

<https://www.halvorsen.blog>



Week Assignment

Software Processes - Agile/Scrum

Hans-Petter Halvorsen

Week Assignment

1. Create/Update Product Backlog Items and Tasks in **Azure DevOps** based on SDP and **SRD** documents
 - Create/Update the Product Backlog
 - (Create/Update the Sprint Backlog) We will make a simplified **Taskboard** on the Whiteboard for remaining Tasks in the Alpha Release. Starting from Beta we will use Azure DevOps 100%
2. Perform a Daily Scrum Meeting
3. Finishing the Alpha Release (Deadline Next Week)
4. **Quiz.** Extended and Important Quiz (30-40 Questions)
(Next Week: Sprint Review & Planning)



Software Processes

Agile/Scrum

Hans-Petter Halvorsen

[Table of Contents](#)

Typical Job Ad

Utviklingsjobb i programvarehus

Vi har for tiden stor etterspørsel etter dyktige .Net utviklere.
Er genuint opptatt av programmering og har relevant erfaring ber vi deg kontakte oss.

Kvalifikasjoner:
Minimum Bachelor med vekt på programvareutvikling
Ønskelig med noen års relant erfaring - solid relevant praksis og konkret erfaring kan kompensere for utdanning

Det er ønskelig at du har god kiennskap til:

- C#, .Net-rammeverk og ASP.Net
- Javascript
- HTML5,
- SQL databaser
- Smidige utviklingsprosesser (scrum etc)
- Gode ferdigheter i norsk og engelsk skriftlig og muntlig er nødvendig

Søknadsfrist: Snarest

Many Job Ads today ask for Agile/Scrum skills!
- And skills within Web, ASP.NET, C#, HTML5, SQL and Database Systems

Agile methods and especially Scrum have become very popular these days!!

More Job Ads

Erfaren og engasjert .NET Systemutvikler

_____ søker etter en systemutvikler med lidenskap og engasjement for teknologi og programmering til vårt utviklingsteam i Bodø. Du bør ha relevant erfaring fra systemutvikling men med de rette personlige egenskapene kan du også være aktuell dersom du er nyutdannet kandidat på master-/ bachelornivå.

Dette er kompetansen vi ønsker:

- Høyere utdanning på høyskole-/universitetsnivå innen IT.
- Erfaring med programmering/utvikling på løsninger på .Net/Java teknologi.
- Erfaring med utvikling/modellering av SQL databaser mot fortrinnsvis MS SQL Server eller andre relasjonsdatabaser.
- Fordel med erfaring fra webutvikling på ASP.NET Webforms/MVC og Visual Studio (C#).
- Kunnskap om web teknologier som XML, Web Services.
- Kjennskap til Scrum/Kanban, Testdrevet utvikling (TDD) og utvikling mot mobile enheter er et pluss.
- I tillegg til faglige ambisjoner må du være kreativ, ha evne til nyskaping og kunne jobbe i team.
- Gode norsk- og engelskkunnskaper, både muntlig og skriftlig er en selvfølge.

SYSTEMUTVIKLER .NET

På vegne av kunde søker Academic Work en fremoverlent og nysgjerrig systemutvikler!

OM BEDRIFTEN:

Vår kunde har spesialisert seg innen HR og Lønn. De leverer IT-løsninger og konsulent- og outsourcingtjenester. Selskapet har over 15 000 kunder fordelt på privat og offentlig sektor og over 45 års arbeidserfaring.

ARBEIDSOPPGAVER OG ANSVAR:

Dine primære arbeidsoppgaver vil bestå av å utvikle selskapets kjerneapplikasjoner (standard lønssystem). Du vil ha muligheten til å inngå i en helhetlig prosess med alt fra å forme, designe, teste og dokumentere data. Teamet du vil inngå i arbeider etter SCRUM modellen. På sikt vil det kunne påregnes noe kundekontakt også.

VI SØKER DEG SOM

- Er utdannet innen IT, bachelor eller master
- Er svært motivert til å bli en .Net-utvikler
- Gjerne har erfaring med .Net rammeverket og Visual studio
- Har noe kjennskap til WCF/WPF/WF/Javascript/HTML5

Som person er du utadvendt og har god driv! Du er løsningsorientert både teknisk og funksjonelt, og evner å sette deg inn i nye arbeidsmetoder raskt. Du takler å jobbe under tidspress og tar ansvar. Personlige egenskaper vil bli vektlagt.

Systemutvikler/ Løsningsarkitekt

Kvalifikasjoner

- Relevant utdanning på bachelor eller masternivå fra universitet eller høyskole. Lang relevant erfaring kan kompensere for utdanningskravet.
- Evne til å sette deg inn i nye arbeidsmetoder og teknologiske trender.
- Erfaring med utvikling ved bruk av moderne og smidig utviklingsmetodikk.
- Erfaring med utviklingsspråk som Java/Kotlin, PHP eller Python
- Erfaring med frontend-utvikling i Javascript og **webrammeverk** som React.js, Bootstrap eller lignende.
- Erfaring med API-utvikling, integrasjoner og mikrotjenester.
- Gode **samarbeidsevner**, og god muntlig og **skriftlig fremstillingsevne**.

Vi ser gjerne at du har

- God forståelse for **løsningsdesign og arkitektur**
- Erfaring med kodeoptimalisering og ytelse
- Kjennskap til søketeknologi og søkemotorer som Sphinx, ElasticSearch eller Solr
- Erfaring med **databaser** og databaseprogrammering (PostgreSQL, mariaDB)
- Kjennskap til eller erfaring med graf databaser (GraphQL, Gremlin, SPARQL)
- Erfaring med **anvendelse av ny teknologi**, inkludert kunstig intelligens og **maskinlæring**
- Erfaring med **DevOps**, automatisert testing, containerbasert utvikling med verktøy som Docker, Kubernetes, og skytjenester

Personlige egenskaper

Som person beskrives du som en **lagspiller**. Du har lett for å dele av din kunnskap og erfaring, er initiativrik og ser etter de gode løsningene i samarbeid med dyktige kolleger. Du er opptatt av å holde deg oppdatert på trender og teknologi.

Traditional Plan-driven vs. Agile Software Development Processes

Traditional/Plan-driven

"Cannot" go to next phase before previous phase is finished

Waterfall

Spiral

V-model

RUP

...

Rational Unified Process

Agile

More flexible, Iterative and Incremental Approach

Scrum

eXtreme Programming (XP)

