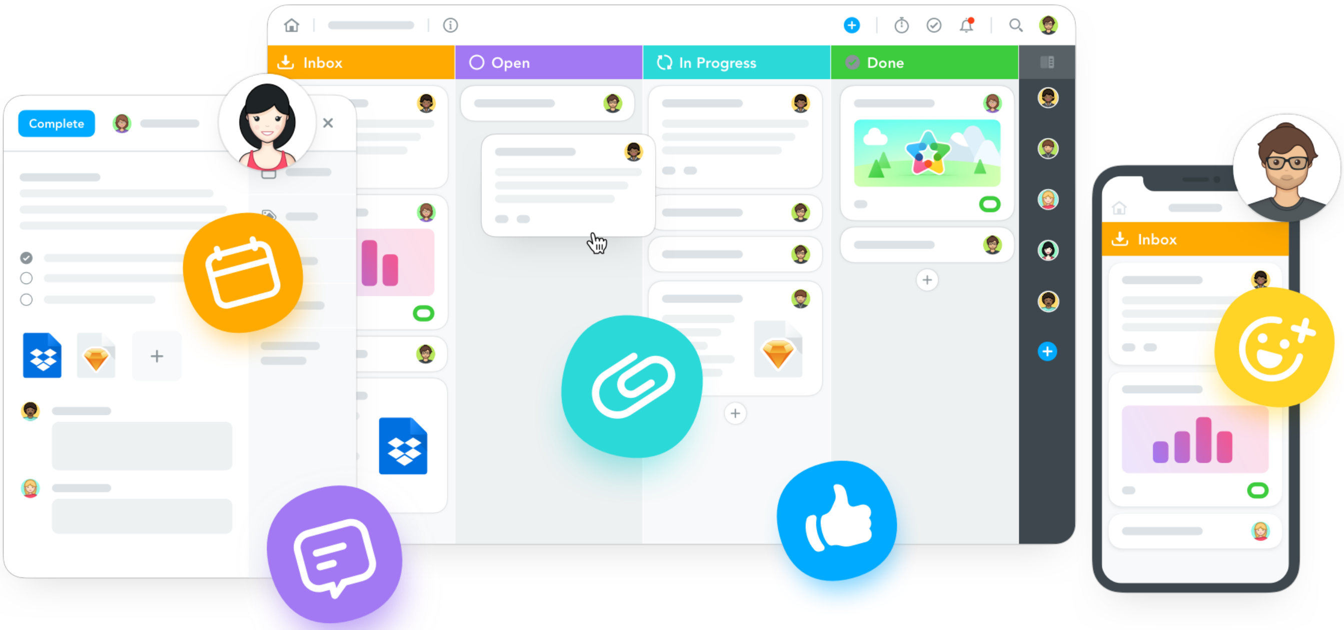
meister task

Staying Agile

5 Best Practices in Software Project Management



Olivia Jardine | Stefan Forster

CONTENTS

INTRODUCTION

- Our five best practices will cover

1. ESTABLISHING THE ‚WHY'

- Beginning Every Project with a Strategy Sprint
- Running a Five-Day Strategy Sprint

2. PLANNING TO PERFECTION

- How to provide a Realistic Project Scope & Budget
- Establishing an Agreed Project Scope
- From Project Plan to Realisitic Budget

3. CUSTOMER VALUES

- Welcoming Client & User Feedback at Every Iteration
- Integrating User and Client Feedback within Sprints
- Creating a Physical Scrum Wallboard
- Managing Project Scope via Online Collaboration Tools

CONTENTS

4. CROSS-DEPARTMENTAL COLLABORATION

- Ensuring Development and Business Colleagues Work Together
- Preventing Siphoning with Online Communication

5. SUSTAINABLE PRODUCTIVITY

- Achieving Team Efficiency Long-Term
- Reviewing Sprint Productivity
- Creating Sustainable Workflows

SUMMARY

- Review of our five best practices
- About the Authors

INTRODUCTION

Ordinarily, software projects are launched for one of two reasons: either a client has come up with a ground-breaking idea, or they need a solution to a pain point they're experiencing.

