**Mobile App Requirements: How to Gather & analyse iOS, Android, or Native Requirements**

## What Are Mobile App Requirements?

Mobile app requirements document the business logic, technical specifications, and development guidelines for mobile app developers to design the application of your business dreams.

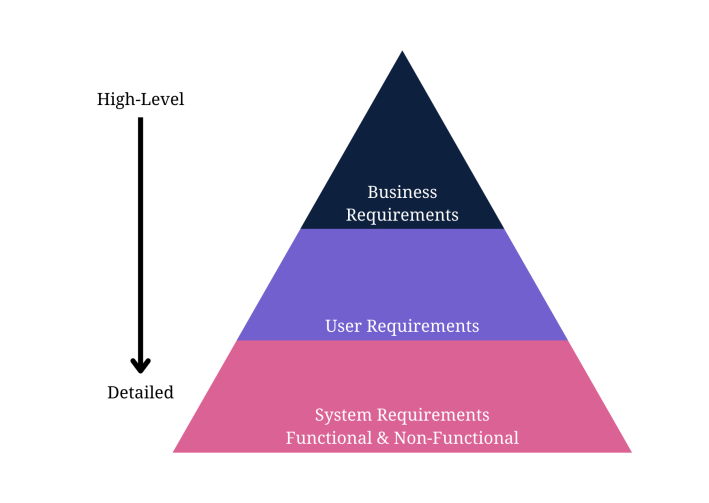
It includes the key app’s features, app user personas, and business goals to ensure that multiple team members are on the same page before the software development process commences.

## How to Gather an App Requirements for Project Success

### Mobile App Requirements Types

Mobile apps have different requirement types to collect, including:

* **Business requirements** are high-level requirements that ensure the app will align with business objectives, and the project’s scope, and identify the key stakeholders.
* **User requirements** are valuable insights into what your target audience needs and wants, how you can solve their problems, and what the audience experiences from your prototyping app.
* **Product or system requirements** are non-functional requirements and functional requirements that include technical requirements and technical specifications for the engineering team.



A functional specification document will outline the product’s features and how users interact with the app. On the other hand, non-functional requirements include the technical specification for the app’s quality attributes.For example, a functional requirement refers to the sign-up button users click on before using your services.However, non-functional requirements refer to how fast the system responds to the user clicking on that button and how the system protects their data while clicking the button.

Next, let’s show the steps to gather requirements for a mobile application to ensure you meet the business requirements and key features necessary to develop a profitable product.Some steps will involve sub-steps to help you gather the right information and requirements for documents.

### Requirements Gathering Process

### Step 1: Define Your App Idea and Purpose

Mobile development requirements-gathering starts with a business idea. The first information you need is the idea or purpose of the mobile app.

What purpose will it serve? Does it offer a solution to a potential problem?

You need to identify a problem the app will solve to recognise the idea or purpose behind it.

[Requirements gathering and management for mobile apps](https://www.requiment.com/requirements-gathering-and-management-for-mobile-apps/) require some effort with massive results.

### Step 2: Gather and Align the App and Business Objectives or Goals

An app idea is fruitless without understanding business needs, business goals, and business rules.

This step encourages to gather business requirements to understand how the enterprise aligns with the idea from the first step.

Gathering business requirements to document involves these steps:

1. Identify the stakeholders for the right mobile application software development based on the business idea.
2. Define clear and concise business goals and objectives to understand the project’s scope.
3. Elicit stakeholder requirements and user requirements with elicitation techniques.
4. Document the requirements in a business requirements document.
5. Validate your requirements with stakeholders for a further transparent and opportunistic process.

So, what requirements-gathering elicitation techniques could be used to gather stakeholder requirements for the business requirements document?

Elicitation techniques work for any requirements-gathering type.

Here are some popular and successful techniques to use:

* Analyse similar documents
* Analyse similar external and internal interfaces
* Brainstorm use cases and user stories
* Create user stories and use cases
* Hold stakeholder focus groups
* Host requirements workshops
* Interview all the relevant stakeholders
* Observe documents and case studies
* Prototype visual examples for feedback
* Reverse engineer the processes
* Use online surveys/questionnaires
* Validate ideas with stakeholders

### Step 3: Run a Market Analysis and Competitor Analysis

Conduct a market or competitor analysis to truly understand the user’s perspective and design the appropriate user personas.