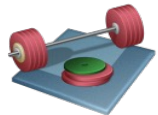
Lean

Kanban

FDD

TDD

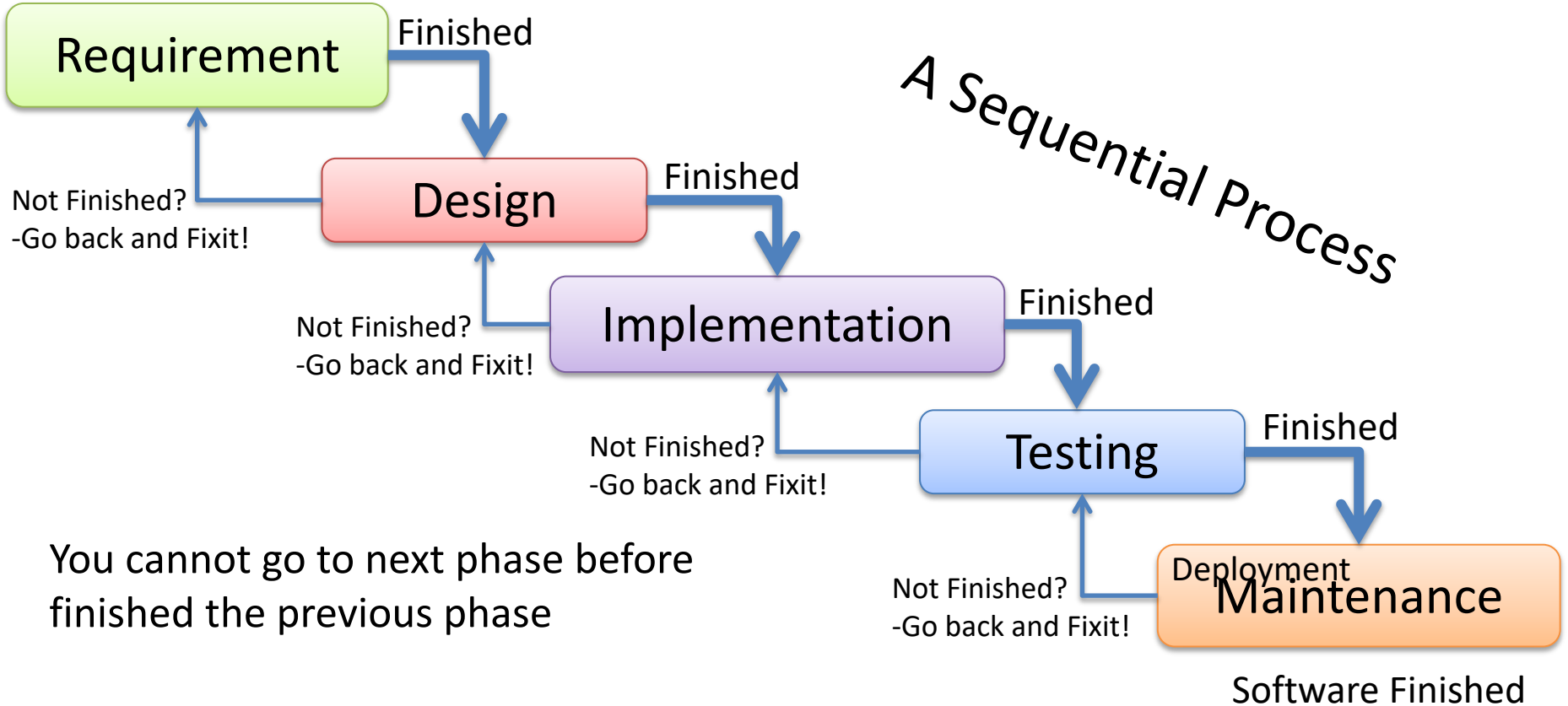
...



We will mainly be focusing on **Waterfall** and **Scrum** in Class. But you should read and get an overview about all these Software Development Processes in the Textbook!

The Waterfall Model

Planning to create a new Software



Agile

(Norsk: Smidig utvikling)

Customer
Involvement

Pair
Programming

Test Driven
Development (TDD)

Incremental
Work in Iterations

Less Documentation.
Only what's necessary

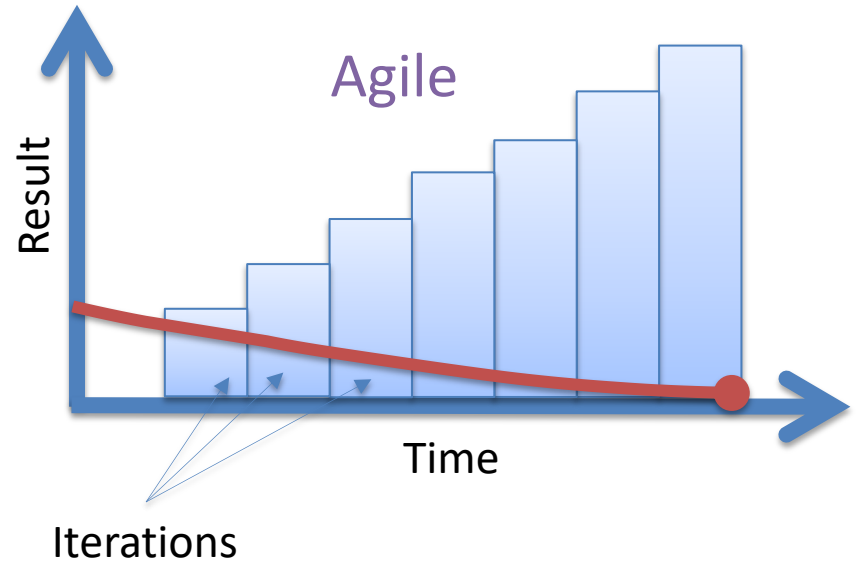
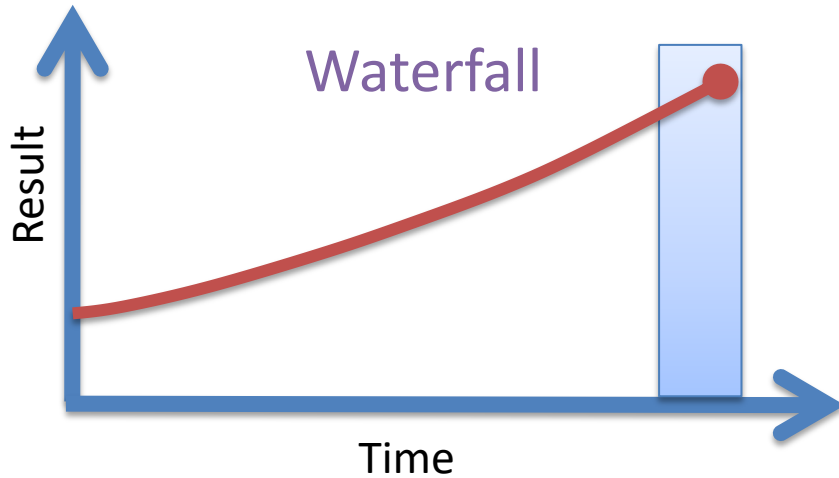
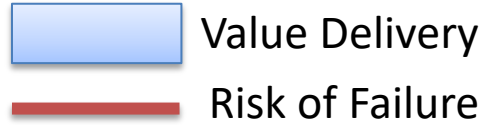
Continuously
Integrate Changes

Communication

Refactoring

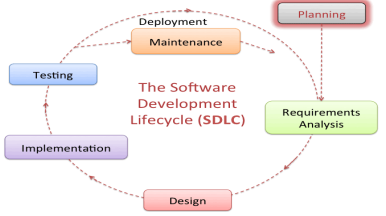
Working Software
at All Times!!