In these projects, just like most areas of business, it's impossible for teams to predict exactly what solution will work best. Perhaps, after the initial project scope, a competitor enters the market, meaning you need to focus on different features that set your product apart. Or maybe the CRM tool that the CTO at your client's firm envisaged is actually far too complex for all staff members to on-board to.

Unforeseeable eventualities like these are common. Your project team needs to have the processes to adapt and evolve your project plan, as new market information and client requirements emerge. To this end, flexibility is essential in delivering a product with the highest value to your client and end-users - and while within budget. This is where Agile project management has provided teams worldwide, in software development and beyond, with the skills and processes to build valuable products.

You may already be aware of the values outlined in the Manifesto for Agile Software Development. When written back in 2001, the twelve Agile principles were vaguely formulated and left a lot of room for interpretation (as well as misinterpretation). However, over the past 16 years, several best practices for applying Agile have been learned, tried out, regularly adapted and improved. In research for this paper, we've spoken to teams worldwide who are taking an Agile approach to their software development projects and are reaping hugely positive results. Most have created strategies that take influence from the 2001 manifesto and updated these values to meet their team's modern workflows.

In this modern guide to staying Agile, we'll share lessons from our combined 15 years of experience in software project management, partnered with strategies taken from other successful Agile software development teams along the way.

OUR FIVE BEST PRACTICES WILL COVER:

- **Establishing the 'Why':**
  How to begin every project with a strategy sprint

- **Planning to Perfection:**
  How to develop a realistic project budget

- **Customer Values:**
  How to engage with client and user feedback at every iteration

- **Collaborating Cross-departmentally:**
  How to enable constant communication between your business and development teams

- **Sustainable Productivity:**
  How to achieve team efficiency long term

Whether you're new to Agile and looking to persuade colleagues about the benefits, or are already using Agile but are hoping to improve your team's workflows, this guide will provide you with tangible lessons to apply with your team.

As a note, throughout the white paper, we use the term 'client'. However, we use the term to refer to both internal and external stakeholders.

1. ESTABLISHING THE ‘WHY‘:

Beginning Every Project with a Strategy Sprint

According to the Agile Manifesto, the number one priority of Agile teams should be “to satisfy the customer through early and continuous delivery of valuable software.” This means providing a product that is either specialist to the pain-point highlighted by your client, or a unique service, which is not yet supported to an equal level by a competitive tool.

To this end, although developers can be keen to dive into user story development from the outset, all projects must begin with a thorough assessment of what’s already available, and why this product is required. This means beginning your project with a strategy sprint, focused on identifying either:

- which pain point(s) the proposed product would be solving,
- which unique experience or service the proposed product would support.

Once one of the above is identified within the strategy sprint, this objective will become the raison d’être for your project. Even as features come and go and business requirements change, the vision must be defined and communicated to the team, meaning everyone is on track with where the project is headed.

Max Dufour, working in product development, shared that careful planning of the sprints initiation phase is, in his view, the key to a successful Agile delivery process.

> “In many cases, it is necessary to go beyond sprint planning and to facilitate a strategy-focused sprint, Dufour shares. This will enable your Agile team to decide what are the best products, modules and features to build from both a business and technology standpoint.”

Max Dufour
Digital Strategy expert

Running a Five-Day Strategy Sprint

For a step-by-step guide of how this strategic planning phase should look, Dufour signposts to Google Venture's five-day sprint. The Google Ventures process is designed to answer critical business questions through a five-day agenda of designing, prototyping, and testing ideas with real prospective customers. The aim of the week is to ensure the proposed product is truly valuable, before expansive development work is begun:

"The sprint gives you a superpower," Google Venture claims. "You can fast-forward into the future to see your finished product and customer reactions, before making any expensive commitments."

To hone in on the key project objective, the Google Ventures initial strategy week features the following agenda:

Day 1: Map out the pain point or need for the product
Day 2: Establish existing competing solutions
Day 3: Use the initial ideas to establish a testable hypothesis
Day 4: Create a rough product prototype
Day 5: Test the prototype on prospective end-users (i.e. real human beings).

