

Apiko

SOFTWARE DEVELOPMENT PROJECT TRANSITION CHECKLIST

TO MAKE SURE YOU ARE ALL SET FOR
A SUCCESSFUL PROJECT TRANSITION
TO A NEW IT VENDOR



PROJECT TRANSITION CHECKLIST

If for any reason you need to change your software development services provider, plan every aspect of project transition in detail. Without a thorough plan, getting any improvements will be a 50/50 luck. And it's not something one can afford when significant budget, time, success of your software solution, and, most likely, business digital transformation are at stake.

Follow this checklist to get your IT project transition fully covered and under control.

- 01 Accept the last product increment delivered by the former team.
- 02 Create a list of sensitive data. It may include security documents, authentication certificates, servers access keys, source code, keybase, including access to App Store and Google Play Market, keys to third-party services integrations, etc.
- 03 Collect environment variables and authentication certificates.
- 04 Collect all the project assets like mockups, design files, written project requirements, and marketing materials (e.g. Adobe XD, Figma files, etc.). Make sure that you are the business logic holder
- 05 Collect project specifications, code/architecture and QA documentation, database diagram, deployment scripts and procedures. If any of these documents are missing, ask your current vendor to create them.
- 06 Collect the root credentials for all the third-party services or ensure their ownership is transferred to you successfully.

CHECKLIST

Follow this checklist to get your IT project transition fully covered and under control.

- 07 Change passwords to all third-party services used, and delete previous vendor's accounts.
- 08 Transfer all the source code root ownership to you. Collect the output code documentation and API calls, often presented using tools like [Swagger](#).
- 09 **Change the keys and remove access rights** from those who no longer work on the project. Set up an encrypted environment to store new development credentials and sensitive files:
 - <https://aws.amazon.com/kms/>
 - <https://www.vaultproject.io/>
 - <https://keybase.io/>
 - <https://bitwarden.com/pricing/business/>
- 10 Agree on a knowledge transfer period. 1-2-hour daily contribution of a software developer previously working on your project will let your new IT vendor get the answers to technical questions regarding current app implementation, and **come up with the necessary improvements**.
- 11 Onboard a new software development team.
- 12 Apply immediate process improvements (check the best practices below 🙌), like daily meetings, scrum hygiene, **strive for clarity, precision and transparency both from your and IT vendor's sides**.

10 BEST PRACTICES TO IMPROVE SOFTWARE DELIVERY

- 01 Access control.**

Securely store the credentials to all required services in one place, like an encrypted storage. You should be the owner of all corresponding accounts. Make sure the two-factor authentication (2FA) is enabled to prevent hacking attempts.
- 02 Demo meetings.**

Conduct regular demo meetings and agree upon the scope of the features beforehand.
- 03 Work planning and reporting.**

The team should plan work instead of getting things done sporadically. As the Product Owner, facilitate the planning by taking into account the interest of other stakeholders. The project manager, in their turn, should compile a weekly written report with a statement of work and hours spent.
- 04 Project management system hygiene.**

Enter all the work items in your project management system (e.g. at Apiko we use Jira), and review them every week to validate their status, priority, blockers, and description.
- 05 Requirements elicitation sessions**

with a Business Analyst (BA) for documenting any change requests or new features. The requirements should be documented, visualized, and approved before taking them into development.
- 06 Design workshops.**

All new requirements should have design references so that developers don't need to make UX/UI decisions on their own.

10 BEST PRACTICES TO IMPROVE SOFTWARE DELIVERY

07 Code commit messages.

When developers upload the code they've written into Gitlab, they should create a meaningful message instead of something like "bug fixes". Ideally, that message should contain the ID of the related Jira issue. Here's an example: feat(AP-1123): Enable user to go "back" to the previous page after login (#235).

08 Release notes.

The software engineering team should create and post a release note in a separate communication channel (e.g. in Slack) 15 minutes before the deployment or sooner. They should clearly indicate the platform type and work items (commit messages) in the release.

09 **Proof of concept (PoC) and minimal viable product (MVP) development.**

PoC development is not a mandatory stage, but it's often required, especially during the project handover phase. It helps to validate certain approaches, making sure the project is on the right track. And if your app has not been released yet, or needs a serious makeover, start with an **MVP development**. The sooner you are able to get real-life feedback from app users, the more chances you have to develop a successful solution.

10 App refinement.

When we at Apiko take over software development projects, in a way, we have to deal with **legacy application modernization**. Software architecture and IT infrastructure checks together with code refactoring are a must to set a solid foundation for your project.

Planning does take time...to **save you way more time, effort, budget, and to reduce risks and overall stress by introducing order, clarity and transparency**. The introduced software transition plan is easy to apply to any app development project, and the attached checklist won't let you skip a bit.

It's not only the project transition itself, but also the way you organize your cooperation with the software development team that are crucial for the best outcome. Implementing the described proven best practices will certainly be a good start. Feel free to **reach out** if you have any questions or feel like you could use a consultation for your particular case.

ABOUT APIKO

Apiko is a software development company that enters markets with digital businesses, using a solid process and clever strategies. We merge imagination and technology to help brands grow in the age of constant change and transformation.

SOCIAL LINKS

[in](#) [Linkedin](#) [Twitter](#) [f](#) [Facebook](#)
[YouTube](#) [Clutch](#)

CONTACTS

Harju maakond, Tallinn,
Haabersti linnaosa, Paevalille tn 6-7, 13517

contact@apiko.com

apiko.com