Waterfall vs. Agile

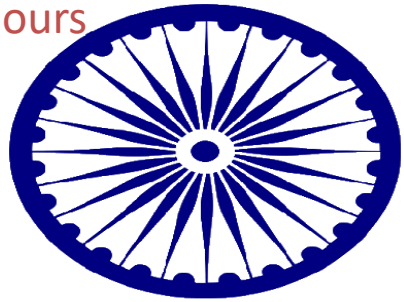


Agile/Scrum Development

Daily Scrum Meetings

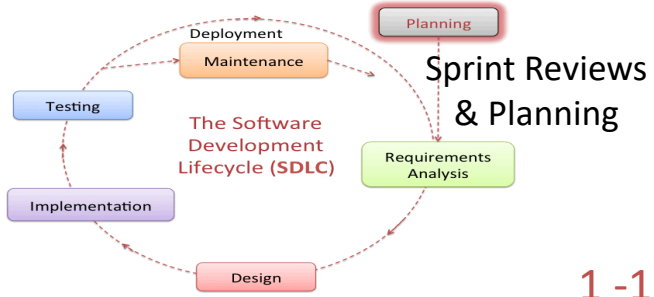


24 hours



Days

Working Software at all times!
Testing every day



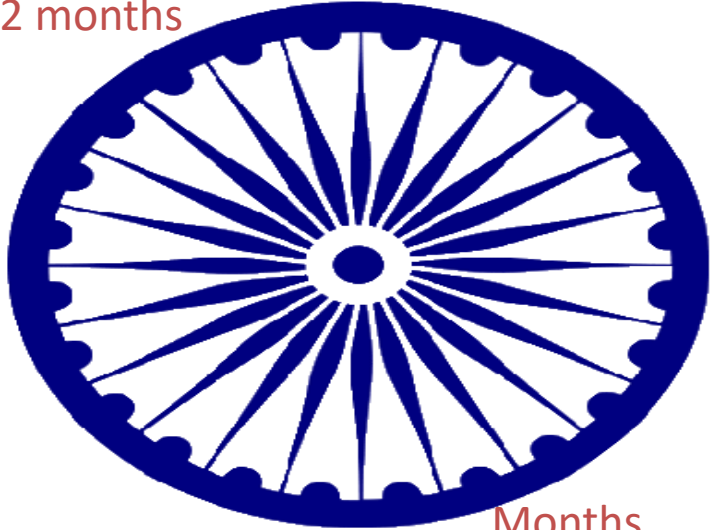
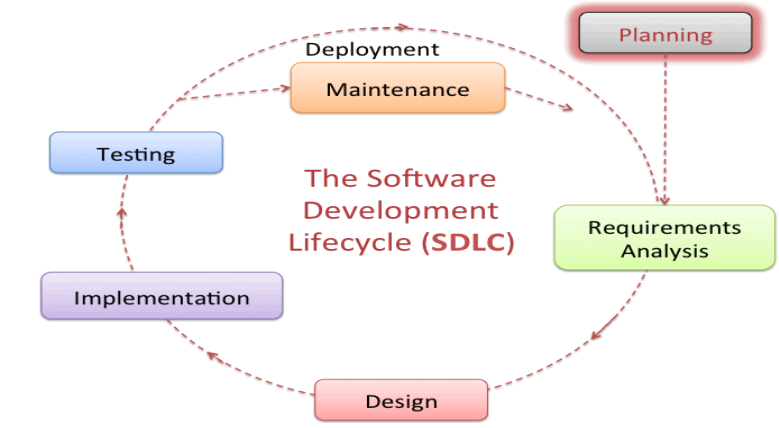
2-4 weeks



Weeks

Iterations/Sprints

1 -12 months



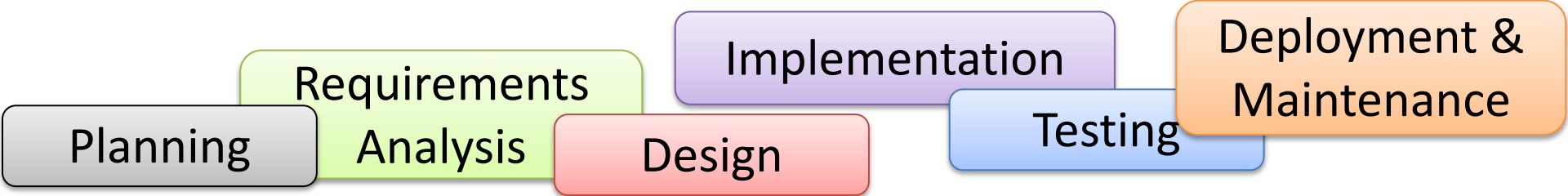
Months

Finished

Which Process are we using in the Project?

- We will use different Development Processes throughout the project
- The goal is to learn about different methods and test out some of them in real-life.
- You should be able to know the main features and be able to choose and apply a proper method for a specific project
- We should know advantages and disadvantages of different methods
- In practice, you typically use a mix of different processes that together suits your needs

Which Process are we using in the Project?



We are using:

Waterfall Waterfall Scrum Waterfall + Scrum mix



We create different documents

We create Product Backlog and use Taskboard



Introduction to Scrum

Hans-Petter Halvorsen

[Table of Contents](#)

Video: Scrum in 2 minutes



Scrum in 2 Minutes:

<https://www.youtube.com/watch?v=WxiuE-1ujCM&feature=youtu.be>

Scrum Overview

Scrum Members:



Stakeholders



Product Backlog



Product Owner



Scrum Master



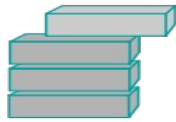
Sprint Backlog

Development Team

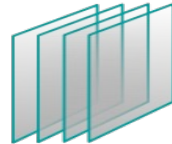
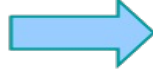
3-9 persons

Scrum Process:

