



Systemutvecklingsprocessen

Att utveckla en app

Vattenfallsmetoden - gammalt sätt



Problem?



How the customer explained it



Manifest för Agil systemutveckling

► <https://agilemanifesto.org/iso/sv/manifesto.html>

Länkar

<https://projektforum.se/det-agila-manifestet/>

<https://andersljungberg.nu/2018/02/08/the-agile-manifesto-borjan-pa-den-agila-eran/>

<https://onbird.se/arbeta-agilt-vad-innebar-det/>

Exempel på agila metoder

- ▶ Scrum
- ▶ Extrem programmering (XP)
- ▶ Lean Software Development

Behov

- ▶ Allting börjar med att det finns ett behov eller någon får en idé
- ▶ Det kan vara ett problem som behöver en lösning



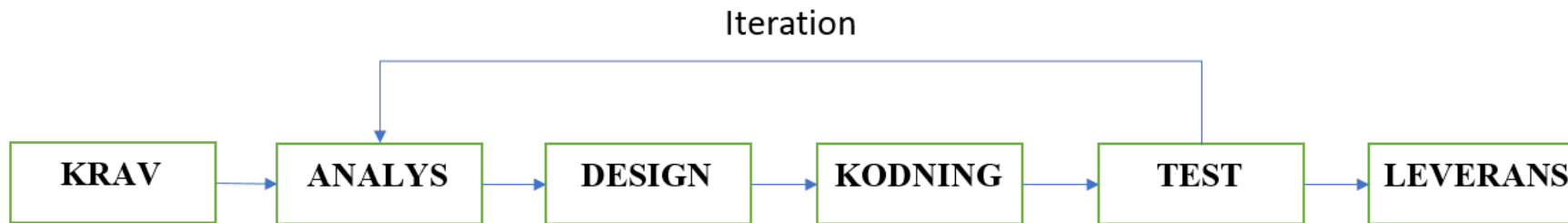
Skapa en bank

Ett bankkonto ska hålla reda på information om namnet på den person som äger kontot och hur mycket pengar som finns på kontot (saldot). Man ska kunna ta ut pengar från kontot och sätta in pengar. Det skall inte gå att plocka ut mer pengar än vad som finns på kontot.



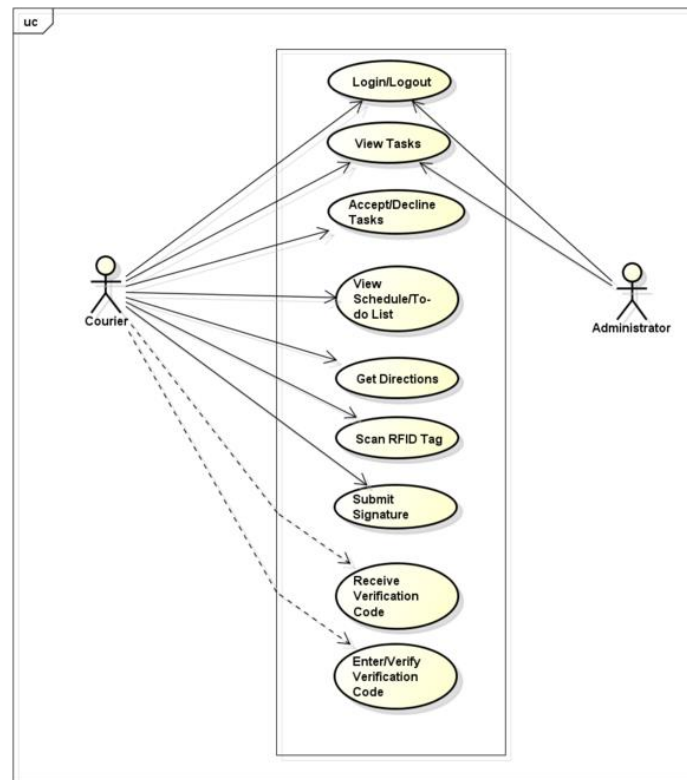
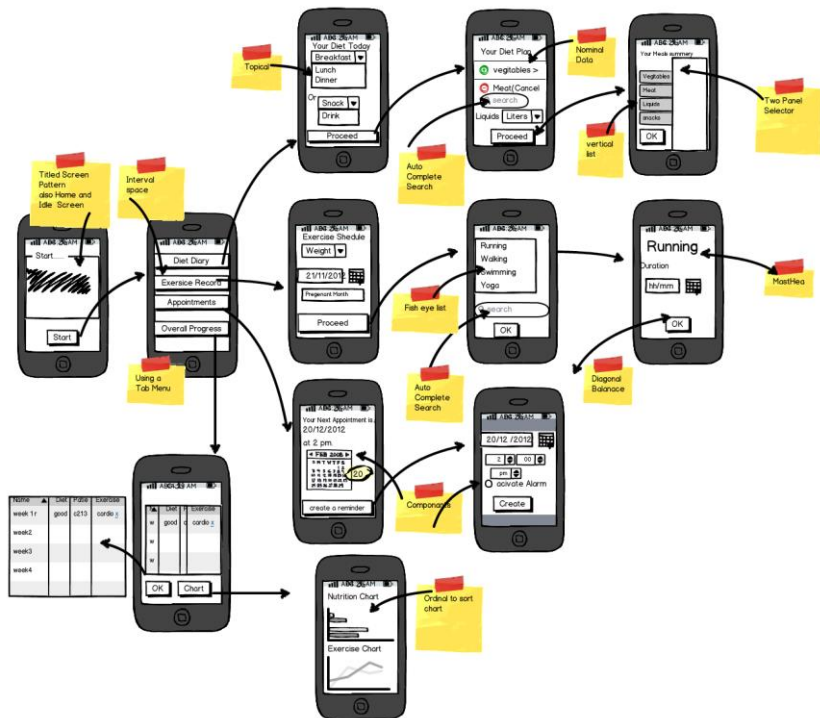
Systemutvecklingsprocessen

