https://www.halvorsen.blog

Student Projects

Guidelines and Resources

Hans-Petter Halvorsen

Contents

- Project Work
- Microsoft Teams
- Project Management
  - Gantt Diagram
  - Microsoft Planner
- Meetings
- Development
- Report Writing

Project Work

Project Work

- Why Projects? Task (challenge/problem) that needs to be solved.
- A Project is typically initiated by your superior and it can be in your company or for an external company (customer). A project group and project manager is then organized to manage the project and deliver according to the project description.
- Work to be done within a limited time, e.g., 5-6 months.
- Need to use a mix of known skills but also it is needed to learn new skills.
- Teamwork. Collaboration is essential!
- A Project Manager is needed to keep track of the overall work made by many people.
- Meetings are needed to discuss and synchronize activities and to go through work that has been done.
- Project Planning and Management needed to find out what to do and when to do it and how much time it takes.
- Documentation: The Project Work needs to be documented in form of project plans, internal working documents, code and a report or other types of delivery documents, e.g., user manuals, etc.

Student Projects

Learning goals in a typical student project:

- Solve a Technical challenge, i.e., build something technical, develop software, etc.
- Project Work, most work in the industry today is based on the project model.
- Collaboration and Team Work.
- Project Planning and Management, and Meetings. Use suitable software tools for this.
- Use existing skills and combine it with new skills that you need to learn by yourself during the project.
- Independent work. No more lectures. Its time to use the skills learned combined with new ones on practical issues and challenges.
- Write Technical Documents of high quality.

Assessments

In a student project the following are typically part of the overall assessments:

- Process including independent work, collaboration, initiative, attitude, being a team player, takes responsibility, being an active member of the team, focus on quality, structured work, etc.
- System/Product – The stuff (hardware/software) you create in the project.
- Report
- Final oral Presentation
- Final Examination with Questions and Answers

Grades Descriptions: https://www.uhr.no/_f/p1/ie0fb9b5f-58fe-4c7b-b420-9d640727c9b6/karakterbeskrivelser_og_vurderingskriterier_bacheloroppgave_i_ingeni_rfag_l_29028.pdf

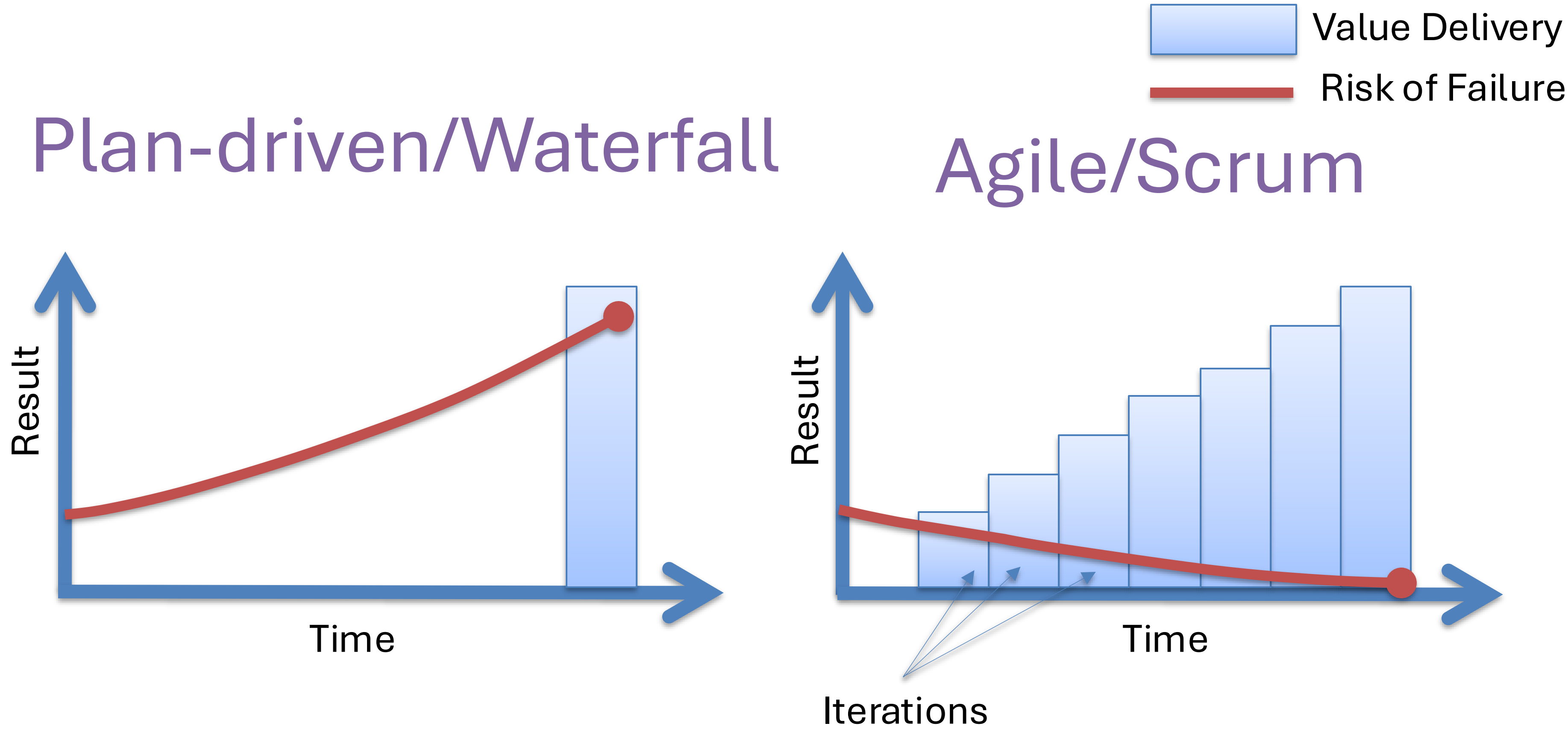
Project Work

Different ways to work in a project:

- **Plan-driven/Waterfall**
  - The project does not need to be finished until 6 months and no deliveries needs to be finished before that day
  - “Old-fashion” way to think of projects and deliveries
- **Agile/Scrum**
  - We work in short Iterations (1-2 weeks) where we produce working software, documents, etc. in each iteration. Basically, assume that the final delivery is tomorrow!
  - “Working software” means that the software (and documentation) can be used by the customer in a production environment!!! – so testing and quality control must be an important part of each iteration!!!
  - Modern way to think of projects and deliveries

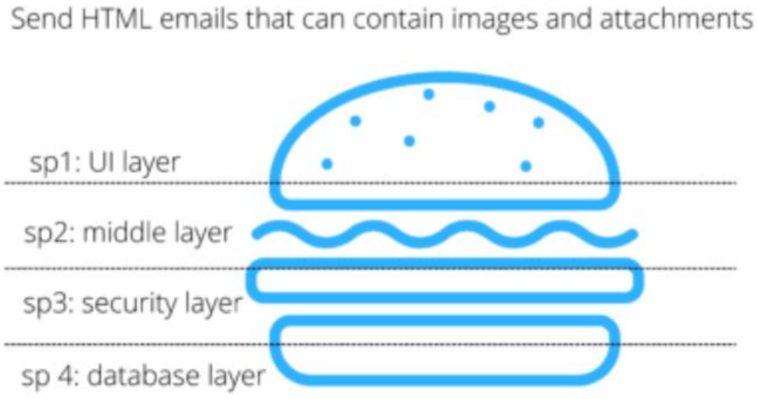
→ For Student Projects and most modern smaller projects, and especially software projects, Agile/Scrum method is recommended!

Waterfall vs. Agile/Scrum

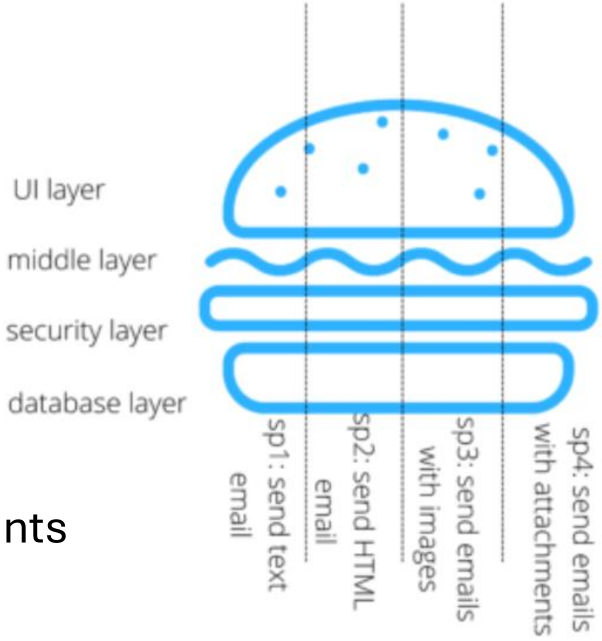


Project Work

Would you rather eat layers (left image) or slices (right image) of a burger?



(Plan-driven/Waterfall)



(Agile)

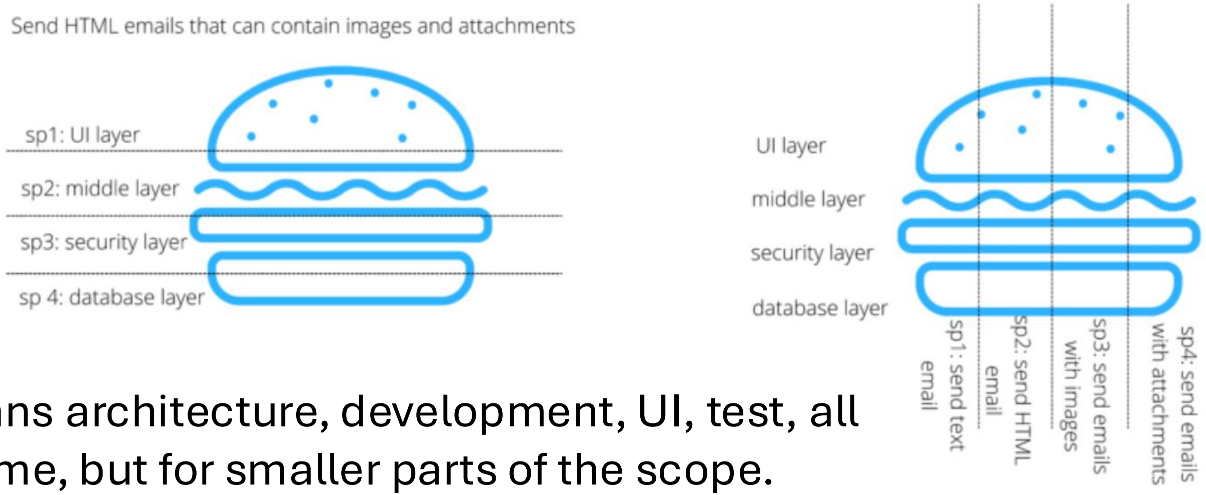
Important Agile principle: Working software, documents and product at all times, which is illustrated with the hamburger to the right.

https://allthingsagile.co/post/agile-principle-7/

Work Breakdown

Layered approach. Imagine a hamburger. Would you accept just the top bun? That's just bread. Or maybe just the lettuce? Well, that's just salad. You don't get the feeling of what the burger with all ingredients taste. What makes them a hamburger is all of them put together!

