eHKAR

Mobile App Requirements Template

A practical project-planning worksheet for businesses and development teams

Version 1.0 | Prepared by eHKAR | ehkar.com

How to use this template

- Complete the sections that apply. If you do not know an answer, write “To be decided” rather than guessing.
- Describe the result you need, not just a technology or feature name. Explain what the user does and what the app should do.
- Separate first-release essentials (Must-have) from improvements that can come later.
- Share the completed document with your development team. Clarify open questions before approving cost and schedule.
- Keep a dated copy when requirements change so everyone works from the same version.

Priority guide

Priority	Meaning	Typical decision
Must-have	Required for the first release to deliver its core purpose.	Do not launch without it.
Should-have	Important, but a safe first release can work without it.	Schedule if time and budget allow.
Later	Useful improvement for a future version.	Add after learning from real users.

Tip: A clear requirement is specific and testable. “The app should be fast” is vague; agree on a measurable loading target under normal network conditions.

1. Project overview

Project name / business

Project name: _____

Business / organization: _____

Contact details

Primary contact and role: _____

Email / phone: _____

Project status and timing

Stage: [ ] Idea [ ] Planning [ ] Design [ ] Development [ ] Existing app improvement

Target launch date (if known): _____

Project summary

In 2–4 sentences, describe the app and what it should help people do.

_____

_____

2. Business goals and success measures

Problem to solve

What problem do customers, employees, or the business currently experience?

_____

_____

Proposed solution

How will the app solve or reduce that problem?

_____

_____

Primary goals

[ ] Increase sales [ ] Improve customer service [ ] Save staff time

[ ] Deliver information [ ] Manage bookings [ ] Improve operations

[ ] Build loyalty [ ] Other: _____

Success measures

Choose measurable outcomes where possible (e.g. 100 orders/month).

Metric 1: _____ Target: _____

Metric 2: _____ Target: _____

Metric 3: _____ Target: _____

Why a mobile app?

Could a website or existing tool solve this problem? Why is an app appropriate?

_____

3. Target users and user needs

User groups

Examples: customers, delivery staff, managers, administrators, service providers.

1. _____ 2. _____

3. _____ 4. _____

Primary audience

Who will use the app most often? Note location, language, accessibility, or device considerations.

_____

User tasks

What should each group be able to accomplish?

User group: _____ Task: _____

User group: _____ Task: _____

User group: _____ Task: _____

Usage estimate

Expected use: [ ] Daily [ ] Weekly [ ] Monthly [ ] Occasionally

Estimated users at launch: _____ Expected growth: _____

4. Platforms, devices and languages

Platforms

[ ] Android [ ] iOS [ ] Web application [ ] Tablet [ ] Other: _____

Device support

Minimum Android / iOS version (if known): _____

Phone / tablet / both: _____

Languages

[ ] English [ ] Bengali [ ] Other: _____

Right-to-left language support needed? [ ] Yes [ ] No [ ] Unsure

Accessibility

Requirements for text size, contrast, screen readers, captions, or simplified navigation:

_____

5. Features and functional requirements

Instructions

List each feature separately. Describe what the user does and what the app should do in response. Mark each Must-have, Should-have, or Later.

Feature requirements

Feature / user action: _____

Expected app behaviour: _____

Priority: [ ] Must-have [ ] Should-have [ ] Later

Feature / user action: _____

Expected app behaviour: _____

Priority: [ ] Must-have [ ] Should-have [ ] Later

Feature / user action: _____

Expected app behaviour: _____

Priority: [ ] Must-have [ ] Should-have [ ] Later

Feature prompts

[ ] Sign up / login [ ] User profile [ ] Search / filters

[ ] Product / service catalogue [ ] Cart / checkout [ ] Online payment

[ ] Booking [ ] Chat / support [ ] Push notifications [ ] Maps / location

[ ] Upload photos / files [ ] Ratings / reviews [ ] Admin dashboard

[ ] Reports / analytics [ ] Offline access [ ] Other: _____

Roles and permissions

Who can view, create, edit, approve, delete, or manage information?

Role: _____ Permissions: _____

Role: _____ Permissions: _____

Important workflow

Describe a key process from beginning to end (e.g. order → payment → delivery).

Step 1: _____

Step 2: _____

Step 3: _____

Step 4: _____

6. Screens, navigation and design

Required screens

[ ] Onboarding [ ] Home [ ] Login / registration [ ] Profile

[ ] Search / listing [ ] Details [ ] Cart / checkout [ ] Payment result

[ ] Notifications [ ] Settings [ ] Help / support [ ] Admin screens

Other: _____

Navigation

Describe the menu, tabs, back behaviour, and movement between key screens.

_____

_____

Design references

List apps or websites whose design or interaction you like. Explain what you like; do not assume their design can be copied exactly.

1. _____ What I like: _____

2. _____ What I like: _____

Brand assets

[ ] Logo available [ ] Brand colours [ ] Fonts [ ] Images

Brand guidelines / asset location: _____

7. Content and data

Content

Examples: product details, articles, service information, prices, images, documents.

_____

Content management

Who manages content? [ ] Client team [ ] Developer [ ] Both

Other details: _____

Data collection

What information will the app collect, and why is it necessary?

