

Impact of agile software development on TMS projects

In recent years, the development of agile software and systems has become an alternative to traditional methods. This includes, for example, the waterfall model for breaking down development projects into fundamental phases such as analysis, specification, design, implementation and testing. Agile development methods, on the other hand, follow a different approach: short release cycles and incremental development based on close collaboration and timely feedback to clients.

The use of ever more agile methods and scrum¹ projects is also observable in Treasury. Especially in conjunction with complex treasury management systems (TMS), ready-made products frequently are no longer sufficient. In the course of implementing larger TMS projects, a slew of technical updates often become necessary in accordance with various requirements of internal departments in addition to external new (regulatory) developments. If it transpires during the analysis phase of such a project that the solution offered by the manufacturer does not meet all the internal requirements, i.e. the manufacturer is in need of new product development, agile working methods are increasingly used.

However, what are the effects of agile system development on Treasury processes and projects, especially when introducing new or upgrading existing treasury management systems? To what extent are the project structure, planned approach (project, task and resource planning) and, not least, the entire corporate structure affected? What fundamental challenges should companies expect?

What are the characteristics of agile system development?

If the analysis of requirements at the beginning of a project shows that there is need for development with regard to the standard solution during the introduction or upgrade of a TMS at one point or another, the next logical step would be a development phase in 'sprints' in accordance with agile system development approach 'scrum'.

During a sprint, a team develops a fully operational part of the application based on a pre-defined work package (increment). Sprints usually last two weeks and are combined into blocks of development and adjustment sprints for expediency. At the end of a sprint or block, the increment (i.e. the implemented functionality), is presented to the client in a sprint review meeting. Based on the client's feedback and new requirements, detailed planning for the next sprints occurs.

It is in the nature of agile methods that shorter development cycles lead to more deliverables (sprints) in shorter time intervals. The traditional separation into intake of requirements and their implementation at the manufacturer followed by delivery to the client (and subsequent integration, testing and inspection until going live), is replaced by an iterative approach which more closely involves the client throughout the process.

What are the consequences for clients?

Agile system development requires intensive communication and close collaboration of the client with the manufacturer, and also more effort with regard to integration and testing of the incrementally developed components of the comprehensive solution. With regard to testing, regression tests will become increasingly important, because it must always be ensured that previously tested and released components continue to function flawlessly. It would be advisable to strive for a high degree of automation of test scenarios, in order to be able to conduct these tests repeatedly with as little manual effort as possible. But, pivotal test phases conducted by end users, such as user acceptance

¹ Scrum is a model frequently used in practice for agile project management and agile software development. It is characterised by an empirical, iterative approach and incremental developments. With only few clearly defined roles and tools, the focus is on flexibility and a dynamic team.

tests, will also have to be integrated – broken down into several steps – into the project and can no longer be planned as one phase at the end of the project, as is traditionally the case.

Agile working methods also require reorganisation at the client. The focus on providing operational software components quickly and avoiding bureaucracy is likely to affect project planning and the control of projects and budgets. In project teams, the client's professional experts and manufacturer's developers will have to collaborate closely and have the authority to make the necessary decisions.

What are the advantages and disadvantages?

Some of the advantages certainly are fast implementation of requirements through shorter development cycles and faster delivery of functions to the client. The risk of taking the wrong path is identified quickly during the course of the project and can be removed relatively unbureaucratically. This pays off especially when the quality of implemented requirements can really only be sensibly assessed in the prototype. Early involvement of the client and demonstrations of new software functionalities at short intervals allow for a flexible response to identified errors, gaps or recent changes to be taken into account. The client is able to influence major development steps at an early stage and actively participates in the decision-making process.

Depending on the extent to which an agile structure is already used in a company or department, the introduction of scrum and Kanban & Co. however requires considerable implementation effort. Agile methods require close cooperation of the client (ideally in smaller teams) with the supplier, reflected in frequent meetings, coordination and more testing of software. This iterative process may entail a major reorganisation of project planning, project structure, planning of time and resources as well as feedback culture, even as far as a fundamental change in thinking among management and staff. If the manufacturer is agile in its thinking and actions, clients must be equally flexible in their response, in order to fully use the potential of agile procedures and benefit equally on both sides.

This approach also has an impact on the project-based audit of TMS projects. Due to the fact that many requirements are not clearly defined in advance and only specified and implemented in detail during the project, it is becoming increasingly difficult to verify requirements and their implementation. The auditor should therefore be continuously involved in the coordination process between the manufacturer and client, in order to be able to follow and audit the development of individual items adequately. In the end, it must be ensured that all requirements are taken into account, correctly implemented and suitable documented – even if the focus of the agile method is on a functioning product and not on elaborate specifications. Short-term changes discussed and coordinated in small cross-functional teams can only be verifiably presented by means of consistent documentation and appropriate reporting mechanisms.

What are the challenges?

Depending on the extent to which agile project management, agile thinking and commensurate structures have been put in place and developed, agile TMS projects present a more or less major challenge to the stakeholders and staff of a company.

The more complex the integration of a TMS within a company's organisation and structure (such as the interconnection of several departments and subsidiaries, multiple interfaces with other systems, suppliers and banks), the greater the impact.

Nevertheless, it will no longer be possible to avoid the use of agile methods in TMS projects, as, in addition to changing professional conditions, ever more regulatory requirements (changes in accounting standards, new regulations and laws, etc.) necessitate great flexibility and timely implementation. This can be greatly facilitated by shorter project phases and smaller software development steps combined with timely validation of the results. Furthermore, close collaboration

with the software provider and its methods and approaches is essential in order to successfully complete joint projects on time.

The key aspects of agile development should be made transparent among all involved at an early stage, clearly defined and coordinated, in order to allow consistent planning and procedures and project progress without mishaps.