Work breakdown: horizontal vs vertical slicing?



"Slicing vertically", which means architecture, development, UI, test, all have to happen at the same time, but for smaller parts of the scope. Instead of breaking work structurally, work should be broken by functionality. That's an important principle in Agile development. You can now eat slices of your hamburger!

https://allthingsagile.co/post/agile-principle-7/

Project Work Guidelines

- You should have “working software”, i.e., product and documents at all times (Agile mindset).
- Work in short **Iterations** (1-2 weeks) where you show what you have produced in weekly/ biweekly meetings with the customer and supervisor.
- Produce software and documents (in student project often a report) so it can be used and tested by the customer after each iteration.
- **Basically, assume that the final delivery is tomorrow.**
- Make sure to get feedback from the customer after each iteration so you know what to improve and focus on in next iteration.
- It is the customer that are going to use the product/software, so make it in the way so it fulfils the customers' needs.
- Quality control: Continuously Test your work, both applications and documents

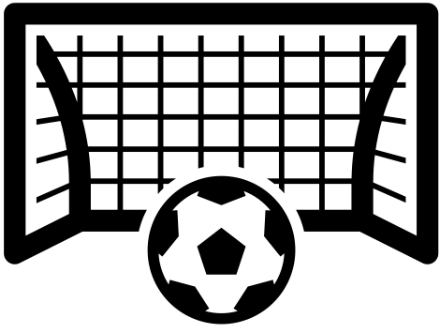
Project Work

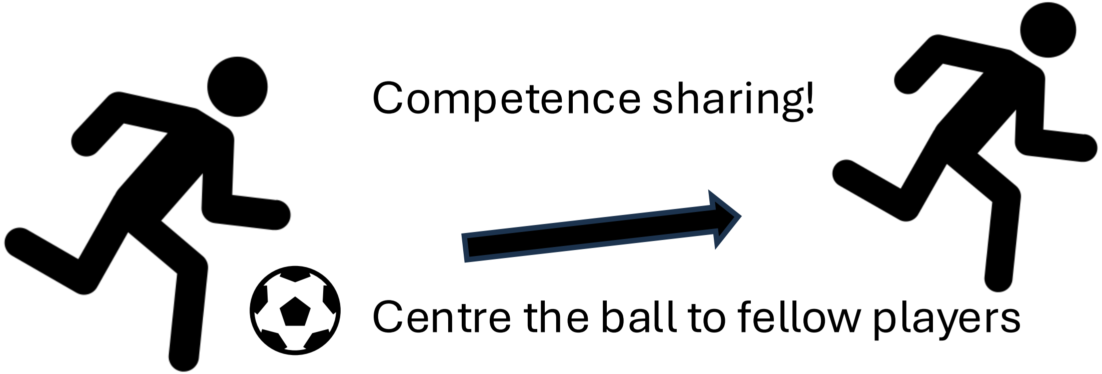
Project Work is like playing football!

Project work is Teamwork where you make each other good.

The team consists of individuals who must work together to succeed.

The team must make a good plan to win the match.
(project planning and documentation)





Competence sharing!

Centre the ball to fellow players

It doesn't help to have Ronaldo in the team if the players can't work together and make each other good. They must communicate well and distribute tasks, know how to play, etc.

The players have different roles (striker, defense, captain, ..), in a project we have Programmer, Tester, Architect, Scrum Master, Project Manager, ..

Project Work

Project Work is running a marathon!

Run smoothly along the way so you don't get too far behind the others but remember to save a little for the last round/lap.

Project Start

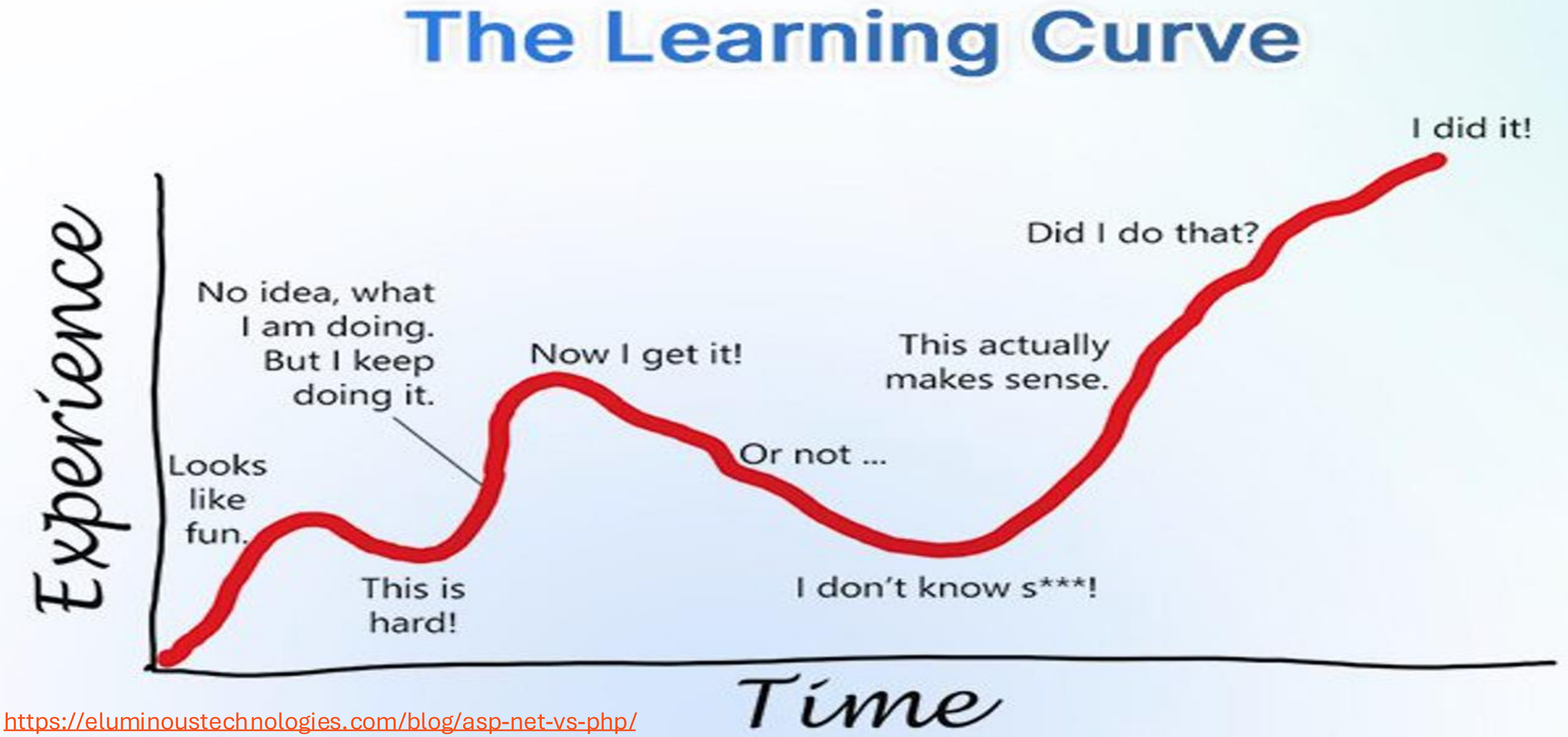
Important to get a good start so that you do not fall too far behind already at the start

It is common to encounter some obstacles/problems along the way, but it is important not to give up

Project Finished

Last round/lap. Important not to back down but stay the course. If you are motivated and the right attitude, you can outrun many in the last kilometer. This means that don’t stop development 2 weeks before but work with the product until the end.

Project period, ups and downs



https://eluminoustechnologies.com/blog/asp-net-vs-php/

Project Work Principles

- **Customer focus** - The highest priority is to satisfy the customer through early and continuous deliveries of software that has value.
- **Customer involvement** – When developer and customer collaborate, we get the best result
- **Positive to changes** - Welcome changes in requirements, even late in development. Ask for feedback continuously and do something about it! Don't say "yes we ..", "We thought it was best to do that..". It is the customer who will use the product, not you.
- **Delivery of frequent increments** – Deliver working software frequently, at 1–2-week intervals. The more often, the better.
- **Face-to-face communication** - The most effective way to convey information to and within a development team is to talk face-to-face.
- **Software that works** - Working software is the primary measure of progress. Which is tested, quality assured and which can be used by the customer in principle
- **Work steadily** – Work the same amount week by week, not wait until the end of the project.
- **User-friendliness and quality** - Continuous focus on quality and good design
- **Simplicity** – Think simple rather than complicated.
- **Continuous process improvement** - At regular intervals, the team reflects on how it can become more efficient and then adjusts its behavior accordingly.

Project Tools

- Microsoft Teams (Online Meetings, Collaboration, Chat, Project Planning, Document sharing, etc.)
- Microsoft Planner (part of Microsoft Teams)
- Microsoft Word (Documentation)
- Microsoft Excel (Gantt Diagram, etc.)
- Microsoft PowerPoint (Presentations, Drawings and Sketches)
- Design and Drawing tools, here you can use Microsoft Project or more tailormade software like Lucidchart, etc.
  - Microsoft PowerPoint is an excellent tool for making simple drawings!
  - Make System Sketch in PowerPoint: https://youtu.be/9mmBXFOjV3s
- Visual Studio and C# (Development)
- GitHub (Share Code between multiple Developers)

Video Resources

- Making **Gantt** diagram with Microsoft Excel: https://youtu.be/L31m3Jf87PY
- **Planner App** in Microsoft Teams: https://youtu.be/LrZK3oUgkL4
- Make **System Sketch** in PowerPoint: https://www.youtube.com/watch?v=9mmBXFOjV3s
- Write Technical **Reports** in Microsoft Word: https://www.youtube.com/watch?v=ao_eDJOEUkA
- **Citation and Referencing** with Microsoft Word: https://www.youtube.com/watch?v=IgH7qmLa_L4
- **Figures and Equations** in Word and PowerPoint: https://www.youtube.com/watch?v=b9f2bb2yn1Y

Microsoft Teams

Microsoft Teams

Use Microsoft Teams for the following:

- Online Meetings.
- Document sharing. Here multiple person can share and work on the same documents in simultaneously.
- Collaboration, like using the Chat
- Project Management, like the built-in App Microsoft Planner.
    - Planner App in Microsoft Teams: https://youtu.be/LrZK3oUgkL4

Microsoft Teams

- Make a good **Folder structure**!
  - It should be easy to find what you are looking for!
- Use the **Chat**! Write messages and reply to messages, create engagement! Show that you have read messages with, e.g., "Thumbs up", etc.
  - If you don't have time to answer immediately, reply e.g. "I will look at it tomorrow"
- **Always On**:
  - Always have Teams up and running on your PC!
  - Turn on **Notifications**
  - Install and use the Teams App for iOS/Android
  - Respond immediately when somebody write messages, etc.

