THE WAY OF WORKING

Release 1.0

CONTENT

»ONLY THREE THINGS HAPPEN NATURALLY IN ORGANIZATIONS: FRICTION, CONFUSION, AND UNDERPERFORMANCE. EVERYTHING ELSE REQUIRES LEADERSHIP.«

PETER DRUCKER

THE WAY WE WORK

This booklet describes our way to run software projects. It is not a manual or recipe, but rather a structured collection of our experience, which tells you a bit about what we expect of you, as our partner. Agile Software Development is about leadership, and the purpose of this book is to allow both of us to focus on the task at hand, by empowering our teams to deliver a great product.

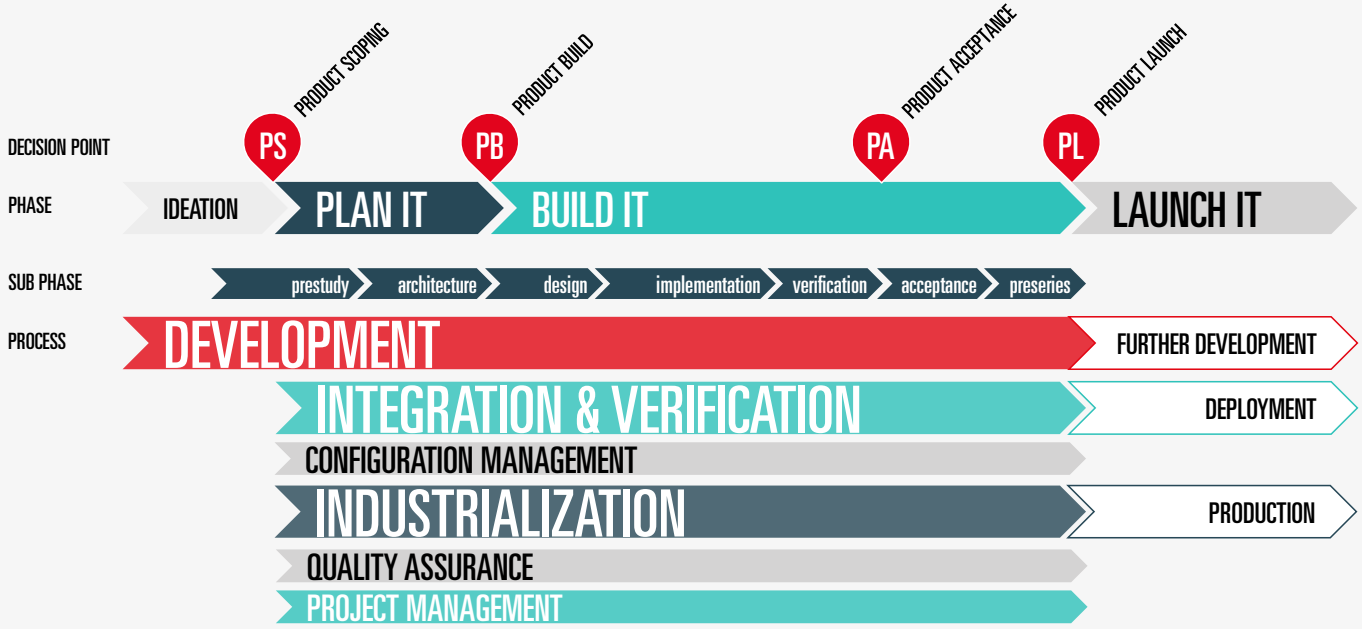
Amer Hadziomerovic & Daniel Thysell, Stockholm, March 2015.


SCOPE
QUALITY
TIME
COST

PROJECT IMPLEMENTATION

We assume a Product Development Model for all project deliveries. The Product Development Model is used as a support to succeed in delivering the right quality within scope, cost and time. The primary challenge of project management is to achieve all of the project goals and objectives while honoring the preconceived constraints.

Project deliveries are quality assured in accordance with our quality assurance system, certified with SS-ISO 9001:2000, environmental assurance system, certified with SS-ISO 14001 and information security assurance system, certified with SS-ISO 27001.


