



**Campus Nyköping**

# SYSTEMUTVECKLINGENS GRUNDER OCH AGIL UTVECKLING

Systemutveckling

# Idag

- Kursintroduktion
- 12 Regler för att lära sig systemutveckling och programmering
  - Kort disussionsgrupparbete med kort redovisning
- Systemutvecklingens grunder
  - Systemutveckling
  - Att arbeta som utvecklare
  - Projektmetoder

# Systemutveckling – Datorns hårdvara

- Hårdvara är den generella termen för fysiska saker. I IT-sammanhang avses de fysiska komponenterna.
- Allting som du kan se och ta på i/på/kopplat till din dator utgör hårdvaran.
- Hårdvaran delas typiskt in i fyra kategorier:
  - 1. Inmatning
    - Tangentbord, kamera, mikrofon, mus, sensorer, rörelse, gps osv
  - 2. Utmatning
    - Skärm, högtalare, skrivare m m
  - 3. Lagring
    - Hårddisk, filsystemet, internminnet
  - 4. Kommunikation
    - Nätverkskabel, Bluetooth, WIFI, NFC

# Systemutveckling – Datorns mjukvara

- Mjukvara (eller programvara) är en generell term som beskriver program och applikationer
- Emellanåt ser vi andra begrepp som program, applikationer/appar, script, kommandofiler, instruktionsuppsättning. Alla dessa faller under kategorin mjukvara
- Mjukvara är utvecklad och konstruerad av en utvecklare (systemutvecklare)
- Mjukvara slits inte
- Installerat på ett datorsystem
- Viktigaste mjukvaran är operativsystemet
  - Microsoft Windows
  - Unix
    - Linux
    - macOS och iOS
    - Android
- [List of operating systems - Wikipedia](#)

# Operativsystem

- Syftet med operativsystemet:
  - Hantering av hårdvara
  - Hantering av processer
  - Minnesallokering
  - Delning av processor och övrigt hårdvara mellan olika program
  - Säkerhet
  - Felhantering
  - Filsystem och lagring
  - Kommunikation mellan processer
  - Abstraktion av hårdvara
  - Utgör ett ramverk för alla övriga program att köras i.