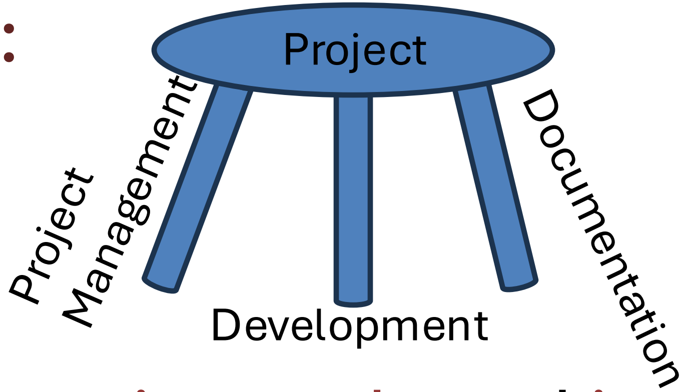
Project Management

Hans-Petter Halvorsen

Table of Contents

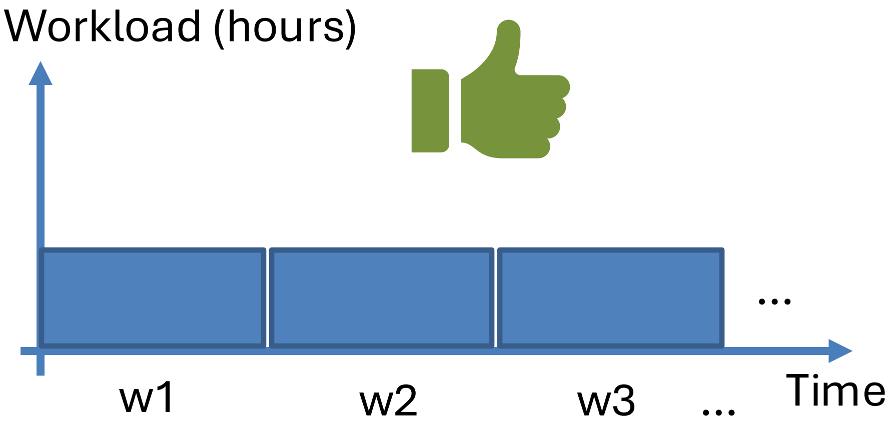
Project Work

- All projects consist of 3 parts:
  - Project Management
  - Development
  - Documentation

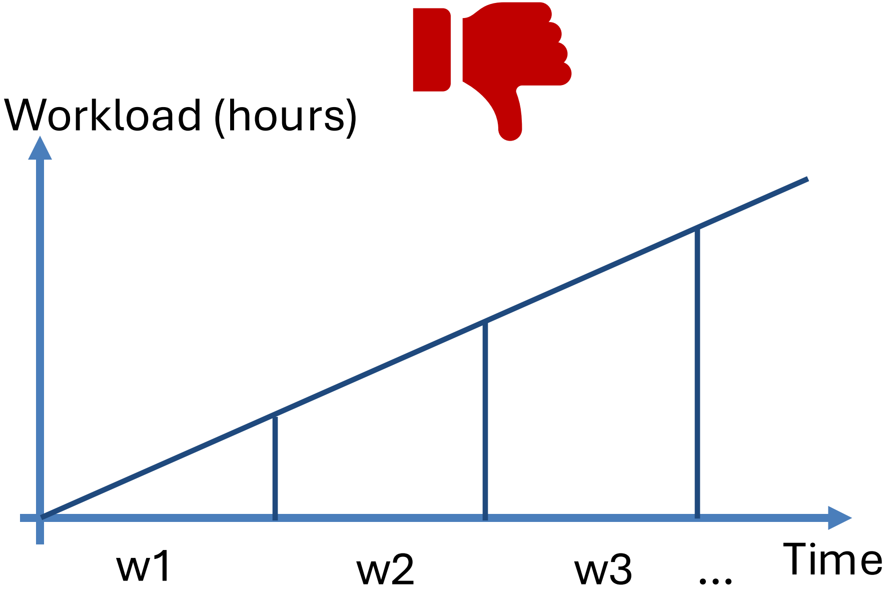


- These MUST be worked with continuously and in parallel throughout the life of project
- Comparable to a table with 3 legs
  - If you take away one (or 2) legs then the whole table (read the project) will collapse

How to Work



Finalize last week's plans and tasks so you don't get backlogs and have to do last week's work in addition to this week's work. Then there will be more and more to do every week.



Project Management Tools

1. Project Plan/Gantt Diagram (e.g., Microsoft Excel)
   - The overall project from start to end. The “big picture”
2. Taskboard (e.g., Microsoft Planner or Azure DevOps)
   - Weekly details
   - Each Task should contain at least the following:
     - Task Name
     - Responsible Person (only 1 person!!)
     - Deadline
     - Duration (max 8 hours)

→ Both need to be updated during the entire project

Agile and Scrum

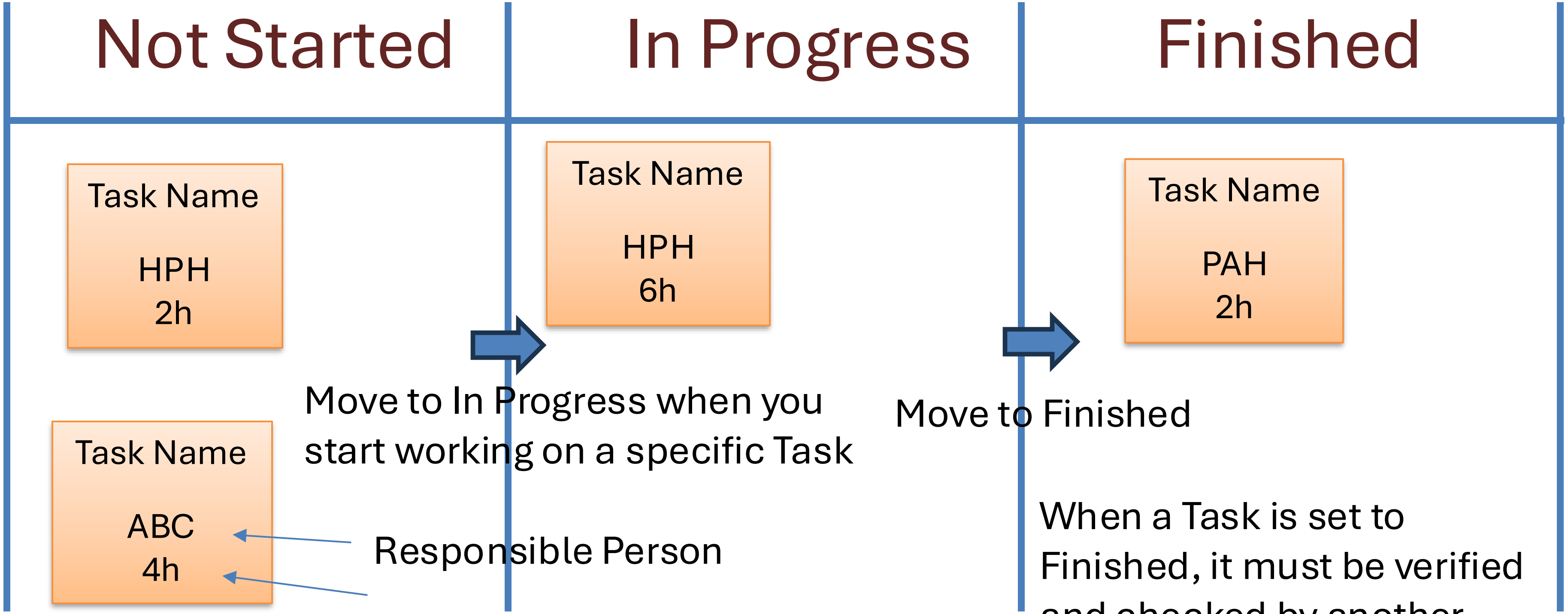
- The traditional way of organize and work in a project is the so-called “**Waterfall**” method are in general “**Plan-driven**” methods.
- **Agile** is a newer and more flexible way to organize and run a project.
- **Scrum** is the most poplar Agile method today.
- In Scrum we work in short **Iterations**.

Scrum and Sprints

- In **Agile** you work in iterations or **Sprints** as it is called in Scrum.
- Like in Scrum you should create new Tasks in the group when a new Sprint starts.
- Taskboard: The during the Sprint you should move the Tasks from “Not started” to “In progress” and then finally to “Completed”.
- When the Sprint is finish, the hopefully all the Tasks should be in the “Completed” column
- **A Sprint can typically be 1 week in Student projects**. Then a new Sprint cam start on a Monday and be finished on a Sunday, then start on a new Spring, etc.

