated as the leading standard for the exchange of information in the mHealth environment.

FHIR uses a REST (Representational State Transfer) architecture to perform data exchange, using structural standards such as XML or JSON. Its implementation allows the use of a server to store data, and a number of standard services to send and retrieve it. To facilitate the implementation of FHIR and the development of applications that adopt it, several Application Programming Interfaces (APIs) have been created, the best known of which are **HAPI** (for Java) and **Firely** (for .Net). These APIs also allow the creation of an FHIR-based server, specifically designed to store its resources.

Security<sup>11</sup> and access control to the generated data is essential for the application to comply with the relevant regulations and provide a secure and trustworthy service to users. The SMART on FHIR specifications cover these needs, enabling a secure integration with EHR (Electronic Health Records), portals, etc. and uses security protocols such as OAuth2 for permissions or OpenID for login to ensure the protection of information.

As indicated above, the basic components of the FHIR standard are called resources<sup>12</sup> and represent the basic unit of information that can be shared. Each of these components represents a specific entity (such as Patient, Health Problem or Clinical Observation) and each has a number of elements associated; for example for the Patient resource, elements such as name, sex, date of birth, which provide content to the resource<sup>13</sup>:

---

<sup>8</sup> <https://www.hl7.org/fhir/>

<sup>9</sup> It is considered Regulatory when the content is stable and is subject to the FHIR version compatibility rules. While changes may be implemented in future releases, they are expected to be infrequent and highly restricted.

<sup>10</sup> It is considered STU when significant changes may be made to future versions of FHIR and cause it to be incompatible with previously published content.

<sup>11</sup> <http://hl7.org/fhir/security.html>

<sup>12</sup> <https://www.hl7.org/fhir/resource.html>

The following is an example of part of the definition of the Patient resource<sup>13</sup>:

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

Image 1: Extract of the patient resource

For each of the elements that make up the resource, its cardinality (the minimum and maximum number of times it can appear in the resource) and its type are specified. FHIR contains many different types of elements, which can range from an integer to encoded information using controlled vocabularies such as SNOMED CT. The list of available resources, as well as their definition, can be found at the following link: [Resource Index](#).

## Semantic interoperability and the SNOMED CT standard



The standards which regulate the semantic layer of interoperability aim to ensure that the information exchanged does not lose its meaning by standardizing it using controlled vocabularies. This standardization allows the concepts to be represented in such a way so that they can be exchanged, compared and exploited, even if they come from different sources.

There are different controlled vocabularies for standardizing information, and it is important to know how to identify what the purpose of each one is in order to use them correctly. The so-called **classifications** such as CIM-10-MC/SCP (diagnoses and procedures), CIAP-2 (primary care), ATC (active ingredients) or NANDA (nursing), are designed to group and exploit information, so the use of these for user-friendly and accurate registration and display is not recommended. By contrast, **terminologies** such as LOINC (laboratory) or SNOMED CT are specifically designed to record information with the highest level of detail, and ensure interoperability.

<sup>13</sup> <https://www.hl7.org/fhir/patient.html>

In the field of semantic interoperability, it is common (and necessary) to make equivalences between different vocabularies so they can be combined as well as used for their own specific purposes.

The Systematized Nomenclature of Medicine – Clinical Terms (hereinafter, SNOMED CT) clinical terminology is an **international and multilingual** semantic standard initially focused on Health, but which is already starting to be applied to other areas such as social care or mHealth. The fact that the conceptual model of this vocabulary is designed to offer a cross-sectional view of the citizen means that it contains concepts that are not strictly clinical. This means that SNOMED CT can be used as “bridge” or “link” terminology between different types of professionals.

SNOMED International (hereinafter, IHTSDO<sup>14</sup>) is the non-profit organization owned by SNOMED CT, and is responsible for updating and distributing it internationally. This terminology was created in 1965 by the College of American Pathologists and already has nearly half a million concepts in different specialties. Specifically, SNOMED CT offers the following 19 top level axes, also called hierarchies:<sup>15</sup>

- ▼ ● 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)

*Image 2: Main axes of SNOMEDCT*

Ideas in SNOMED CT are represented through hierarchically organized **concepts**, which can have different **descriptions** (with synonyms) and which are connected to each other through **relationships**. SNOMED CT provides different mechanisms to facilitate its adoption, such as the use of **subsets** and **extensions**. The former are designed to group components (concepts, descriptions or relationships) for a specific purpose and therefore allow the use of only those elements that are really needed. Extensions, on the other hand, can be seen as local versions of terminology and provide the flexibility to respond to real needs. Within an extension, concepts, descriptions, subsets, etc. can be created following the guidelines of SNOMED International and without damaging the interoperability of the contents. The Interoperability and Standards Office of the TIC Salut Social Foundation maintains the Catalan extension of SNOMED CT, with almost 3,000 concepts that are not in the vocabulary, and that is distributed either once or twice a year, depending on the demand.

<sup>14</sup> <https://www.snomed.org/>

<sup>15</sup> <https://browser.ihtsdotools.org/>

The following are some examples of the concepts that are part of the subset displayed through the SNOMED-CT viewer:

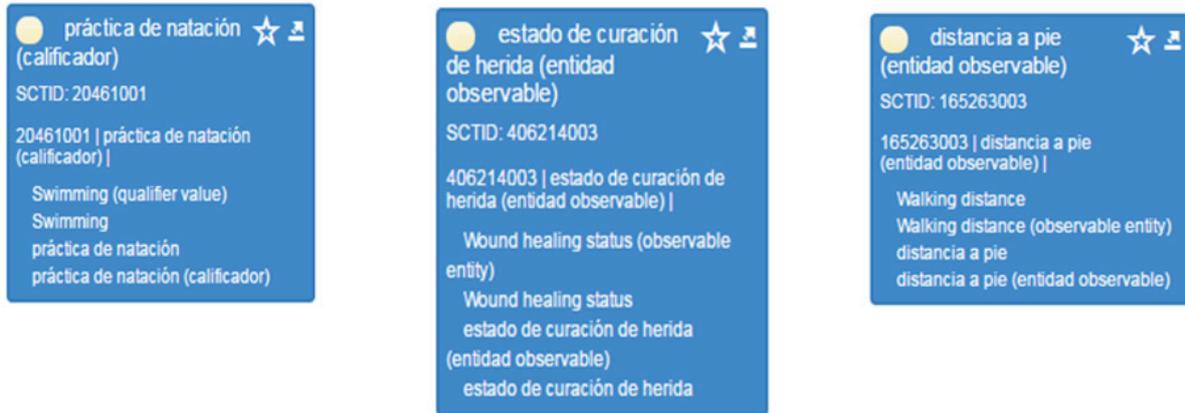


Image 3: Examples of subset data

## 2.4 Usability, accessibility and user experience

To take full advantage of the benefits of technology, the application must be **intuitive**, with a **design** appropriate to the required function and guarantee **universal and inclusive** access for people with functional diversity. For this field, the perfect combination is an application that is usable and also offers a **good user experience**, ensuring the final product is well received and consequently used by more people and with a greater benefit to all.



**Usability and  
accessibility**

### Accessibility to guarantee universal and inclusive access

Accessibility in the digital field means ensuring that all the available information, both on the internet and in the application, as well as the use of the technological device itself, are available to all people, regardless of their condition, characteristics or capabilities. When it comes to mobile app development, many designers and developers overlook the importance of accessibility. Accessibility is crucial in today's mobile world, since creating a mobile App with accessibility in mind will dramatically improve the user experience and maximize revenue. This type of accessible access must be possible with or without support systems, such as **VoiceOver**<sup>16</sup> screen readers for iOS or **TalkBack**<sup>17</sup> for Android.

Operating systems such as iOS or Android now offer many accessibility options to enhance the user experience, and it is important to ensure that the App is compatible with these.

Adapting programs, websites, and apps to make them accessible is often seen as an expensive and complex process. This complexity, and the resulting cost, can be significantly reduced or completely eliminated if basic accessibility guidelines are taken into account from the outset. The most important aspect to bear in mind about accessibility is that it should be included as part of the process when designing the wireframes<sup>18</sup> and in the functional analysis of the App.

Taking into account accessibility from the beginning of the process is vitally important given that about 1 billion people, or 15% of the world's population, experience some form of disability, according to a publication by the WHO on 1 December 2020 (Disability and Health<sup>19</sup>). Given that mobile apps are playing an increasingly important role in our daily lives, it follows that 1 in 7 people may not have the same level of access as other app users.

---

<sup>16</sup> VoiceOver Screen Reader for iOS: <https://www.apple.com/es/accessibility/vision/>

<sup>17</sup> TalkBack Screen Reader for Android: <https://support.google.com/accessibility/android/answer/6283677?hl=ca>

<sup>18</sup> Wireframe or page schematic: This is a visual guide that represents the skeleton or visual structure of a website.

<sup>19</sup> WHO publication on Disability and Health data: <https://www.who.int/es/news-room/fact-sheets/detail/disability-and-health>



The concept of usability, understood as the capacity for a digital solution to be understood, learned, used and be attractive to a user, must also take into account those most vulnerable groups, a fundamental aspect if we want the App be inclusive.

Meanwhile, another aspect to keep in mind when developing a Health App is to ensure that it complies with the principles of the Easy To Read Association<sup>20</sup>. The **Easy To Read Association** (hereinafter, ETR) is aimed at everyone and especially people with temporary reading difficulties (immigration, late reading, poor schooling, etc.) or permanent reading issues (learning disorders, functional diversity, senility, etc.). Texts that appear in an App, for example, are considered ETR, if they follow the international guidelines of the International Federation of Library Associations and Institutions (hereinafter, IFLA<sup>21</sup>) and of Inclusion Europe<sup>22</sup> regarding language, content and form. The Easy To Read Association reviews ETR materials and certifies them with the ETR logo.

The international body responsible for promoting Internet accessibility is the World Wide Web Consortium (hereinafter, W3C<sup>23</sup>), in its Web Accessibility Initiative working group (hereinafter, WAI<sup>24</sup>). This organization publishes a number of accessibility standards for consideration that are contained in the Web Content Accessibility Guidelines<sup>25</sup> (hereinafter, WCAG). Meanwhile, another aspect to keep in mind is that according to Royal Decree 1112/2018, of 7 September<sup>26</sup>, all websites and mobile applications of public organizations must comply with an AA level of accessibility.

There are tools (some of them open) that can be used to check the level of compliance with these standards, as well as to test Apps with different types of filters to meet all visualization needs. There are also others that test the quality of the accessibility of the Apps and that offer proposals for change. When developing an App, it's important to consider add-ons that are specifically designed to improve the use of apps, such as screen readers, dictation tools, or those that increase text size.

---

<sup>20</sup> Easy To Read Association: <https://www.lecturafacil.net/>

<sup>21</sup> International Federation of Library Associations and Institutions: [http://www.cobdc.org/publica/directrius/directrius\\_if.pdf](http://www.cobdc.org/publica/directrius/directrius_if.pdf)

<sup>22</sup> Inclusions Europe: [http://www.lecturafacil.net/media/resources/ILSMH\\_catal%20C3%A0.pdf](http://www.lecturafacil.net/media/resources/ILSMH_catal%20C3%A0.pdf)

<sup>23</sup> World Wide Web Consortium: <https://www.w3.org/>

<sup>24</sup> Web Accessibility Initiative: <https://www.w3.org/WAI/>

<sup>25</sup> Web Content Accessibility Guidelines: <https://www.w3.org/WAI/standards-guidelines/wcag/>

<sup>26</sup> [https://www.boe.es/diario\\_boe/txt.php?id=BOE-A-2018-12699](https://www.boe.es/diario_boe/txt.php?id=BOE-A-2018-12699)

In October 2017, the TIC Salut i Social Foundation collaborated in the publication of the guide “**Digital Accessibility: ICT for everyone**<sup>27</sup>”, drafted by the Table of Entities of the Third Social Sector Of Catalonia through its **m4Social**<sup>28</sup> project. This guide offers best practices and recommendations for developing accessible websites, apps and content, as well as a list of references for existing tools and add-ons.

Entities such as the Organización Nacional de Ciegos Españoles (hereinafter, ONCE), which was established with the main purpose of improving the quality of life of blind or visually impaired people throughout Spain and currently also strives to improve the quality of people with all kinds of disabilities, are aware of the high degree of penetration that ICT tools have had in recent years and have begun a study of technological surveillance in order to understand the innovation trends in the accessible ICT sector (**Inclusive technology**<sup>29</sup>).



In addition, they also propose various initiatives not only in the field of mobile apps, but for technology resources in general, including something as essential as the **design of a document in PDF format**.

The first requirement for a PDF document to be accessible is that it must be a text file and not an image of text. However, even some documents in which the content is text, so the Screen Reader (technical aid used by blind people) can indeed access the information, cannot be considered accessible because the text cannot be properly read. This lack of accessibility may be for a number of reasons; because the order in which the content is presented to the user does not correspond to the logical and visual sequence, because the reading of some paragraphs or lines of text is omitted, because it does not display existing content structures (lists, tables, headers, et.) that facilitate their understanding, because the texts that are links are not identified, because an alternative text is not associated to the images for which the content is relevant, etc.