# Filer och mappar

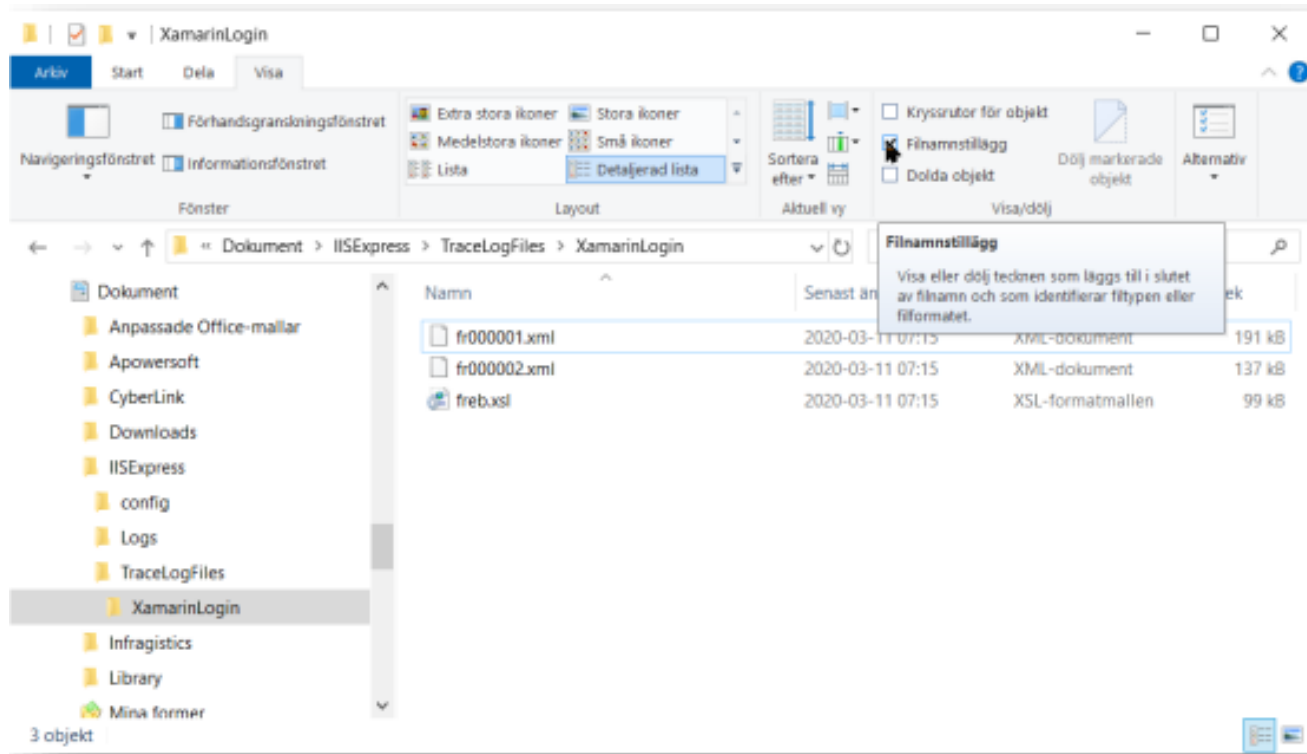
- Du måste följa filnamnskonventioner när du sparar filer
- Skiftlägeskänsligt - stora och små bokstäver är olika
  - Det är sant i Linux- och Unix-variationer, inte i Windows
  - Maximal längd (Windows 260 tecken)
  - Mellanslag tillåtna
  - Siffror tillåtna
  - \ / : \* ? " < > | inte tillåtet
  - Vissa filnamn är ej tillåtna (con, nul, prn)
  - Filnamnstillägg (ändelser) ger ledtrådar om filinnehållet
  - OS använder tillägg för att veta vilket program som skapade filen och filens interna format
    - Den informationen sparas i OS egna inställningsfiler – du behöver oftast inte bry dig

# Filtyper

<b>Tillägg</b>	<b>Dokumenttyp</b>	<b>Applikation</b>
.doc eller .docx	Ordbehandling	Microsoft Word
.xls eller .xlsx	Kalkylark	Microsoft Excel
.ppt eller .pptx	Presentation	MS PowerPoint
.accdb	Databas	Microsoft Access
.gif, .jpg, .png	Bilder	Windows bildvisare
.mp4, .mp3	Video, audio	Windows Media
.zip	Komprimerad fil	Utforskaren / WinZip
.pdf	Portable Document Format	Adobe Acrobat
.htm or .html	Webbsida	Webbläsare

# Mini-Code-along

- Windows standard är att INTE visa tillägg, men vi vill kunna se dem
- Öppna ett fönster med Utforskaren (Windows + E)
- Klicka på Visa i menyn
- Kryssa för rutan Filnamnstilllägg under avsnittet Visa/Dölj till höger





# Filer och mappar – Heta Tips

- Spara INTE alla filer från kursen huller om buller på skrivbordet
- Skapa **EN** mapp på datorn, helst i Onedrive-mappen (om du använder det).
- Döp mappen till “**Kursen**” eller något annat lämpligt.
- I den mappen skapar du underliggande mappar med namnet på just den kursen vi befinner oss i. Idag ska mappen t ex heta “**SystemutvecklingAgil**”
- I den mappen sparar du varje lektions arbete i separata mappar, t ex “**Lektion 1**” eller “**Systemutveckling**”. Imorgon kan nya mappen heta “Lektion2” eller “**SCUMKanban**”
- I den mappen sparar du sedan pdf-er, projektfiler, anteckningar m m så du lätt hittar det senare.