Taskboard

The Taskboard is for short term use, e.g., weekly, i.e., what you shall do within the next week



Not Started

In Progress

Finished

Task Name
HPH
2h

Task Name
HPH
6h

Task Name
PAH
2h

Task Name
ABC
4h

+ Deadline

Move to In Progress when you start working on a specific Task

Move to Finished

Responsible Person

Duration

When a Task is set to Finished, it must be verified and checked by another person

Gantt Diagram

Gantt Diagram

- A Gantt diagram, also known as a Gantt chart, is a project management tool
- It visually represents a project schedule over time.
- It uses horizontal bars to illustrate the
  - start and end dates of tasks,
  - their duration,
  - and their dependencies
- Making Gantt diagram with Microsoft Excel: https://youtu.be/L31m3Jf87PY

Use of Gantt diagram

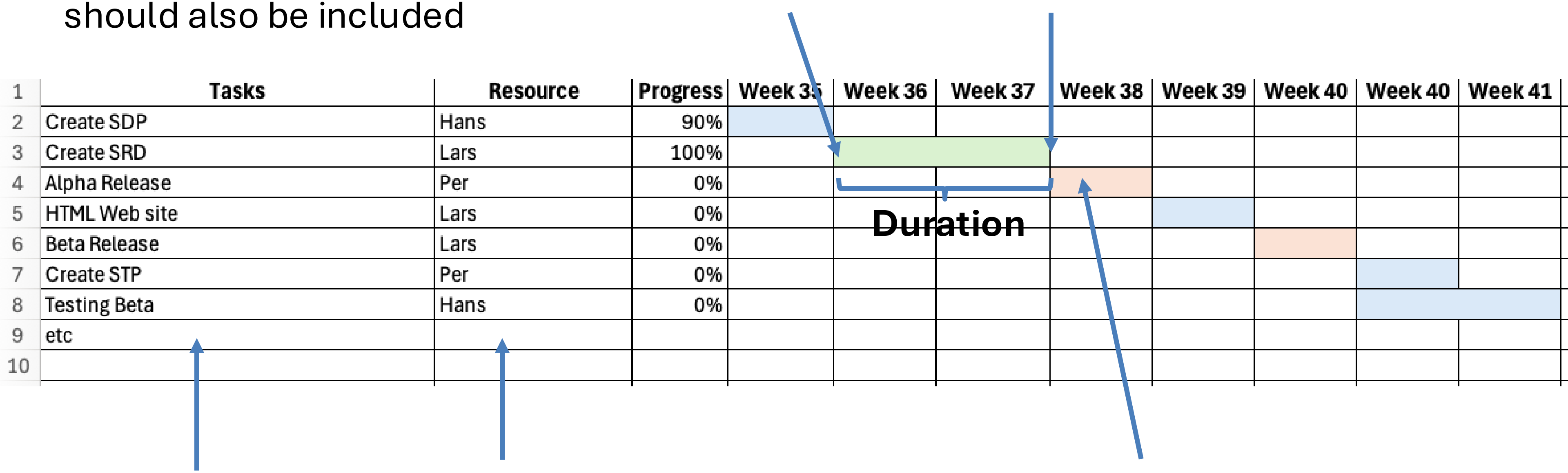
- We typically create a **Gantt diagram** in the beginning of a project to get an overall idea of the main parts that shall be done in a project (i.e., no details).
- It must typically to be updated weekly.
- We typically use the Gantt diagram for the overall project planning, i.e., the ”big things” not all tiny details.
  - The Task duration in a Gant diagram is typically many days or weeks.
- While we may use other tools for the details and the daily work, such as a **Taskboard**. The Taskboard is then used and updated daily.
  - The Task duration in a Taskboard is typically a few hours.

Gantt Diagram

A column with **Estimated Hours** should also be included

Start Date

End Date



1	Tasks	Resource	Progress	Week 35	Week 36	Week 37	Week 38	Week 39	Week 40	Week 40	Week 41
2	Create SDP	Hans	90%	🟦							
3	Create SRD	Lars	100%		🟩	🟩					
4	Alpha Release	Per	0%				🟧				
5	HTML Web site	Lars	0%					🟦			
6	Beta Release	Lars	0%						🟧		
7	Create STP	Per	0%							🟦	
8	Testing Beta	Hans	0%							🟦	🟦
9	etc										
10											

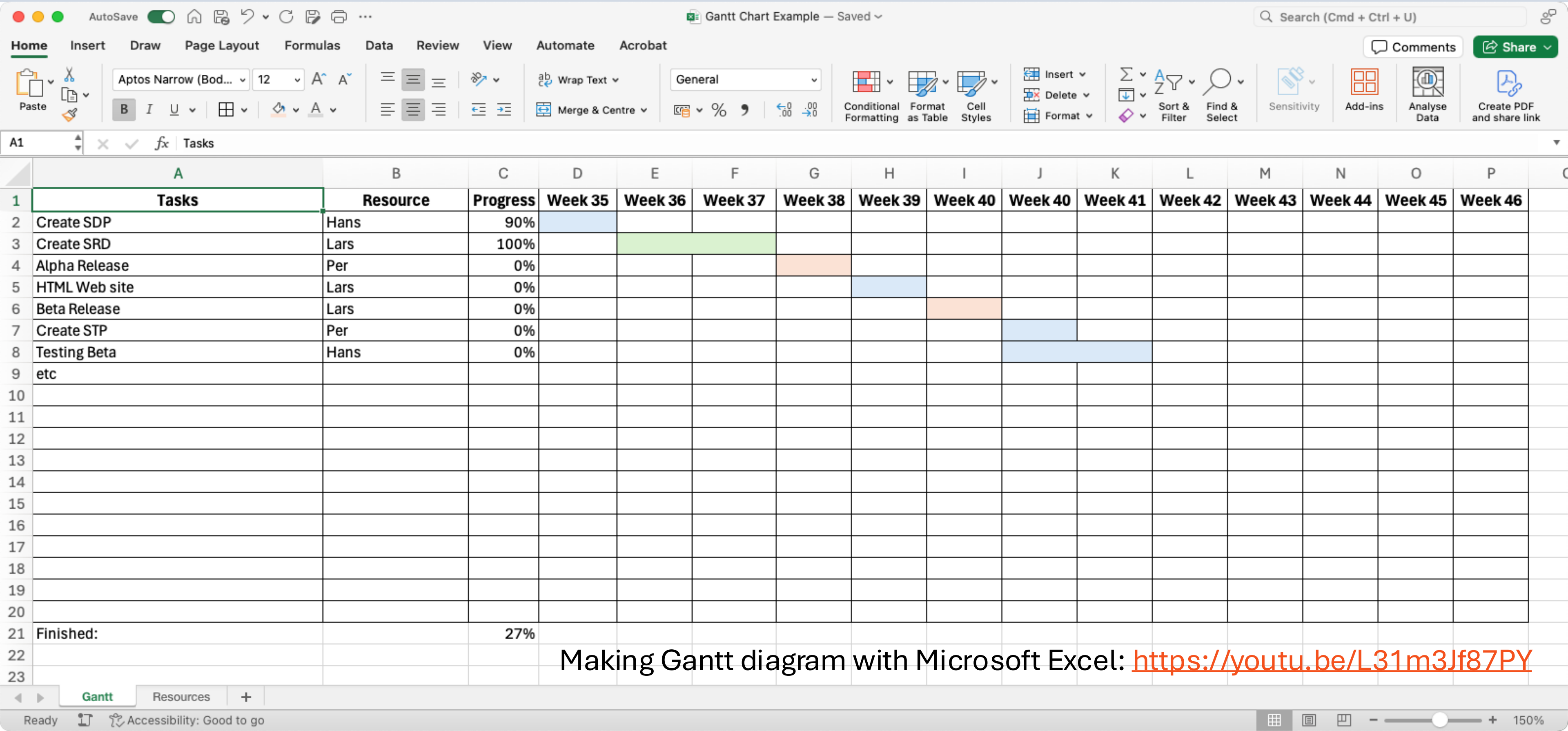
Duration

Tasks (What should be done)

Resources (Who is responsible for the Task?)

Dependency. Task 4 cannot/should not start before Task 3 is finished

Excel Gantt Diagram Example

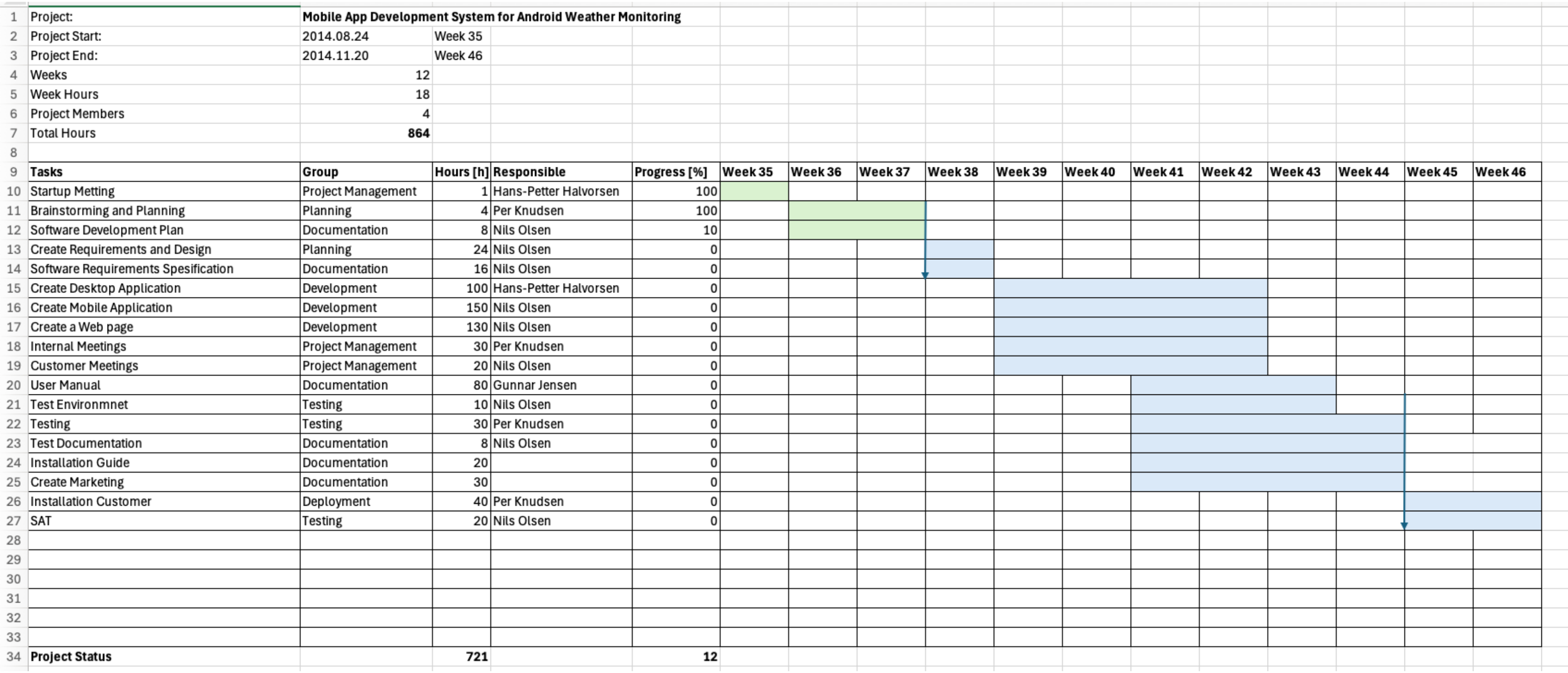


Tasks	Resource	Progress	Week 35	Week 36	Week 37	Week 38	Week 39	Week 40	Week 40	Week 41	Week 42	Week 43	Week 44	Week 45	Week 46
Create SDP	Hans	90%	🟦												
Create SRD	Lars	100%		🟩	🟩										
Alpha Release	Per	0%				🟧									
HTML Web site	Lars	0%					🟦								
Beta Release	Lars	0%						🟧							
Create STP	Per	0%							🟦						
Testing Beta	Hans	0%							🟦	🟦					
etc															
Finished:		27%													

Making Gantt diagram with Microsoft Excel: https://youtu.be/L31m3Jf87PY

Add more Features and Functionality

By using some time and using the features in MS Excel, you can easily add a more advanced Gantt diagram that has extended functionality.