In addition, the Ministry of Economic Affairs and Digital Transformation published in December 2017 a *Guide to Accessibility of Mobile Applications (Apps)*.<sup>30</sup> The main objective of this guide is to provide the resources needed for developers to create accessible mobile applications. puguin crear aplicacions mòbils accessibles.

<sup>27</sup> Digital accessibility guide: ICT for everyone: <https://www.m4social.org/ca/academy/2017/guia-accessibilidad-digital>

<sup>28</sup> m4social project: <https://www.m4social.org/ca>

<sup>29</sup> Study of technological surveillance for the accessible ICT sector: <https://www.discapnet.es/areas-tematicas/tecnologia-inclusiva/tendencias-en-tecnologia-accesible>

<sup>30</sup> Guide for Accessibility of Mobile Applications (Apps): [https://administracionelectronica.gob.es/pae\\_Home/dam/jcr:3746627f-da12-40af-a5f5-20c42bb8c453/2017\\_Guia\\_accesibilidad\\_aplicaciones\\_moviles\\_apps.pdf](https://administracionelectronica.gob.es/pae_Home/dam/jcr:3746627f-da12-40af-a5f5-20c42bb8c453/2017_Guia_accesibilidad_aplicaciones_moviles_apps.pdf)

## Tips for making Apps accessible

When taking into account the needs of people with functional diversity who have impaired sight or hearing, the App should be made compatible with the voice assistant or vibration functions that many of these devices offer. The developer must tag the content using texts, descriptions for images, correctly name buttons, use text hierarchies and complement the confirmations of actions. Therefore, aspects such as adaptation and context, gesture control and receiving a response to the actions performed when interacting with the App must be kept in mind when creating an accessible App.

- **Adaptation and context:** The content should be adapted to the psychomotor skills of the user. It should speak the same language as the user with words, phrases, and concepts that are familiar and adapted to their skills and the purpose of the application. It is also important to keep in mind how the content is structured, with a relationship that is logical and natural for the user. Finally, universal conventions should be used both from other applications and the real world, which are familiar and known to the user.
- **Gesture control:** If the device allows gesture control, for example, zooming or swiping in the case of mobile devices, this should be implemented in ways known to the user, though always **complimented with a button that performs the same function**.
- **Feedback:** Each time the user performs an action, there should be a response from the App or device. At least a small sound or some form animation or message related to the action. An obvious case where there should be movement on the screen is the loading of information, since **waiting times exceeding 5-10 seconds can be a source of frustration** for the person using the application. A small vibration could also be emitted if the device allows it. The type of alert must be configurable by the person using the application.

In conclusion, developing an App designed with accessibility in mind provides a good experience for the user regardless of whether or not they have a disability. Accessibility includes a wide range of potential financial, moral and legal benefits. As we have seen, both Android and iOS platforms offer various tools to incorporate accessibility solutions into new Apps. Therefore, accessibility is not only seen as an additional feature but a must-have in all mobile apps.

## Usability and the user experience

**Usability** is defined as the ability of a person to use any artificial object or tool to achieve a goal. In this case, the object or tool is an App, which must follow certain criteria so **it can be used easily and intuitively**.

**The user experience** is the set of elements related to the interaction of the person used to achieve a **positive perception of an application**. It is a factor that not only depends on usability, accessibility or the visual design, but also includes aspects related to emotions, the brand or the perceived trustworthiness of the product.

The following are a series of recommendations and considerations aimed at ensuring that the App being developed is usable with regards to browsing, readability and consistency, content and planning.

## Browsability

Browsability in an App is the ease with which a person can browse through all the screens of the App. To achieve this goal, the App must offer resources and an optimal design for locating the information. Answering questions like, “Where am I? Where have I been?” and “Where can I go?” help to endow Apps with good browsability. Concepts such as efficiency, effectiveness, clickable area, affordance<sup>31</sup>, are key elements when browsing through the App.

- **Efficiency and effectiveness:** An App should provide the user with an **intuitive, fast, and easy** way to interact. Whenever possible, an attempt should be made to comply with **the 3-click rule** so that the user can access all the information in a maximum of 3 steps. It is important to keep the **user’s learning process** to a minimum; and if minimal learning is essential, guide it step by step through a tutorial. It is also necessary to decide whether **the interface will be displayed both horizontally and vertically**.
- **Clickable area:** The buttons should not be too small. They should be at least **9 mm wide and tall** (the equivalent of about 44px high), a ratio determined by the proportions of users’ fingers. The separation between the buttons should also be considered to avoid undesirable interactions such as pressing two buttons at once. Finally, it is essential that every button, link or clickable area has the appearance of what it represents. Either as the user would expect it to be or using a colour in the app that is part of a colour code established within the universe of the platform.
- **Visibility and status:** The application should let the user know where in the application structure they are and the status of the current process. This information should be clear and should be provided in a timely manner. These actions help the user to follow the process and browse correctly, avoid or correct errors, and enhance the feeling of control and familiarity with the application.
- **Affordance:** Affordance is an important feature that enhances the usability and browsing of the application. This term refers to the ability of the different elements of the application to represent their function. In other words: that a button looks like a button, that a text field in a form indicates what it is and that it can be edited; that an icon for displaying a list of items tells the user what it is or its status; or that an error message is visible, obvious, and indicates the type of error.

---

<sup>31</sup> Affordance: Term that refers to the ability of the different elements of the application to represent their function.

## Readability and consistency

Text elements should be easy and simple to interpret for anyone interacting with the App, the text should make sense and be understandable. Aspects such as layout, colour, typography, icons, and operating system are elements to consider in the design of this part of the App.



- **Minimalist design:** The design of the interface should be as simple and **practical as possible**, eliminating elements that do not add anything, something which does not preclude an attractive design. It is easier for the user to remember and interact with a simple structure than a complex one with many options and information that can generate stress, screen noise and erroneous decision making. All actions and interactive elements must be easy to find.
- **Colour:** The use of colour is an important factor in helping to prioritize information and tidy up content within the application. The **“60-30-10” rule** should be followed, where: 60% is in the main colour (a background colour that helps reading); 30% is in a secondary colour that helps to complement and contrast the visual aspect; and finally, 10% is reserved for accent colours (links, buttons, important items, etc.). Colours should be part of a colour range that offers high contrast and clearly defines the main elements. Chromatic resources that follow the universal colour code can be used, such as red for error or cancellation, and green to denote correctness or confirmation.
- **Typography:** It is important that text is highly readable so that messages can be conveyed easily and clearly. It is recommended that the fonts of the operating system itself be used, since these have been specially designed for this purpose. In addition, the fonts should be limited to between **1 and 3 types** with maximum of **3 different sizes** or styles. All user profiles and their needs should be considered, such as older people who need a larger text size compared to younger users. **Different styles and hierarchies** also need to be applied to help tidy up the information. For examples titles, subtitles, text fields, etc. should be used.
- **Icons:** If iconography is to be used to represent the usual functions of an App, it is best to imitate the standard design of applications that are well known to the user instead of implementing a completely new design. These icons should have the same graphic and functional consistency, so that the user can easily identify and relate to them, following the universal meaning that they have in other types of applications.
- **Consider the operating system:** If the operating system has buttons for browsing, they should be supported in the App. For example, for Android, the OS Back button should always work, even if there is already a specific button to perform this function within the application.

## Content Organization

The content of an App can be structured in different ways, depending on whether it is hierarchical, linear, both or networked. Depending on its ultimate goal, the interaction with the user can be:

- **Hierarchical:** There is an **index or main screen** from which the rest of the pages are accessed, which can also link to sub-pages, creating a hierarchy. In this type of organization, there must be a menu that allows browsing.
- **Linear:** From any particular screen **the following and previous screens can be accessed**, as if they were the pages of a book. This is not recommended if the number of pages is very high, but it could be used for example for a tutorial.
- **Linear with hierarchy:** This is a hybrid structure, organized in a hierarchical way, but with the option of browsing linearly along pages at the same level.
- **Networked:** There are no apparent order. It is not a recommended structure when there are a large number of pages because it disorients the user.

In any of these cases, the content must contain **concise, precise and clear descriptions**, avoiding redundancy; especially considering the limited space available on mobile phones and tablets compared desktops.

## Planning

In order to offer an optimal experience, a set of **functional specifications** should be drawn up that detail all the functionalities that the App must include in order to **meet the needs of the people to whom it is addressed, from a functional point of view** (not technical). These functional specifications include the following elements:

- ✓ Detail **critical situations** involving the functionality of a particular product.
  - ✓ Describe the **functions** to be specified.
  - ✓ **Avoid technical details**, unless strictly necessary.
  - ✓ Provide **concise, accurate and clear descriptions, avoiding redundancy**.
- **Detecting and minimizing possible user errors:** It is necessary to detect in advance the possible mistakes that people who will use the application may make to avoid future frustration. This will allow prevention systems to be designed to prevent these errors. An example is Google search engine's autocomplete function (which corrects the proposed search term) or form validation. It is also necessary to take into account mistakes made by the users themselves and not system errors, allowing users in these cases to quickly find the option that undoes the action they have just performed.

- **Flexibility and adaptation:** An App can be used by all kinds of people. If the App has advanced features, the inexperienced user should not be forced to use them. For example, the Google search engine incorporates operators to better filter searches, but the user does not need to know them to use them.
- **Help and documentation:** The App **should not need documentation** in order for the user to be able to use it, since one of the basic concepts of usability is that it is intuitive. However, support should be provided through the website of the application or the application itself, via a section of **frequently asked questions** or displaying **question marks next to certain functions**. Therefore, documentation on the features of the App and how they are used should be available.

## User experience

The user experience is the set of elements related to their interaction that leads to their positive or negative perception of an application. It is a factor that not only depends on usability, accessibility or the visual design, but also includes aspects related to emotions, preferences or the perceived trustworthiness of the product.

The following are aspects that can influence the final perception that the user will have of the application:

**Consider the first-time user experience:** The information that will be displayed when the user has not yet entered data must be defined. It is usually not a good idea to leave parts of the screens simply “empty”, without prior information or related instructions. Examples include displaying the “Favourites” tab when the user has not yet defined these, the appearance of the shopping cart when it is empty, or the absence of search results. These issues can be resolved with information messages such as: “The favourites list is empty” or “No products have been added to the basket”, but it is more useful to use the space to teach the user how to fill them in.

**Error messages:** The application must anticipate as many possible errors as possible that might occur and not display descriptions that are incomprehensible to the user. In this regard, it is necessary to identify elements that can trigger these errors, such as mandatory form fields, accepted data type restrictions, or the maximum file size; and implement prevention systems. In the event of an error in the application, the user must be informed of it with an indication on how to fix it. The user must know at all times what the state of the system is, to avoid feeling confused or lost.

**Offer the option to undo changes.** Any action, such as deleting or modifying an item, should always take into account that the user may have made a mistake or may want to undo the changes. Therefore, it is a good option to incorporate a button to “undo” or “cancel” the action and/or indicate how the user can restore or reactivate these archived or deleted items. Informing the user that they have the option to undo the actions they have taken makes them more confident when using the App and not afraid to explore it.

**Include confirmation messages.** Any action that does not allow the undo option must include a message for the user to confirm that they really want to perform the action and inform them that they will not be able to undo this action. A simple “Accept” or “Save” button is often displayed, the consequences of which are not known. So, reminding the user that the action they are taking is permanent, is a good option to make them more comfortable using the App.

**Make buttons and menus familiar.** Design them in a way that is intuitive and familiar to the user. For example, using the same icons used by the most popular applications or those in the same field is a good option. Similarly, using a graphical interface similar to the majority of applications or following a pattern within the application will enable an easier and more intuitive use of the app.

**Manage waiting times.** Inactivity is one of the most annoying things for users, who do not like to wait for the results of their action. Users must be notified in some way that the app is performing actions that may take some time (to prevent the user from concluding that the App has frozen, for example). The most common solution is to use an element that shows progress in the form of a bar, clock or throbber, these wasted moments can also be livened up with more attractive content such as jokes, tips for use or interesting facts about the App.