- Krav
- Analys
- Design
- Kodning
- Test
- Leverans



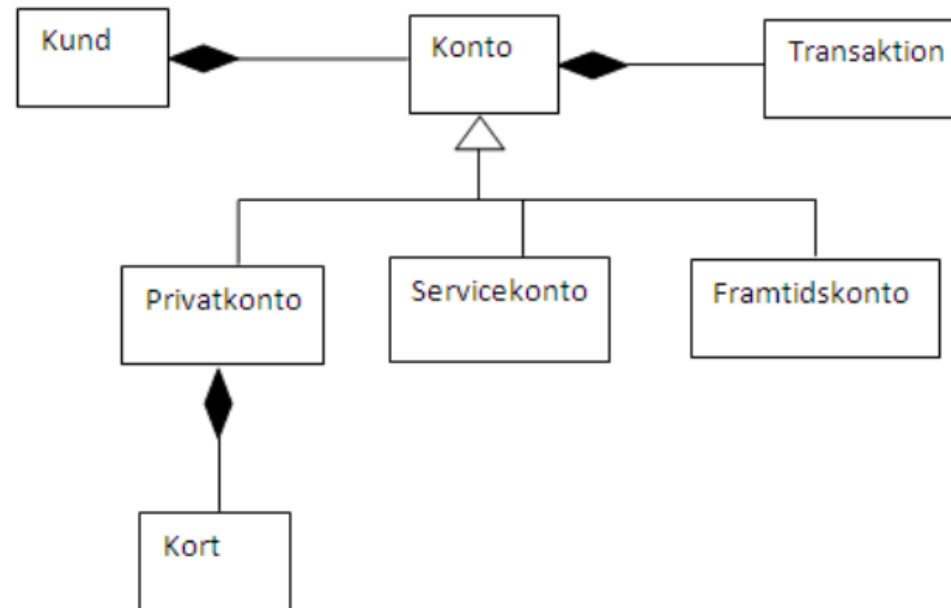
Krav

- Krav på appen, vilken funktionalitet. Vad ska appen göra?
- Även krav på t.ex. svarstider, tillgänglighet, aktualitet, säkerhet (icke funktionella krav).
- Beskrivs t.ex. med storyboards eller användningsfall



