

Software requirements specification template

This template structures the requirements for a software project before you request quotes. Replace the placeholders in brackets with your own information and remove any sections that don't apply to your project. The more specific your answers, the easier it is to compare quotes from different agencies.

1. Current situation

1.1 Company and industry

Briefly describe your company, its industry, and its size so agencies can understand the context. [...]

1.2 Reason for the project

Explain why the project is happening now, for example a relaunch, a new product, or an expiring contract. [...]

1.3 Current solution

Describe the solution you use today and its biggest shortcomings from your point of view. [...]

2. Goals and success criteria

2.1 Project goals

State the three to five most important goals of the project in your own words. [...]

2.2 Success criteria and metrics

Define how you will know after launch that the project succeeded, for example load time, conversion rate, or processing time. [...]

2.3 Out of scope

Describe what is explicitly not part of the project, to avoid misunderstandings later. [...]

3. Users and roles

3.1 Target groups

Describe who will use the solution: customers, employees, or external partners. [...]

3.2 Roles and permissions

List the planned roles, for example administrator, editor, or guest, and what each role is allowed to do. [...]

3.3 Expected usage

Estimate the expected number of users and their access patterns, for example peak times or seasonal spikes. [...]

4. Functional requirements

4.1 Core functions

Describe in a few sentences what the solution needs to do at a basic level. [...]

4.2 Requirements list

List individual requirements with a priority. Must-have requirements are a disqualifying criterion if unmet, should-have requirements are important but negotiable, and could-have requirements are desirable extras.

No.	Requirement	Priority (Must/Should/Could)	Notes
1	Users can sign in with email and password	Must	Two-factor authentication optional
2	Export reports as PDF	Should	Confirm the format with the relevant department
3	[...]	[Must/Should/Could]	[...]
4	[...]	[Must/Should/Could]	[...]

4.3 Excluded functions

Name functions that are deliberately not part of the project, even if they seem like an obvious fit. [...]

5. Non-functional requirements

5.1 Performance

Define expectations for load times, response times, and performance under load. [...]

5.2 Security

State requirements for authentication, encryption, the permission model, and any certifications needed. [...]

5.3 Accessibility

Specify which standard applies, for example WCAG 2.2 Level AA, and whether it's a legal requirement. [...]

5.4 Data protection

Describe which personal data is processed, where it is stored, and whether a data processing agreement (DPA) is needed. [...]

6. Interfaces and data migration

6.1 External systems

List the systems the new solution needs to work with, for example ERP, CRM, or a payment provider. [...]

6.2 Data migration

Describe which data needs to move from which legacy system, and what condition it's currently in. [...]

6.3 Interface formats

Name any required standards or protocols, for example REST, SOAP, or file exchange via CSV. [...]

7. Constraints

7.1 Budget

Provide a budget range, even a preliminary one, so agencies can put together realistic quotes. [...]

7.2 Timeline and milestones

Name the desired project start, key interim dates, and the target launch date. [...]

7.3 Technology requirements

Note whether specific technologies, frameworks, or platforms are required or excluded. [...]

7.4 Hosting

Describe whether hosting will be handled by the agency, an existing provider, or in-house. [...]

8. Operations and maintenance

8.1 Responsibilities after launch

Clarify who is responsible for monitoring, updates, and bug fixes after go-live. [...]

8.2 Response times

Define the response times expected for incidents, by severity level. [...]

8.3 Further development

Describe whether, and to what extent, further development is planned after launch. [...]

9. Rights and handover

9.1 Code ownership

State that usage rights to the code created transfer fully to you after acceptance and payment. [...]

9.2 Documentation

Name the documentation that must be delivered, for example technical documentation, an operations manual, or training materials. [...]

9.3 Handover

Describe how code, access, and knowledge will be handed over at the end of the project or if you later switch providers. [...]

10. Acceptance criteria

10.1 Acceptance process

Describe how, and by whom, acceptance will be carried out, for example through a test catalog or a defined test phase. [...]

10.2 Acceptance criteria

Define the conditions that must be met for acceptance, for example that all must-have requirements are fulfilled and critical functions are free of defects. [...]

11. Contacts

11.1 Client contact

Name the responsible person on your side, with their role and how to reach them. [...]

11.2 Agency contact

The agency adds this after the contract is signed, naming the person responsible for the project. [...]

Template by techagenturen.de — free to use.