PRODUCT SCOPING
PRODUCT BUILD
PRODUCT ACCEPTANCE
PRODUCT LAUNCH
DECISION POINT
PS
PB
PA
PL
PHASE
IDEATION
PLAN IT
BUILD IT
LAUNCH IT
SUB PHASE
prestudy
architecture
design
implementation
verification
acceptance
preseries
PROCESS
DEVELOPMENT
FURTHER DEVELOPMENT
INTEGRATION & VERIFICATION
DEPLOYMENT
CONFIGURATION MANAGEMENT
INDUSTRIALIZATION
PRODUCTION
QUALITY ASSURANCE
PROJECT MANAGEMENT

DECISION POINTS

The following decision points are defined for a project:

- PS Initiate Product Scoping
- PB Initiate Product Build
- PA Initiate Product Acceptance test
- PL Initiate Product Launch

DELIVERY PLAN

The overall strategy for the development of the system is that it will be done using agile methodologies and Scrum. The development will be done in two week sprints. After each sprint a working software will be delivered.
The project will have the following phases:

- PLAN IT Initial scoping
- BUILD IT Development and test
- LAUNCH IT Deployment and continuous development

DELIVERY PLAN

PS PRODUCT SCOPING

PLAN IT

Initial Scoping

The first phase consists of scoping the work in a way that makes it possible to start building the product, as well as focusing on elimination of any non-technical risks.

Goals

- Set limitations and exceptions of the team and the product.
- Produce a well-structured backlog.
- Setup of all necessary infrastructure such as development tools, work areas and test tools.
- Create and approve a system test plan.
- Create and approve a system architecture specification.
- Perform a risk analysis.
- Create and approve system and sub-system requirement specifications.

PB PRODUCT BUILD

BUILD IT

Build it – Development & Test

Product development is done in an iterative process in two week sprints to deliver stories from the backlog. This phase is done when the stories remaining in the backlog are deemed to not add any more value to the product.

Goals of each sprint

At the sprint planning a goal for each sprint is set.

Deliverables

- Working software.
  This is ensured by testing.
- Implemented stories are documented and tested.
- List of remaining stories. Remaining stories will be pushed back to the backlog for consideration in the next sprint planning.

Acceptance Test

During the acceptance test the project deliverables will be tested and integrated by the customer's engineers to validate the functionality of the system. These tests will follow the test plans and test specifications as agreed upon in the initial scoping. The forms for this part will be agreed upon during the initial scoping and form the grounds for MS PL.

Deployment and Continuous Development

When the product is accepted it is ready for launching. This phase will include work of setting up production environment as well as maintaining the product and developing new features. When developing new features, the same iterative process as the development phase is used.

METHOD

Self-organizing Team

To solve all the tasks and issues that the team will face, the team will have to act as a self-organizing unit. Therefor the team will consist of skills from several areas to obtain a cross- functionally skilled team.

Competence

The following competence is required of the team.
One person in the team can possess one or more of these skills.

Software design. This skill means that the person has experience building software systems in a wide variety of environments.

Software development. This skill means that the person is an accomplished developer and have full professional competence in the tools required.

Testing. This skill means that the person is experienced in testing and validation of software functionality, is also experienced in automation and final testing.