# Säkerhet – Heta tips

- Se till att ditt operativsystem på datorn uppdateras automatiskt.
- Tacka alltid ja till uppdateringar av kända program som du redan har installerat på din dator.
- Ha ett uppdaterat antivirusskydd på din dator.
- Se till att din dator är utrustad med en brandvägg.
- Tänk på vilka webbplatser du besöker och vad du laddar ner, installera inte okända program.
- Skydda alltid datorn med ett lösenord så att den är låst för obehöriga.
- När du har utfört dina ärenden på en webbsida (ex: bankärenden) bör du logga ut från sidan.
- Lämna inte datorn i bilen eller på andra platser där den är lätt att stjäla.

# Backup – Heta tips

- Spara alltid värdefull information på minst två olika ställen.
- Hitta den lösning för säkerhetskopiering som passar dig. Du kanske vill göra säkerhetskopior på en extern hårddisk eller lagra via en webbplats.
- Information som är oersättlig bör du spara på två ställen som inte påverkas av samma riskscenario till exempel en brand eller översvämning.
- Information på mobila enheter bör du vara extra noga med att säkerhetskopiera då risken att tappa bort en mobil enhet är större än en stationär dator.
- Automatisk säkerhetskopiering till molnet är väldigt bekvämt
  - Google Drive, DropBox, OneDrive, ...

