**Waterfall Development:**

Sekventiell, man modellerar och designar allt i en fas och bygger sedan programet

requirements -> Design -> implementation -> verification -> maintenance

**Iterative Development:**

Man arbetar iterativt och bryter ner arbetet i fler iterationer där man planerar en eller två iterationer framåt. Man bygger de kritiska delarna i projektet först och gör en release som kunden sen bedömer. I en iterativ arbetsprocess är processen inte planerad långt i förväg och kan lätt ändras. Man arbetar med nya requirements, implementation och design, testning och utvärdering i varje iteration.

**Incremental Development:**

Man designar, bygger och testar i mindre portioner som ökar gradvis

**Agile:**

Flexibilitet vid tillverkning av programvara

Agile är ett samlingsnamn för olika lättrörliga sätt att utveckla system på

Agila metoder använder en tidsram för iterativ and evolutionär utveckling. lätt anpassad planering, ökade leveranser och uppmuntran till metoder som är agilla.

Agila, svarar på förändring under utveckling.

**Scrum:**

Scrum är ett sätt att fördela arbetsuppgifter i tiden med bibehållet fokus på levererad affärsnytta. Tanken är att få ordning i projekt med kontinuerliga förändringar i önskemålen om funktion.

**Kanban:**

Är en metod för att hantera kunskapsarbete med betoning på "just in time"-leveranser utan överbelastning av gruppmedlemmarna. I denna process visualiseras uppgiften för deltagarna så att de kan se vad som behöver göras. Gruppmedlemmarna tar sedan arbete från denna backlog.

**Minimum Viable Product:**

Minsta möjliga som behöver göras för att få en fungerande presenterbar produkt

**Work breakdown Structure:**

Projekt delas upp i mindre delar vilket gör det enklare att uppskatta tid för de enskilda delarna och ange uppgifter till rätt personer.

**GANTT chart:**

Ett sätt att hålla reda på projektets nuvarande och olika uppgifter, samt start, deadline och tiden som just den uppgiften får ta. Gantt charts listar också ansvarspersoner för varje uppgift.

**Earned Value Chart:**

Ett diagram vilket har axlarna tid och framsteg där grafer med uppskattad framsteg enligt tid och aktuell framsteg enligt tid presenteras.

**Burn charts:**

Ett diagram vilket flippas till att visa progressen med start på 100% och slutpunk på 0%.

**UML:**

Unified Modelling Language

Ett sätt för utvecklare att kommunicera med andra utvecklare genom olika typer av design diagram som förklarar hur systemet fungera.

**Unified Process:**

UP är ett ramverk som används inom mjukvaruutveckling.

Hantera hög risk uppgifter i tidiga iterationer

Bygg kärnan i arkitekturen i tidiga iterationer

Involvera kunden för att bättre förstå kraven och kundens bild av den färdiga produkten

Testa ofta och realistiskt

**XP:**

En typ av agil utveckling där releases släpps ofta och utvecklas inom en kort utvecklingscykel.

**OOA**/**D:**

Objekt Orienterad Analys och Objekt Orienterad Design används för att bygga förståelse mellan utvecklare och kund. Metoderna används bäst i samband med iterativa utvecklings metoder där man i varje iteration gör en analys och design för att bättre förstå vad som ska bli

Analysen lägger vikten vid att utreda problemet och kraven istället för att leta efter en lösning.

Design lägger vikt vid en konceptuell lösning som möter kraven istället för en implementation.

**User Stories:**

En kort informell beskrivning av en eller flera system funktioner ur en användares perspektiv.

**Use Case:**

En lista av händelser vilka definierar interaktionerna mellan en roll(ex. användare) och ett system.

**TDD:**

Test Driven Design

Handlar om att man som utvecklare skriver testet först och sedan implementationen för metoden som ska testas. Man får tänka igenom vad metoden ska göra och koden blir oftast snyggare och mer lättläst samt att produkten hela tiden måste passera testet, håller man koll på att inga ändringar skadar produkten.

**Code Smell:**

Går även under namnet “Bad smell” vilket refererar till något symptom i koden som kan indikera ett djupare problem.