SIGMA
SIGMA

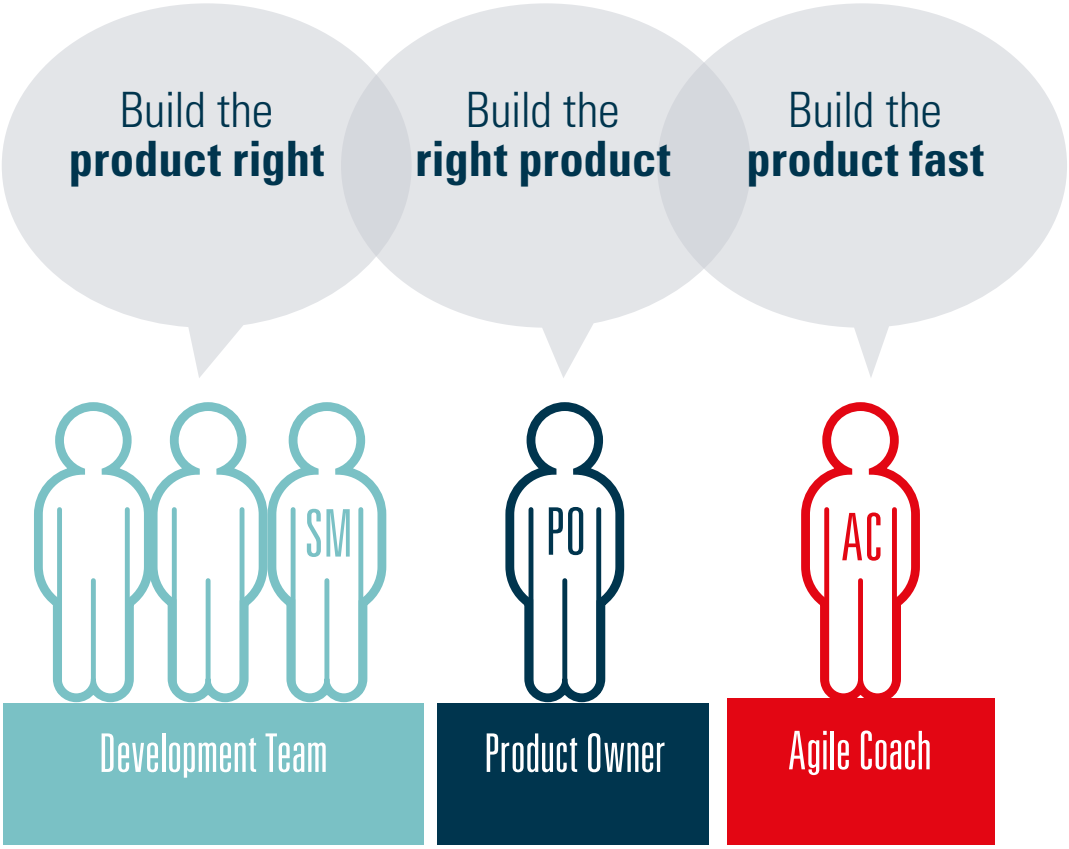
ROLES

To get all the skills to cooperate and work as a unit the following roles are defined.

Development team. The development team is jointly responsible for the product being tested, has the right quality and is built correctly. Each development team has an appointed **Scrum Master (SM).** SM is part of the development team and is responsible for the scrum process and the teams impediments. SM is also the person responsible that the team has the expertise and resources needed to complete the task.

Product Owner (PO). PO is not part of the development team and is appointed by the customer. PO shall be a person with extensive experience of the product and how it is used in the real world. PO is responsible that unspoken requirements on the system are communicated to the development team. PO owns the product backlog and is responsible to prioritize it. PO also makes sure that each item in the backlog is defined. PO should be available for the team to easily answer any questions about specific solutions. PO is responsible for the product having the desired functionality.

Agile Coach (AC). AC is not included in the team but helps the team by making sure that the team uses all the tools available to them, follows best practices and continuously improve their way of working.


Build the
product right
Build the
right product
Build the
product fast
SM
PO
AC
Development Team
Product Owner
Agile Coach