## 3. Aspects related to the App development process

### 3.1 Development life cycle

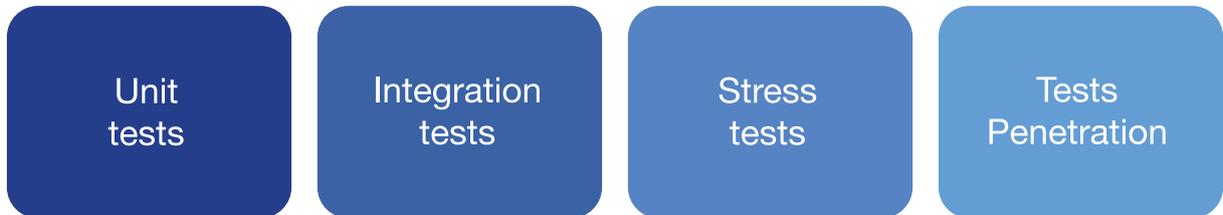
The software development process or lifecycle has various characteristics depending on the scope, functionality, or risks associated with the type of application being developed. Depending on these variables, aspects such as **complexity, duration, or cost may differ** significantly. This section describes the methodologies that can be applied to develop applications, with the corresponding phases and roles in each of them

#### Phases of the development process

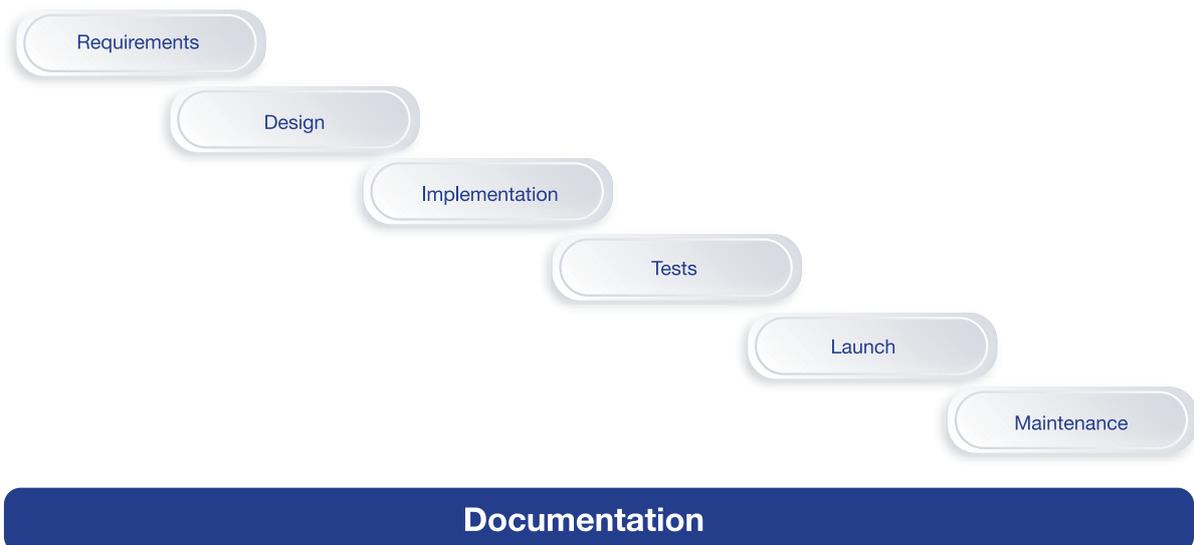
The development process can be managed following different methodologies, the choice of which will be made according to the type of project being carried out. However, there are some basic steps that are common to almost all software development processes, which are detailed below:

- **Definition of requirements:** Initially, the **use cases or functionalities** to be covered by the App are identified and **defined**, though these may be modified throughout the development process if deemed appropriate. Modifications can be due to different factors such as changes in the needs of potential users, legal changes or resource limitations among others.
- **Design:** A **functional and technical analysis** is performed to determine which requirements will be implemented and how they will be applied.
- **Implementation:** The **functionalities**, graphic elements, etc. specified in the design phase are **developed**.
- **Testing:** All the necessary tests are performed to ensure that the software meets the needs identified in the technical and functional design:
  - **Unit tests** are those that evaluate functionality in isolation from the rest of the system, such as a method or function.
  - **Integration tests** involve testing how two or more elements work together, for which the relevant unit tests should have previously been carried out.
  - **Stress tests** focus on the performance of the software when it is taken to the limit, to see to what extent it continues to operate normally and what happens when this threshold is exceeded. This type of testing also helps calculate the scale of the hardware required.

- **Penetration tests** are tests that simulate a malware attack which, using every technique available, attempts to breach the security of the application to extract data, corrupt its operation, etc.



- **Launch:** The software is configured, deployed, and distributed so that users can start to use it.
- **Maintenance:** Process in which measures are applied to correct detected errors in the application, as well as improvements to meet new requirements that may appear over time.
- **Documentation:** This is a cross-cutting task to be performed throughout the product life cycle (including the maintenance phase), which involves documenting the functionalities and use cases for which the App aims to provide a solution, the internal design of the software, the modifications that are carried out in the course of the process, etc.



## 3.2 Development methodologies

There are different methodologies for developing applications, which are classified into paradigms based on their main characteristics. Two of the most widely used paradigms today are:

- The **traditional cascade methodology**, in which the phases follow on from each other linearly. This is an inflexible approach, since once one phase is over the next one begins, with the aim of not going back to the previous phase at any point in the future. Cascading methodologies are suitable for projects where the requirements and scope are very clear and the team of developers has extensive experience in the technology being used.
- **Agile methodologies** seek to offer great flexibility, allowing the development to be adapted to any change that is required. In this sort of approach, the modification of requirements is constant and common, and is seen as a means to increase the chances of success of a project and prioritize the software that works. Working in this way means not all the requirements need to be defined or implemented in order to make the application available to users, reducing the time-to-market.

### Agile Methodologies

Agile methodologies are based on the principles of the so-called agile manifesto<sup>32</sup>, which can be grouped into the following four points:

- **Prioritize individuals** and interactions over processes and tools.
- **Prioritize working software** over comprehensive documentation.
- **Customer collaboration** over contract negotiation.
- **Responding to change** over following a plan.



<sup>32</sup> Agile manifesto: <http://agilemanifesto.org/iso/ca/manifesto.html>

It is very important to keep in mind that prioritizing does not mean neglecting aspects such as documentation or contract negotiation because if, for example, the contract is too ambiguous or the documentation is incomplete it is very likely this will lead to lots of issues during the software lifecycle.

In an agile project the cost can also vary, as it depends on the modifications needed to meet the changing requirements during the course of the project. Another of the main characteristics of these methodologies is the role that customers play in them: a relationship of collaboration and trust must be established to be able to define the requirements and carry out the validations of the developed functionalities.

These methodologies are suitable for less defined, quite changeable projects in which the customer is very involved and willing to invest their efforts. Some of the best known agile methodologies are:

## Scrum

This framework defines incremental, flexible stages that allow the development to respond to changing needs. It is suitable for fairly complex projects in which not all the requirements can be defined at the start and in which these are susceptible to being modified or added to. Regarding the composition of the team, it should have between three and ten members.

Scrum<sup>33</sup> is organized into iterations called sprints<sup>34</sup>, at the end of each of which a product increment should have been developed that is potentially deliverable. The duration of a sprint is usually between one and four weeks.

A Scrum team has three well-defined roles:

- **Product Owner:** This role acts as a liaison between the customer and the team, looks for funding and decides what features and functionalities are essential for the product to be delivered. Non-essential tasks are ordered by assigning a priority.
- **Scrum Master:** Responsible for training and guiding the team in the principles and concepts of Scrum as well as facilitating the process. This person must ensure that the Scrum is understood and properly applied within the team.
- **Development Team:** Relatively small multifunctional and self-organized team. If the team is very large it will become difficult to achieve this self-management due to the lack of hierarchy within the team. On the other hand, if it is very small it will be difficult for it to contribute all the knowledge needed for the project. The team is in charge of negotiating with the Product Owner what developments will be carried out in each sprint and of implementing them.

The Scrum lifecycle begins with the creation of a list by the Product Owner of priority-ordered requirements called the product backlog<sup>35</sup>. The Product Owner then negotiates with the

---

<sup>33</sup> Scrum methodology: [https://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development))

<sup>34</sup> Sprint: Basic unit of development in Scrum. The work that will be developed during this stage is identified and at the end of each sprint the increment achieved to fulfil the final goal of product delivery must be presented. [https://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)#Sprint](https://en.wikipedia.org/wiki/Scrum_(software_development)#Sprint)

<sup>35</sup> Spring backlog: Detailed document defining the tasks required to fulfil the requirements assigned to the current Sprint. [https://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)#Sprint\\_backlog](https://en.wikipedia.org/wiki/Scrum_(software_development)#Sprint_backlog)

development team what requirements will be developed in the next sprint. Once the sprint has begun, the development team is in charge of managing the included tasks. A quick stand-up meeting is held daily with the team, the Scrum Master, and the Product Owner to clarify questions, troubleshoot, discuss issues, and more.

Once the sprint is over, the team presents the development and reviews it. The iteration ends with a sprint retrospective, in which possible improvements are assessed based on the experience. Then another sprint begins, and the process is repeated until all the requirements have been developed.

### Extreme Programming (XP)

This methodology has a dual purpose, on the one hand to improve the quality of the software, and on the other, that of the conditions of the development team. To achieve this dual goal, the following 5 basic values are defined:

- **Communication:** software development is considered a team effort and therefore communication, preferably face-to-face, between its members is considered essential.
- **Simplicity:** only what is absolutely necessary should be developed and the design of the solution should be as simple as possible, to facilitate the maintenance, support and review of the product.
- **Feedback:** the large amounts of feedback received from the various sources should be used to identify areas for improvement and review the testing carried out.
- **Courage:** developers need to act bravely and not be constrained by fear. For example, they should be bold enough to accept that a process is not being done correctly and try something new or detect and report organizational problems.
- **Respect:** it is key that the members of a team respect each other to be able to work together, enabling communication and collaboration.

### Kanban

This methodology aims to achieve a quality product and minimize bottlenecks that may occur throughout the project. It follows these basic rules:

- **Accept change:** as is common in agile methodologies, the idea is to motivate people to see change as something positive and encourage them to change what doesn't work well or can be improved.
- **Leadership:** the leader must take initiative and properly manage the assigned team or task.
- **Respect the active process and responsibilities:** each member of the team must be clear about their role at each stage of the project, so these need to be defined in advance.

- **Kanban does not set out how a task should be done**, but it helps to decide if it is being done in the best way or if something can be improved.

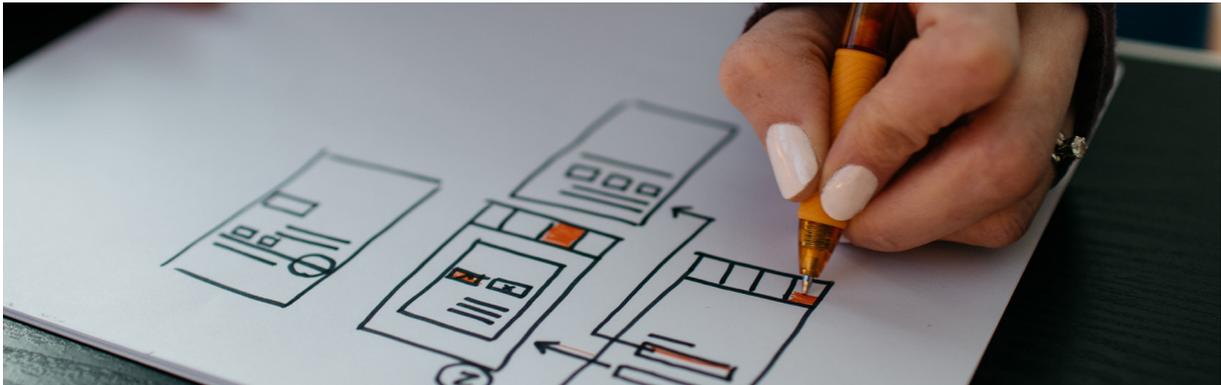
### Selecting a methodology

The decision on which methodology best suits the needs of a specific project should take into account factors such as the characteristics of the company or entity, the characteristics of the specific software or the size and characteristics of the team that will be involved. This decision must be made specifically for each project, and it may be the case that different methodologies are used in the same organization depending on the project.

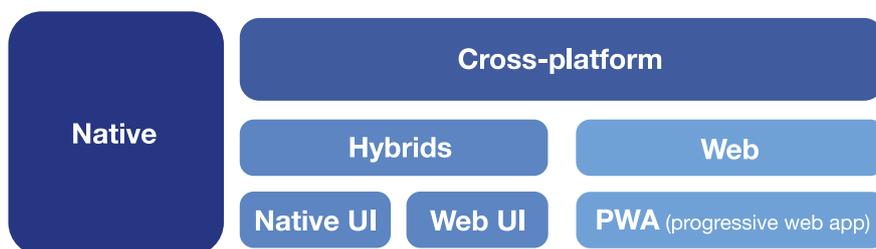
The volatility of the mobile environment and the Apps market means that the initial requirements of an App are constantly changing. In many cases, minimizing the time-to-market is a priority, which requires a functional product to be ready as soon as possible. For these reasons **agile methodologies are more often associated to the development of this type of application, since they allow the software lifecycle to be adapted to this highly changing environment.**

### 3.3 Development types and technologies

When starting an App development project, one of the decisions that will determine which skills are required from the **development team**, the **cost** of the project, or the **end solution** is **choosing the development type and technology** that best suits the needs of the project. To make this decision, a rough outline of the design of the App and of the main features and functionalities that it needs to cover should be drawn up.