Day 1 **Day 2** **Day 3** **Day 4** **Day 5**

By mapping out a clear idea of what the aims are from the outset, you can provide structure to your initial sprints and ensure that all developers are on the same page with the long-term goals. What's more, all of this can be done before development has begun, which, as Parag Sharma, Founder and CEO of software development team Mantra Labs, shares, "goes a long way in creating the correct architectures and saving money in the long term."

Agile Practice 1: Dedicate your first week of work to implementing a strategy sprint, by designing, prototyping and testing your product, in order to identify your project vision.

✓ 2. PLANNING TO PERFECTION:

How to Provide a Realistic Project Scope & Budget

We’ve established that the main objective of Agile project management is to deliver a valuable product to your client. However, how a client defines software as valuable may well change as the project progresses and business requirements evolve. It might be that after a certain sprint, consultation with a group of prospective users reveals that a favorite new feature is in fact superfluous, and that instead an additional guidance feature is needed to make the UX clearer.

Inevitably, as a project moves forward, new features will emerge and priorities might shift course. When a software development company has provided an initial cost estimate, trying to account for additional features within budget can become a minefield.

As a result, the second best practice in implementing a successful Agile project is to develop a realistic project scope and estimate.

In the Manifesto for Agile Software Development, a key value is to prioritize customer collaboration over contract negotiation. However, prioritizing a realistic, agreed budget at the beginning of the contract does not mean needing to obsess over contracts. It simply requires a fair and honest prediction of what can be achieved within the remit of a certain budget, in order to ensure that both the client and Agile team are happy with that estimate before progressing. It also requires a means of enabling your scope to evolve, by demonstrating to your client how features can be swapped in and out of the plan while remaining within budget. We’ll come to how that can be implemented a little later in the paper, with a transparent project roadmap.

Establishing an Agreed Project Scope

To ensure that their bespoke software development team are on the same page as their clients in regards to project scope and budget, Agile software team IT Enterprise create a project mind map for each new client during the project consultation stage:

“Having client requirements displayed in a visual way immediately demonstrates the complexity of the project to both us and the client. Linking different sections of the mind map allows the team to visualize interactions between different project areas, while defining the client’s user story and how we will go about achieving it.”

A challenge that the IT Enterprise team had previously experienced was that additional client requirements had occasionally slipped through the gaps at the initial consultation stage. This meant that some features were missed from the initial project plan and estimate, leading to difficult conversations and cost recalculations along the way.

By sharing a visual idea of project scope with the client ahead of any agreements being signed, either in the form of a mind map or another shared document, both teams can ensure that their expectations are aligned. IT Enterprise can then go on to develop a realistic estimate for the project, ensuring that all key features have been considered in the price.

From Project Plan To Realistic Budget

U.S. Agile development company Praxent takes this costing process one step further with their process for providing a realistic quote to potential clients:

> "We use the Fibonacci sequence to size new feature requests. But staying within scope begins with accurately pricing the product from the very beginning."

Tim Hamilton
Praxent CEO

As a result, the team has developed a product development pricing process based on the cone of uncertainty.

"This entails a progressive conversation with clients across three stages," Hamilton explains. The three phases, covering product definition, product requirements, and UX design, establish the following:

1. Product definition phase: In this phase, no prices are provided

2. Detailed requirements phase: In this phase, the team estimates the general cost of product development based on the client's ability to pay and perceived difficulty of the project.

3. User interface design phase:

- Praxent creates a clickable software product prototype within the estimated budget.
- Praxent re-prioritizes project objectives and features where appropriate, in collaboration with the client.
- Praxent finalizes the quote, using findings from the prototype discovery and usability testing.