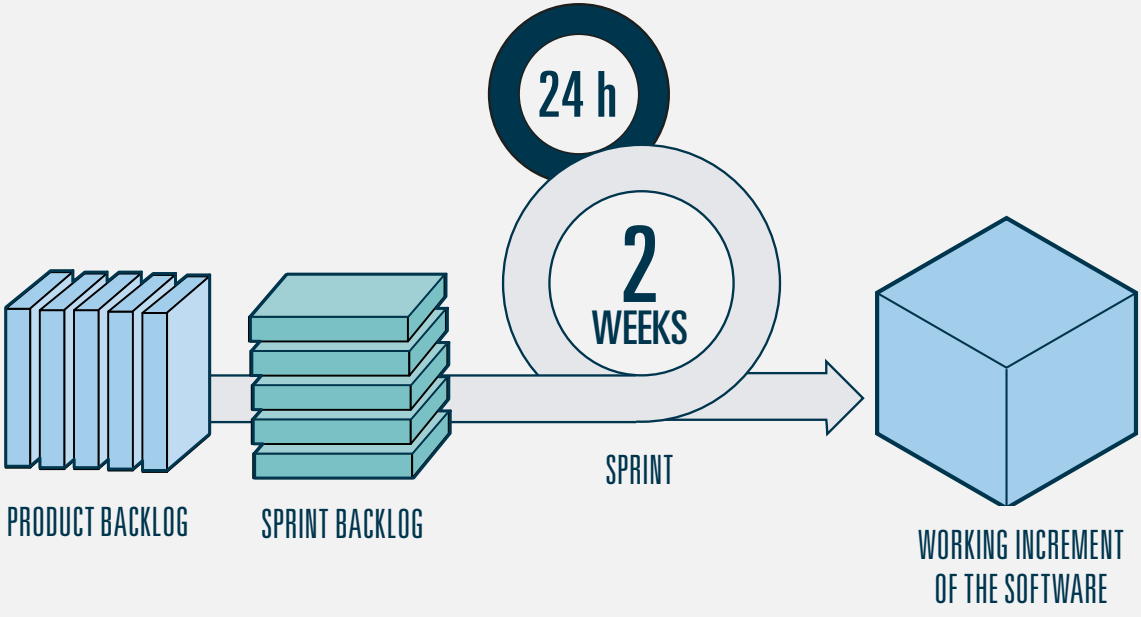
PRODUCT BACKLOG

The product backlog is a document that from a user perspective describes the functionality desired of the product. Every part of the product backlog is called a story and looks as follows.

PO is responsible that all columns except "Estimate" is properly described, the SM is responsible for estimates produced for each story.
PO and SM is jointly responsible that the team understands every story, which is why "Definition of Done" is important. It basically constitutes the acceptance criteria for a particular function.

The development team will work with the product owner to break down the product backlog to work packages in the form of sprints.


PO
PO
PO
SM
SM
PO
Title
Description
Priority
Estimate
Definition of Done
S
T
O
R
Y



SCRUM PROCESS

The work will be done iteratively in the form of sprints, each sprint is two weeks. The following activities are included in a sprint.

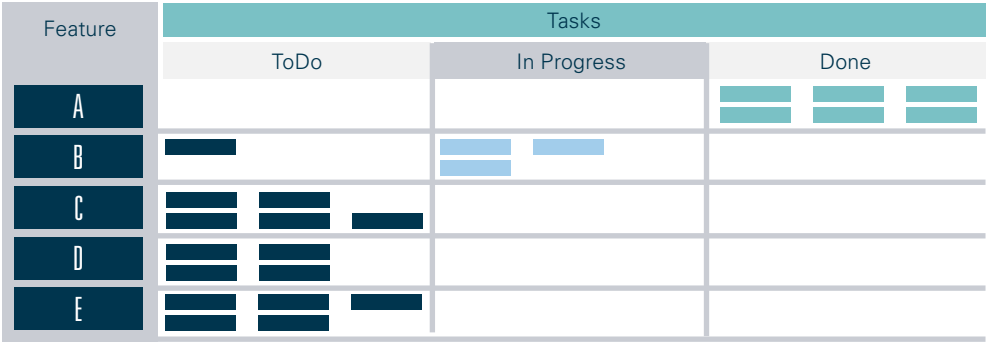
SPRINT PLANNING

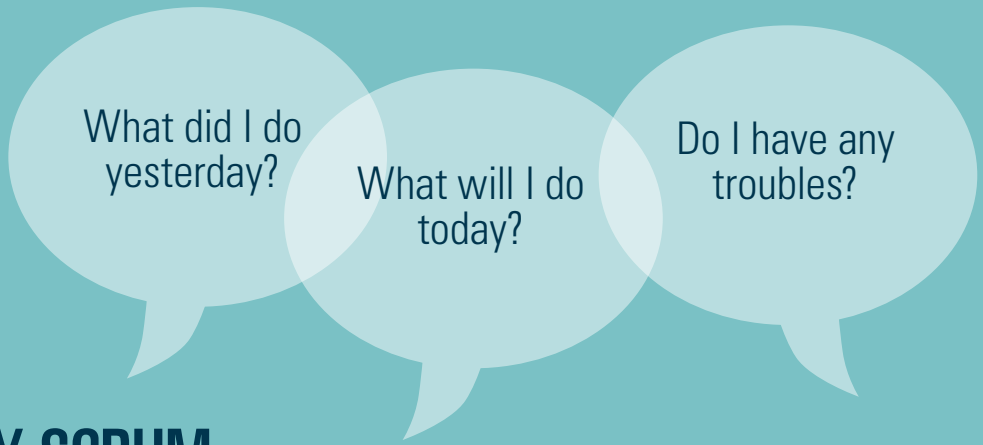
A sprint starts with a sprint planning meeting where the team together with the PO selects a number of stories from the backlog to work on during the upcoming sprint. PO should in advance have prepared this meeting by going through the backlog to ensure that the priority of all stories is desired and that they are clearly defined.