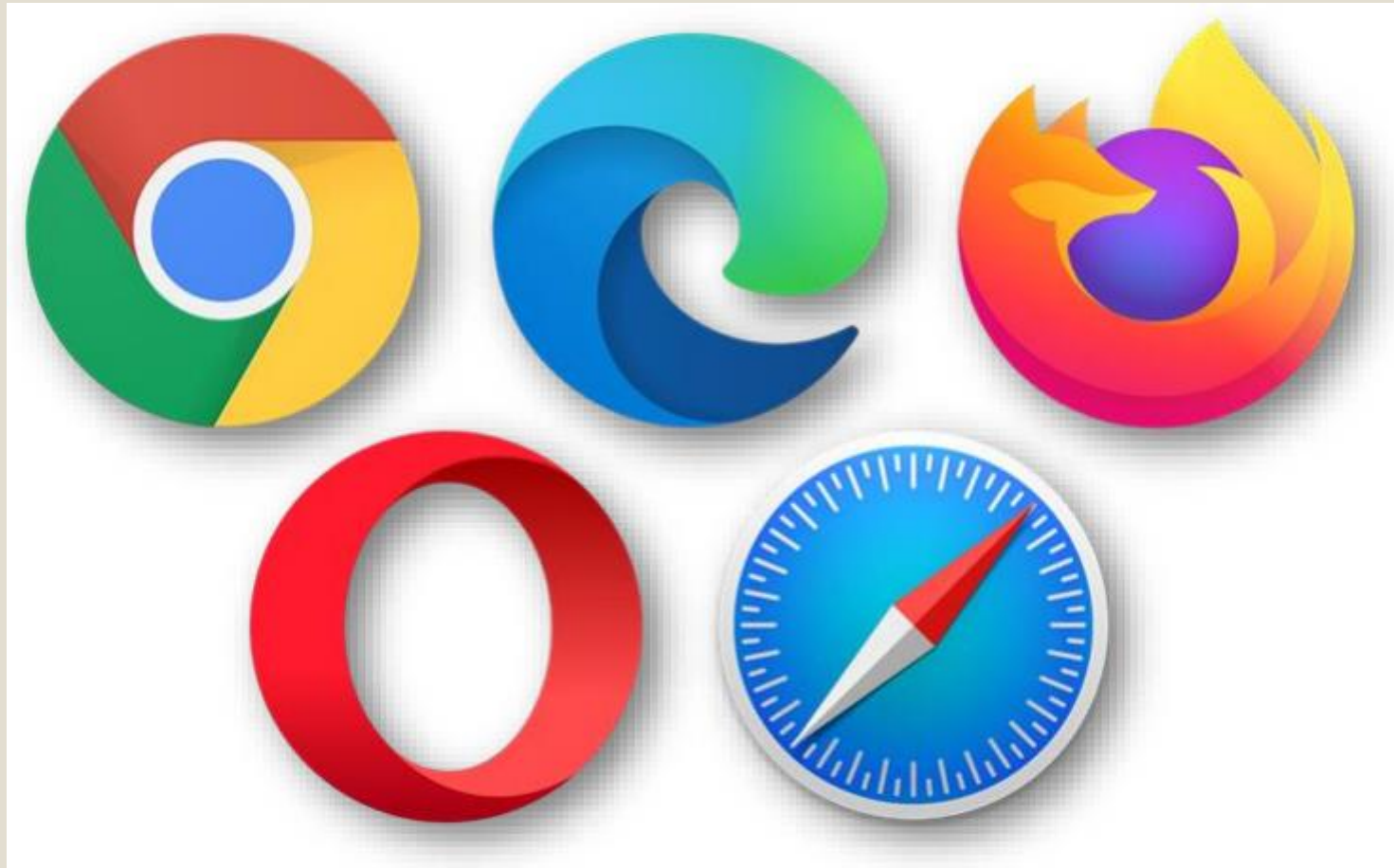
# Säkerhet för kunden

- Ur en utvecklares perspektiv innebär säkerhet att
  - Skydda information/data
    - Själva applikationen är sällan särskilt intressant
  - Systemets/kundens data är heligt
    - Skydda mot olovlig läsning (kopiering/spridning)
    - Skydda mot olovlig skrivning (skapande/ändring)