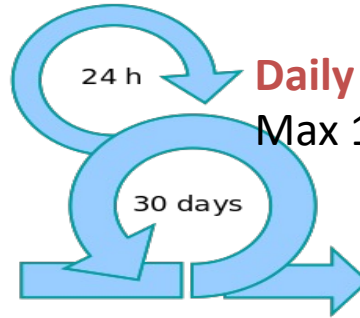
Sprint Planning Meeting



Product Backlog



Sprint Backlog



Sprint

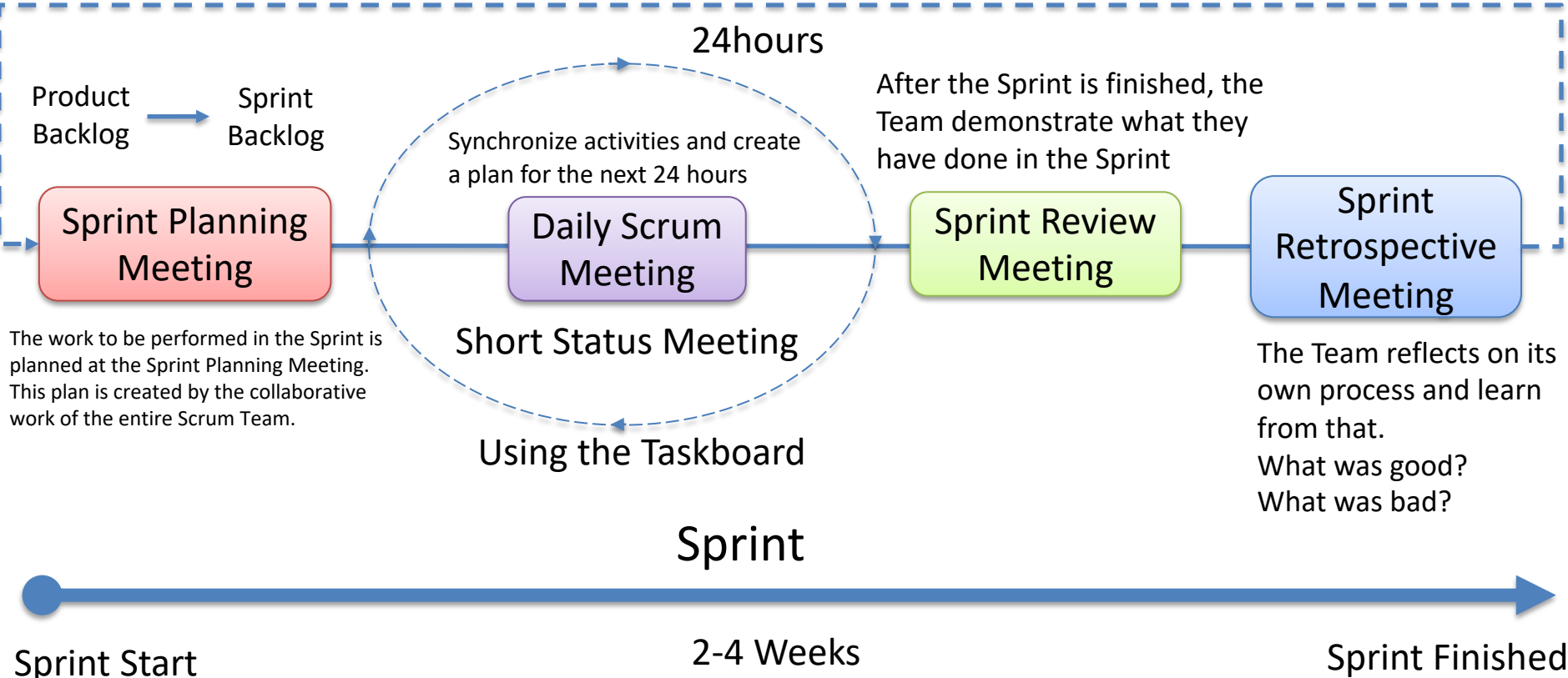
Sprint Review



Working increment
of the software

We will practice these meetings during the course

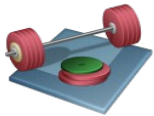
Scrum Meetings



The Official Scrum Guide



<http://www.scrumguides.org/scrum-guide.html>



Make sure you read this (at home) – This guide gives a very good overview of Scrum! The Scrum Guide is available in many languages



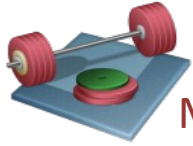
Product Backlog Items and Tasks

Product Backlog Items and Tasks

- Create/Update your Product Backlog Items (PBI) and Tasks in DevOps
 - Use the SRD document as a foundation for creating the Product Backlog Items and Tasks
- Create/Update your Sprint Backlog for the Alpha release/iteration
 - Make sure to include Responsible Person and Estimated Hours
 - Class Exercise: We will also make a simplified Taskboard on the Whiteboard using Yellow Sticky Notes to get started

See Next Slides for more details...

Sprint Task Board Example



Make a Taskboard for Remaining Work in “Alpha Release”

We divide the Whiteboard into subparts and each Team gets some Sticky Notes they can write on. Include Task Name, Responsible Person and Estimated Hours

To Do

In Progress

Done

Create Web Interface

HPH

4h

Create Database

OD

7h

Create UML

NOS

3h

Create GUI

HPH

8h

GUI Mockup

OD

6h

Task Name

Responsible Person

Estimated Hours

The Taskboard has to be updated on a daily basis

Start using Scrum in your Project

Start using Scrum within your Project:

Give access to and send the Project URL on e-mail to the supervisors

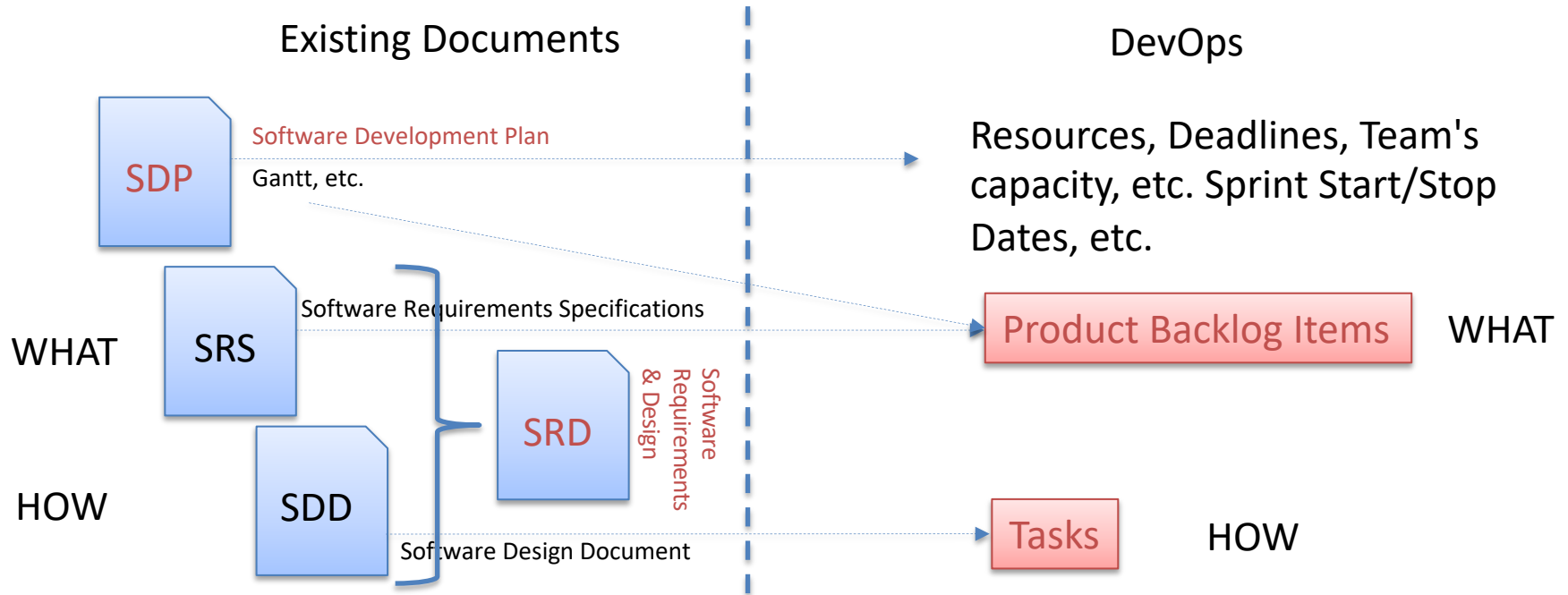
- Create/Update **Product Backlog**
- Create the different Iterations for the entire Project with Start and End Dates
 - Start Create Sprint Backlog for the Next Iteration (Beta Release)
- Start using the Taskboard

Cont. Next Week

See Next Slides for more details...