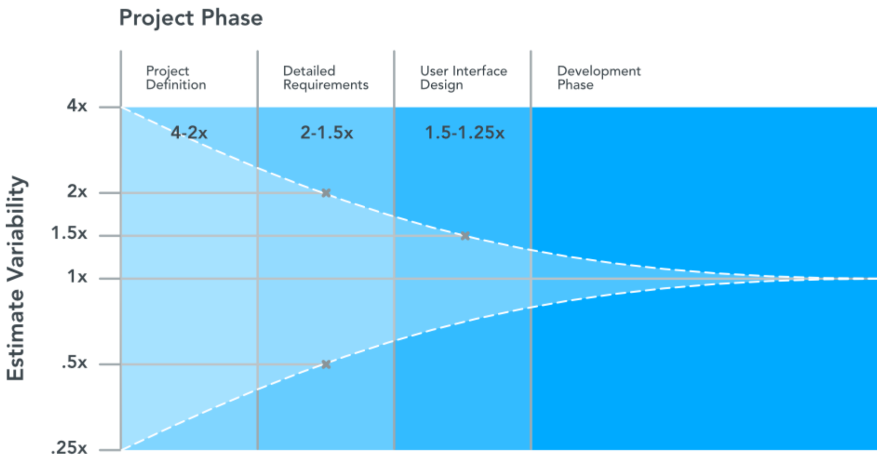


Illustration based on the *Cone of Uncertainty* graph developed by Barry Boehm and Steve McConnell.

Praxent strive to enable their clients to make informed decisions, through proactive project updates and weekly client meetings. With this approach, Hamilton shares that the Praxent team has been able to manage and even exceed budget and scope expectations.

Agile Practice 2: To ensure you're able to stay within project scope and budget, begin by agreeing on a visual roadmap of key requirements with your client, and then utilize the cone of uncertainty method to drill down on a realistic project estimate, before development work begins.

3. CUSTOMER VALUES

Welcoming Client & User Feedback at every iteration

The next key value of the Agile Manifesto that we'll focus on is that "changing environments are embraced at every stage of the process to provide the customer with a competitive advantage."

This ability to adapt to evolving markets and changing business requirements is what makes Agile unique. Welcoming client and user feedback at each iteration not only ensures that you're on track to produce a valuable product but enables your team to make changes before the final product is launched, saving time and money.

Actively encouraging and listening to client and user feedback at every iteration is therefore essential to taking an Agile approach.

Accessing User Feedback

Dorian Simpson, Project Management Strategist and author of Agile for the Enterprise, shares that there are three key steps for achieving rapid and regular end-user insight:

1 Access:

Teams must find and identify a set of target customers that can be relied upon to provide accurate and timely insight. "Successful Agile requires developing a well-maintained customer panel or advisory board," Simpson shares.

2 Listen:

Once a customer panel or advisory board has been secured, teams must build skills to actively listen to them. This should go beyond a single focus group or survey. Simpson explains that although these methods can be used, "having high-quality interaction with your customers - in person or through collaboration tools - is critical, as is probing customers for real needs, problems, desires and objective feedback." This stage should involve sharing early designs with customers to find out which elements of the solution they would prioritize, helping to also cut back on unnecessary features.

3 Communicate:

Ensure that what is learned through this listening process is clearly converted into prioritized use cases that explain the value of each feature, by communicating user insights to the entire project team.

Integrating User and Client Feedback within Sprints

Via Simpson's three-step process, an Agile team can gather information from prospective end users, which, alongside client feedback, should then be integrated at every Sprint.

One way to achieve this is through taking a Scrum approach to Agile project management. If you're new to Scrum, the approach involves a Product Owner who retains a bird's-eye view of the project. This is achieved through meeting with the development team and 'Scrum Master' on a daily basis, to communicate evolving business goals and customer needs. These goals and needs will then form each Sprint, and following each Sprint, Scrum teams hold a Sprint Review to present on what has been completed to all stakeholders. This process pushes team accountability and ensures that each Sprint's completed tasks align with the project's business, client and user goals.

Creating a Physical Scrum Wallboard

> In Scrum, the Development Team and Product Owner can move and change items in the Product Backlog at any time.