There are four types of technological solutions for developing mobile applications:



The first solution has a more native user interface<sup>36</sup> (hereinafter UI) compared to the fourth solution which is the least native. **Native development** involves programming that is linked to the operating system (hereinafter, OS) of each platform. It is a good option especially if the App has at a very specific goal and if it is being developed for a single platform (iOS, or Android, or Electron, etc.). It's also a great option if the App needs as native a UI as possible, and to maximise the use of the functionality of the mobile phone. Meanwhile, **cross-platform development** consists of a single development that will be used for all platforms, i.e. a single code is developed that can be compiled on different operating systems. There are three different options each with their advantages and disadvantages: some offer better UI, others require a less specialized development team, others allow the code to be kept together, and in others the maintenance is native, etc. It is the most widely used option today by many mobile app developers, as it reduces costs and development time compared to native development, as long as the ultimate goal is to reach all available platforms. In order to understand in more detail the functionalities offered by the different types of developments available and the technological possibilities that exist on the market, it is necessary to have a good understanding of the following elements:

<sup>36</sup> The user interface (UI) is the display that allows a user to interact effectively with a system.

- **The type of user interface that will be displayed:** The user interface is the display that allows a user to interact effectively with a system. It is the sum of the *information architecture* plus the *visual elements* that comprise it and the *patterns of interaction*. It is often confused with the concept of usability which simply refers to those aspects that facilitate the use by the user.
- **The desired degree of user experience:** The user experience (hereinafter UX) refers to the perception of the user when interacting with the product or service. A good UX is achieved by designing useful, usable, and attractive products, which in turn provide satisfaction to the user. In the health sector the user may be the patient, the health professional, or administration staff, among others, and the solutions that are developed must be oriented to the target profile.
- **The ecosystem in which the information will be shared:** An information system is a system made up of the set of resources (people, data, activity, etc.) that process an organization's information. Knowing the information systems used in the health ecosystem will allow the designer to anticipate synergies or possible collisions with each type of development.
- **The operating systems on which it will be used:** An operating system is a set of programs and functions that hide the details of the hardware and offer the user a simple and flexible way to access the system. Some examples of operating systems are: *Android, iOS, Windows, macOS, Linux, etc.*
- **Compatibility with the different browsers available in the case of a progressive web app:** A web browser is a computer program that allows the user to retrieve and reproduce hypertext documents, usually written in HTML, from web servers located anywhere in the world. This network of documents is known as the World Wide Web. Some examples of web browsers are: *Chrome, Explorer, Firefox, Edge, Safari, etc.*
- **The programming languages required for each type of development:** A programming language is a computer language used to control the behaviour of a machine. Each language has a series of strict syntactic and semantic rules that must be followed to write a computer program, and which describe the structure and meaning respectively. These rules allow the programmer to specify both the type of data that the program will work with and the actions it will perform. While some languages are defined by a formal specification (a document), others are unofficially defined by a specific implementation (a compiler). Some examples of programming languages are: *Java, Swift, PHP, JavaScript, HTML, CSS, SASS, Typescript, Python, C#, .NET, C++, etc.*
- **The libraries, APIs, SDKs, and Frameworks available for each type of development:** A **library** is a set of functional implementations, coded in a programming language, that provides a well-defined interface for the functionality being invoked. Programming languages often have libraries for all sorts of tasks such as data processing, graphing, text analysis, and so on. Once included, libraries streamline the programming process, as they avoid rewriting existing functions. The number of people in the community involved in the maintenance and evolution of libraries and how often they are updated should be

taken into account. Some examples of libraries are: Math, Retrofit, Parcelable, JSON.parse, Alamofire, Lifecycle, etc.

**API (Application Programming Interface)** is the term that refers to the “face” of the library, since it is accessible to the programmer. An API is a logical representation of what is in the library and the relevant documentation that explains what the programmer can do with the library. The difference is that the library refers to the code itself, while the API refers to the interface. Examples of APIs include the Google API, the Facebook Messenger API, etc.

An **SDK (Software Development Kit)** is a complete set of software development tools for a specific platform. This “kit” can include all kinds of things, such as: libraries, APIs, IDEs, documentation, etc. Examples of SDKs are: Android SDK, iOS SDK, Windows SDK, etc.

A **Framework** is a generic structure that provides a skeletal architecture with which specific software can be implemented. Abstraction allows the reuse of common design patterns, while leaving the specific details to the developers. Reusing common design patterns means having the general structure to solve similar types of problems. The Framework can be manifested as functions and classes that must be implemented, such as the *run ()* method in *Java Swing*, which requires the user to conform to the design pattern that the framework deals with. Some examples of frameworks are: *React*, *Java Swing*, *Model-View-Controller*, *React Native*, *Ionic*, *Spring*, *Xamarin*, etc.

- **The integrated development environments (IDEs) depending on the type of programming language, libraries, frameworks, APIs, or SDKs:** An IDE (Integrated Development Environment) is an application that helps during the code writing process by automating many useful processes such as debugging, refactoring, code generation, etc. Some examples of IDEs are: Eclipse, IntelliJ IDEA, NetBeans, Visual Studio, etc.

Once these elements have been identified, the following decisions must be made in order to determine the type of development that best suits the needs of the project:

- 1. **Decide** which platform(s) must be reached: smartphones, tablets, e-books, smart watches, medical devices, wearables, etc.
  2. **Prioritize** the criteria for selecting which type of development (Native, Native Hybrid, Web Hybrid, PWA) best suits the needs of the project.

3. **Choose** according to the type of development, the technology that best suits the needs of the project. It is important to find out about the characteristics and differences between the frameworks and the IDEs that can be used for the same type of development, since it will be necessary to choose the most appropriate one. For example, for a Native Hybrid development, the developer can choose between different frameworks such as Xamarin, React Native, Native Script, etc. These will also determine the programming language to be used, the IDE, etc.

## Gartner

An example of a tool that facilitates these decisions is the methodology offered by Gartner<sup>37</sup> in its article “*Decision Point for Selecting Your Mobile App Architecture*”. This methodology provides selection criteria when choosing which type of development best suits the needs and priorities determined by the project team. Team members must establish the priorities of the App based on its needs. The priority scale is categorized into 5 levels, as shown in the following table:

Priority	Value
1	Not important
2	Mildly important
3	Important
4	Very important
5	Essential

This value will be used to determine the priorities in the decision criteria list in Table 1, where the **priority** value to be established for each decision criterion must be added to the priority column (e.g. if an App needs to have a very high level of security, the criterion “**security**” will be valued with a priority value of 4 or 5 if it is considered an essential element). This value will serve as a multiplier in the value code identified for each type of feature according to the type of development, defined in colours, in table 2. The colour code in Table 2 is detailed below:

Colour	Value code	Order
RED	1	Low
ORANGE	2,3,4	Intermediate
GREEN	5	High

<sup>37</sup> Decision Point for Selecting Your Mobile App Architecture, Gartner: <https://www.gartner.com/en/documents/3889491/decision-point-for-selecting-your-mobile-app-architectur>

According to Gartner, the decision criteria used in this tool are as follows:

DECISION CRITERION	DESCRIPTION AND FOCUS	PRIORITY (Scale from 1 to 5, with 1 being the lowest priority criterion and 5 the highest)
Portability	The ability to deliver an app to multiple platforms using a single development tool, framework and programming language.	(Select from 1 to 5)
Maintenance	The ease with which a development team is able to make changes to a mobile app.	(Select from 1 to 5)
App distribution and updates	The ease of delivering an updated version of a mobile app to the mobile device. This includes the ease of initial deployments to user devices, ease of managing app upgrades and support for continuous deployment.	(Select from 1 to 5)
App Store Presence	Determines the presence of an app in an app storefront, such as Apple's App Store or Google Play.	(Select from 1 to 5)
Access to device features	Ease of access to common capabilities found across all mobile platforms, including camera, GPS, calendar and email.	(Select from 1 to 5)
Ease of native integrations	Ease of access to proprietary capabilities found on a single mobile platform. Examples include Android TV on Android, and Face ID and watchOS on iOS.	(Select from 1 to 5)
Interapp communication	Identifies the ability for an app to invoke another app and share data.	(Select from 1 to 5)
User interface	Determines the control a developer has on native UI controls, and ensuring the app conforms to basic native characteristics.	(Select from 1 to 5)
Security	Reflects the ability to build and/or link security and identity capabilities into the app and to code the app in a secure way,	(Select from 1 to 5)
Performance	Indicates the rendering, responsiveness, and execution speed of the application.	(Select from 1 to 5)
Architectural agility	The ease with which a developer is able to switch between alternative architectures.	(Select from 1 to 5)
Simplicity	The overall simplicity of the architecture.	(Select from 1 to 5)
Use of web skills	Indicates the extent to which the architecture makes use of the existing web development skills of the development team.	(Select from 1 to 5)
Use of native skills	Indicates the extent to which the architecture makes use of the existing native development skills of the development team.	(Select from 1 to 5)

**Table 1.** Description and prioritization of the reference criteria for the type of development

Based on the priorities identified in the table above, Gartner proposes the following comparative table for each type of development. Depending on what the App development team has chosen, the result will be the best type of development that best suits the needs of the project.

	Natives	Híbrides UI Nativa	Híbrides UI Web	Web Progressives
Portability	2	4	4	5
Maintenance	2	3	4	5
App distribution and updates	2	4	4	5
App Store Presence	5	5	5	1
Access to device features	5	5	4	3
Ease of native integrations	5	4	2	1
Interapp communication	5	5	3	1
User interface	5	5	2	2
Security	5	4	3	2
Performance	5	4	3	2
Architectural agility	2	3	4	5
Simplicity	3	2	4	5
Use of web skills	1	3	5	5
Use of native skills	5	4		1
FORMULA TO OBTAIN THE RESULT (Best Type of development = highest sum)	$\Sigma$ (Value code column * priority of each criterion)	$\Sigma$ (Value code column * priority of each criterion)	$\Sigma$ (Value code column * priority of each criterion)	$\Sigma$ (Value code column * priority of each criterion)
Example (Best result = highest sum)	$2^*p + 2^*p + 2^*p + 5^*p + 5^*p + 5^*p + 5^*p + 5^*p + 2^*p + 3^*p + 1^*p + 5^*p = \text{result}$	$4^*p + 3^*p + 4^*p + 5^*p + 5^*p + 4^*p + 5^*p + 5^*p + 5^*p + 4^*p + 4^*p + 3^*p + 2^*p + 3^*p + 4^*p = \text{result}$	$4^*p + 4^*p + 4^*p + 5^*p + 4^*p + 2^*p + 3^*p + 2^*p + 3^*p + 3^*p + 4^*p + 4^*p + 5^*p + 2^*p = \text{result}$	$5^*p + 5^*p + 5^*p + 1^*p + 3^*p + 1^*p + 1^*p + 2^*p + 2^*p + 2^*p + 5^*p + 5^*p + 5^*p + 1^*p = \text{result}$

"p" refers to the priority given to that criterion with that value code.

Legend: **RED** (negative), **ORANGE** (can be improved), **GOOD** (positive).

**Table 2.** Table of characteristics of the development type.

This is an example of a tool to support decision making, but it should be said there are **no wrong or incorrect solutions and technologies** for developing a mobile app. Choosing a particular type of development or technology will **depend on the needs identified** by the project team.

For each type of development, there are different technologies that will determine which platforms, web browsers, programming languages, frameworks and IDEs to use. The following table lists some examples of development technologies for each type of development:

Development type	Platforms	Web browsers	Programming Languages	Frameworks	IDEs
PWA (Progressive Web Apps)	Web browser	<p><b>Compatible:</b> Chrome / M, Firefox /M, UC Mobil, Opera, Xiaomi Mobil, 360explorer / M, Baidu Mobile, Edge, Sogou Mobile, Quark Mobil, WeChat Mobile.</p> <p><b>Partial support:</b> QQ Browser / M, Samsung Mobile.</p> <p><b>Bug/error support:</b> Cheetah, Safari / M. (M: means the indicated browser, plus the smartphone/mobile version)</p>	HTML, CSS, Javascript, PHP, etc.	Symphony Django, Spring, Laravel, React, Angular, Express	VisualStudio, PyCharm, PHPStorm, WebStorm, NetBeans, KomodoEdit, Brackets, etc.
UI Web hybrids	Android, + iOS, + Electron, + PWA (can be generated by adding two lines of code)	In the case of generating PWA check compatibility.	HTML5, CSS3, JavaScript, Typescript, SASS, etc.	Ionic, Angular, Capacitor, Node.js, Cordova, Phonegap, etc.	Visual Studio, Webstorm, Angular IDE, Brackets etc.
UI Native hybrids	Android, + iOS, + Electron, + PWA (can be generated by adding two lines of code)	In the case of generating PWA check compatibility.	JavaScript, .Net, C #, AngularJS, JSX, etc.	React Native, NativeScript, Node.js, Vue.js, Xamarin, Flutter, etc.	Visual Studio, PyCharm, JetBrains Rider, etc.
Native	Android or iOS	N/A	for Android (Java, Kotlin), for iOS (Objective C, Swift), etc.	for Android (Jetpack) for iOS (SwiftUI)	Android Studio, xCode

**Table 3.** Examples of development technologies for each type of development

## For Native apps *Android Operating System*



Initially the reference programming language for Android app development was Java<sup>38</sup>. In 2017 *Kotlin* was offered as an alternative and in May 2019 Google decided that the reference language for Android would be ***Kotlin***<sup>39</sup>. These languages are used to implement the app's functionalities, storage management and data access, and hardware interaction (such as camera or GPS). As for the user interface in **native applications**, this is created with a wizard provided by the development environment, which is responsible for generating the corresponding **XML**<sup>40</sup> design and configuration. Through the wizard (and/or this XML) the elements of the application such as the position, appearance or distribution of buttons, forms and images can be arranged and customized.