Product Backlog

Create/Update Product Backlog Items and Tasks in DevOps based on SDP and SRD documents



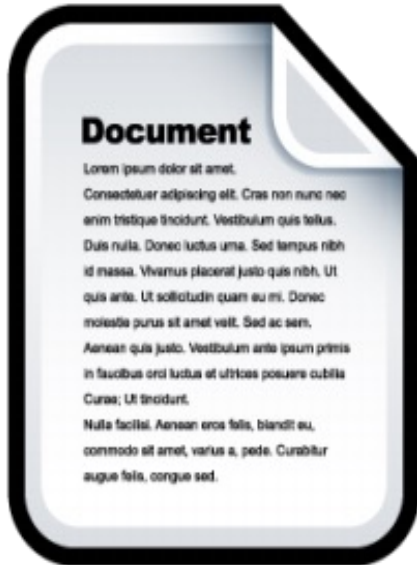


Product Backlog, Sprint Backlog

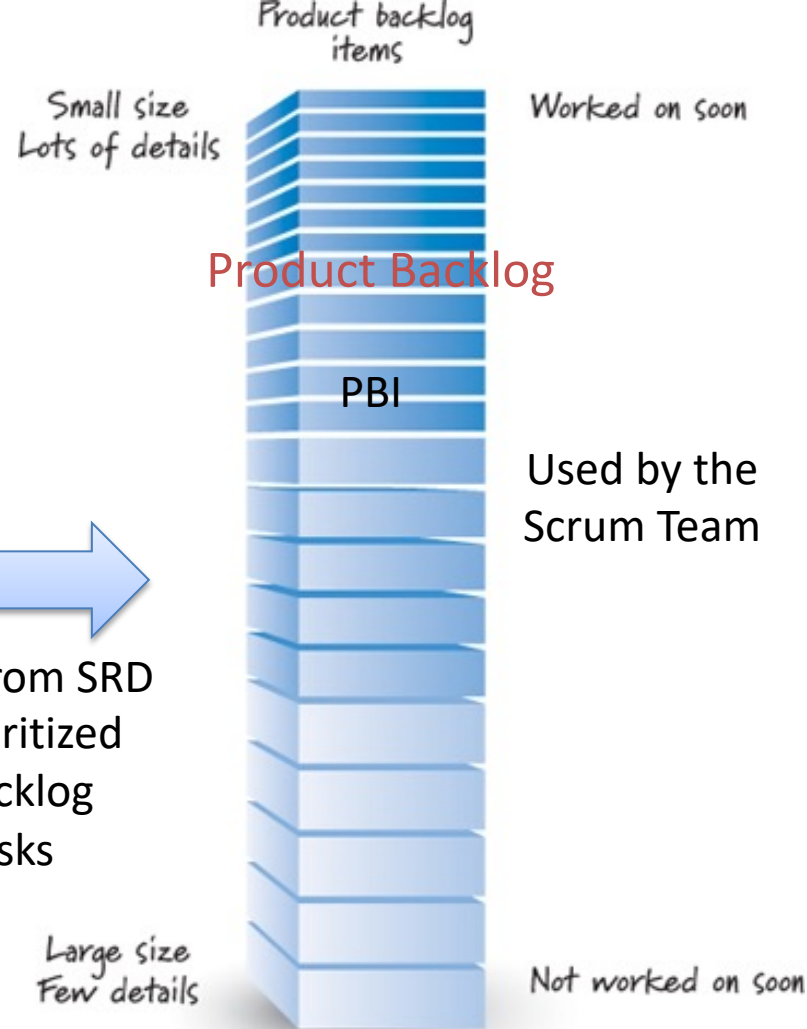
Product Backlog

Software Requirements and Design (SRD)

Used by the Customer, Managers, etc.



Create all Requirements from SRD into Backlog Items (in prioritized order). Then break the Backlog Items into Manageable Tasks



The Product Backlog in DevOps

Azure DevOps

software-usn / MySoftware / Boards / Backlogs

Search

MySoftware

Overview

Boards

Work Items

Boards

Backlogs

Sprints

Queries

Repos

Project settings

MySoftware Team

New Work Item View as board

Backlog items

Order	Work Item Type	Title	State	Effort
1	Product Backlog Item	The system should store the data in a Dataabse	New	
	Task	Create SQL Server	To Do	
	Task	Create Tables	To Do	
2	Product Backlog Item	A Web Application should be created	New	
	Task	Install Visual Studio	To Do	
	Task	Learn ASP.NET	To Do	
3	Product Backlog Item	The System should be properly documented	New	
	Task	Create User Manual	To Do	

Product Backlog Example

Order	Work Item Type	PBI	Title
▼ + 1	Product Backlog Item	▶	▶ Patient Data should be saved
	Task		▶ Create GUI for patient Data
	Task		▶ Create Class for Saving Data
	Task		▶ Create Database Table
2	Product Backlog Item	▶	▶ The Nurse needs Medical Information
	Task		▶ Create GUI that list Medical Prescriptions for a specific Patient
3	Product Backlog Item	▶	▶ The Doctor should get Reports
	Task		▶ Create Search GUI
	Task		▶ Create search Class and Methods
	Task		▶ Create Report "Get Patient Data"
...			

A prioritized list of all Features/Requirements for the System to be made

The Sprint Backlog in DevOps

Visual Studio Online / Software Development

HOME CODE **WORK** BUILD TEST*

Backlogs Queries

Features

Backlog items

Current

- Sprint 1

Future

- Sprint 2
- Sprint 3
- Sprint 4
- Sprint 5
- Sprint 6

Backlog items

Backlog Board

New [Add] [Remove] | Create query | Column options | [Refresh]

Type: Product Backlog Item

Title: [Input field] [Add]

Order	Work Item Type	Title	State	Effort
1	Product Backlo...	Create GUI for Web App	New	
2	Product Backlo...	Add Database Communi...		

Visual Studio Online / Software Development

HOME CODE **WORK** BUILD TEST*

Backlogs Queries

Features

Backlog items

Current

- Sprint 1

Future

- Sprint 2
- Sprint 3
- Sprint 4
- Sprint 5

Software Development Team Sprint 1

Backlog Board Capacity

[Add] [Remove] | Create query | Column options | [Refresh]

Title	State	Assigned To	Rema...
Create GUI for Web App	New		
Add Database Communication with Server	New		

Just Drag Items from the Product Backlog to the Sprint Backlog

Title



Create GUI for Web App

GUI Design

Add Database Communication with Server

Creating Tasks in DevOps

Break down Product/Sprint

Backlog Items into workable Tasks

Task 3: GUI Design

Tags Add...

GUI Design

Iteration Software Development\Release 1\Sprint 1

STATUS	DETAILS
Assigned To Hans-Petter Halvorsen	Remaining Work
State To Do	Backlog Priority
Reason New task	Activity
Blocked	Area Software Development\Web

DESCRIPTION

HISTORY LINKS (1) ATTACHMENTS

Save Save and close Close

Max 8 hours

How to

- Work**
Create and manage your product backlog in minutes
- Code**
Learn about working with your code
- Build**
Learn about builds in the cloud
- Test**
Learn more about Test Manager and test planning

Work

Backlog
Task board
Queries

Task Board

Sprint 1

0 h

Members



Visual Studio Online / Software Development

HOME CODE WORK BUILD TEST*

Backlogs Queries

Features
Backlog items

Current
Sprint 1

Future
Sprint 2
Sprint 3

Software Development Team Sprint 1

Backlog Board Capacity

Create query Column options

Title	State	Assigned To	Rema...
Create GUI for Web App	New		
Add Database Communication with Server	New		