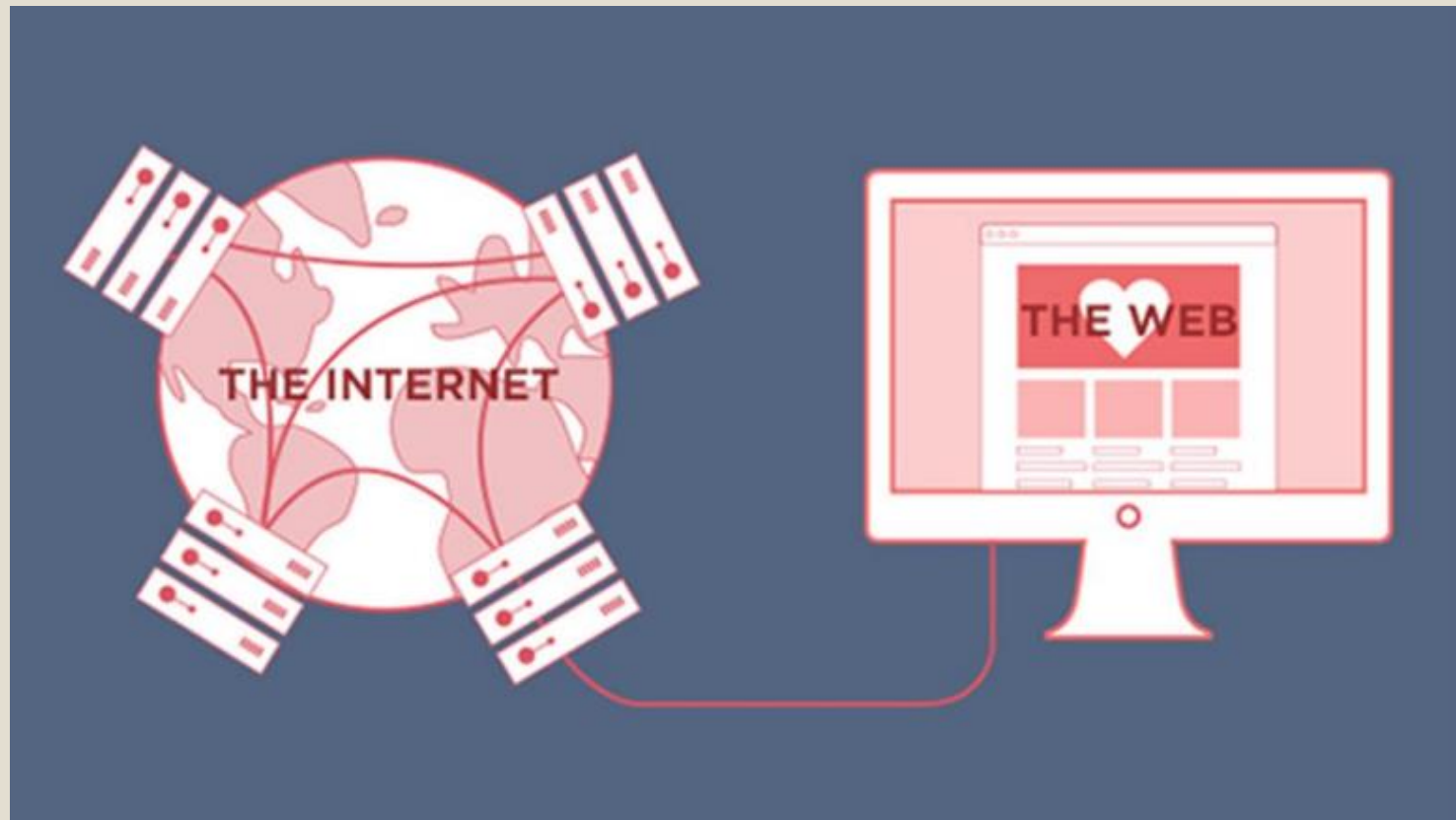
# Mjukvaruexempel

- Industriell automatisering
  - AutoCAD, Sketchup
- Affärsprogram
  - Visma, SAP
- Dataspel
  - Fortnite, Pokemon Go, Elden Ring...
- Kontorssystem
  - MS Word, PowerPoint, Google
- Utbildningssystem
  - Albert, Pluralsight
- Mediaproduktion
  - Photoshop, After Effects, Cubase/Logic
- Kommunikation
  - Webbläsare, Emailklient, Sociala medier...
- Förr i tiden installerade man alla program/applikationer på datorn
- Idag körs de flesta applikationer på nätet/molnet.

# Viktigaste mjukvaran - Webbbläsaren



# Internet och webben



# Webbläsarens uppgift

- Visa HTML sidor
- Spela upp ljud och bild
- Köra lokala skript
- Tillhandahålla funktionalitet för upp och nedladdning
- Tillhandahålla anonymitet
- Parallellitet
- Testfunktioner
- Systemutvecklarens uppgift är att bygga applikationer, som körs på en server, som sedan kan överföras via Internet, så de blir synliga och funktionella i webbläsaren.
- Internet kommer vara er primära källa till kunskap!
  - Tveka aldrig att "Googla" något du vill veta mer om eller vill få förtydligat.
- [Internet Basics: Using a Web Browser \(gcfglobal.org\)](http://gcfglobal.org)
- (Webb kommer vi inte prata om förrän nästa år)



# Molnet

- Ett annat ord för "någon annans dator"
- Vi utvecklare kommer ägna allt mer tid åt att driftsätta våra applikationer i molnet.
- Oftast är det en serverlösning, som utmynnar i en webb-applikation/webbsajt eller en mobil-app.
- Vilka är skillnaderna mot att bygga lokalt installerade applikationer:
  - Ingen skillnad, egentligen
  - Mest webb-baserade system
  - Utveckling lokalt
  - Publicering/driftsättning i molnet
  - Visual Studio har mycket bra stöd för Azure

# Molnet – För och nackdelar

- Köra i molnet
  - + Billigt – du betalar bara för det du använder
  - + Enkelt – någon annan sköter systemadministration och serverdrift
  - + Förutsägbara kostnader
  - - Ingen kontroll över data eller hårdvara
  - - Kräver internetåtkomst
- Kör i egna servrar
  - + Kan vara billigt – om du optimerar din serverpark hårt
  - + Full kontroll över dataåtkomst och -säkerhet
  - - Oftast dyrare
  - - kräver personal för drift och övervakning
  - - Bundet kapital i serverpark
- Den primära drivkraften: PENGAR
- Vi kommer prata molnet i NET Del 2.