Jonathan Roger
Operations Director and Certified Scrum Master
at software development company And Plus

This can, as mentioned, lead to competing priorities and a surplus of features. Roger explains that in order to keep the project from growing beyond scope and budget, the AndPlus team creates a physical board for their Scrum task cards, which is shared with the clients to track project progression:

“One way that we keep track of scope with our clients is by using different colored task cards on our project management wallboard to represent items that were added after a release plan was created,” Roger shares. “This makes it visually easy (for both us and the client) to identify items or features that were added after the initial project scope, so need to be traded with another feature of equivalent development hours in order to stay within budget, or postponed to a later release.”

Managing Project Scope via Online Collaboration Tools

Software development team Praxent takes a similarly visual approach to scope management and have developed their own online collaboration tool to share with clients. CEO and Founder, Tim Hamilton, shares that the Praxent team developed a custom reporting and analytics dashboard, in order to place feature management in the hands of their clients. Praxent’s collaboration tool displays all data to keep clients up-to-date on project momentum, budget health, scope management, and rate of spending:

> The dashboard provides a play-by-play on feature development progress, allowing clients to see the impact it would have on production should they decide to shift direction or re-prioritize.

Tim Hamilton
Praxent CEO

“Additionally, we provide them with data on how far their budget dollars will stretch, allowing them to control feature scope as they choose to add or swap in new features.”

By enabling your clients to track project progress transparently, while controlling feature and budget management, you can ensure their opinion is integrated at every stage. Equally, it will help to prevent the project from spiraling out of hand with additional features or end up over budget.

While not all Agile teams will have their own custom-made dashboard to play with, there are many collaboration tools available online that enable teams to take a similarly transparent approach to Scrum or Kanban project management. Look for a tool that allows you to share your project roadmap with your client. Collaboration tools that enable you to include details within each task, such as estimations, will also enable transparency over costs. This way you can provide your client with the ability to manage feature prioritization while remaining informed of how the changes will affect the budget.

Agile Practice 3: Establish a clear means of gaining regular, valuable feedback from both your client and end-user. Then integrate this feedback at every iteration, with transparent feature management, via a shared collaboration tool.

4. CROSS-DEPARTMENTAL COLLABOARTION:

Ensuring Development and Business Colleagues Work Together

The next essential Agile value is the need for business analysts and developers to work together on a daily basis, throughout the project.

Speaking on the importance of constant collaboration between business analysts and the development team, project management strategist Dorian Simpson shares the following:

“To guide the development team’s iterative approach, the marketing and sales teams must be clear on what customers deem most important and how market dynamics are impacting on solution requirements, to guide Agile efforts with every Agile Sprint.” Simpson explains that the bottom line is that “any Agile approach used by the development team must support all business needs and address all stakeholder concerns.”

When the original Agile Manifesto was written back in 2001, the team argued that the best cross-departmental collaboration is done via face-to-face meetings. With developers traditionally writing user stories and planning Sprints using highly technical tools, often inaccessible to business colleagues, there’s no wonder that face-to-face meetings were seen as the antidote to departmental siphons. However, 16 years on, distributed teams are now commonplace across the technology sector, making the Agile value to prioritize in-person meetings difficult to implement.

Preventing Siphoning with Online Communication

The Software Product Partner team, NewOrbit, came up against a similar challenge when managing Agile projects across their business, operations, product management and development teams. When they began taking an Agile approach to their software projects, the teams were experiencing siphoning, as the developers were using a highly technical tool that other staff members were unable to on-board to. In many cases, this meant that the business and operations staff were unable to access the Sprint roadmaps and priorities, creating a breakdown in project communication.

In order to create a bridge between departments, NewOrbit's Support Consultant, Matthew Dowding, shared that the team sought and found an online collaboration tool that all teams could easily on-board to:

> "As a result, our new, simpler task management workflow acts as the bridge between our different departments, enabling everyone to remain on the same page with project progression, via a solution that is easy for everyone to use."