Software Development Team Sprint 1

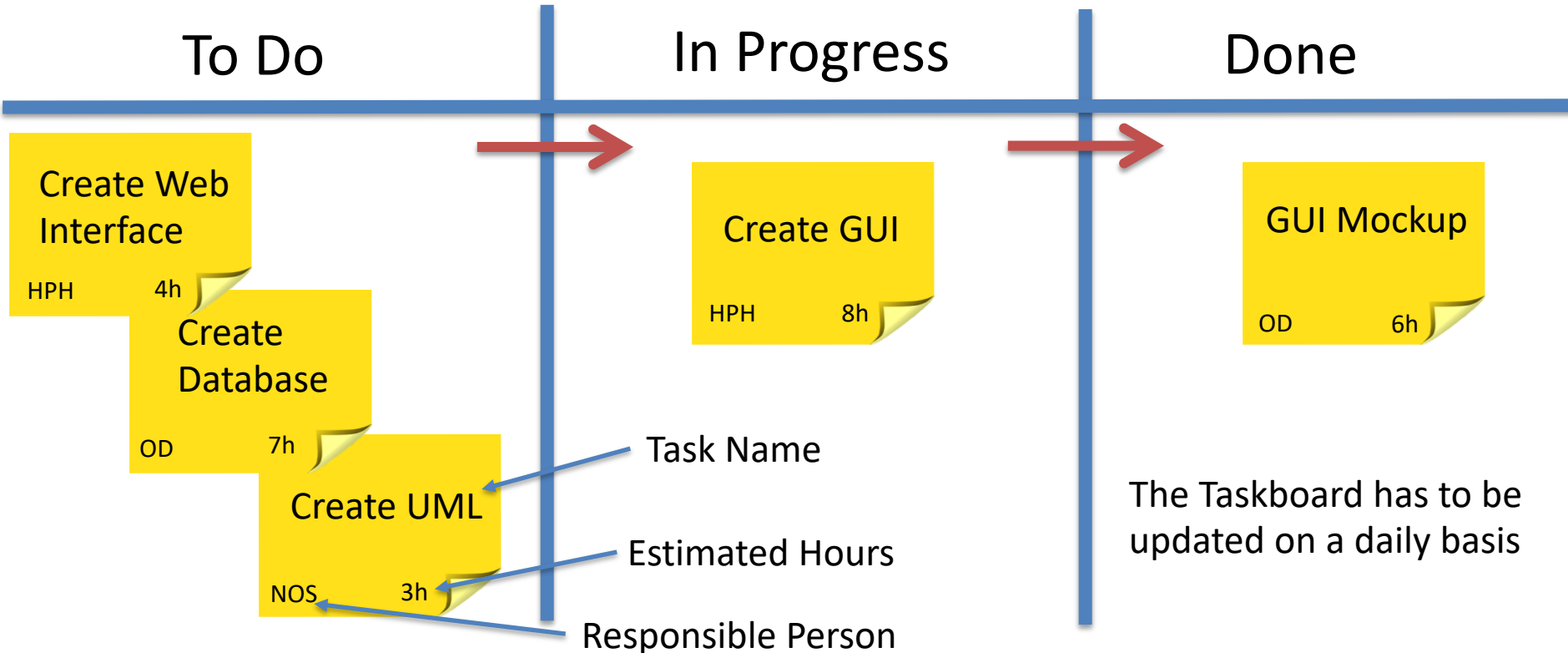
Backlog **Board** Capacity

Group by Backlog items Person All

	TO DO	IN PROGRESS	DONE
Create GUI for Web App	GUI Design Hans-Petter...		
Add Database Communication with Server			

Drag and Drop Tasks

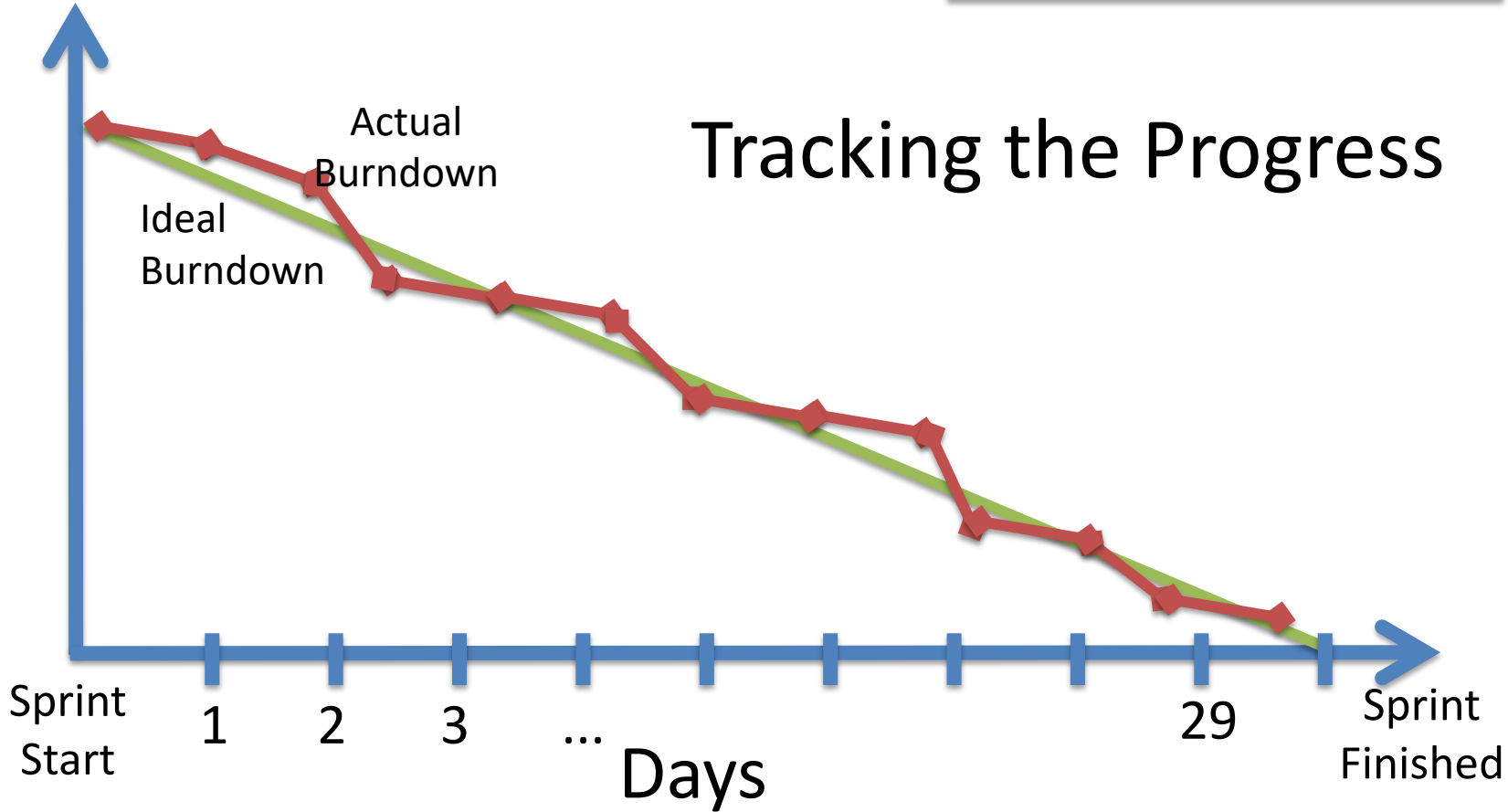
Sprint Task Board Example



Burndown Chart

Important Project
Management Tool in Scrum

Remaining Work (Hours)





Daily Scrum Meeting

Daily Scrum Meeting

- **Prepare for your first Daily Scrum Meeting**
 - We divide the whiteboards in subparts where you draw a simple **Taskboard** (3 columns: “To Do”, “In Progress”, “Finished”)
 - Create a set of remaining Tasks for the Alpha iteration on yellow Sticky Notes and place them on your Taskboard.
 - Prepare the “Answers” for the 3 Questions (“Have done?”, “Shall do?”, “Problems?”)
- **Perform your first Daily Scrum Meeting**
 - Each in the Team should answer the **3 Questions**
 - Move Tasks on the **Task Board** (“To Do” -> “In Progress” -> “Finished”)
 - Try also to create a “**Parking lot**” if necessary
 - Try also to draw the **Burn down Chart**