1	Project:	**Mobile App Development System for Android Weather Monitoring**															
2	Project Start:	2014.08.24	Week 35														
3	Project End:	2014.11.20	Week 46														
4	Weeks	12															
5	Week Hours	18															
6	Project Members	4															
7	Total Hours	**864**															
8																	
9	**Tasks**	**Group**	**Hours [h]**	**Responsible**	**Progress [%]**	**Week 35**	**Week 36**	**Week 37**	**Week 38**	**Week 39**	**Week 40**	**Week 41**	**Week 42**	**Week 43**	**Week 44**	**Week 45**	**Week 46**
10	Startup Metting	Project Management	1	Hans-Petter Halvorsen	100	🟩											
11	Brainstorming and Planning	Planning	4	Per Knudsen	100		🟩	🟩									
12	Software Development Plan	Documentation	8	Nils Olsen	10		🟩	🟩									
13	Create Requirements and Design	Planning	24	Nils Olsen	0				🟦								
14	Software Requirements Spesification	Documentation	16	Nils Olsen	0				🟦								
15	Create Desktop Application	Development	100	Hans-Petter Halvorsen	0					🟦	🟦	🟦	🟦				
16	Create Mobile Application	Development	150	Nils Olsen	0					🟦	🟦	🟦	🟦				
17	Create a Web page	Development	130	Nils Olsen	0					🟦	🟦	🟦	🟦				
18	Internal Meetings	Project Management	30	Per Knudsen	0					🟦	🟦	🟦	🟦				
19	Customer Meetings	Project Management	20	Nils Olsen	0					🟦	🟦	🟦	🟦				
20	User Manual	Documentation	80	Gunnar Jensen	0							🟦	🟦	🟦			
21	Test Environmnet	Testing	10	Nils Olsen	0							🟦	🟦	🟦			
22	Testing	Testing	30	Per Knudsen	0							🟦	🟦	🟦	🟦		
23	Test Documentation	Documentation	8	Nils Olsen	0							🟦	🟦	🟦	🟦		
24	Installation Guide	Documentation	20		0							🟦	🟦	🟦	🟦		
25	Create Marketing	Documentation	30		0							🟦	🟦	🟦	🟦		
26	Installation Customer	Deployment	40	Per Knudsen	0											🟦	🟦
27	SAT	Testing	20	Nils Olsen	0											🟦	🟦
28																	
29																	
30																	
31																	
32																	
33																	
34	**Project Status**		**721**		**12**												

Important Information in Gantt Chart

- Project Title
- Overview of available Resources/Persons
- Project Start and Project End
- Task Title (make sure to use a descriptive Title)
- Grouping/Category
- Estimated Hours
- Resource/Responsible Person
- Progress (%)
- Start Date and Stop Date for each Task

Microsoft Planner

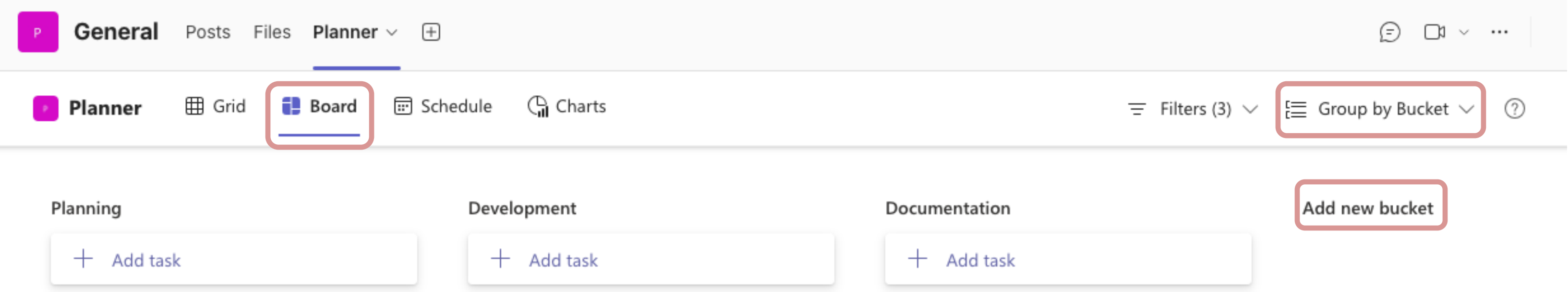
Hans-Petter Halvorsen

Table of Contents

Microsoft Planner App

- Planner App is made by Microsoft and integrated in Microsoft Teams.
- The Planner App can typically be used to manage small projects.
- The Planner App supports Agile approaches like Scrum and Kanban
- The Planner App is excellent for Student group projects!
- Can also be used independent of Microsoft Team from this location: https://planner.cloud.microsoft
- Planner App in Microsoft Teams: https://youtu.be/LrZK3oUgkL4

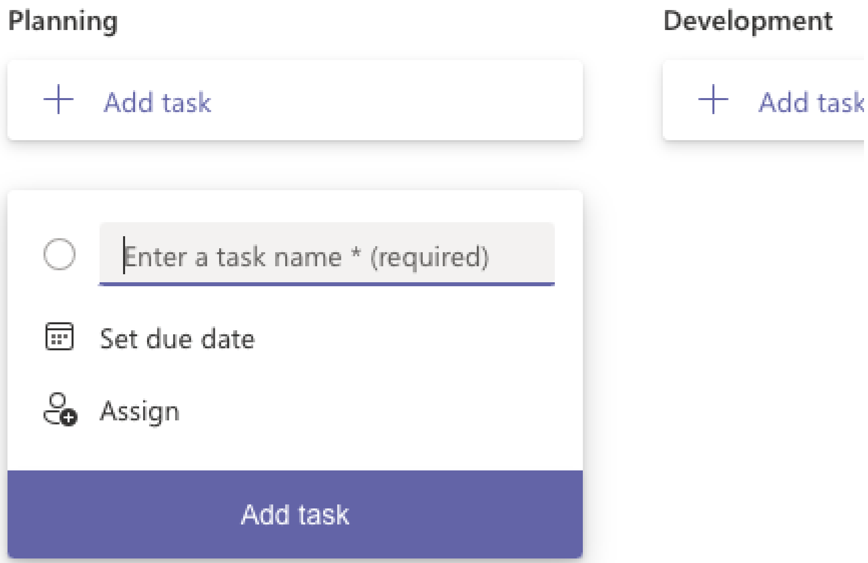
Organize with “Buckets”



Your can structure your Tasks by organize it in different “Buckets”. “Buckets are just a fancy name for Groups or Folders.

I recommend 3-4 buckets like “Planning”, “Development” and “Documentation”.

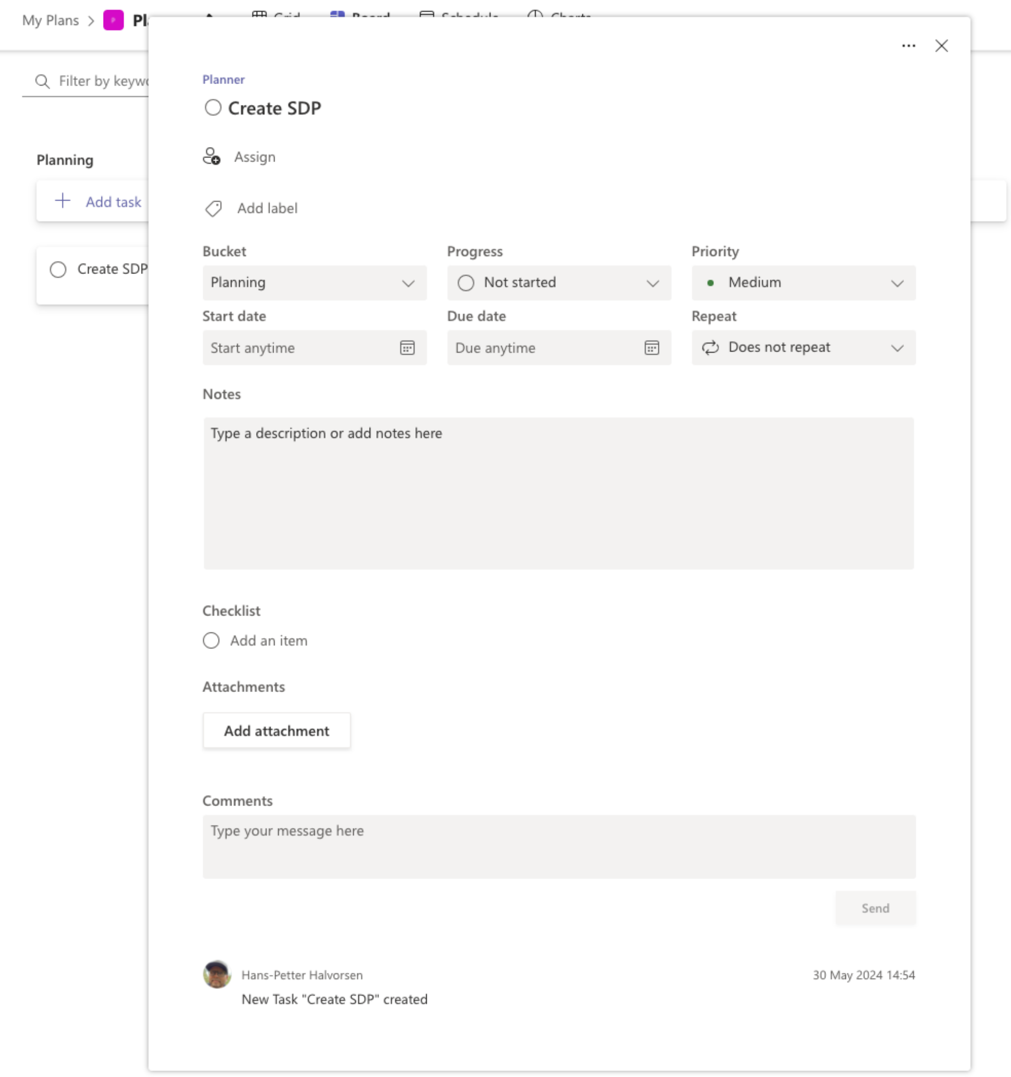
Add Task



Planning

Development

○ Enter a task name * (required)