# Systemutvecklarens viktigaste länkar

- [Google](#)
- [Stack Overflow - Where Developers Learn, Share, & Build Careers](#)
- [Utvecklarverktyg, teknisk dokumentation och kodexempel | Microsoft Docs](#)

# Systemutveckling = utveckla mjukvara

- En systemutvecklars huvudsakliga jobb är att planera, strukturera, bygga och felsöka program/mjukvara.
- Det är ett kreativt, krävande, komplicerat och roligt jobb...oftast
- Ger vanligen en bra lön, är efterfrågat och innehar en viss status.
- Yrket kräver dock att du är uppdaterad, öppen, strukturerad och kreativ.
- Nackdelar är att det ibland kan bli stressigt, komplicerat och "svettigt".

# IT-branschen

- Det finns en plats för alla inom IT-branschen.
- Allt från megaextroverta säljare till introverta utvecklare
- Alla kompetenser och personlighetstyper behövs.
- Oerhörd kompetensbrist inom alla områden - trots Corona
- Goda möjligheter till fina karriärer - speciellt när ni tar examen

# Yrkesroller

- Det finns väldigt många olika roller med många coola engelska namn.
- Det viktigaste är dock att komma ihåg att tekniken fungerar inte utan rätt människor.
- Här är några exempel:
  - IT-chef (ofta kallad CTO)
  - Account Manager
  - Projektledare (Project Manager)
  - Scrum-master
  - **Utvecklare**
  - HR/rekryterare
  - Designers
  - **UX:are**
  - **Testare**
  - Nätverkstekniker
- Väldigt många kockar - förhoppningsvis god soppa

# Anställning eller konsult

- Troligt att ni börjar som anställda – svårt att bli konsult som junior
- Båda har fördelar:
  - Konsultjobb mer varierande och ofta bättre betalt – speciellt om man driver eget
  - Anställningar är tryggare
- Båda har nackdelar:
  - Konsultjobb är osäkrare
  - Anställningar ofta sämre betalda och beror mycket på arbetsgivaren

# Några “krav” på en utvecklare

- Konsultmässighet:
  - Att förstå att kunden är viktigare än projektet
  - Att samtidigt förstå att kunden inte alltid har rätt
  - Att vara en teamspelare
  - Att vara hel och ren
  - Tänka utanför sin bubbla – affärsnytta!
  - Att alltid lära sig nytt och vara öppen för förändring
  - Att inse sitt värde och utnyttja det på ett ödmjukt sätt
- Varför all detta?
  - För att det gör DITT liv enklare och bättre
  - Hur du agerar kan vara viktigare än vad du kan!
  - För att konstant röra dig mot att göra mer av det du tycker är kul och mindre av det du inte tycker är kul



# Att vara utvecklare



## THE PROBLEM ABOUT BEING A PROGRAMMER

My mom said:

"Honey, please go to the market and buy 1 bottle of milk. If they have eggs, bring 6"

I came back with 6 bottles of milk.

She said: "Why the hell did you buy 6 bottles of milk?"

I said: "BECAUSE THEY HAD EGGS!!!!!"

# Hur får man ordning på soppan?

- Med många kockar i ett utvecklings-team, och ofta komplexa applikationer som ska byggas så måste man bestämma HUR man ska jobba.
- Detta kallas för en Projektmetod.

# Vad är en projektmetod?

- En projektmetodik kan definieras som "ett system med metoder, tekniker, förfaranden och regler som används av dem som arbetar inom ett yrke".
- Gemensamt för de mest använda projektmetoderna är att de är baserade på många års erfarenheter och tester.
- Grunderna lär du dig snabbt, men därefter finns det mängder av utbildningar och certifieringar för att utvecklas vidare. Flera av de mest använda metoderna är så välanvända och etablerade att många helt enkelt betraktar dem som "vanlig projektledning".
- Om du tror att du inte kan något om projektmetoder och projektmetodik så har du därför troligen fel, du har bara inte tänkt på det!