It also helps your team gather more user requirements for the development company.

The following steps explain the process of gathering user and competitor requirements:

1. Identify the direct, indirect, secondary, and substitute competitors for the mobile app. Remember to recognise any businesses offering similar mobile app services or products and those offering different products in a broader niche umbrella.
2. Gather competitor information, including products, descriptions, pricing structures, geographic reach, engaging promotions, target market positioning, business reputation, user profiles, and key partnerships to understand what your product needs to compete against.
3. Use a SWOT analysis to determine your competitor’s strengths, weaknesses, opportunities, and threats. We could learn from another app’s mistakes to improve our requirements and identify possibly unique features other apps don’t provide.

The SWOT analysis in a table will help to see what the new product needs to do to compete better with the top market competitors.

Rank each competitor from 1-10 on each key element.

Then, rank what your mobile app aims to have in the requirements, looking for opportunities to improve the numbers.

| **Attribute** | **Competitor 1** | **Competitor 2** | **Competitor 3** | **Your Product** | **Action Needed** |
| --- | --- | --- | --- | --- | --- |
| Quality | 7 | 6 | 5 | **9** | Maintain leadership |
| Pricing | 6 | 8 | 4 | 7 | Match Competitor 2's value |
| Place | 5 | 7 | 3 | 6 | Expand distribution channels |
| Promotion | 4 | **9** | 6 | 5 | Boost marketing campaigns |
| Unique Features | 5 | 6 | 2 | **7** | Highlight differentiators |
| Positioning | 6 | **8** | 5 | 7 | Refine brand messaging |
| People | 7 | 5 | 4 | **8** | Enhance customer support |
| Reputation | 6 | 7 | 5 | **8** | Leverage positive testimonials |
| Partnerships | 3 | **5** | 2 | 4 | Build strategic alliances |

Next, determine what competitive advantage we hold over other apps. For example, the sample analysis shows that your product quality is far superior to others. The pricing is also better than competitors. Choose at least three elements in which you wish to compete with other apps.

Fun fact: [Requirements gathering is the most undertaught area in software development](https://www.requiment.com/requirements-gathering-is-the-most-undertaught-area-in-software-development/).

### Step 4: Determine Scenarios and a User Persona

The next major step in mobile app requirements-gathering is to design user personas and scenarios to guide the requirements.

A user persona fictionalises the target users for the mobile app.

It should describe the ideal person who uses the app, with some flexible aspects for alternate users.

The ultimate user person could include the following details about target users:

* Age (also, typical generational qualities)
* Behavioural considerations
* Gender (including non-binary if relevant to the product)
* Geographic location
* Goal or problem the app addresses or solves
* Goal quotes or principles
* Goal-related frustrations
* Motivation to use the app
* Range of hobbies and daily activities
* Typical occupation range

How to transform user personas into scenarios?

Create a persona scenario or storyboard by focusing on the goals, how their typical behaviours affect them, and how the persona’s background motivates them to respond differently.

Write the scenario as a short paragraph for starters, as you’ll design user stories later. Meanwhile, [identifying stakeholders for requirements gathering](https://www.requiment.com/identifying-stakeholders-for-requirements-gathering/) means creating personas.

### Step 5: Gather and Prioritise Functional and Non-Functional Requirements

Your user and business requirements are shaped through the initial steps of app requirements gathering.

You still need to document them, but you’ll do that soon enough.

Meanwhile, start prioritising the functional and non-functional requirements for the technical details, which also design use cases.

[Functional and non-functional requirements](https://www.requiment.com/what-are-functional-and-non-functional-requirements/) differ.

First, determine which functional app requirements the project needs.

Here are some examples of functional mobile requirements:

* A complete description of a feature the app offers or software interfaces.
* How the app allows users to sign up, verify accounts, or subscribe to a newsletter.
* Buttons and dashboards users interact with to complete a specified task.
* External and internal interfaces users interact with on the app.
* The necessary administrative functions for different user classes.
* Transaction adjustment, correction, and cancellation functions.

Secondly, determine the non-functional requirements necessary to run your app.

Here are some examples of non-functional requirements in mobile development:

* How fast the app responds to user input.
* How the app protects user and business data.
* Whether the app can work on multiple platforms.
* How much data does the app store and is it scalable?
* How reliable and maintainable the app remains.
* Does the app comply with local laws and regulations?