○ Create SDP

Bucket: Planning

Progress: ○ Not started

Priority: ● Medium

Start date: Start anytime

Due date: Due anytime

Repeat: Does not repeat

Notes: Type a description or add notes here

Checklist

Attachments

Comments: Type your message here

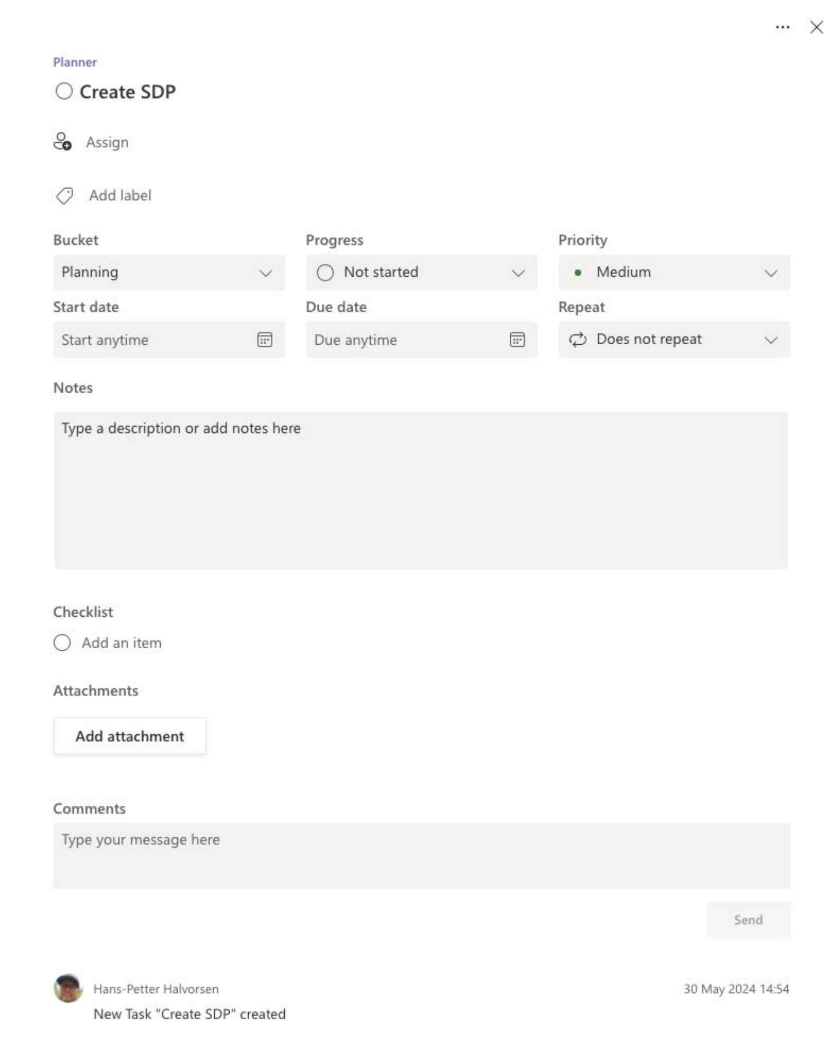
Hans-Petter Halvorsen 30 May 2024 14:54

New Task "Create SDP" created

Task Details

Make sure to fill out as many fields as possible. Important fields are:

- “**Title**”
- “**Assign**”
- “**Bucket**” – Group your work into a “Category“
- “**Progress**” (“Not started”, “In progress” and “Completed”)
- “**Priority**”
- “**Start date**”
- “**Due date**”
- “**Notes**”



Planner

○ Create SDP

Assign

Add label

Bucket: Planning

Progress: ○ Not started

Priority: ● Medium

Start date: Start anytime

Due date: Due anytime

Repeat: Does not repeat

Notes

Type a description or add notes here

Checklist

○ Add an item

Attachments

Add attachment

Comments

Type your message here

Send

Hans-Petter Halvorsen 30 May 2024 14:54

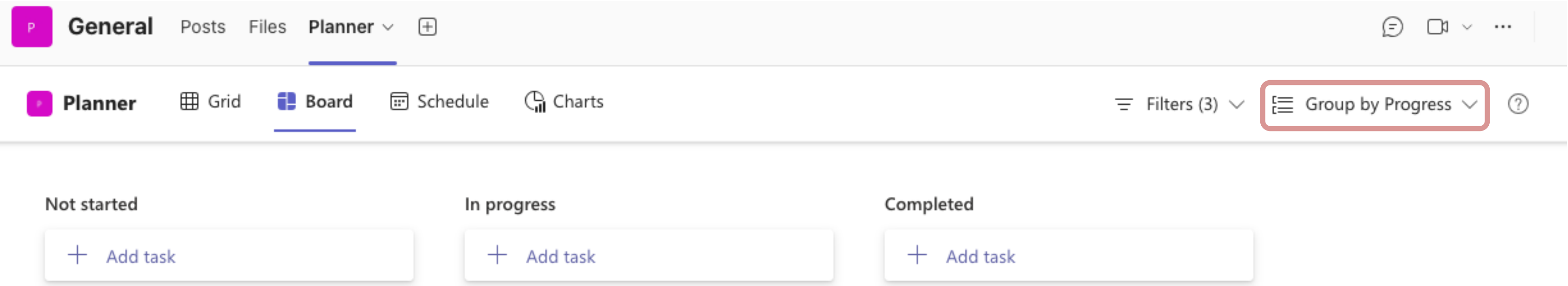
New Task "Create SDP" created

Best Practice!!!

- **Assign only 1 Person to a Task**.
  - If there are more than one person there will be none in charge of the Task and it will most likely not be done at all
  - Then better to divide into multiple Tasks, one for Person A and another Task for Person B
- Better to **Create many small Tasks** than a few large Tasks
  - Example "Update Report" is NOT a good Task, will it take a week or a month? and what shall be done?
  - Example "Update Chapter 2.3 with Results from the Simulations" is much more specific and time limited Task
- Create Tasks that has a **Max duration of 8 hours**
- **Work in Iteration/Sprints that last 1 week**, this means you come together in a Meeting and setup all the necessary Tasks for the current iteration/Spring.
  - Make sure to create enough Tasks for the entire Iteration/Sprint for all members. 20 credit is 270hx2xnumber of members, divide by number of weeks to find the weekly workload.
- **Update Tasks when working**
  - Make sure to immediately update your Task from "Not started" to "In progress" when you start working on that specific Task.
  - Make sure to immediately update your Task from "In progress" to "Completed" when you are finished with that specific Task.
- Make sure to have a **Project Manager** or **Scrum Master** that keep track of all the Tasks and have an overall control of the situation and status

Group by Progress

This is like a “Kanban board” used in Agile/Scrum where you have 3 columns; “**Not started**”, “**In progress**” and “**Completed**”

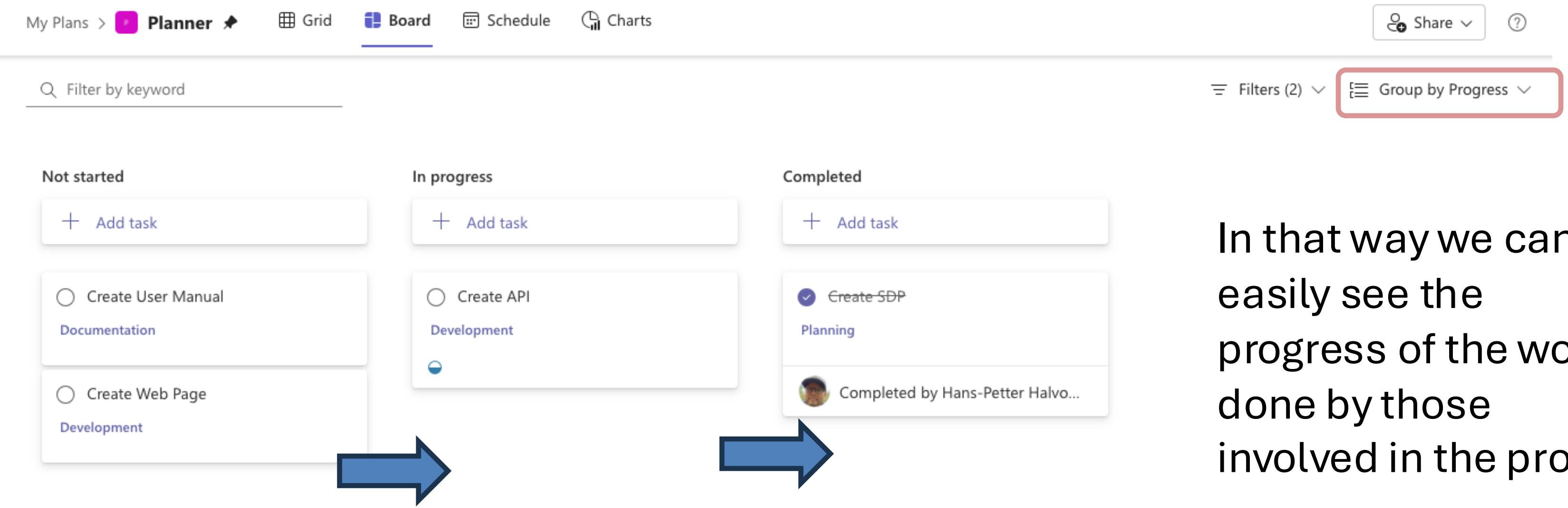


Drag and Drop from “Not Started” to “In Progress”

Drag and Drop from “In Progress” to “Completed”

Update your Progress

In your “Kanban board” make sure to move your Taks from “Not started” to “In progress” and then finally to “Completed”.



My Plans > Planner Grid Board Schedule Charts Share ?

Filter by keyword Filters (2) Group by Progress

Not started
+ Add task
○ Create User Manual
Documentation
○ Create Web Page
Development

In progress
+ Add task
○ Create API
Development

Completed
+ Add task
● ~~Create SDP~~
Planning
Completed by Hans-Petter Halvo...