The results from the sprint planning is a well-defined work package that the team believes that they will be able to complete during the next sprint. This includes breaking down each story into tasks that will be have to be completed. Usually the first task to be defined should be an acceptance test to satisfy the DoD-criteria.

SCRUM BOARD

For every sprint a scrum board will be organized, with the high priority stories at the top. A Story is done when all the tasks for the story is done.





DAILY SCRUM

The daily scrum is a short stand-up meeting scheduled every morning where each team member answers three questions.
Any questions regarding specific solutions that does not concern the whole team is pushed until after the daily scrum. The SM is responsible for time management of this meeting. Should be no more than 15 minutes.

BACKLOG GROOMING

Backlog grooming is an activity scheduled by the PO in the sprints during which the team leaves the sprint work and focuses on defining stories further down in the backlog. Working to make the future easier by making sure we have all the information we need. The grooming process aims to start breaking down stories into tasks, prioritizing and re-estimating the size of the story.

BUGBASHING

Bugbashing is an optional part that aims to minimize the amount of confirmed bugs in the system. For example the SM can schedule Bugbash every Tuesday morning to lunch where no work is allowed on the backlog but only on confirmed bugs.

DEMO

At the end of each sprint the team will hold a demo of the implemented features for the PO and any other interested parties. The demo should focus on the newly implemented stories and be kept as short as possible. Preferably all the acceptance tests for each story is run. The PO and other stakeholders should offer feedback on the system and preferably test it themselves if possible.

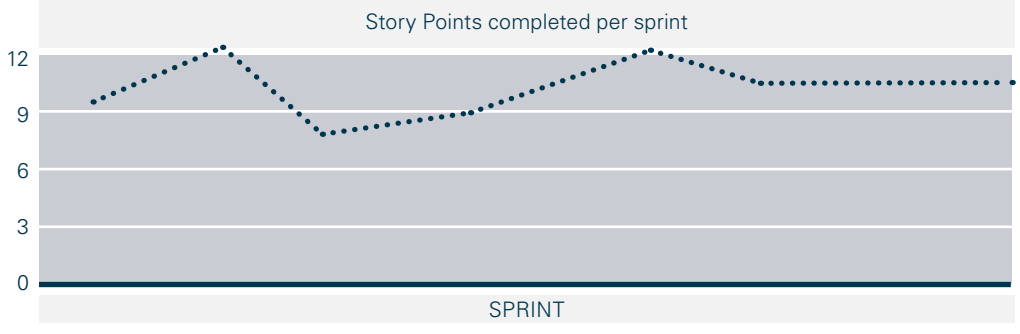
RETROSPECTIVE

The retrospective is an activity that the SM organizes for the team. This activity should result in a complete review of the last sprint and focus on how to make an even better job the next sprint. Where the demo focuses on "This is what we have done" the retrospective focuses on "This is how we did it, how can we improve?".