Next, you’ll prioritise the non-functional requirements (NFRs) and functional requirements for an app.

Priorities determine the tech stack and importance of each function.

The MoSCow prioritisation technique helps with any requirements prioritisation before documenting the requirements.

The technique requires you to put every technical requirement into one of four categories:

* **Must Have** – The highest-level requirements are critical to the requirements document to ensure the project’s success.
* **Should Have** – The second highest-level specifications are necessary for the project but won’t delay the progress of development or success.
* **Could Have** – The medium-level specifications could enhance user experience but aren’t dealbreakers if you don’t develop them right away.
* **Won’t Have** – The low-level requirements aren’t important to stakeholders at the time of requirements documentation and won’t affect the development process.

Include requirements for user experience (UX) and user interface (UI) with your functional and non-functional requirements.

It helps to have these requirements in place before designing use cases and documenting a key user experience and user flow requirement for app development.

[Requirements prioritisation](https://www.requiment.com/requirements-prioritisation-making-informed-decisions/) simplifies decision-making for your app.

### Requirements Analysis Process

The next step is the requirement analysis, pun intended. Some techniques also help managers represent the requirements to stakeholders. The following requirements analysis techniques are commonly used for the actual analysis process:

#### Common Requirement Analysis Techniques

**Gap Analysis:**A gap analysis identifies performance gaps in a software application to determine whether the business requirements are met. The gap analysis depicts the difference between the present state and the target state of a software application.

**Gantt Charts:** Gantt charts outline the schedule of tasks and timelines in which developers must perform them. The Gantt charts show a project team the complete timeline with tasks.

**Integrated Definition of Function Modeling (IDEF):**The integrated definition of function modeling (IDEF) method in requirement analysis techniques represents process functions and the integrated set of relationships between parent and child systems.

**Role-Activity Diagrams (RAD):** This method offers a high-level view of diagrams that capture role structure and dynamics within an enterprise. Roles are used to group responsibilities and activities into units in project planning and systems engineering.

**Data Flow Diagrams:** A data flow diagram lets project managers represent complex processes that would otherwise be challenging to visualise from the text. Also, a data flow diagram shows analysts possible gaps.

**The Flowchart Technique:** A flowchart technique outlines the sequential flow of information and the control logic of an activity set. The flowchart technique uses different formats, including top-down, cross-functional, and linear graphs.

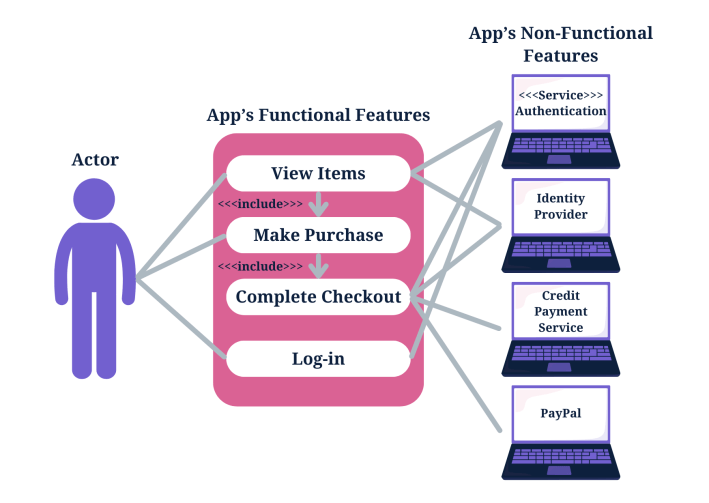
**Unified Modeling Language (UML):** The unified modeling language (UML) method uses an integrated set of diagrams to specify, construct, and document software system artifacts. Graphical notations also present the software project for systems engineering.

**Business Process Modeling Notation (BPMN):** The business process model technique uses process flowcharts with unique elements and symbols to depict the business process in graphs. Business process modeling is a process improvement method in the software industry.

#### ***CASE STUDY:*** *How to Design a Use Case for App Requirements Documents*

A use case diagram lets everyone at the development company visualise an overview of how users will interact with the app.

It’s an overview that includes actors, how actors interact with the app, and the sequence of interactions actors will deploy.