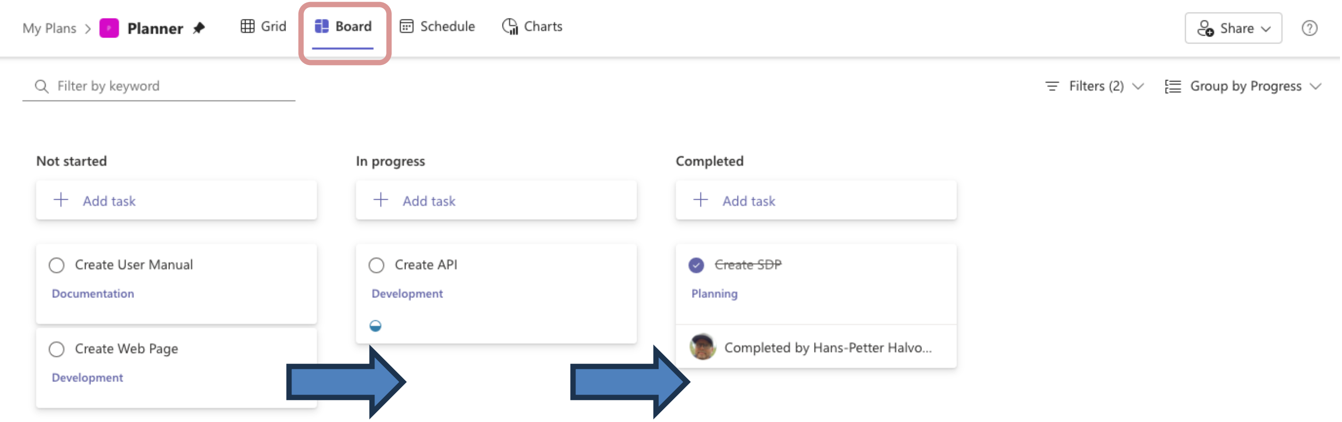
In that way we can easily see the progress of the work done by those involved in the project.

Drag and Drop from “Not Started” to “In Progress”

Drag and Drop from “In Progress” to “Completed”

“Taskboard” in Microsoft Planner

You can use a Taskboard in Microsoft Planner by using the “Board” view



Not started

+ Add task

○ Create User Manual
Documentation

○ Create Web Page
Development

In progress

+ Add task

○ Create API
Development

Completed

+ Add task

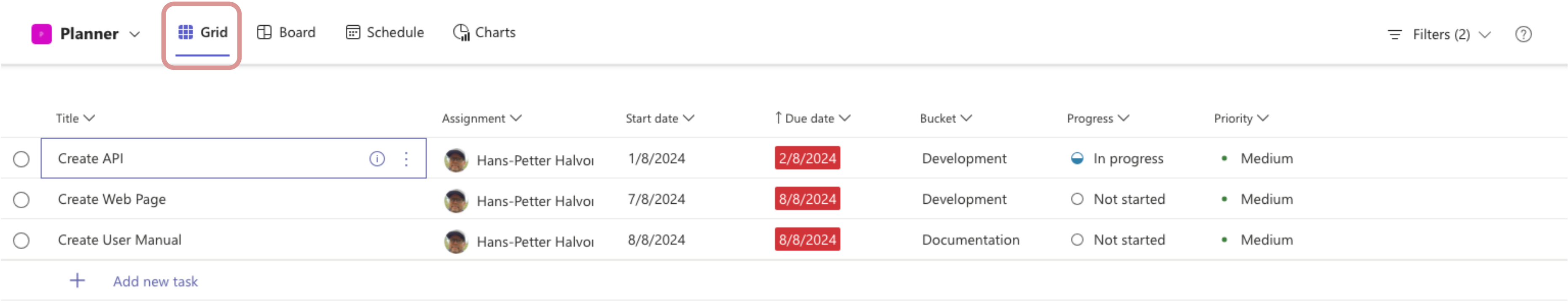
~~Create SDP~~
Planning
Completed by Hans-Petter Halvo...

“**Group by Progress**” mode: You can easily drag and Drop Tasks from “Not started“ to “In progress” and from “In Progress” to “Completed”

“**Group by Bucket**” mode: You can also easily drag and Drop Tasks from one Bucket (Group/Category) to another Bucket (Group/Category)

Grid View

In the “Grid” view you will see alle the Tasks in your Project:

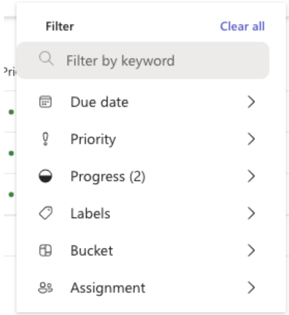


Planner ∨ | Grid | Board | Schedule | Charts | Filters (2) ∨ | ?

	Title ∨	Assignment ∨	Start date ∨	↑ Due date ∨	Bucket ∨	Progress ∨	Priority ∨
○	Create API	Hans-Petter Halvoı	1/8/2024	2/8/2024	Development	In progress	• Medium
○	Create Web Page	Hans-Petter Halvoı	7/8/2024	8/8/2024	Development	○ Not started	• Medium
○	Create User Manual	Hans-Petter Halvoı	8/8/2024	8/8/2024	Documentation	○ Not started	• Medium

+ Add new task

Use the “Filter” to select just some of the Tasks in your Project:



Filter Clear all

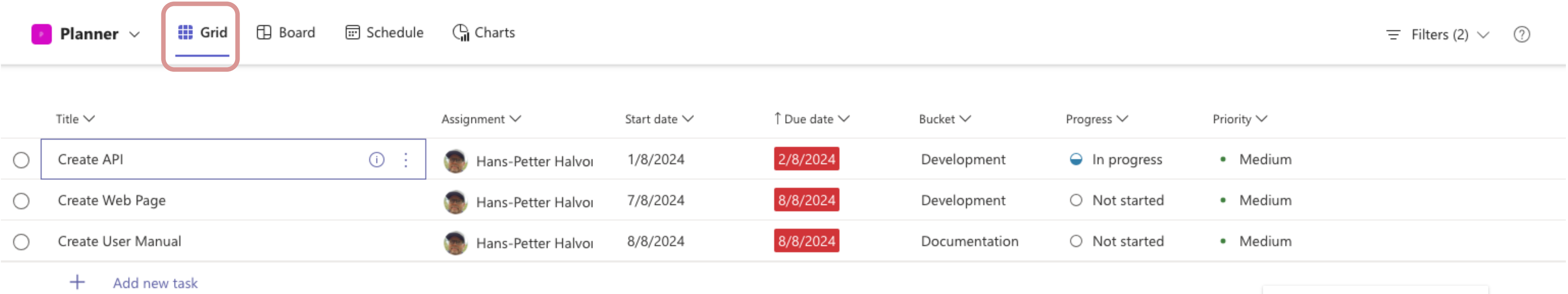
Filter by keyword

- Due date >
- Priority >
- Progress (2) >
- Labels >
- Bucket >
- Assignment >

“Product Backlog” in Microsoft Planner

You can setup a list of Tasks in Microsoft Planner, which will be your Product Backlog or Sprint Backlog (“Grid” view)

You can sort the list by Priority , Start Daye or Due Date, etc.



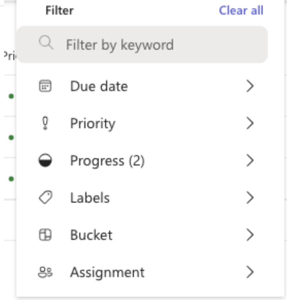
Planner | Grid | Board | Schedule | Charts | Filters (2)

	Title	Assignment	Start date	↑ Due date	Bucket	Progress	Priority
○	Create API	Hans-Petter Halvoı	1/8/2024	2/8/2024	Development	In progress	• Medium
○	Create Web Page	Hans-Petter Halvoı	7/8/2024	8/8/2024	Development	○ Not started	• Medium
○	Create User Manual	Hans-Petter Halvoı	8/8/2024	8/8/2024	Documentation	○ Not started	• Medium

+ Add new task

You can easily add new items (Task) to your list (“Add new task”)

Use the “Filter” to select just some of the Tasks in your Project:



Filter — Clear all

Filter by keyword

- Due date >
- Priority >
- Progress (2) >
- Labels >
- Bucket >
- Assignment >

Planner Tool Summary

- Use "**Taskboard**" (-> "Group by Progress")
  - Not Started -> In Progress -> Finished
- "Buckets" are used for grouping (but not too many groups)
  - Example: Project Management, Development, Documentation
- Use and update the Taskboard daily!
- Each **Task** must and should only be **assigned** to one Person (this person is then responsible for ensuring that the task is performed within the given deadline, "**Responsible person**").
- Set **Deadline** for **All** Tasks
- A Task should have a **duration** of 1-8 hours (Never more than 8 hours, then you have to create several tasks instead)
- Don't create general tasks that never finish (e.g., "Write report"), create small, specific tasks. In this way, you get a clear progression in the project, and this provides motivation!

Meetings

Meetings

Meetings are important part of all projects and typically involves:

- Planning and discussions.
- To synchronize and agree on activities between the team members.
- Track Progress. Go through Project Plans and update them.
- Status updates. The team members need to show during the meetings what they have done since last meeting.
- Questions and Obstacles, discuss and find solutions together

Meeting Types

- Internal meetings inside the development team, 1-2 times a week?
  - Discuss details and Development issues
- Meetings with Supervisor, once a week or twice a month?
- Customer Meetings, once a month?
  - The Customer must have regular updates
  - What have been done?
  - Will the project be delivered on time? Obstacles?
- Scrum Meetings
  - Meetings according to the Scrum methodology
- ..