Matthew Dowding
NewOrbit's Support Consultant

Whether an Agile team is distributed or working in one office, sourcing a transparent and simple to use online project management tool will assist in ensuring that business and development remain connected and collaborative throughout the project.

Agile Practice 4: Ensure constant communication between developers and business analysts by using workflows and solutions that all staff can access, including non-technical team members.

5. SUSTAINABLE PRODUCTIVITY:

Achieving team efficiency long-term

Within Agile, teams need to know how to work in a streamlined and productive way, but it's essential that teams avoid burning out.

Achieving sustainable productivity stems from two of the key Agile principles: Firstly, that "at regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly." Secondly that "Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely." Where these two Agile values intersect is the ability to develop workflows that enable sustainable productivity, long-term.

Reviewing Sprint Productivity

As mentioned before, Sprint retrospectives can help in not only realigning business goals with development tasks, but also in assessing how the sprint progressed and where bottlenecks, or other productivity challenges, emerged.

The software development team, AndPlus, review team efficiency after each sprint by judging three primary elements: client satisfaction, cycle time of particular backlog items, and QA kickback rate. In terms of reviewing and raising productivity, the second focus on cycle time and workflow is key:

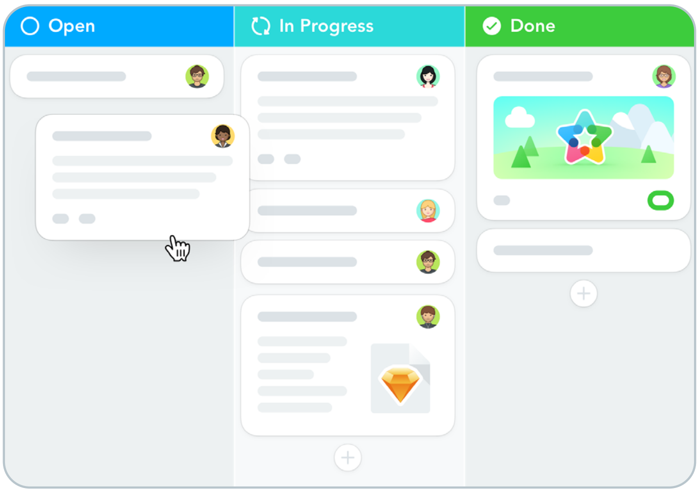
> "A useful productivity metric is cycle time - the amount of time a particular backlog item (user story, use case, task, bug, etc.) takes to get through your workflow...
>
> ...If you find items are sitting in your "In Progress" swimlane for a long time, look into why - what's the bottleneck? Improving cycle times are a good sign that your team is improving."

Jonathan Roger
AndPlus Operations Director

To assess where bottlenecks are being created, Agile teams can utilize the Kanban project management method. If you're new to Kanban, Kanban boards visually present a roadmap of a project, as tasks move through sections, classically including 'Upcoming', 'In Progress' and 'Completed'. Tasks are then dragged between these different stages until they are implemented.

Teams can of course name their stages however they wish, for example, if they'd prefer to name each stage by the corresponding team, with the task moving from design to development, they can name their sections as such.

With 'Work in Progress Limitations', teams can define a maximum number of tasks that may be added to any given section. This is done to enforce a steady, collaborative workflow and avoid the possibility of more tasks being opened than can be completed at the same time.



As a result, Kanban boards provide a means of gathering a visual overview of how tasks are progressing and whether bottlenecks are beginning to build up.

Creating Sustainable Workflows

A key step in ensuring that your team can retain a constant level of productivity is by ensuring that their workload is consistently manageable.

One contributing factor in this is ensuring that your team is not overloaded with features within each Sprint. The ultimate responsibility of this comes down to the Product Owner in their ability to be strict in not over-burdening Sprints. This involves being consistently open and honest with clients, when ensuring that not too many features are added, returning to issues already mentioned:

Firstly, it is the responsibility of a Product Owner to ensure that the project vision is defined from the outset and followed throughout the project. This vision should provide the Product Owner with the ability to be strict when recognizing that certain features could move too far from this project vision and are therefore unnecessary.