The official development environment for both native and hybrid Android applications is ***Android Studio***<sup>41</sup>. This software offers a code editor, Android and Java libraries (SDKs), a real-time application viewer, and a virtual machine that allows the App to be run and tested. Other environments, such as ***Eclipse***<sup>42</sup>, can also be used if specific components are previously installed in order to be able to compile and run Android.

Android applications can also be developed with other languages such as *GoLang*<sup>43</sup> and/or C++, using Android Native Development Kit (NDK) for developing apps with low-level functionality.

---

<sup>38</sup> <https://www.java.com/es/>

<sup>39</sup> Kotlin: <https://kotlinlang.org/>

<sup>40</sup> XML tutorial: <https://www.w3schools.com/xml/>

<sup>41</sup> Android Studio development environment: <https://developer.android.com/studio>

<sup>42</sup> Eclipse development environment: <https://www.eclipse.org/>

<sup>43</sup> Golang: <https://golang.org/>

## iOS operating system



In the case of the iOS operating system, apps can be programmed using **Objective-C**<sup>44</sup> or **Swift**<sup>45</sup>. *Objective-C* is the language initially used in iOS app development, but in 2014 Apple announced the launch of a new, easier, more powerful and intuitive language called *Swift*. However, it is still possible to develop apps using only *Objective-C*, and in fact, the major libraries for iOS are written in this language, including the API-specific *Cocoa Touch*. *Swift* is also based on this same API, making it easy to migrate an *Objective-C* application to *Swift*, and both languages can be used together in the same app development.

In **native applications**, both *Objective-C* and *Swift* are used for functionality development, data management, and hardware access, but also for implementing the UI presentation layer.

The official development environment for iOS is *XCode*<sup>46</sup> (for the *MacOs* operating system) and it offers a code editor with version control, as well as test and simulation tools.

---

<sup>44</sup> Objective-C for Apple: <https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>

<sup>45</sup> Swift for Apple: <https://developer.apple.com/documentation/swift#2984801>

<sup>46</sup> XCode development environment: <https://developer.apple.com/xcode/>

## 4. Aspects rated to markets, publishing and classification

App stores, also known as “markets”, are web platforms that allow you to sell or distribute mobile applications. Depending on the operating system they target, Apps will be released in a specific market, with the **Google Play Store**<sup>47</sup> for Android and the **App Store**<sup>48</sup> for iOS the most used. There are also other app stores such as: **Amazon App Store**<sup>49</sup> which is the main alternative to Google Play Store; **F-Droid**<sup>50</sup> with the restriction that all the applications it offers have been developed using free code which can be accessed and/or contributed to; **GetJar**<sup>51</sup> which is a cross-platform store (Android, iOS, Windows Mobile); and **Microsoft Store**<sup>52</sup> which is the official store of Microsoft.

This section describes in detail the Google Play Store and App Store markets, highlighting some tips for publishing applications.

### Google Play Store



Google Play is Google’s official store which offers digital books, music, movies, news and publications, as well as apps and games. The specific section for mobile apps is called the Google Play Store and it contains more than 3.7 million apps organized into different categories, including “Health and Fitness.”

To publish an application on the Google Play Store, you must comply with the **developer program policies** and agree to the *developer distribution agreement*<sup>53</sup>. The company offers advice on how to comply with these policies, relating to content, identity theft, intellectual property, privacy, security and fraud, revenue or the presence of advertising. These tips can be found at the following link: [Developer policy centre](#)<sup>54</sup>.

You need to create a file associated with the App: title, short description, screenshots, high resolution icon, featured image, category, content rating, contact information, specify privacy policy, specify whether to apply payments, distribution. The following aspects should be considered when publishing an app on the Google Play Store:

1. Draw up a **comprehensive test plan** that includes both unit and integration tests, to minimize possible errors that may arise and to ensure the **user experience is altered as little as possible**.

---

<sup>47</sup> Google Play Store: <https://play.google.com/store/apps>

<sup>48</sup> App Store: <https://www.apple.com/es/ios/app-store/>

<sup>49</sup> Amazon App Store: <https://www.amazon.com/gp/mas/get/amazonapp>

<sup>50</sup> F-Droid: <https://f-droid.org/en/>

<sup>51</sup> GetJar: <https://www.getjar.com/>

<sup>52</sup> Microsoft Store: <https://www.microsoft.com/es-es>

<sup>53</sup> Google Play Developer Distribution Agreement: [https://play.google.com/intl/ALL\\_es/about/developer-distribution-agreement.html](https://play.google.com/intl/ALL_es/about/developer-distribution-agreement.html)

<sup>54</sup> Google Play Developer Policy Center: [https://play.google.com/intl/ca/about/developer-content-policy/#?!?modal\\_active=none](https://play.google.com/intl/ca/about/developer-content-policy/#?!?modal_active=none)

2. The application can be published in alpha or beta mode to ensure its quality before being published in production.
3. Review and ensure that the App **complies with all market policies**.
4. Ensure that the **description** of the App is correct for all audiences, and that it is **understandable and concise**.
5. Ensure that you have the **copyright** for all images used, including those displayed in the market.
6. If the App uses user data, ensure that a **privacy policy** is included that describes the data that is collected and its purpose.
7. Ensure that the App and the ads displayed on it do not contain **inappropriate content** (such as sexually explicit content, hate speech, threats, or harassment). Also, if ads are included in the app, they should not be annoying nor misleading.
8. Include all the **basic and relevant information about the app (with translations if you want to publish in different language domains)**: an accurate description, images that show how it works, the corresponding **PEGI classification** and, if necessary, a demonstration video.<sup>55</sup>
9. Choose an icon that follows the Android recommendation guides
10. Ensure that the **contact email associated with the App is correct** so that Google can communicate any issues with the app.
11. **Sign the App in release mode**.
12. **Correctly choose the package names since they are unique, permanent and cannot be reused in the future**.
13. Different applications (APKs) may be published in the same App file for different device configurations and/or compatibility.
14. App Optimization: Disable unnecessary logs, unused resources, etc.
15. Verify that the permissions defined in the manifest are actually required.
16. Verify the chosen minimum version.
17. **Correct any errors** reported or contact Google to report any disagreement or additional information needs.
18. Use tools like **Crashlytics**<sup>56</sup> or **Google Analytics**<sup>57</sup> to simplify the management of corrective and evolutionary maintenance and to analyse how the App is being used.

---

<sup>55</sup> PEGI classification: Code that allows you to classify video games by age group. [http://www.acpb.cat/infancia-i-joventut/videojocs-noves-tecnologies/#El\\_Codi\\_PEGI\\_de\\_l8217etiquetatge\\_del\\_videojoc](http://www.acpb.cat/infancia-i-joventut/videojocs-noves-tecnologies/#El_Codi_PEGI_de_l8217etiquetatge_del_videojoc)

<sup>56</sup> Crashlytics tool: <https://firebase.google.com/products/crashlytics/>

<sup>57</sup> Google Analytics tool: <https://analytics.google.com/analytics/web/provision/?authuser=0#/provision>

## App Store



The App Store is another digital distribution platform but is owned by Apple Inc. In this market you can buy or download more than 2.1 million Apps, also organized by categories such as “Health and Fitness”.

All apps that appear in the App Store are pre-reviewed by an Apple expert, but the requirements are quite similar to those listed for publication in the Google Play Store. However, the following particularities must be taken into account:

1. So that Apple experts can review the App, the **Apple developer program**<sup>58</sup> linked to a **profile on iTunes Connect**<sup>59</sup> should be used and all relevant information regarding the App should be provided in order to give a clear idea about its functionality.
2. A Mac computer and the Xcode<sup>60</sup> application must be used to generate the binary code for publication
3. Each App must have signed Apple certificates (development and distribution) generated by the developer account or Xcode.
4. Screenshots that are incorporated into iTunes Connect must **match exactly those of the final product**, otherwise the security validation will not be passed.
5. **No reference or logo from Google** or other platforms such as Windows or Amazon may be included in the application being published.
6. “In House” applications (internal distribution) cannot be deployed in this store. These types of applications must be deployed in Apple’s Enterprise program.<sup>61</sup>
7. Use tools like Crashlytics or Google Analytics to track app usage.
8. The validation process can take a few days and if publication is rejected, the app stays registered in iTunes with its current status and errors,

---

<sup>58</sup> Apple Developer program: <https://developer.apple.com/programs/>

<sup>59</sup> iTunes connect login: <https://itunesconnect.apple.com/login>

<sup>60</sup> Xcode application: <https://apps.apple.com/es/app/xcode/id497799835?mt=12>

<sup>61</sup> Apple Enterprise Program: <https://developer.apple.com/programs/enterprise/>

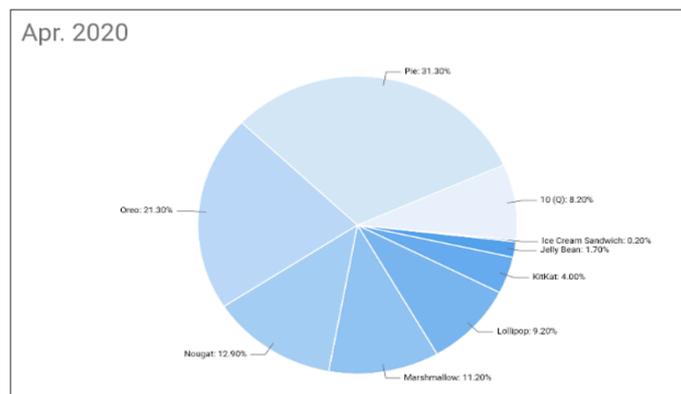
Apple makes the **TestFlight**<sup>62</sup> tool available to developers to beta test their apps. This App allows you to publish the trial version of an App, and **invite up to 10,000 users** to download and test it, so you can obtain their feedback. The tool also allows for internal validation by the development team, with a limitation of up to 30 devices.

The link below details the recommendations and best practices specified on the App Store website, relating to aspects such as reviewing applications, the user interface or marketing: <https://developer.apple.com/app-store/guidelines/>

### Tips for planning and design

The best form of planning should **take into account the design of the user interface** from the beginning of the project, as leaving it for later may mean adapting the functionality subsequently. For this reason, the developers and the design team must be in constant coordination and work with a joint vision from the start.

Another important factor to consider before starting the development of the App is **choosing the version (API) in which it will be programmed**. Choosing the latest version could rule out a large number of users using lower versions due to incompatibility issues. However, choosing a much lower version will prevent the newer features from being used, which may cause the App to become outdated. In the case of Android, you should study which version to use in order to reach as many users as possible, but ensuring a version is used that is new enough that allows the use of the latest features. For iOS, it is enough to program using a version that is two versions prior to the latest one because, as shown in the graph below, this will cover practically the whole market <sup>63</sup>



**Figure 1.** Distribution of Android versions as of April 2020<sup>64</sup>

<sup>62</sup> App Store Testflight Tool <https://developer.apple.com/testflight/>

<sup>63</sup> This is because Apple releases updates for all devices, so everyone has the latest version regardless of the version of the device.