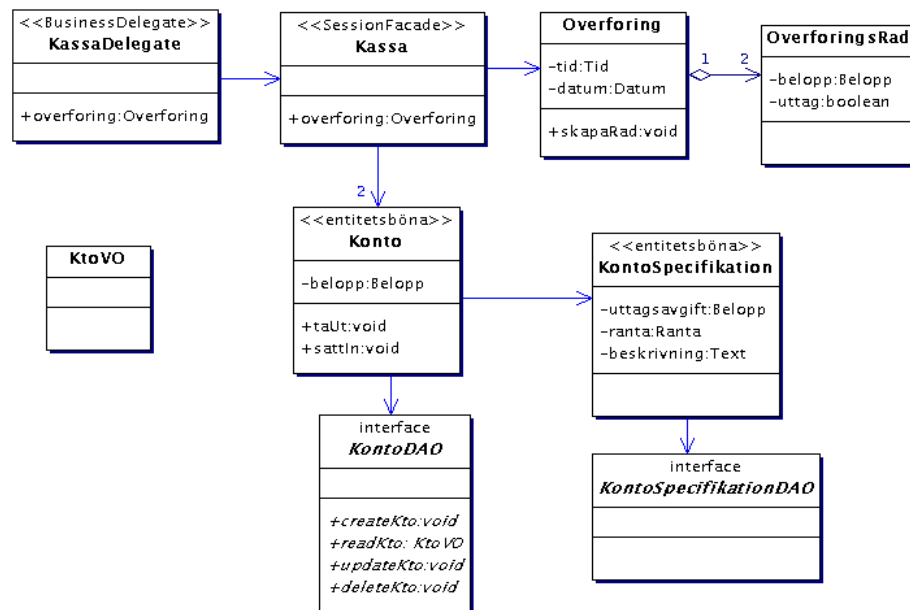
Analys

- Ser systemet utifrån, **vad** ska göras?
- Försöker förstå problemområdet. Vilka objekt finns och relationerna mellan dem.
T.ex. en bank app har objekt som kund, konto.
- Framställer dokument som ska användas i designarbetet, t.ex. klassdiagram och terminologi (språket).



Design

- ▶ Ser datasystemet inifrån, **hur** ska det göras rent tekniskt?
- ▶ Hur objekten kommunicerar med varandra.
- ▶ Tillräckligt med information för kodning, mer detaljerade klassdiagram.
- ▶ Om ny teknik används kan det finnas behov av en POC (Proof of Concept), ett sätt att se om idén kan förverkligas.
Jämför med prototyp (som kan användas i designprocessen)



Kodning

- ▶ Kodning är att programmera.
- ▶ Här skrivs appen i något programmeringsspråk.
- ▶ Dokument från designen ska göras om till programmeringskod.



```
main.xml x
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:id="@+id/frameLayout">

    <GridLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/gridLayout"
        android:layout_gravity="center"
        android:orientation="vertical">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Login:"
            android:id="@+id/textView"
            android:layout_row="0"
            android:layout_column="0"
            android:paddingRight="20dp" />
```

The screenshot shows the Android Studio IDE with the 'main.xml' file open. The 'Text' tab is selected at the bottom, displaying the XML code for the layout. The code defines a FrameLayout containing a GridLayout, which in turn contains a TextView. The TextView is configured with the text 'Login:', a specific ID, and a padding. The interface also shows a 'Design' tab and a vertical toolbar on the left.

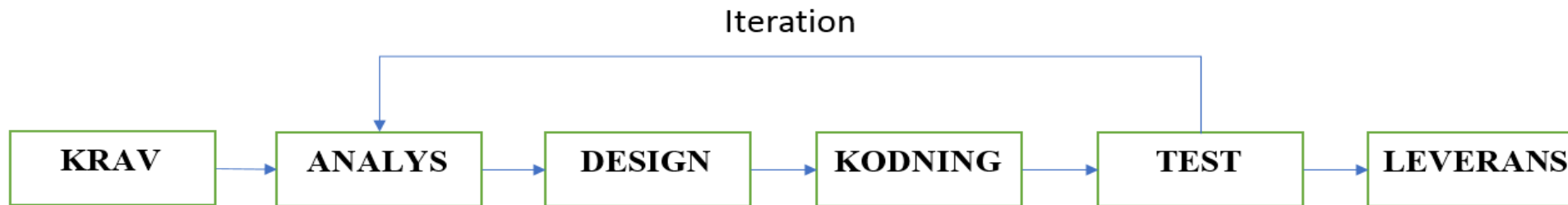
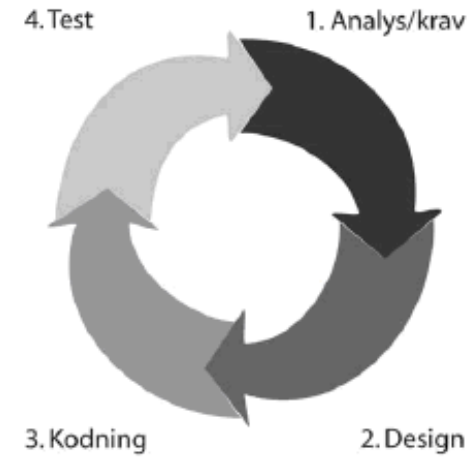
Test

- ▶ Kontrollerar kvalitén, att appen fungerar som det är tänkt.
 - Ett fel i en app brukar kallas bugg
- ▶ Använd användningsfallen/story boarden från kravanalysen och testa.
- ▶ Tester utförs av programmerare, testare och några användare



Iteration

- Bygga appen bit-för-bit
- Varje iteration tillför ny funktionalitet
- Varje iteration kan medföra förändringar/förbättringar. Det går inte att bestämma allt i förväg.



Leverans

- ▶ Installera på användarnas datorer/mobiler
- ▶ App store och Google Play
- ▶ Marknadsföring
- ▶ Support
- ▶ Det finns ingen kostnad att kopiera en digital produkt/tjänst (t.ex. appar). Jämför med kostnaden att tillverka en fysisk produkt (t.ex. brödrost).