SPRINT TIMETABLE EXAMPLE, 10 DAY SPRINT

FORECAST

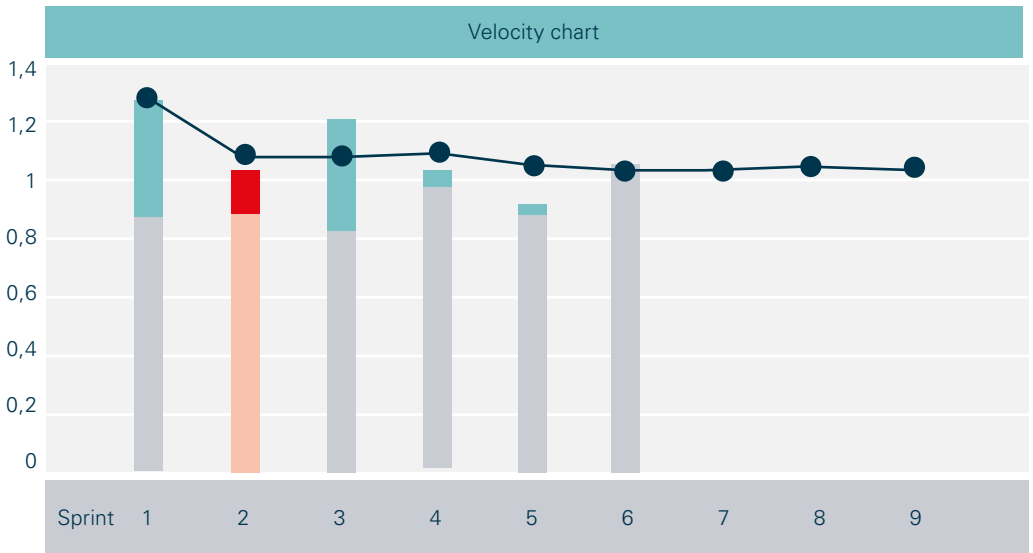
The easiest way to forecast deliveries is to measure the team velocity. The team velocity is a measurement of how many stories the team is able to complete each story. The following graph shows an example teams velocity through a couple of sprints. The size of a story is measured in story points (SP).



The team has done a minimum of 8 SP and a maximum of 12 SP per sprint. This means that if the backlog is 100 SP big, then the team will have finished the backlog in 8-13 sprints.
If the question is how many features will be completed in four sprints the answer is 32-48 SP.

VELOCITY CHART

Below is an example of a velocity chart. The darker colors represent the change of scope during the sprint. The green are successful sprints with increased scope and the red is failed sprints and the difference towards the planned amount. This particular graph has story points normalized to man days.



PRIORITIZE AND FOCUS – TO MAXIMIZE RETURN ON INVESTMENT

Prioritaztion is central to the agile process to allow the team to work on the most important features first. We focus on the current issue and not on serveral at the same time to ensure that we finish the most important first. This allows for flexibility and ability to change priorities since feedback can be gathered faster.

TRANSPARENCY

We value showing what we are working on at any given point and feel that transparency is the best way to build trust. We incorporate methods to make progress and current status easily accessible and understood by project stakeholders outside the team.

WALL BOARD – JIRA

JIRA allows for easy overview of sprint progress, management of product backlog and quick status updates of what is going on. Below is an example of a scrum board in JIRA

SPRINT PLAN

The sprint plan is condensed into a page that allow for quick overview of the plan for the current sprint. After each sprint a Sprint review document is created. This document includes the feedback on how the latest sprint went, along with updated velocity charts. This document also includes the actions decided on during the sprint retrospective.

SPRINT 4 –PLAN

SPRINT GOAL

➤ Users, bootloader and bug-fix

SRINT BACKLOG (estimate)

PROJ-75	Fix bug when using the buttons	–
PROJ-41	Develop a log-in	15
PROJ-51	Create a user model	5
PROJ-52	Create user access model	1
PROJ-53	Firmware for USB	10
PROJ-66	Bootoader	5

SCHEDULE
Period: 16/2–27/2
Demo: 27/2, 13.00 @ The Cafeteria
Daily scrum: 08.45 @ Team room

TEAM
Daniel T (SM)
Daniel S
Christian

ESTIMATED VELOCITY
36

Current Status – JIRA

ITERATIVE AND INCREMENTAL DEVELOPMENT CYCLE

We want to receive feedback on the implemented features as soon as possible. We do that by delivering functional software often. By building the software in small increments, we allow for better decisions about the future development.

COOPERATION

To deliver functionality requested by the market the development team have to be in constant communication with the business development team. By working together and finding a common language to understand each other, the possibility to build the correct functionality is greatly increased. We value developers who understand and are interested in the business side of the product development and encourage them to interact with project stakeholders. This increases knowledge of the system and allows the development team to make better decisions.