<sup>64</sup> Source: <https://androiddistribution.io/>

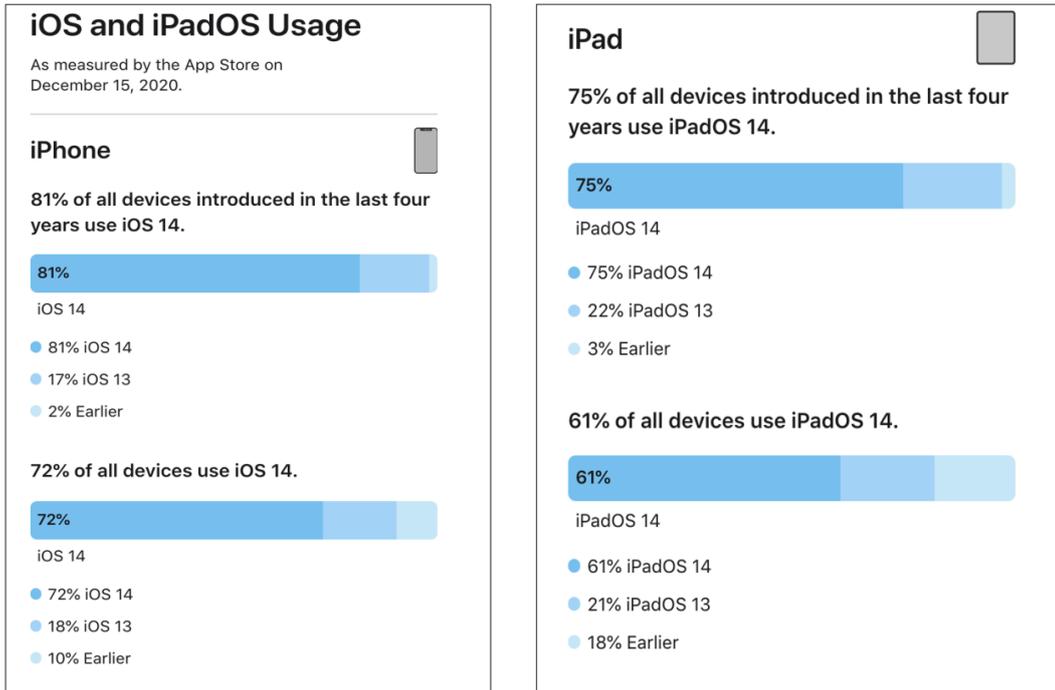


Figure 2. Distribution of iOS versions as of December 2020<sup>65</sup>

## Tips during development

The **official Android and iOS developer guides** should be used to discover the features of each operating system and resolve any programming questions that may arise during development. It may also be useful to join a **community of developers**, such as Stack Overflow<sup>66</sup>, where you can share questions with other programmers and find answers those that have already been resolved. The following aspects should also be incorporated into the development model:

- Use of a version control system (for example based on GIT).
- Automated generation of products in a single step using continuous integration systems.
- Use a database of incidents or bugs.
- Perform tests (including usability). This involves having people trained in this field.
- Use software quality analysis tools, such as **Android Lint**<sup>67</sup> or **SonarQube**<sup>68</sup>, to identify problems or vulnerabilities.

<sup>65</sup> Distribution iOS as of December 2020. <https://developer.apple.com/support/app-store/>

<sup>66</sup> Stack Overflow Developer Community: <https://stackoverflow.com/>

<sup>67</sup> Android Lint software quality analysis tool. <https://developer.android.com/studio/write/lint>

<sup>68</sup> SonarQube software quality analysis tool. <https://www.sonarqube.org/>

## Consells per a la publicació

An App may contain **bugs that may go unnoticed**, since some errors only occur on certain devices or on specific versions of them. Because it is very difficult to test the App on all devices and versions on the market, an initial release should be created as a pilot, **restricted to a particular region**. For example, if the goal is to publish an App internationally, it should only be published domestically for a few months, to ensure that the application does not contain errors. This will prevent the App from receiving negative ratings that could lower its market positioning and lose users. Once all errors have been corrected, the App can be published in a larger region so that end users can rate the App bug-free.

Also, it is worth **checking that the translations of the App are correct** in all languages, as a bad translation could provide incorrect information and lead to negative comments that would also lower the positioning of the App.

## How to make the App more visible

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

1. The App **icon** is the first thing users see. It should **stand out**, be **easy on the eye**, and be **relevant** to the search.
2. The **images** displayed on the market must represent the **main functionality** of the App, **be comprehensible** in the language of the user and **be sorted** by priority.
3. A **video can be used to show the key features** of the App to convince the user that their needs will be met.
4. The **right keywords need to be chosen** for the App so it appears in the top positions of searches.
  - a. **Investigate the competition** and ensure that the **description highlights features that differentiate the App** from the rest.
  - b. The App Store allows you to add keywords that can help to find the App. It is important to **choose the most appropriate ones**.
5. If the App is published in multiple regions, make sure it is **properly translated into the main languages** of each location.
6. **Reviews and ratings are important**. If they include the App's keywords and are positive, they will increase the position of the App in market searches.
7. **Positioning tools** can also be used to help you gain downloads:
  - a) **Google Play Developer Console** is a free tool provided by Google that allows you to easily and intuitively define and optimize all the elements of the App file, as well as view statistics.

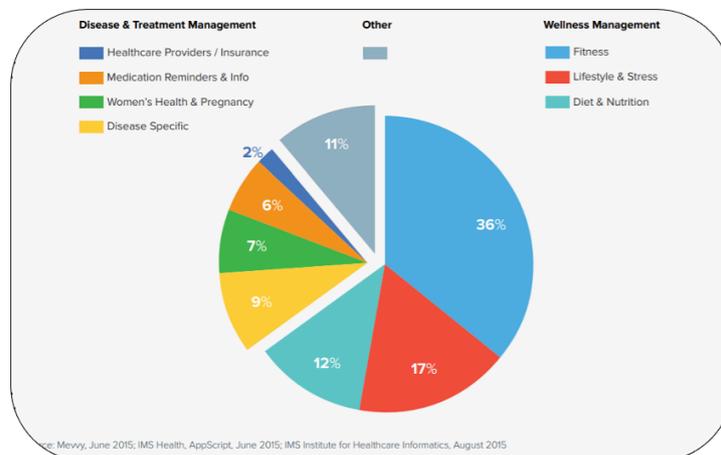
b) **App Annie** has free and paid features. It allows you to track app rankings by country and estimate your App's revenue.

c) **The Tool** is a paid tool that allows you to monitor keyword rankings, user ratings and manage advertising.

d) **Sensor Tower** is a payment tool that helps to increase the positioning of the application in searches, thanks to the study of keywords.

### Health Apps classification

The classifying of health applications is not fully defined and limited in all existing application stores, and sometimes a different classification of applications is made in each of them. According to research consultancy Research2Guidance, on Google Play, most health applications are classified in the "Health & Fitness" section and in the "Medical" section. Meanwhile, IMS Institute for Healthcare Informatics states that most health-related applications (around 53%) focus on overall well-being (lifestyles, stress, diet, exercise, etc). The rest focus mostly on specific health conditions (9%) such as mental health and diabetes, information on prescribed medication and reminders (6%), and female health and pregnancy (7%).



**Figure 3.** Distribution of mHealth Apps by category

Font: IMS Institute for Healthcare Informatics. Patient adoption of mHealth: use, evidence and remaining barriers to mainstream acceptance. 2015. <https://www.iqvia.com/-/media/iqvia/pdfs/institute-reports/patient-adoption-of-mhealth.pdf>

Part of the success of an App lies in appearing in the Top 25 and Top 100 in the App Store or Google Play store, both in the overall ranking and in the individual categories. And this substantially influences the number of installations of an App. It will not always be possible to appear as a featured App in the overall ranking, so it is very important to choose well the category to which it belongs so that it appears as featured in the category selection.

Leaving aside the success of the App, the most important thing about App Store Optimization (ASO) is to ensure that people who are searching for an App can easily find it. In this regard, categories and keywords will help to define the main features of the application and therefore to position it in the right place.

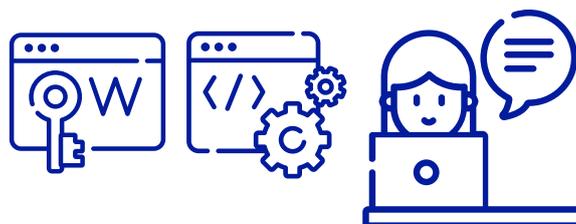
## Categories

There are **26 categories of applications** in the major app stores. This diversity can lead to hesitation when choosing the most appropriate one. The most important categories in the field of health in the main mobile app stores are:

- **Medicine:** This category includes applications focused on education for health professionals, clinical information management, and decision-making aids for both patients and health professionals. For example: Diagnostic guidance apps, anatomy atlases, HIS registration, chronic disease control, symptom management, control and registration of external devices (sphygmomanometers, ECG, pulse oximeters, etc.), remote health monitoring.
- **Health and Fitness:** This category includes applications related to healthy lifestyles, including stress management, fitness and recreational activities. For example: Apps for information on food, yoga, meditation, remote training, running, cycling, stress management, pregnancy, weight loss, Pilates, acupuncture, menstrual control.

## Keywords

Once the category in which the App will be included has been decided, it is important to define some keywords that allow it to be found during a general search in the market. It is important to remember that users search for apps based on the **benefits they expect from them** and not so much based on their features.



For example: Some applications use texts such as “unlimited diagnostics”, “more than 1,000 food products!” or “Monthly charts”. These messages do not answer the question “**What can it offer me?**” but are simply list the features. Instead of features, it is best to list the **functionality or benefits it offers**: “Diabetes Monitoring”, “Menstruation tracking”, “Online diagnoses” or “Healthy eating advice”.

Once it has been decided which words a potential user would use to describe the app, it is important to make sure they are visible in the description text.

## 5. Aspects related to the applicable regulations and quality

### 5.1 The mobile app as a medical device

The growing use of health apps has led to a growing debate about the point at which they should be considered medical devices. When a health app is considered a medical device, it is subject to strict security controls equivalent to those carried out in any other solution that affects the health of the public. Therefore, identifying in the design phase whether or not the App being developed is a medical device is essential because if it is, it will need to comply with the applicable legislation which focuses on the safety of the patient.

At the legislative level, the rules governing medical devices in Europe date back to the 1990s; 1990 through the Active Implantable Medical Device Directive and 1993 through the Medical Device Directive. As everybody knows, the pace of scientific and technological evolution in the health sector has not been matched by these standards. Both regulations to be updated to adapt the demands of the sector to the important **advances that have been made at the scientific and technological level.**

The three directives on which the European regulatory framework for medical devices has been based since 1990 are:

- Directive 90/385/EEC on active implantable medical devices (AIMD).
- Directive 93/42/EEC on medical devices (MDD).
- Directive 98/79/EC on in vitro diagnostic medical devices (IVDD).

As mentioned, the sector is constantly evolving and these directives are being replaced, harmonized and updated by:

- Regulation (EU) 2017/745 on medical devices (MDR)
- Regulation (EU) 2017/746 on in vitro diagnostic medical devices (IVDR)

On 26 May 2020, Regulation (EU) 2017/745 was scheduled to begin its mandatory application to the following medical devices:

- Class I products under the MDD, which will continue to be Class I under the MDR and therefore do not need to involve a notified body,
- Class III custom-made implantable devices,
- Devices using human tissue derivatives,
- Combination products (those which are part medicinal product and part medical device) under Article 117,

As an exception, 6 months from the date of adoption of the common specification published by the Commission instead of 26 May 2020, for:

- Annex XVI devices for non-medical purposes, such as lasers, liposuction equipment, etc.

On 24 April 2020, Regulation (EU) 2020/561 of the European Parliament and of the Council of 23 April 2020 amending Regulation (EU) 2017/745 on medical devices, as regards the dates of application of certain of its provisions, was published. Regulation (EU) 2017/745 on medical devices became applicable from **26 May 2021**.

This postponement was to ensure the permanent availability of medical devices in the European Union market, including those that are of vital importance in the context of the COVID-19 outbreak and the resulting public health crisis.

The main goal of this major change is to improve **patient safety** and public health, as the new regulations are a major qualitative and quantitative leap forward. With its entry into force, more thorough evaluations will be established for introducing a medical device into the market, and those products that are currently self-certified may now, under this Regulation, need a notified body certify them. Below is a list of other changes introduced by the regulations:

- More clinical data and more research are required.
- Higher post-marketing monitoring requirements
- Creation of **EUDAMED (European database on medical devices)** which aims to centralize the monitoring of all medical devices, from their marketing to surveillance, as well as clinical studies.
- UDI (Unique Device Identification) requirement, that is, each medical device must have a **unique identification number**.
- Implant card, which is delivered to the patient to provide them with information on the implant manufacturer and specific traceability data.
- The manufacturing organizations and authorized representatives must now have a person responsible for compliance with the legislation.
- The regulation also affects cosmetics devices, such as lasers or liposuction equipment.
- Notified bodies must go through a recertification process and will be subject to strict monitoring as well as unannounced audits.

The regulations also target all agents involved in the launch on the market of a medical device or service (manufacturer, authorized representative, importer or distributor). They too are guarantors of compliance with regulations, traceability and surveillance.

This section of the guide provides a number of key points that can help determine whether or not an App is a Medical Device.

## Definition of a medical device

Regulation (EU) 2017/745 defines a medical device as:

**Medical device:** Any instrument, apparatus, appliance, **software**, implant, reagent, material or other article intended by the manufacturer to be used, alone or in combination, for human beings for one or more of the following specific medical purposes:

- Diagnosis, prevention, monitoring, prediction, prognosis, treatment or alleviation of disease,
- Diagnosis, monitoring, treatment, alleviation of, or compensation for, an injury or disability,
- Investigation, replacement or modification of the anatomy or of a physiological or pathological process or state,
- Providing information by means of *in vitro* examination of specimens derived from the human body, including organ, blood and tissue donations, and which does not achieve its principal intended action by pharmacological, immunological or metabolic means, in or on the human body, but which may be assisted in its function by such means.

