



SKAPA ETT SPEL

HUR KOMMER VI IGÅNG?



GRUNDLÄGGANDE PROCESS

IDÉ

- Vision
- Historia
- Yttre inspiration
- Illustrationer
- Brainstorming
- Eller vad som helst...

PROGRAMMETS EKOSYSTEM (IT-ARKITEKTUR)

- Hur ska programmet vara uppbyggt?
- Vilka delar finns med i ekosystemet?

IT-ARKITEKTUR

- IT-arkitektur är designprinciper, ramverk, teknologier och standarder som används för att bygga och hantera IT-system effektivt.
- Med andra ord:
IT-arkitektur handlar om hur vi bygger och hanterar IT-system på ett smart sätt. Det handlar om att bestämma hur olika delar av systemet ska prata med varandra och hur information ska röra sig igenom systemet. Det handlar om att välja hur man gör saker samt vilka verktyg och regler som ska användas.

IT-ARKITEKTUR KAN INNEFATTA

1. **Systemkomponenter:** Hårdvaruenheter (servrar, lagringssystem och nätverksutrustning) och programvarukomponenter (operativsystem, databaser och applikationer).
2. **Dataarkitektur:** Design och hantering av data inom systemet, samt datamodeller, databaser etc.
3. **Applikationsarkitektur:** Design och struktur av appar inom systemet, inklusive användning av programmeringsspråk, ramverk och utvecklingsmetoder.
4. **Nätverksarkitektur:** Design och konfiguration av nätverksinfrastrukturen, inklusive protokoll, kommunikationskanaler och nätverkssäkerhetsåtgärder.
5. **Säkerhetsarkitektur:** Designa säkerhetsåtgärder och protokoll för att skydda IT-systemet mot obehörig åtkomst, dataintrång och andra säkerhetshot.
6. **Integrationsarkitektur:** Definierar hur olika komponenter i IT-systemet integreras och kommunicerar med varandra, samt användande av API:er och meddelandesystem.

DEFINIERA DELARNA

- Skissa upp vad som ska hända i ditt program.
- Från det visuella till det praktiska
- Hur ska programmet se ut?
- Vilka bibliotek behövs?
- Vilka funktioner eller klasser ska du ha?
- Ska du ha databaser eller filhantering av något slag?

UTVECKLING

Arbeta del för del och försök inte att utveckla alla funktionalitet samtidigt.