Meetings

- Microsoft Teams meetings have become the new meeting standard
  - Make sure to learn to use Teams, have meetings, share documents
  - Everyone must have a **camera** and **microphone** (no need to turn this off during the meeting) so that it is almost like a normal physical meeting and that everyone can participate actively in the meeting.
- Notice of Meetings (NoM)
  - Make NoM in "good time" before the meeting! At least one week in advance
  - Write a short **agenda** (bullet points) in the meeting request itself (normally you do not need to create a separate Word document for this)
- Minutes of Meeting (MoM)
  - Always write MoM, but don't necessarily need to spend a lot of time on this. Index entry
  - Make sure to include with Tasks, Responsibilities and Deadlines
  - Should be written ASAP after the meeting is finished (either separate document that is added (and then added to the chat so that everyone involved can read this) in Teams or just message with a few key words in the chat)

Online Meeting Equipment

Examples:

- Earphones
- Headset with built-in Microphone
- External Camera
- Table Microphone and Speaker

Notice of Meetings (NoM)

Norwegian: “Notice of Meeting” 👉 “Møteinnkalling”

- Notice of Meetings are made with Microsoft Outlook/Teams.
- Make NoM in "good time" before the meeting! Typically, at least one week in advance.
- Try to keep the meetings short, typically max 1 hour.
- “Unwritten rules”:
  - Call for a meeting no less than 1 week before the meeting shall take place
  - Don’t call for meetings before 9:00, especially not on Mondays
  - Don’t call for meetings that ends later than 15:00, especially not on Fridays
- **Agenda**: Write a clear Agenda for the meeting.
  - Make sure to prepare a good agenda that focus on solving challenges and problems and not on things that goes smooth
  - Write the Agenda inside the NoM in Microsoft Outlook/Teams.
- **Always Respond to the NoM** (Accepted, Tentative Declined) so we now if you will participate or not. This is Done in Microsoft Outlook/Teams
- Make sure to have an updated version of the **Report** ready for the meeting (typically one hour before the meeting starts)
- Gantt diagram and Planner Tasks should also be updated before the meeting

Minutes of Meeting (MoM)

Norwegian: “Minutes of Meeting” 👉 “Møtereferat”

- What is the purpose with the MoM?
  - Do we need the MoM 2 weeks after the meeting is finished?
  - The MoM is needed so everybody that participated know and agree on the decisions made in the meeting
- MoM should be written ASAP after the meeting is finished (No later than 24 hours after the meeting)
  - Good practice: Write the MoM during the Meeting
- Make sure to include with Work to Do/Tasks, Responsibilities and Deadlines that shall we have agreed on in the meeting.
- Update “Microsoft Planner” with the new Tasks based on the Meeting and MoM
- Send the MoM ASAP to all participants as soon as possible
  - This can be done in different ways, Send MoM on Email, Use the Chat in Microsoft Teams, etc.

Development

Requirements and Design Analysis

- Before you start development you need to figure out what to make and how to make it in a systematic way.
- Start with a Brainstorming session and then use a more systematic approach for the stuff you talked about in the brainstorming session.
- Write down a list of Functional and Nonfunctional Requirements.
- Make Sketches and Diagrams.
- In Scrum we make a Product Backlog, which is a list of all Requirements in a prioritized order.

Diagrams Examples

- System Sketch – Start with a sketch that shows the big picture and not too much technical details
- Architecture Sketch(es) – more detailed sketch(es) showing the technology to be used
- UML Diagrams, like Use Case Diagrams
- Flow Charts
- Database Diagram(s) (Entity Relationship Diagram, ERD)
- Class Diagram(s)
- +++

Development

- Visual Studio and C#
  - https://visualstudio.microsoft.com
- Visual Studio Code
- Python
- LabVIEW
- Database Systems (SQL Server, MySQL, ...)
- GitHub
  - https://github.com
- GitHub Desktop
  - https://github.com/apps/desktop
- Microsoft Azure

Testing

- Testing must be an important part of any development projects.
- You should test your work daily but should also have milestones during the project where you put things together and test everything more thorough.
- It is important to test in a structured way, so make sure to make a simple Software Test Plan.
- Documentation should also be tested.
- “Eat your own dogfood”. Even if you make the system for others, try to use it yourself, in that way improvements to be made.
- Everything works on my PC. Make a separate Test Environment.
- Test the system with the eyes of the Customer, because it is the customer that shall use the system.
- Different types of Testing: Unit Testing, Integration Testing, System Testing and Acceptance Testing.
- Things to Test: Does the system fulfils the Requirements? Good Design? User-friendly? Robust? Acceptable Performance?

Installation and Deployment

- The system/software you create shall be used by the Customer.
- It most be easy for the Customer to Install and start using the system/software.
- Need to create Executable files? Need to create a Setup or installation package? Download from Internet? Download from AppStore? Use in a Cloud platform? Database Installation? How to update the installed software?
- Desktop App or Web App? Need different types of Installation and Deployment.
- The Installation and Deployment process is not something you can start working on the last week, start thinking about it already from day 1!

Report Writing

Table of Contents

Hans-Petter Halvorsen

Report Writing

- Starting already the 1 . week. Fill in the Title, Name(s), create chapter structure, write Introduction (always chap. 1), Background, general Theory (about the topics in the thesis), etc.
- The report must be updated "daily"
- Use the report as the only source (i.e., do not write on 10-20 different documents)
- **IMRaD** structure!
- Basic elements: **Chapter numbering** (max 3 levels), **Figure numbering**, figure title and refer to the texts (DO NOT wait with this, done WHEN figure is entered in the document), References, Spelling and hyphenation, etc.
- Enter **References** continuously both in text and in the reference list - do not wait with this
- Write in the same way and use the same formatting throughout the report, e.g., one cannot use "round" bullet points and another a "dash", one cannot use font size 12 and another 14, etc. => Agree in advance! Make/use a Template and stick to it!
- **Quality Control**!! You MUST check each other's work and what you have written, e.g., quality, typos, formatting, appearance, etc.

Document Writing

All types of documents should have 3 parts

1. Introductory part
   - Introduction
   - Background
   - Theory, Material and Methods
2. Main part
   - Development
   - Results
3. Closure/Ending/Summary part
   - Discussions
   - Further Work
   - Conclusions
   - References

Write Technical Reports in Microsoft Word: https://www.youtube.com/watch?v=ao_eDJOEUkA

Technical Report Structure

This MUST be included:

IMRaD structure:

Introduction

Methods

Results

and **D**iscussions

Typically, you end up with something like this:

- Title Page
- Table of Contents
- Introduction
- Background
- Methods
- Results
- Discussions
- Conclusions
- References
- Appendices

Note! The IMRaD structure indicates what's need to be included, and not necessarily the names of the Chapters. You can, e.g., have multiple chapters that covers the Results part, etc.

Report Structure

- **Title** page – Title of the work + your name (+ relevant figure)
- **Table of Contents** (TOC) – Autogenerated by MS Word
- **Introduction** - This is the first chapter in the report (1. Introduction). Introduce your work and include a system sketch. Include background, problem description, goal, constraints, etc.
- **Theory** – Theory that gives background for the work. One or more chapter.
- **Materials and Methods** – One or more chapter that gives an overview of the Materials (e.g., hardware/software) and Methods used. Methods could be theory, algorithms or frameworks used, etc. In software development projects you may include use case diagrams, flow charts, etc. The chapter(s) can have other names than "Methods".
- **Results** – One or more chapter that gives an overview of the results of your work. Figures and tables are central in this section. Explain and interpret your work and results. In software development projects you could include figures and explanations of your application(s), both GUI and some code snippets. The chapter(s) can have other names than "Results".
- **Discussions** – Overall discussions of the results. In the discussion section, you must explain and interpret your results. What do the results mean? You may also compare your results with the results of others.
- **Further Work** – Give overview and suggest what should be done later
- **Conclusions** - Finally, after the discussions you should include an overall conclusion or summary of the results/findings. In software development projects you may name the chapter "Summary" instead.
- **References** – You always need to include a list of references used inside your report
- **Appendices** – Additional details that does not fit into the main report. Appendix A, Appendix B, etc. Note! No need to add tons of code listing here. The main code structure (including some relevant code snippets) should be documented in the main report. If you think the reader need to see all the code details, link to a GitHub repository or something.

System Sketch

- A System Sketch should always be included in a Technical Report
- The System Sketch gives an overview of the System that has been made, the different parts, and the relationship between those
- You typically put the System sketch in the Introduction part of the report
- PowerPoint is a great tool for making different types of sketches and figures.
- Typically, these sketches may include some Mathematical Expressions. So here you can use the Formula Editor in PowerPoint
- Make System Sketch in PowerPoint: https://www.youtube.com/watch?v=9mmBXFOjV3s

Chapters

- Introduction chapter is always Chapter 1: “1 Introduction”
- Use descriptive chapter names, use a whole sentence, not just one word
- Max 3 levels within a chapter
  - 3 Chapter Name
  - 3.1 Subchapter Name
  - 3.1.1 Subsubchapter Name
- Never start a chapter/subchapter with a figure/bullet points -> Always start with some introduction text

AI - ChatGPT/Copilot

- You can use of ChatGPT, Copilot, or similar.
- Feel free to use it, but with caution!
- Beware of references!!
- AI Tools:
  - ChatGPT: https://chatgpt.com
  - Copilot: https://copilot.microsoft.com
    - Used in the web browser, preferably Microsoft Edge but can also be used in other browsers. If you use another web browser than Edge, you need to use the Bing search engine where Copilot is integrated. Make sure to log in with your Microsoft account.
  - Sikt KI-chat: https://ki-chat.sikt.no
    - Login using your USN account (Feide)

Report Writing - References

- Give credit to authors
- Avoid being caught for plagiarism and cheating
- Add references continuously when writing!
  - Don't wait to do until the end, then it's easy to forget where things were found, etc. -> Easier to get caught for cheating/plagiarism
- MS Word has built-in functionality for this
  - Citation and Referencing with Microsoft Word: https://www.youtube.com/watch?v=IgH7qmLa_L4
  - Also possibly to use 3rd party tools for this

Referencing Tools

- «Kildekompasset» - www.kildekompasset.no
- «Søk & skriv» - https://www.sokogskriv.no
- EndNote
  - EndNote is a program with many features, which is widely used by researchers.
- Zotero
  - Zotero is a similar program that has the most important features that you need.
  - Zotero is easier to learn and has fewer technical challenges than EndNote. If you are going to use a reference management tool for the first time, Zotero is recommended.

Source: https://bibliotek.usn.no/skrive-og-referere/endnote/ and https://bibliotek.usn.no/skrive-og-referere/zotero/

Report Appendices

- Appendix A – Project Description
- Appendix B - Project Plan (Gantt)
- Appendix X - Details that do not fit into the main report
  - Typical proprietary material
  - Things that are not easily found on the Internet, then you use references in the report and the reference list
  - Budget details
  - Detailed drawings and sketches
  - NOT 50 pages of code listing - > then use a link to a OneDrive/Dropbox folder or GitHub
- Any attachments must also be processed and have the necessary quality and include the Title/descriptive heading as well as a textual description of the appendix.

Resources

- «Kildekompasset» - www.kildekompasset.no
- «Søk & skriv» - https://www.sokogskriv.no
- «Skrive og referere» (USN) https://bibliotek.usn.no/skrive-og-referere/
- «Bruke og referere til kilder» (NTNU) https://i.ntnu.no/oppgaveskriving/bruke-og-referere-til-kilder

Document Resources

- Write Technical Reports in Microsoft Word: https://www.youtube.com/watch?v=ao_eDJOEUkA
- Make System Sketch in PowerPoint: https://www.youtube.com/watch?v=9mmBXFOjV3s
- Citation and Referencing with Microsoft Word: https://www.youtube.com/watch?v=IgH7qmLa_L4
- Figures and Equations in Word and PowerPoint: https://www.youtube.com/watch?v=b9f2bb2yn1Y

Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

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

Web: https://www.halvorsen.blog