The following products shall also be deemed to be medical devices: - devices for the control or support of conception, - products specifically intended for the cleaning, disinfection or sterilisation of devices as referred to in Article 1(4) and of those referred to in the first paragraph of this point.



In addition, Regulation (EU) 2017/746 defines an *in vitro* diagnostic medical device as:

***in vitro* diagnostic medical device:** Any medical device which is a reagent, reagent product, calibrator, control material, kit, instrument, apparatus, piece of equipment, **software or system**, whether used alone or in combination, intended by the manufacturer to be used *in vitro* for the examination of specimens, including blood and tissue donations, derived from the human body, solely or principally for the purpose of providing information on one or more

of the following: (a) concerning a physiological or pathological process or state; b) concerning congenital physical or mental impairments; c) concerning the predisposition to a medical condition or a disease; d) to determine the safety and compatibility with potential recipients; e) to predict treatment response or reactions; f) to define or monitoring therapeutic measures. Specimen receptacles shall also be deemed to be in vitro diagnostic medical devices.

Regulation (EU) 2017/746 on in vitro diagnostic medical devices is applicable from **26 May 2022**.

### How to identify if an App is a medical device

The main decision criteria and regulatory requirements for medical devices are set out below.

#### Decision criteria

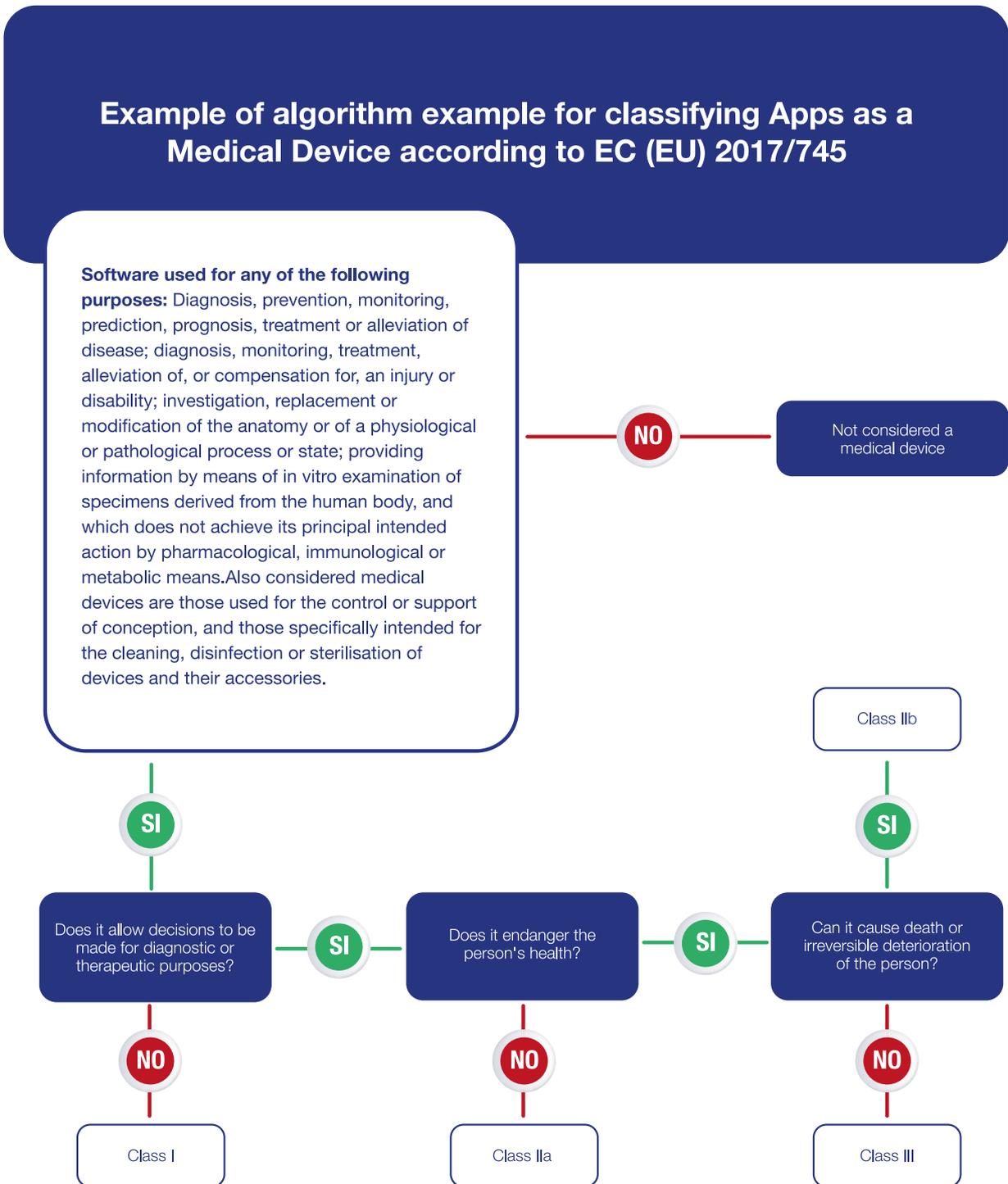
Mobile health apps that are considered medical devices are often called Mobile Medical Apps, while the rest are considered Wellness Apps. The purpose of the application (intended use) is crucial to determine whether or not it is a medical device: if it is to **diagnose, support the diagnosis or clinical decisions, make calculations to determine the diagnosis or treatment, or used for any medical purpose, the application could be considered a medical device**. For this reason, an accurate and complete definition of the intended purpose of the application is needed to help determine whether or not it is a medical device.

Mobile applications considered medical devices must comply with the definition of a medical device (according to Directives 93/42/EEC, 98/79/EC or according to new regulations EU 2017/745 or EU 2017/746). However, they may act as an accessory for a medical device or directly convert the mobile platform into a medical device Conversely, **if the application only stores, files, transmits, performs simple data searches, or presents data results without altering or manipulating the data, it is not considered a medical device**.

Below is a set of questions which, along with the criteria outlined above, help to determine whether or not the **application is a medical device**:

- Is the application intended to interpret (or facilitate the interpretation of) data by modifying or representing individual health-related information?
- Does the application interpret or alter data? Or does it just manage static information?
- Does the app calculate doses for ingestion/injection?
- Does the application indicate a medical condition, illness, or individual percentage of risk of suffering an illness?

As an example, the following algorithm has been designed which may be interpreted and the classification of which should always be carried out by a competent body.



In any case, the same App may or may not be a medical device depending on the intended purpose. For example: an App that monitors the heart rate which is intended exclusively for sports use will not be a medical device; on the other hand, the same application intended for medical use may be considered a medical device.

Other examples of medical devices are Apps that transform wearables data into graphs or electrocardiograms, since they process the data obtained from the user, or calculators that take into account the clinical factors of the user.

Not falling under the definition of medical devices are those Apps whose intended purpose is to improve communication between patients and caregivers, which allow storing and searching patient data or which allow the display of information, conducting research, and if the purpose is not for the health benefit of patients.

The document MDCG 2019-11 (Guidance on Qualification and Classification of Software in Regulation (EU) 2017/745 – MDR and Regulation (EU) 2017/746 – IVDR), contains a decision diagram that provides guidance on the necessary steps to follow to decide whether or not a particular software can be considered a medical device. This guide is directly adapted to the requirements of the MDR and IVDR regulations. As the same guide indicates, Medical device software is: Software that is intended to be used, alone or in combination, for a purpose as specified in the definition of a “medical device” in the MDR or IVDR, regardless of whether the software is independent or driving or influencing the use of a device.

### Regulatory requirements for medical devices

Once it has been concluded that the App is a medical device, it is necessary to determine to **which class it belongs** among those indicated in Annex VIII of the regulations (MDR or IVDR as appropriate). This classification is based on the potential risk that the device represents for the user, and will determine the steps to be followed to assess its conformity. Medical devices can be classified as follows:

- Class I low risk
- Class IIa moderate risk
- Class IIb serious risk
- Class III high risk

It is important to note that this classification does not apply to *in vitro* diagnostic medical devices. Indeed Annex II of Directive 98/79/EC defines a list of devices that require a specific conformity assessment route. The above-mentioned Regulation (EU 2017/746) introduces changes at this point, establishing a complete classification which includes *in vitro* diagnostic medical devices.

- Class A low risk
- Class B moderate risk
- Class C serious risk
- Class D high risk

All medical devices must obtain the **CE marking** before they can be sold. This implies that the manufacturer (or authorized representative in the event that the manufacturer is outside the European Union) ensures and declares that the product complies with the relevant current legislation. Once compliance has been certified, the medical device must be labelled with the CE marking.

Broadly speaking, in order to sell an App considered a medical device in Europe, the manufacturer must:

- Draft a technical dossier covering all the requirements of the relevant Directive.
- Have a quality management system in place.
- In accordance with article 100 of General Health Law 14/1986 of 25 April, natural or legal persons engaged in the manufacture, import, grouping or sterilization of medical devices, and the facilities in which these activities are carried out, require an operating license, granted by the **I'AEMPS** (Spanish Agency of Medicines and Medical Devices).

The key aspects to consider when talking about a mobile application as a medical device are:

- Technical documentation standards
- Clinical evidence/evaluation standards
- Possible need for intervention, depending on the classification of the product, by a body (notified body) declaring conformity of the product in addition to the manufacturer's declaration
- Standard on product classification
- High level of product traceability both during development and during the different stages of manufacture (product life cycle)
- Rigorous control of product changes (software versions)
- Handling of incidents or complaints
- After-sale surveillance and monitoring

It should also be noted that the aforementioned new Regulations introduce additional requirements, such as increased transparency through the creation of EUDAMED (European Database on Medical Devices).

In addition, the legislative framework on medical devices has been revised and 2 new regulations will be progressively applied during 2021 and 2022. The new standards contain specific requirements for medical device software, applications, cybersecurity, and clinical and performance evaluation. The MDCG 2019-11 guidelines, a report entitled *Guidance on Qualification and Classification of Software in Regulation (UE) 2017/745 - MDR and Regulation (UE) 2017/746 - IVDR* <sup>69</sup>. The criteria specified in this document apply to Apps running on mobile phones, in the cloud, or on other platforms.<sup>69</sup>

---

<sup>69</sup> Ec.europa.eu. 2021. Guidance on Qualification and Classification of Software in Regulation (UE) 2017/745 - MDR and Regulation (UE) 2017/746 - IVDR <sup>69</sup>. Available from: [https://ec.europa.eu/health/sites/health/files/md\\_topics-interest/docs/md\\_mdcg\\_2019\\_11\\_guidance\\_en.pdf](https://ec.europa.eu/health/sites/health/files/md_topics-interest/docs/md_mdcg_2019_11_guidance_en.pdf)

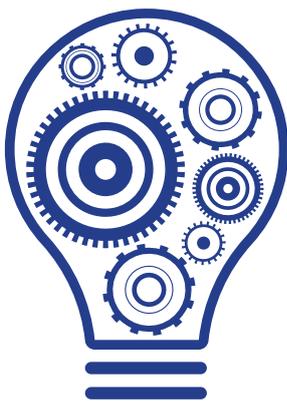
## 5.2 TIC Salut Social Foundation quality seal for mobile applications (TICSS seal)

The TIC Salut Social Foundation has a certification model for mobile applications in the health sector that allows applications to be certified as suitable for facilitating and improving the monitoring of the health of citizens, offering a guarantee of quality and safety.



Applications are subject to a set of criteria<sup>70</sup> in the public domain which can be consulted on the Foundation's website. By promoting the safe use of quality applications and publicizing those with high added value, it contributes to improving the empowerment of citizens, making them aware and co-responsible for their own health.

The design of the certification model was based on a cross-cutting consensus among different professionals and agents in the sector: experts in the field of technology, health professionals (doctors, nurses, psychologists and social workers), health communication professionals, patients experts and institutional and citizen representatives. This diversity has contributed to creating a model that encompasses every perspective from different areas, obtaining a set of criteria that can be grouped into **four areas of expertise**: *usability and user experience*; *technology and reliability of the application*; *data security and privacy*; and *functionality and evaluation of content*. These areas of expertise are assessed separately and are independent of each of other and adaptable to the different demands of the sector.



### Design, usability

Accessibility, user experience and visual aesthetics.



### Content and functionality

Content quality and usefulness of the features included in the App.



### Technological requirements

Technological reliability and adaptability.