Video: Microsoft Daily Scrum Meeting



Microsoft Daily Scrum Meeting:

<https://www.youtube.com/watch?v=YR84qH6d7QE>

Product Backlog
Sprint Backlog

Taskboard



Daily Scrum Meeting



Daily Scrum Meeting

3 Questions:

1. What did you do yesterday (since last meeting)?
2. What shall you do today (until the next meeting)?
3. Any Problems?

Purpose with the Daily Scrum Meeting:

- Synchronize activities and create a plan for next 24 hours.
- Track Progress

Video: Daily Scrum Meeting – Bad Example



Daily Scrum Meeting – Bad vs Good Example:
https://www.youtube.com/watch?v=q_R9wQY4G5I

(we stop at 3:30 in class - watch the rest at home)

Daily Scrum Meeting - Summary



- Max **15 min.**
- Same Time and Place every day
- Don't be late
- No phones, no checking E-mails, etc. (No distractions)
- Each answer **3 Questions** (Don't go into details!).
- No others can talk while answering the questions.
- Update your Tasks and the Electronic Task Board before the meeting (if sticky notes, you can update the Task Board while answering the 3 questions)
- Stand up (don't sit down) during the whole meeting
- Show the Burn-down Chart at the end of the meeting

Task Board

Sprint: "Finishing Alpha Release"

To Do

In Progress

Done

Oppdater SRD med UML

HPH 2h

Legg inn PBI/Tasks for Database-utvikling

NOS 1h

Oppdater folderstruktur i Azure DevOps

NOS 0.5h

Opprett konto

HPH 0.5h

Installer FTP program for opplasting til Web Server

NOS 0.5h

Sjekker inn all dokumenter i Azure DevOps

OD 0.5h

Oppdater ER modell

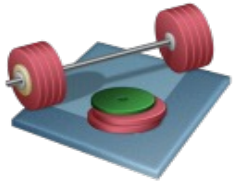
OD 1h

OSV.

OSV.

OSV.

Hver person på gruppa bør lage 3-10 gule lapper



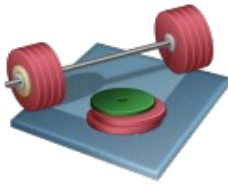
Daily Scrum Meeting

Max 15 Minutes!



- Select a **Scrum Master**
- Prepare for the **3 Questions** (Write the Answers/ Keywords down on a piece of paper)
- Use your Whiteboard **Taskboard** and have your first Daily Scrum Meeting with your Team.
- No Discussions or Problem Solving in the Meeting. Use the “sidebar/**parking lot**” to stay within the 15-minute time box (i.e., Write it down and take it right after the meeting)

Regular Daily Scrum Meetings



- Your Team should have regular Daily Scrum Meetings during your Project
 - We only work 33% so obviously you cannot have it every day – but you should at least have it once a week
 - It is the Scrum Master in your Team that is responsible to facilitate these meetings



Finishing the Alpha Release

Hans-Petter Halvorsen

[Table of Contents](#)

Finishing the Alpha Release

These are the minimal Requirements!

1. Documents: The following Documents should be “finished”:

- Software Development Plan (SDP)**
 - Product Description, Team Description and Project Organization, Gantt Chart, Tools and Templates, etc.
- Software Requirements and Design (SRD)**
 - System overview, System and User Requirements, **Functional and Non-Functional Requirements**, **GUI Sketches**, **ER diagram**, **UML Diagrams**
- SDP and SRD can easily be found in Microsoft Teams, and you have made a good folder structure

2. Software/Programming:

- Latest **Visual Studio** has been installed
- Tables Implemented in **SQL Server** (on everybody's Development PCs) – Generated from erwin Script
- You know how to **Communicate with your SQL Server Database from C# Code** (You have made a small examples to for storing and reading data from the database used in your project)
- Database API**. You have made one or more common Views/Stored Procedures that can be used by all the application. Sql scrips are in Azure DevOps
- You have started to use/made some **ASP.NET** (or WinForm) Database Communication (e.g., “Hello World App”, basic CRUD App, PoC, etc.)
- You have created **Classes** and Methods (according to the UML diagrams) - not the contents in the Methods, just the Names/declarations
- ALL Team Members have installed MS Project (or similar), erwin, SQL Server, Visual Studio, UML software (needed in project and exam!)
- ALL Team Members have started to do some Programming in Visual Studio!
- ALL Team Members have been given a clear responsibility (e.g., a separate module or application) when it comes to Programming!

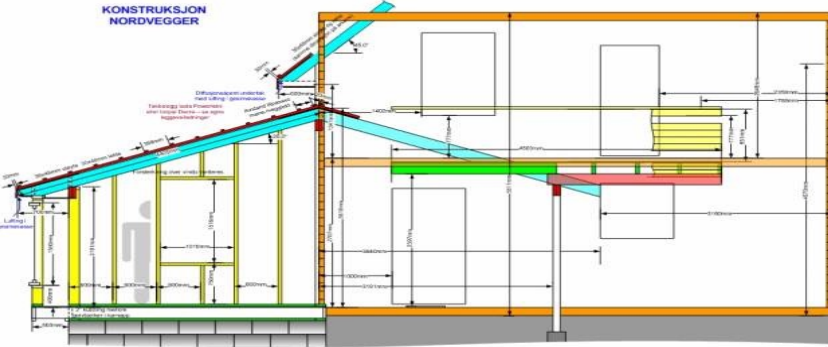
3. Azure DevOps (An Azure DevOps Project has been made + give access to HPH):

- A good Folder structure has been made + Iterations (Alpha, Beta, RC, RTM). Make sure to have a good structure in Microsoft Trams as well
- Product Backlog** has been made (A List of Requirements for your System) in Azure DevOps
- ER Diagram, Database Scripts**, etc. have been uploaded/Checked-in into Azure DevOps
- PoCs/Code Examples/Prototypes have been uploaded/Checked-in + Same with Database Scripts (Tables, Views, Stored Proecures)

4. Project Web Page

- A simple **HTML Web Page** with an Introduction (Text and Figures) + Links to SDP and SRD (PDF) +++ (work next week)

Requirements/Design



Plans made and approved

Alpha



Foundation finished, building structure started
A "proof" that you can do it, PoC (Proof of Concept)

Beta



Building structure finished,
Inside work on track

RC



Furniture, Flowers and
small adjustments missing

RTM



Ready for Sale or Move in

Preparation for Next Week: Prepare for Web Site

- Next week we will create a **Web Site (HTML, CSS)** that introduces the product you are creating, including documents you have created (SDP, SRD, ...)
- This is an individual activity, meaning all team members should create such a web site.
- You need to know basic **HTML**. A good source is:
<https://www.w3schools.com/html/>
- Recommended HTML Editor: **Visual Studio Code** (or you can use Visual Studio, but VS is not well suited for HTML pages)
- The best web site within each group should then be selected and used further in the project.
- We will use the available Web Server at the University. The Web address (URL) will be like this: <https://web01.usn.no/~hansha>

Resources:

<https://min.usn.no/programvare/eget-webomrade-web01-usn-no-article217606-32428.html>

Simple Web Page in Visual Studio Code

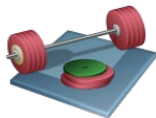
The image shows the Visual Studio Code editor interface with a file named 'webpage.html' open. The code editor displays the following HTML code:

```
1 <!DOCTYPE html>
2 <html>
3
4 <body>
5
6     <h1>Heading 1</h1>
7     <h2>Heading 2</h2>
8     <h3>Heading 3</h3>
9
10    <p>My first paragraph.</p>
11
12    <p>
13
14        <a href="https://www.w3schools.com">This is a link</a>
15
16    </p>
17
18
19    <p>
20
21        
22
23    </p>
24
25
26 </body>
27
28 </html>
```

The preview window on the right shows the rendered web page. It features three headings: 'Heading 1', 'Heading 2', and 'Heading 3'. Below the headings is a paragraph: 'My first paragraph.' followed by a link: 'This is a link'. Below the link is a placeholder for an image with the text 'Weather System' and 'The Weather System...'. The image placeholder shows a weather station and a smartphone displaying weather information. Below the image placeholder is a section titled 'Documents' with two items: 'Software Development Plan' and 'Software Requirements & Design'.