Data item: _____ Purpose: _____

Data item: _____ Purpose: _____

Data item: _____ Purpose: _____

Existing data

Existing database, spreadsheet, website, or system to connect or migrate?

[ ] No [ ] Yes. Details: _____

8. Backend, integrations and third-party services

Backend / admin needs

Does the app need a server, database, admin panel, content management, or reporting?

_____

Integrations

[ ] Payment gateway [ ] SMS / OTP [ ] Email [ ] Maps

[ ] Shipping / delivery [ ] CRM / ERP [ ] Analytics

[ ] Push notifications [ ] Social login [ ] Other: _____

Integration details

Service: _____ Purpose: _____

Account / API access available? [ ] Yes [ ] No [ ] Need guidance

Account ownership

Who will own and pay for hosting, domains, API subscriptions, and app-store accounts?

_____

9. Security, privacy and legal requirements

Authentication

[ ] Email / password [ ] Phone OTP [ ] Multi-factor authentication

[ ] Social sign-in [ ] Other: _____

Sensitive information

Will the app handle personal, financial, health, location, children's, or other sensitive data?

[ ] No [ ] Yes. Describe: _____

Device permissions

What permissions are needed (camera, location, contacts, files, microphone)? Explain why.

_____

Policies and compliance

Countries / markets: _____

Known legal, industry, data-protection, or app-store requirements: _____

Privacy policy / terms available? [ ] Yes [ ] No [ ] Need to prepare

Security expectations

Consider encrypted network traffic, secure authentication, role-based access, backups, audit logs, secure API access, and protection against common vulnerabilities.

_____

10. Performance, reliability and support

Expected scale

Users at launch: _____ Users in 12 months: _____

Peak simultaneous users (if known): _____ Transactions/day: _____

Performance

Which actions must feel fast? Any agreed maximum loading time?

_____

Availability and recovery

Required availability (if known): _____ Backup frequency: _____

Recovery expectations: _____

Maintenance and support

Who provides support after launch? _____

Support hours / response target: _____

Who handles bug fixes, OS updates, and dependency updates? _____

11. Testing and acceptance

Testing required

[ ] Functional [ ] UI / usability [ ] Device [ ] API [ ] Security

[ ] Performance [ ] Accessibility [ ] User acceptance testing

Test devices

List important real devices, OS versions, and screen sizes.

_____

Acceptance criteria

A requirement is accepted when its expected result can be checked.

Requirement: _____ Pass condition: _____

Requirement: _____ Pass condition: _____

Requirement: _____ Pass condition: _____

Approval

Who approves the app? Name / role: _____

Approval date: _____

12. Deliverables, schedule and budget

Deliverables

[ ] Android app [ ] iOS app [ ] Web app [ ] Backend / API

[ ] Admin panel [ ] UI/UX designs [ ] Source code [ ] Documentation

[ ] Store listing assets [ ] Deployment [ ] Training [ ] Other: _____

Milestones

Requirements: _____ Target date: _____

Design approval: _____ Target date: _____

Development complete: _____ Target date: _____

Testing / acceptance: _____ Target date: _____

Launch: _____ Target date: _____

Budget

Estimated budget and currency: _____

Budget: [ ] Fixed [ ] Flexible [ ] Not decided

Includes hosting, subscriptions, taxes, store fees, maintenance?

[ ] Yes [ ] No [ ] To be discussed

Payment preferences

Milestone / amount / due date:

_____

_____

13. Assumptions, exclusions and risks

Assumptions

What are we assuming (e.g. client supplies content, API access is approved, internet is available)?

_____

_____

Out of scope

What is explicitly not included in this project or first release?

_____

Risks and dependencies

Examples: third-party API approval, payment-provider onboarding, delayed content, app-store review.

Risk: _____ Impact: _____ Mitigation: _____

Risk: _____ Impact: _____ Mitigation: _____

14. Priorities and first release (MVP)

Must-have features

List only what is required for the app to deliver its core value.

1. _____

2. _____

3. _____

4. _____

Future features

Useful features that can wait until after launch.

1. _____

2. _____

3. _____

Open questions

What decisions or information are still missing?

_____

15. Review and approval

Business representative

Name: _____ Signature: _____

Date: _____

Development representative

Name: _____ Signature: _____

Date: _____

Change control

After approval, document proposed changes and assess their effect on cost, schedule, security, and scope. Obtain approval before proceeding.

Important note

This template is a planning aid, not a legal contract, security certification, or final technical specification. A qualified development team should review requirements, identify risks, and clarify estimates before work begins.

Before you send this to a developer

[ ] Have we explained the business problem and main user groups?

[ ] Are first-release features clearly separated from future features?

[ ] Have we identified platforms, languages, integrations, and data?

[ ] Have we discussed security, privacy, backups, and ongoing maintenance?

[ ] Can each important requirement be tested and approved?

[ ] Have we discussed ownership of source code, accounts, hosting, and subscriptions?

[ ] Are assumptions, exclusions, risks, budget, and milestones documented?

Plan clearly. Build confidently.

eHKAR | Digital products and software solutions