Here’s a simple example of a use-case diagram that shows an overview of how actors interact with functional features while non-functional features interact with the app from the back-end stack:

### Write an App Requirements Document

Delivering a proper app requirements document means you need to know how to write a mobile app requirements document.

Mobile application development relies on the requirements document to design proper flow or the best app features and hit the right target audience.

#### a: Formulate the App’s Idea Statement

Every app requirements document should include an idea statement that lets every stakeholder and software developer understand the document before diving into the details.

Start your app requirements document with a simple single-sentence statement that aligns with the app’s idea.

#### b: Document All Relevant App Details

A detailed description of development plans in the requirements document is instrumental to completing the documentation.

A successful mobile app requirements document includes more than the details of an application and its functions for the development team.

##### Descriptions to Include in a Mobile Requirements Document

A mobile application requirements document should include a detailed description of functional and technical requirements and the app’s functionality to properly capture and represent the project’s scope.

* Business requirements
* User requirements
* Software requirements specification
* Technical specifications
* Functional specifications
* Non-functional requirements
* Hardware interfaces
* A list of must-have features
* Unique app features
* Internal and external interfaces
* Non-functional key metrics
* User stories
* Acceptance criteria

#### c: Prepare a Navigation Sequence

The development team requires a simple navigation sequence they can follow during software development.

The mobile app requirements document outlines the sequence in which the software development process flows.

#### d: Add Requirements Formats for Visuals

Add your user stories from a user’s perspective and the use case overviews you designed to the app requirements document to help stakeholders and the development team understand every aspect of the app requirements document.

Successful software development means knowing the intended users.

#### e: Add Cost Optimisation Details

The development team, stakeholders, and the client will appreciate a cost-benefit analysis to ensure cost optimisation throughout the software development process.Business analysts also insist on adding a cost-benefit analysis to an app requirements document to meet the business needs and have a greater chance of success against competing apps.

#### f: Add Communication Protocols

Add communication methods for a collaborative process. Collaboration relies on dependable communication.

The [importance of effective communication in requirements gathering](https://www.requiment.com/the-importance-of-effective-communication-in-requirements-gathering/) outlines why you need it.

### Deploy Prototyping and Wireframing

Prototyping and wireframing let you design the user flow of user interfaces and basic app functions.

It also lets you test and validate layouts and transitions between app pages.

Here are the steps to wireframe your requirements for an app, which you will then validate in the next step:

1. Map the target user flow.
2. Sketch the flow’s core part.
3. Set a mobile wireframe.
4. Determine the layout with boxes.
5. Use design patterns.
6. Add intended copy.
7. Connect the app’s pages to design a flow.
8. Design a prototype.
9. Release the initial design to gather feedback.

### Validate the App Requirements

Validation is a quality control process you use before launching the final product based on your requirements.

The prototype app collects feedback from stakeholders, and you can invite stakeholders to verify that the app meets the documented requirements. Use the feedback for the final step.

The [benefits of collaborative requirements gathering](https://www.requiment.com/the-benefits-of-collaborative-requirements-gathering/) share insights about feedback and collaboration.

### Apply Agile Methodology

Agile methodology in requirements-gathering means you’ll always adapt the requirements document as per the feedback from stakeholders, testing, and initial product releases.

Agile methodology focuses on user experience and constant testing and validation to further improve your application.

## Mobile Requirements FAQs

### What Are Android Mobile App Requirements?

An Android app requirements document will need the following additional technical details:

* APK file size
* App permissions
* App version
* Debug and test specifications
* Operating system and SDK version
* Restrictions

### What Are iOS Mobile App Requirements?

An iOS app requirements document will need the following additional technical specifications:

* A business model
* Accurate meta descriptions for all app features
* Additional documents for UIKit, AppKit, WatchKit, iOS data storage, and app extensions
* An innovative and unique design sample
* Extensive safety features
* Legal and regulatory requirements list
* Performance metrics
* SDK versions and permissions
* User experience design

### What Are Native Mobile App Requirements?

A native iOS app requires programming languages like Swift or Objective-C, whereas a native Android app requires programming languages like Kotlin or Java.

The tech stacks in native apps depend on the operating system’s environment, which means adding more details to product requirements documents.

### What Are Web App Requirements?

The minimum requirements necessary for a web app include a user interface, database, security features, and backend infrastructure.

Testing and prototyping are instrumental to web app development projects and requirements gathering to ensure your product meets user needs.