# Varför bör man arbeta med en metod?

- Det finns många goda anledningar att använda sig av standardiserad metodik vid projektarbete.
- En av de stora fördelarna är att projektledaren får tillgång till modeller och verktyg som ger en tydlig struktur för hur arbetet ska drivas framåt, från planering och budgetering till uppföljning.
- Det ger även samtliga inblandade ett gemensamt arbetssätt, vilket bland annat underlättar kommunikationen och effektiviserar det dagliga arbetet. Något som i sin tur skapar bästa möjliga förutsättningar för ett **lyckat slutresultat**, oavsett målsättning.

# Olika projektmetoder

- Så vilka metoder finns det att välja på? Hur många som helst!
- Men det finns några som lyckats bevisa sig vara lite bättre än de andra, några med anor från 1900-talets början.
- **Agil projektledning**
- **Scrum**
- **Kanban**
- Lean
- **Vattenfall**
- Six Sigma (Sex Sigma)
- PMI / PMBOK
- PRINCE2
- Praktisk projektstyrning (PPS)

# Vattenfall

- En av de mer traditionella projektmetoderna. I Vattenfall-metoden rör **sig förbättringar linjärt framåt i en riktning**, ungefär som ett vattenfall. Ursprunget kommer från den tidiga industrin där det var mycket kostsamt att ändra på saker i början av tillverkningsprocessen, vilket innebär att det inte fanns någon anledning att stanna upp processer när de väl satt igång. **Full fart framåt** med andra ord.
- I Vattenfalls-modellen är **dokumentation mycket viktig**. Tanken är att överlämningar mellan olika personer och avdelningar ska kunna ske sömlöst genom tydliga nedskrivna instruktioner.
- Bäst lämpad för:
  - Större projekt som kräver strikta deadlines och tydligt mallade steg, eller projekt som utförts många gånger på samma sätt och där **incitament till förbättringar i processen saknas**.
- Typisk fras:
  - "Läs in dig på underlaget och sätt igång"

# Agil projektledning

- Den kanske mest berömda och på senare år uppmärksammade projektmetoden. Den ligger även till grund för många andra populära metoder. När du jobbar agilt så har du stora möjligheter att **improvisera och anpassa arbetet efter nya förutsättningar**, men utan att förlora strukturen och planen framåt i arbetet. Ordet "agil" kommer från engelskans "agile" som betyder smidig, vig eller lättrörlig – precis som ett bra projekt ska kännas!
- Den agila projektmodellen **skapades ursprungligen för utvecklare av mjukvara** och handlar i grunden om att dela upp ett projekt i flera delmoment, istället för en stor slutleverans. Målet är att arbeta snabbt och flexibelt genom att hela **tiden testa sig fram och tillsammans i projektteamet skapa både problemformuleringar och lösningar**.
- I ett agilt projekt arbetar **självorganiserande och tvärfunktionella team**, helst tillsammans med den beställande kunden.
- Bäst lämpad för:
  - Projekt som kräver flexibilitet och har en komplexitet eller osäkerhet. Till exempel en produkt eller tjänst som påverkas av många externa faktorer.
- Typisk fras:
  - "Vi måste iterera den agila processen"

# SCRUM

- Scrum är en utvecklad variant av det agila arbetssättet. Målet är att utveckla, leverera och upprätthålla komplexa produkter genom **samarbete, ansvarsskyldighet och iterativa framsteg**.
- Rollerna består av en **Produktägare** som ansvarar för resultatet av projektet, en **Scrum Master** som mest liknar en projektledare och **Teamet/utvecklarna** som är de huvudsakliga utförarna av arbetet. Arbetet sker i form av olika händelser där de viktigaste delarna är det dagliga **Scrummötet** som hålls 15 minuter varje dag och **Sprinter** som är kortare delprojekt där teamet under en begränsad tid jobbar intensivt för att färdigställa något.
- Bäst lämpad för:
  - Projekt som består av team på mindre än sju personer som behöver en flexibel strategi för att leverera en produkt eller tjänst.
- Typisk fras:
  - "Vad gjorde du igår för att driva projektet framåt, vad ska du göra idag och vilka hinder ser du för att lyckas med ditt arbete?"