The status bar at the bottom of the editor shows: Ln 23, Col 9 Spaces: 4 UTF-8 LF HTML

Web Site Example



Create a similar Web Site for your Product

Presentation (Figures and Text) of your Software Product

Note! We should use only **HTML** (not ASP.NET! – because the Web Server we are using at the University don't support it). In addition, you may use **JavaScript** and **CSS**

Documents available for download (Hyperlinks)

Weather System

The Weather System ...

ambient weather

6:25 PM 47% MEMOry 75% 0% 0% 0%

Direct DWT 5. 4TH 24h 4.12 inch

0.7 31.27 82.2°F 89% RH 27.0°F 51% RH

Documents

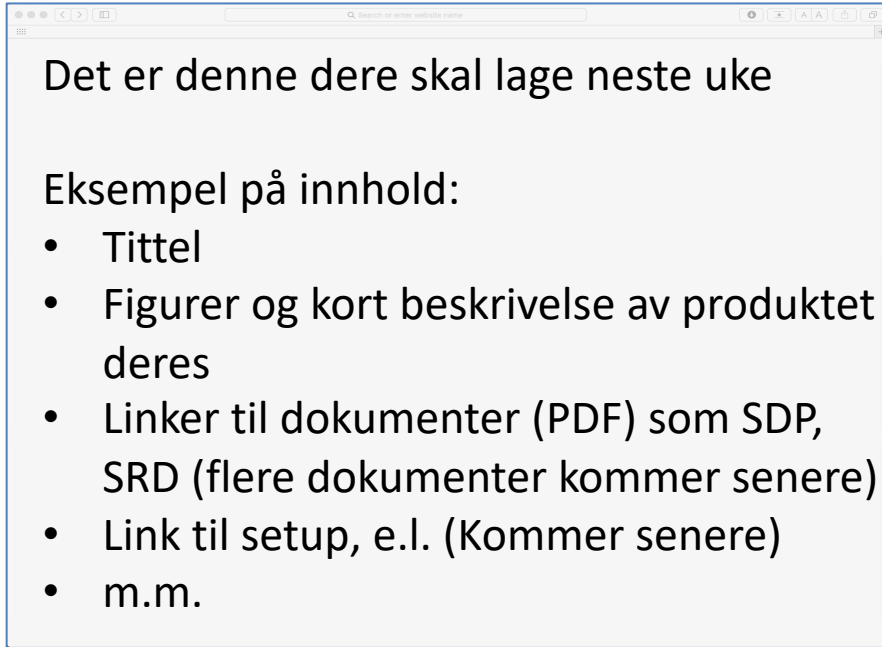
- Software Development Plan
- Software Requirements & Design

Note! Always use PDF Files!!!

Web Development in your Project

1

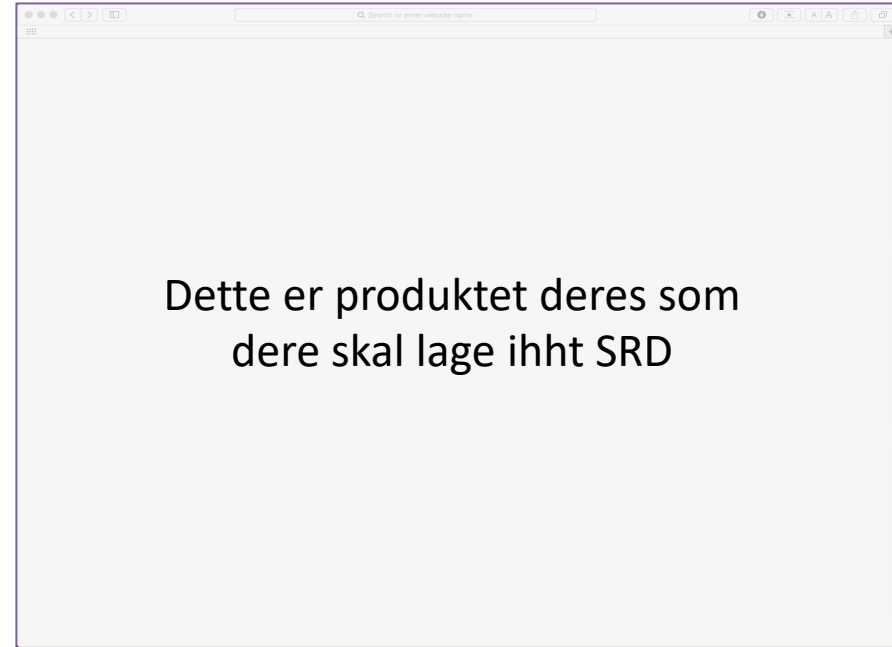
HTML



“Commercial” Web Page with information about your Product, you can download setup files, documents, etc.

ASP.NET

2



Your web-based Product -> Web Application

vs.

Illustrative Example

1

The screenshot shows the homepage of itslearning.eu. The header is blue with the 'fronter' logo and navigation links: 'Our Solution', 'News', 'Training', 'Blog', 'Support', 'About Us', and 'Contact Us'. A search bar and 'Sign-in' button are on the right. The main content area features a large blue banner with the text 'itslearning acquires Fronter' and a sub-headline 'Making itslearning Europe's largest provider of digital learning platforms'. A 'Read more' button is positioned at the bottom right of the banner.

Fronter is all about Education

To improve education and make it more available, we build and deliver a teacher friendly learning platform. It is a service we provide to make it easier for teachers to engage students and allow them to take an active part in their own learning.

Read more

Latest News



2

The screenshot shows the Fronter web application interface. The top navigation bar includes 'Rooms' and 'Tools' menus, and a user profile for 'Hans Petter Halvorsen'. Below the navigation, there are tabs for different courses: 'IA4412 - Syste...', 'PRH612 Bachelo...', 'SCE2006 Indust...', 'FMH606 Master...', 'Bachelor V17 - L...', and 'EPH - Master Th...'. A sidebar on the left contains a navigation menu with items like 'Room', 'Home Page', 'Diverse', 'Hand-in', 'Skype', 'Members', 'News', and 'Statistics'. The main content area displays a 'Description' for a course, detailing its project-based learning approach and mentioning 'Temaer' (topics) and 'Ukesoppgaver' (weekly assignments). Below the description, there is a 'News' section with two articles: 'Tema 4: Unified Modelling Language (UML)' and 'Systemutvikling - Web/database-eksempel gjennomgått tirs 24/1-17 publisert'. Each news item includes a brief summary and a 'Read more' link.

Product Information Web Site

vs.

Product Web Application

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: <https://www.halvorsen.blog>