Secondly, it is also the responsibility of the Product Owner to ensure that these boundaries are communicated to the client, to prevent the team from being overloaded. By sharing the project roadmap and feature management with the client transparently, the Product Owner can demonstrate to the clients where the Sprint limits lie and, potentially, what would need to be given up, to make room for new features.

By not overloading the project (and your team) with unnecessary features, this will also help your final product to be more refined. Jói Sigurdsson, Founder of software company CrankWheel, found that cutting back on unnecessary features enabled their team to align their business and user priorities, and ultimately discover their product market fit:

> We found our product market fit through relentless sales pitches to prospective customers and then from studying customer feedback following sales," Sigurdsson shared. "We discovered that many of our intended features were actually unwanted by our prospects. We'd hear the phrase "We can turn off that feature" a lot, so we did, and ended up building a much simpler, more targeted product that we would have otherwise.

Jói Sigurdsson
Founder of Crankwheel

Once you have strong feedback from your user base on the essential features, a Kanban board can again assist with this process of feature management. By laying out all tasks within a project Sprint, and sharing this roadmap with your client, you can define a limit on the amount of feature tasks per Sprint and 'swimlane'. This will ensure that your project is not overloaded with tasks and that bottlenecks emerging in certain swimlanes will be recognized and addressed. In turn, you can ensure that your team's workload is manageable, and therefore sustainable, while ensuring that your end product is both refined and targeted.

Agile Practice 5: Utilize a Kanban Board to identify where bottlenecks are emerging and ensure that you are not overloading your team.

SUMMARY

Here’s a review of our five best practices for remaining Agile in software project management:

- Dedicate your first week of work to implementing a strategy sprint by designing, prototyping and testing your product, in order to establish your project vision.
- To ensure you’re able to stay within project scope and budget, begin by agreeing on a visual roadmap of key requirements with your client and then utilize the cone of uncertainty method to drill down on a realistic project estimate before development work begins.
- Establish a clear means of gaining regular, valuable feedback from both your client and end user. Then integrate this feedback at every iteration, while enabling your client to work transparently with you on feature management, via a shared collaboration tool.
- Ensure constant communication between developers and business analysts by using workflows and solutions that all staff can access, including non-technical team members.
- Utilize a Kanban Board to ensure that you are not overloading your team with tasks and to identify where bottlenecks exist.

Having seen how software development teams all over the world are implementing Agile and receiving fantastic results, we hope you have received some inspiration for how you, too, could be taking an Agile approach in your team.

ABOUT THE AUTHORS

STEFAN FORSTER

Stefan Forster is a value-driven product manager with more than 11 years of experience leading cross-functional teams to plan, build, launch and manage world-class apps and SaaS innovations. He combines technology skills with extensive Agile experience, a marketing orientation and analytical abilities to ensure products meet or exceed client expectations.

As a Product Owner and Scrum Master he has led all phases of diverse technology projects in the mobile games, gambling, video streaming and app development industries for clients like bwin.party digital entertainment and Red Bull. In 2016, he joined Meister as product manager for MeisterTask.

OLIVIA JARDINE

Since receiving her degree in Anthropology & History from the University of Sussex and the University of California, Berkeley, Olivia has worked in a number of marketing and communications roles, with a focus on project development.

With over four years of experience in project management specifically, Olivia has coordinated a variety of Agile projects, including three for the UK Government. To reach out to Olivia, connect with her via LinkedIn.

AGILE TASK MANAGEMENT FOR TEAMS

MeisterTask is an intuitive multi-platform collaboration tool that raises the standard in efficient—and even enjoyable—task management. Using MeisterTask, teams organize and manage tasks in a beautifully designed, customizable environment that perfectly adapts to their needs.

Project boards can be set up and modified to support software sprints, Kanban boards, and many other agile workflows, making them suitable for virtually all departments and industries. Tasks offer all the information needed for team members to complete them efficiently, and provide a space for real-time communication.

Try MeisterTask for free, at www.meistertask.com