

Agile project management – always on the right track



Agile development is a widely common and proven strategy used in the management of complex software projects. Unlike software engineering, the people involved and the communication between them play a central role rather than processes and specifications. The advantages of an agile methodology can be immediately seen if unclear or changing demands are expected, and if the course of the project is likely to be affected by different or unpredictable factors.

Key factors for success are a sense of responsibility and professionalism of those involved. The employees of BeCompany have made your satisfaction and trust their highest priority. Their motivation comes from the anticipation that the software they make will be accepted and used. The entire development process is focused on this goal.

Accepting and being able to respond to change

It is normally impossible to know and to specify all of the requirements and framework conditions before beginning with implementation. It is often the case that a clear picture of the software to be created is formed only after evaluating the first prototypes. Confrontation with usable versions at an early stage brings about new ideas. Some features are more difficult to implement than expected. In return other features are covered by new products that come onto the market during the course of the project.

Under these conditions it would be unwise to follow rigid specifications or to refuse wishes to change. We do not consider changes to be a deviation, but rather a correction of the path. The goal is not foremost to implement an originally planned specification, but rather to provide a useful and qualitatively superior product at a reasonable price.

Communication is the key to success

You as customer and future user of the software product not only play a central role during the initial specification phase but also during the entire course of the project. We work closely with you on a regular basis to inform you of the current stage of development and to provide you with the opportunity to react to problems and to make changes when necessary. If anything is ambiguous or obscure, an expert is available for immediate consultation so that no incorrect functionality is implemented or that the course of the project is delayed. Iterative development

We carry out our projects in several stages or so-called iterations. This way we are able to provide functional and complete versions of the software product in a timely manner and on a regular basis. You have the opportunity to assess the progress of the project at any time. Your users can practice the functions already implemented and can influence development by providing feedback. An iteration usually lasts one to four weeks and includes the following phases:

- **Reflection and adaptation of the methodology**
The project methodology must be monitored with scrutiny and if necessary adapted to changing requirements.
- **Planning of the features to be implemented with the decision makers**
Based on recommendations of our experienced developers, the decision makers choose which features are to be implemented in this iteration. Prototypes help to assess the feasibility and costs.
- **Requirements analysis**
Because the framework conditions can change during the course of a project, the concrete requirements are determined only right before they are implemented. Direct communication with potential users and experts, e.g. in workshops, and acceptance tests using (paper) prototypes has proven to be effective in this phase.
- **Implementation**

During implementation we focus on the features planned for this iteration. At the same time, the existing code is revised and adapted to new requirements and data. In order for questions to be answered quickly, the developer must be able to communicate with experts throughout the entire implementation phase.

- **functional tests, integration and stress tests**

In addition to continuous automated tests, specific tests are carried out that focus on aspects crucial to the current iteration.

- **Usability and acceptance tests**

Regular user tests help to confirm the correct implementation of requirements as well as early detection of usability and accessibility problems.

- **Communication of the results and project progress, decision maker feedback**

The progress of a project should be transparent and comprehensible in order to ensure continuous trust of the decision makers, which is why after each iteration the current status, problems and time delays are openly communicated so that there is enough time to react accordingly.

Test-driven development

Our developers have recognized and internalized that the implementation of tests is not a necessary evil, but rather leads to quicker success in developing new functionality and to a reduction of errors from the start. We use tests not only to continuously ensure functional correctness, but also as specification and documentation for use of programming interfaces. The focus on test-driven development also improves the quality of the architecture since the system is divided into easily manageable and functionally clear layers and components.

Methods and tools

Also in software development first class products can only be produced using high-quality and well-maintained tools. Our developers use proven and sophisticated environments and methods that mesh perfectly and complement each other.