# Kanban

- Även Kanban är baserat på den agila metoden. Konceptet utvecklades redan på 1940-talet av Toyota för att **effektivisera produktionen**.
- I Sverige har, bland många andra, Scania tagit metoden till sig och arbetar med Kanban i hela företaget – från produktionen till kommunikationsavdelningen.
- Det som utmärker Kanban är att metoden är mycket **visuell där arbetsflöden och processer ska synas och vara tillgängliga för alla**. Genom att ha ofta återkommande korta utvärderingsmöten (ibland så ofta som en gång i timmen) ser man till att processen rör sig framåt och att undvika eventuella flaskhalsar.
- Bäst lämpad för:
  - Precis som Scrum så passar Kanban för projekt med mindre team som behöver en flexibel strategi för att leverera en produkt eller tjänst.
- Typisk fras:
  - "Samling vid Kanban-tavlan om 5 minuter."

# Kursen

- Vi kommer under de här tre veckorna (och resten av skolgången) jobba enligt det agila arbetssättet, med Scrum och Kanban.
- Syftet är att förenkla, förtydliga och minska stress och problem med de uppgifter ni ska lämna in.
- Imorgon ger vi oss in i den agila världen...

# Gruppdiskussion - Hårdvara

- Känner du att du har koll på din egen hårdvara?
  - Identifiera vad du har för hårdvara i din dator:
    - Processor
    - Minne
    - Disk
    - Bildskärm och upplösning
  - Fungerar alla prylar?
  - Saknar du någon sak?

# Gruppdiskussion - Mjukvara

- Är du väl bekant med hur du:
  - Skapar och sparar filer + mappstruktur?
  - Skyddar dig mot virus?
  - Uppdaterar ditt OS och dina program?
  - Hur hanterar du lösenord?
  - Gör du backup?
  - Vet du hur du komprimerar och packer upp filer?
    - ZIP och RAR
- Vilka programvaror använder du mest?
  - Vilken kategori av programvara tillhör den
- Är det någon förkunskap du tror dig sakna?

# Gruppdiskussion - Yrket

- Vilka yrkesroller känner du till, i utvecklarbranschen?
  - Något du skulle tycka var roligt att jobba som?
- Vilka av kraven på oss utvecklare tycker du är rimliga/orimliga?
- Vilken persontyp är du?
  - Extrovert/Introvert
  - Trivs att jobba i grupp eller enskilt?

# Gruppdiskussion - Projektmetoder

- Var och en i Gruppen funderar på ett projekt ni tidigare deltagit i, där mer än du själv deltog. Berätta kort om projektet för varandra.
- Berätta också hur ni jobbade i projektet, planering, avstämningar osv. Om ni inte hade någon förbestämt struktur så nämner ni det.
- Hur gick projektet?
- Vad fungerade bra och dåligt?
- Ett projekt kan vara vad som helst, att ni målat ett hus, åkt på en resa, inrett en lägenhet, byggt en stol, lagat en bil, komponerat en sång, planerat en fest eller något annat valfritt.
- Ta helst INTE ett programvaruprojekt som exempel.

# Resten av dagen

- Samlas i era grupper och gå igenom samtliga diskussioner:
  - Hårdvara
  - Mjukvara
  - Yrket
  - Projektmetoder
- Låt alla komma till tals, lär känna varandra och någon av er antecknar de viktigaste sakerna ni kommit fram till.
- Läraren går runt och tjuvlyssnar och klargör eventuella frågor.
- Kl 15.00 samlar vi ihop oss och diskuterar tillsammans vad ni kommit fram till.

# Länkar

- [List of operating systems – Wikipedia](#)
- [Internet Basics: Using a Web Browser \(gcfglobal.org\)](#)
- [Google](#)
- [Stack Overflow - Where Developers Learn, Share, & Build Careers](#)
- [Utvecklarverktyg, teknisk dokumentation och kodexempel | Microsoft Docs](#)
- [Digitalt självförsvar - en introduktion | Internetstiftelsen](#)
- [IT-läraren - Videor \(itlararen.se\)](#)
- [Projektmetoder som ger resultat – vilken väljer du? - Blikk](#)