ENCOURAGE AND WELCOME CHANGE

Since the world is ever changing the product development must change too, to ensure that the resulting product is modern and what the market is asking for right now. By using short and incremental development cycles we ensure that the cost for changing direction is as low as possible at all times.

SIMPLE TOOLS

We use tools that support the development team, not teams that support tools. We have a toolbox of development tools that are available and possible to tailor to specific needs.

Scrum board – for issue tracking and Agile management, can be done via a whiteboard or JIRA.

Revision control systems like Subversion and Mercurial – for Source code and version management

Unit test framework – for quality assurance

Continuous integration servers like Bamboo and TeamCity – for managing Continuous Integration and running tests

CONTINUOUS IMPROVEMENT

We believe that there is always room for improvement on any process and team. We make sure to have continuous meetings that review what is functioning well and what needs improvement. It is also the way we lift problems right away to get the out of the way.

HIGH QUALITY

Errors and poor design are costly to fix, so we minimize this. We can never promise that there are no defects on the system, but we have processes to minimize the risk of defects. We differ on internal quality and external quality.

Internal quality is each and every developer's responsibility to maintain a high enough standard. By internal quality we mean the format of the code, design patterns and the actual design. We use techniques like pair-programming, code review and unit tests to make sure that the design is good enough.

External quality is everything that is perceived by the users of the system. Matters of external quality is a business decision if it will be launched or not.

An example might be if a user interface is not very user friendly. That is external quality, and it might be good to launch the feature anyway because you want to test other features. However the code foundation that the user interface is built upon is a matter of internal quality.

»THE CODE FOUNDATION THAT
THE SOFTWARE IS BUILT UPON
IS A MATTER OF INTERNAL QUALITY«

MANAGEMENT BY OBJECTIVES

Any decision should be made where it has the best chance of being the correct decision. For example the development team should be allowed to make decisions regarding development and design.

DEFINITION OF DONE AND DEFINED

We use a systemw where we have a checkpoint for backlog items called Definition of Defined that must be passed before the backlog item is allowed into a sprint. This definition is agreed upon by the development team and the product owner and ensures that the team fully understands the product owners' vision for a certain item. The scope of this definition can vary between projects but usually includes at least an acceptance criteria and an acceptance test.
The definition of done is also agreed upon by the development team and the product owner. This definition includes at least a test report from the acceptance test.

TECHNICAL STORIES

We usually like to treat technical stories, like setting up a automated build server, the same as user stories. This way we will prioritize them together with all stories in the product backlog.

SIX GOOD REASONS TO WORK WITH US

Sigma Technology has more than **20 years of experience** in providing services within R&D.

Sigma Technology is more than just an ordinary consultancy provider. We annually deliver more than **850,000 hours in multi-site Managed Services**.

Sigma Technology is a **trusted strategic partner** supporting our clients around the world to develop, improve, and implement new products in fully-tailored solutions.

Sigma Technology has been chosen as one of the preferred suppliers by many global leaders in different industries.

Sigma Technology's **global presence** ensures the competence and capacity to offer high-quality local insights with a global delivery capability. We call this **Local Drive – Global Strength**.

Sigma Technology is a multicultural company that attracts **smart, cooperative, and professional** experts from all over the world ensuring long-term partnerships with our customers.

WE ARE TECHNOLOGY Sigma Technology is part of Sigma, and is a global supplier of product information, software and embedded solutions, and offshore development. We are experts with a passion for technology and information, and take pride in delivering quality and constantly improving our deliveries.

Our philosophy is "Local Drive – Global Strength", therefore we have offices worldwide to be close to our customers, as this is the most effective way to work; both for us and for our customers. We have more than 800 highly-talented and dedicated employees. We have offices in Sweden, China, Hungary, Ukraine, and USA, as well as partner offices in India and Malaysia, and our head office is located in Göteborg, Sweden. Sigma Technology is a part of the Sigma Group – a leading consultancy group, with the objective to make our customers more competitive.

www.sigmatechnology.se

SIGMA Technology