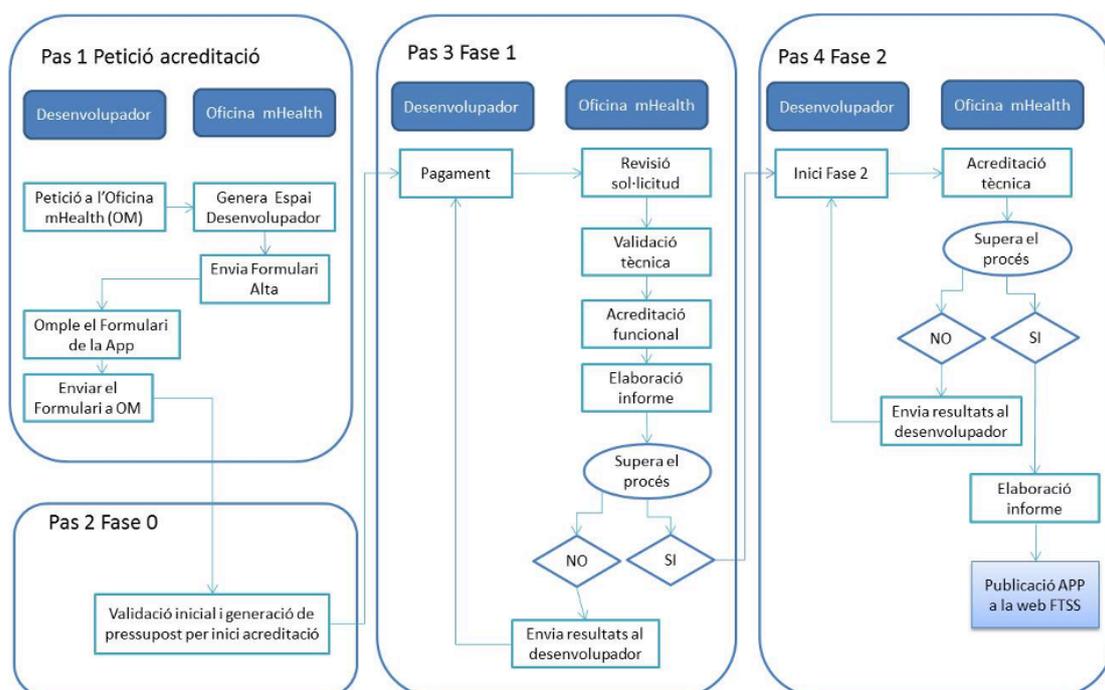
### Privacy and data security

Data management and processing policies.

<sup>70</sup> <https://ticsalutsocial.cat/wp-content/uploads/2018/08/criteris-acreditacio.pdf>

The Official College of Nurses of Barcelona (COIB), the Official College of Physicians of Barcelona (COMB), the Official College of Graduates in Physical Education and Physical Activity and Sports Sciences of Catalonia (COPLEFC), the Association of Family and Community Nursing (AIFiCC), the Catalan Society of Specialists in Clinical Psychology (SCEPC), the Catalan Society of Family and Community Medicine (CAMFIC), are some of the entities that collaborate in the evaluation of these areas.

To certify an App it must pass through the different phases shown in the figure below:



To begin the certification process, the owners/developers of the application must send an email to [oficinamobilitat@ticsalutsocial.cat](mailto:oficinamobilitat@ticsalutsocial.cat) requesting the certification of their App.

Once the mHealth Office receives the application, it will send an email with the New App form to be filled in by the applicant company with the technical and functional data of the App to be certified. These forms must be returned to the mHealth Office duly completed.

As soon as the mHealth Office receives the completed form, it will prepare the budget for the certification of the application. To draw up the budget, the classification level of the App will be reviewed as well as the data filled in on the submitted form.

The form will allow the mHealth Office to determine the minimum certification criteria that must be met, and will enable the process to be started.

Submitting the App to the TIC Salut Social Foundation certification process means that it must pass the following phases:

- **Phase 0: Review of the request:**
  - Receipt of the request:
  - Verification of the data provided in the form
  - Economic budget of the certification process
- **Phase 1: Initial validation and functional certification:**
  - Initial technical validation where a first test of the application is performed.
  - Functional certification of the application by the Committee of Functional Experts of the mHealth Office (entities such as [COMB](#), [COPLEFC](#), [COIB](#), [SCEPC](#), [AiFICC](#) and [CAMFiC](#)) which reviews the App and evaluates the content criteria.
- **Phase 2: Technical accreditation:**
  - Specifically in this phase, the usability block, the technology block, and the security block are reviewed.

Once the three phases have been completed, the owners/developers will be informed of the result of the evaluation through a report of results for each criterion.

The certification process is complete once the mHealth Office has evaluated the three phases and their corresponding blocks. If the App does not pass all of the phases, the developers will be informed through a report and the steps necessary pass the process will be indicated. Applications that pass the certification process receive a trust seal and are published on the Foundation's website, to give them visibility and make them available to the sector for consultation and use.

The TIC Salut Social Foundation is committed to a certification process that is open and transparent to all types of initiatives with the aim of generating trust among the public and health professionals.

All the documentation related to the certification process, as well as the criteria can be consulted in the mHealth section<sup>71</sup> of the TIC Salut Social Foundation.

---

<sup>71</sup> <https://ticsalutsocial.cat/recursos/mhealth/manuals-acreditacio-apps/>

## 6. Impact of 5G technology on Apps

5G technology involves significant technological improvements that offer a wide range of applications in the healthcare environment, both for improving existing services (e.g. telecare) and for making possible certain scenarios that were impossible until recently (e.g. video-assisted surgery).

### What is 5G?

5G is an improvement to the current mobile communications system known as 4G. This improvement can be seen in two areas: in the provision of the services that already exist (higher speed, availability, etc.) and in enabling things that could not be done with the current technology (tactile communication –remote experiences with a delay so short that they are comparable to a physical interaction–, increased number of devices, network segmentation, etc.). Mobile networks began as an extension of fixed telephony networks and have been evolving into a network that will provide all kinds of connectivity, in which fixed communications will be the extension. 5G is revolutionary in that it is the first step towards this new concept of networks. Achieving this goal requires more capacity (radio spectrum, base stations, etc.) and improvement in backbone networks to make them more flexible and secure.

1G	These were the first mobile voices which could only be used for talking.
2G	Text messaging was then introduced, which enabled mobile phones to be more complete communication tools.
3G	The technology evolved to allow mobile web browsing, which provided an internet connection from anywhere. Since then, one of the main concerns of operators has been to improve network connectivity, serve more users and have more devices connected.
4G	This offers a higher speed of data, and enables video consumption via mobile.
5G	Evolution in improving the speed of data, creating a network similar to the fibre optic network but without the need for it to be installed physically, but rather through the mobile networks.

**Table.** Examples of how mobile technology has evolved over time

### Main features of 5G

The descriptions below cover the most relevant aspects of this technology, such as the radio spectrum, 5G coverage and the different areas of application that 5G addresses.

## Radio spectrum

In terms of the radio spectrum, 5G could work on any frequency band that is already used today. Nevertheless, new and wider bands are preferable. To date, operators have begun to use the 3.5GHz band where each operator, of the four that exist (in Catalonia), can use 80 or more MHz. The 700 MHz band is expected to be tendered soon, which may offer better coverage. This second band is intended to be used to provide access in rural areas or for Internet of Things (IoT) devices. There is another band, called the millimetre wave band, which is located at 26 GHz. This may provide very high speeds, but with more complex coverage. Therefore, a larger number of base stations will be required.

It is possible that in the future, some of the bands that now have 2G or 3G systems can be reused by 5G systems, which are more efficient and therefore allow better use of the spectrum. Combining these different frequency bands means that **5G will enable better coverage across the territory (rural and urban) and in all environments (outdoor and indoor). In addition, it will allow operators to offer more capacity (greater number of users).** It will also help reduce the levels of exposure to electromagnetic radiation, since it is transmitted for less time (by increasing the speed) and with less power (since the base stations will be closer).

## Scope and capabilities

5G offers three basic capabilities:

- Enhanced Mobile Broadband (**eMBB**) enables high-speed communications.
- Ultra-Reliable and Low Latency Communications (**URLLC**) gives reliability (99.999% better) and low delays (in the order of milliseconds).
- Massive Machine Type Communication (**mMTC**) will allow device densities in excess of one million per square kilometre.

These capabilities can be combined to meet any connection requirement. For example, they will enable:

- Connectivity anywhere guaranteeing the ability to send data at all times regardless of location, both static and when moving.
- A reduction in delays in interaction and allow, for example, remote surgery
- Connecting millions of devices. Ensuring that a device for monitoring a person with a disease can work in a hospital, healthcare or primary care centre and that, when the patient goes home, they will not experience any problems due to network congestion and will not be disconnected.
- Stream quality or very high quality video. This can be used for remote interventions, but also for observing the person being cared for during a remote visit.
- Reliability in communications, guaranteeing the delivery of data to the other end as if it were a wired communication. This will facilitate, for example, remote monitoring or control of remote devices, such as continuous infusion pumps or oxygen therapy.

## Example of a 5G application

An example of the application of this technology is in the field of what is known as the **connected vehicle**. Connecting a vehicle makes a lot of sense since it allows the driver to download information about maps or traffic. It also vehicle maintenance data to be sent or automatic alerts if an accident has occurred. The use of 5G offers a wider range of possibilities. For example, vehicles will be able to exchange information with each other directly to help prevent collisions. Also, warnings can be sent that an emergency vehicle is approaching or even traffic lights can be controlled to prioritize the passage of an ambulance and improve traffic time when covering an accident or transporting a seriously ill person.

4G and 5G mobile technologies enable the same functionalities to be offered, but this fifth generation substantially improves the connectivity experience.

The main differences between 5G and 4G technology are listed below:

- Data transmission speed is higher with 5G. 5G allows downloads that are up to 20 times faster than with 4G.
- The latency of 5G is 1 millisecond, compared to the average of 4G which is 40 milliseconds.
- The 5G network supports the connection of more than 1,000,000 devices per kilometre. Meanwhile 4G allows the simultaneous connection of 10,000 devices.<sup>72</sup>

## New functionalities compared to 4G

5G also contributes modifications to the network that allow better functionality. Specifically, the following three features stand out: 1) virtualization and network slicing<sup>73</sup>, 2) positioning services and 3) security.

### Virtualization and network slicing

With virtualization, the functions that were previously performed on a specific computer can be performed on a general-purpose computer. This offers a lot of flexibility allowing you to replicate an element and run it in different places. As a result, it is called slicing. It involves different networks (slices) on a single physical network. The resulting slices can be configured differently and can be dedicated to different user groups. A slice can be created, for example, for the Health System and guarantee specific features independently of other users. This functionality improves security and ensures services are maintained in an emergency.

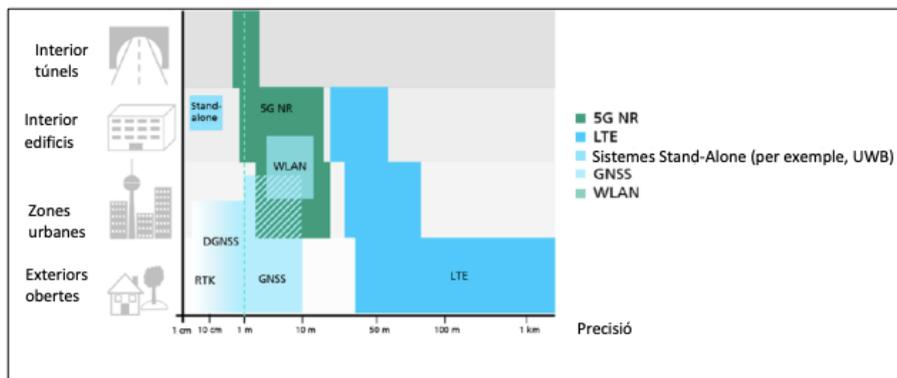
---

<sup>72</sup> <https://economyatic.com/tecnologia-5g/>

<sup>73</sup> Network Slicing: A form of virtual network architecture that enables the creation of multiple virtual networks on a common shared physical infrastructure. Virtual networks are customized to meet the specific needs of applications, services, devices, customers, and operators. <https://centralizate.es/telefonía-ip/que-es-el-network-slicing/>

## Positioning services

Another improvement offered by the 5G network is the ability to offer positioning services. Currently the most widespread approach is the use of satellite systems (GNSS) that only work outdoors. Cellular systems, and in particular 5G, can offer improvements to positioning indoors as well as outdoors. Efficient resource management in hospitals or accessing an injured person who has called 112 requires a positioning system that works in all environments, both outdoors and indoors. The following figure shows the expected accuracy of 5G systems for different locations.



**Figure.** Accuracy of positioning technologies according to the environment (Source: Fraunhofer IIS)

## Security

Cellular mobile networks, unlike Wi-Fi or Bluetooth, have proven to be secure. 5G offers security improvements that already existed in 4G. These include authentication, encryption, integrity and privacy. These forms of protection are essential for communications in the field of health.

## Availability

Not all of the functionality of 5G is available today because standards are evolving, manufacturers need time to produce network equipment and terminals, and operators need to upgrade their networks. To date, there are 33 5G networks in operation in Europe and 109 worldwide. This technology is being introduced in the mobile phone market, where the best-known manufacturers have released or will soon release models that incorporate 5G.

5G technology offers functionality and enhancements that could be very useful in the field of health. For example, as part of the 5GBarcelona initiative, various pilots and demonstrations of use cases have been carried out (such as the 5G connected ambulance or the remote surgeon) that demonstrate the possibilities of 5G in a healthcare environment. This technology is in an emerging state and will evolve over the next few years.