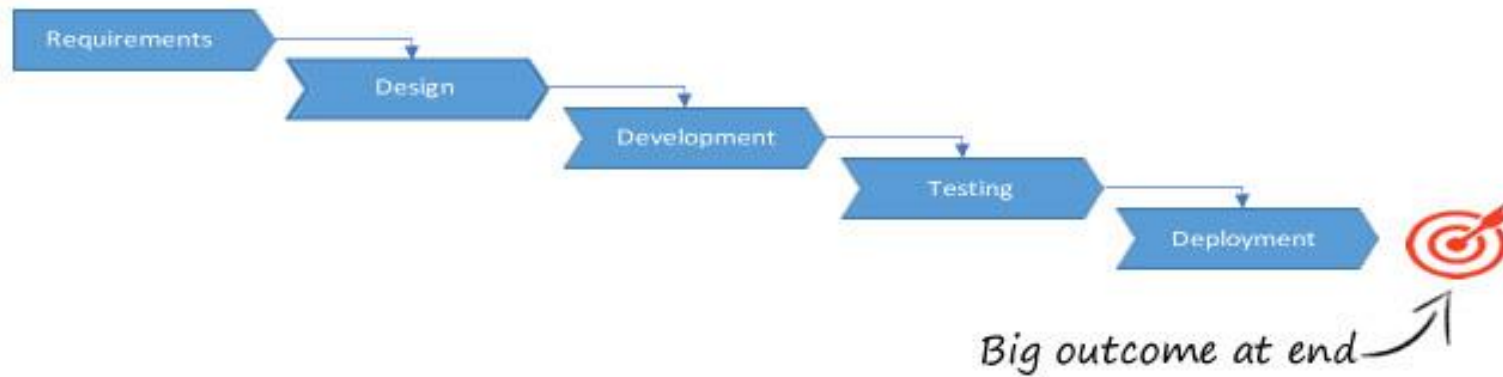
Listor

Bryta ner uppgifter

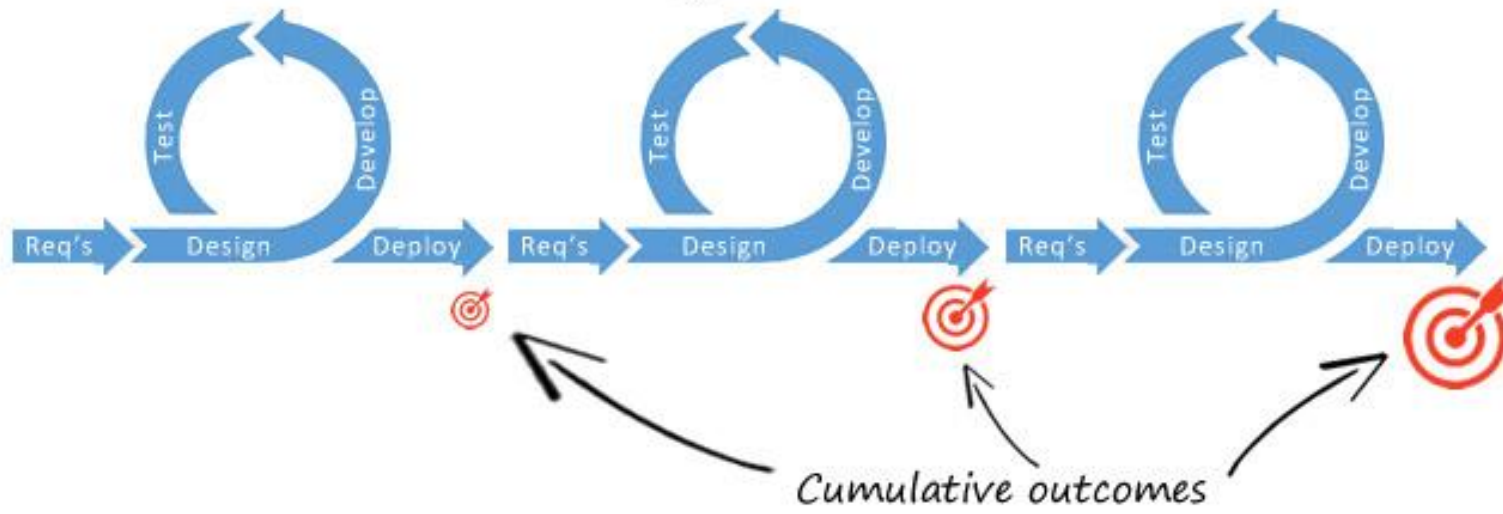
Ta hjälp då man fastnar

AGIL OCH VATTENFALL

Vattenfallsmetod



Agil metod



TESTNING

- Testning är något man gör kontinuerligt genom projektet.
- Börja med att skapa en MVP – Minimum Viable Prototype – den minsta del som behövs för att spelet ska vara spelbart.
- Testa själv (eller låt någon annan testa om du vill)
- Lös buggarna innan du går vidare med att programmera mer



Hur kommer
jag på en idé?

Hur ser
ekosystemet
ut?

Definiera
programmets
delar

Utveckling

Testning

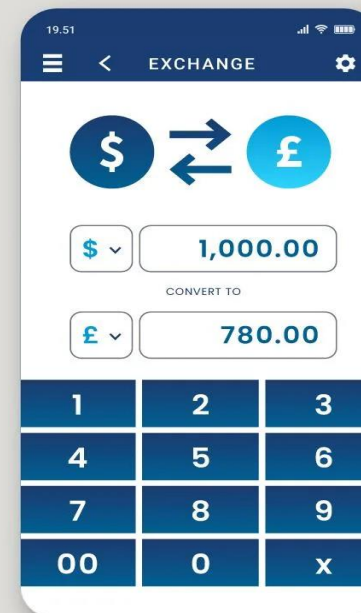
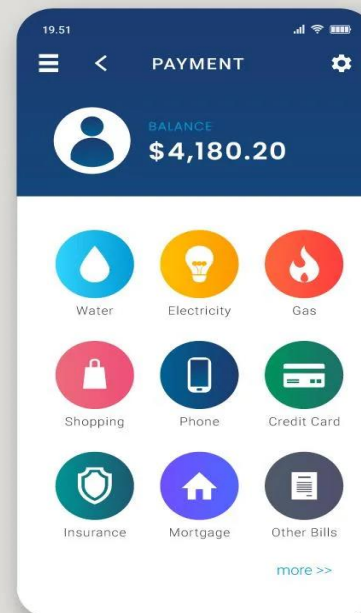
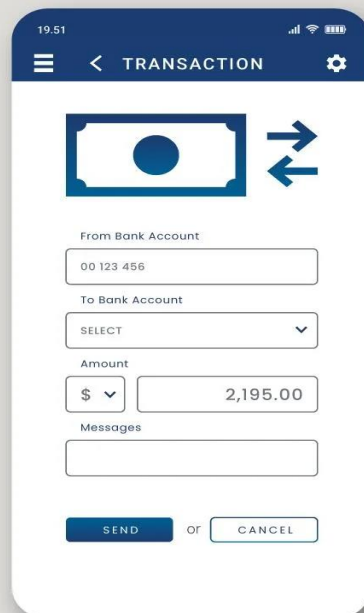
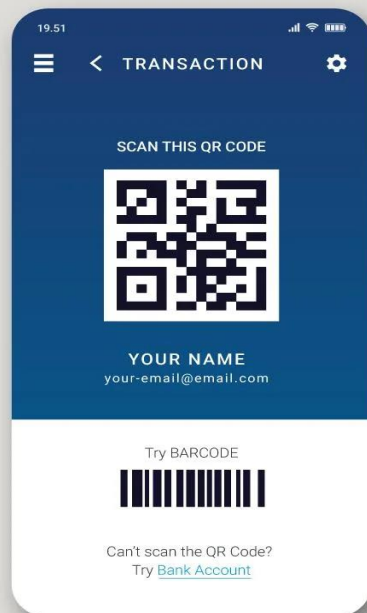
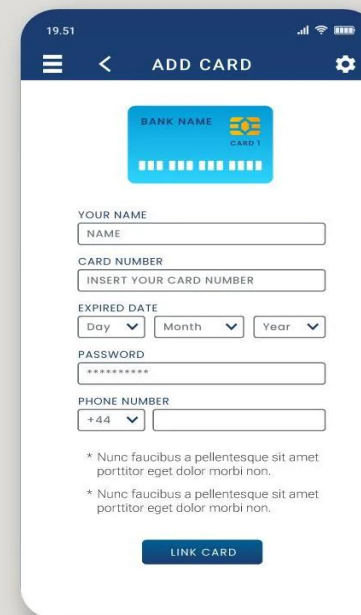
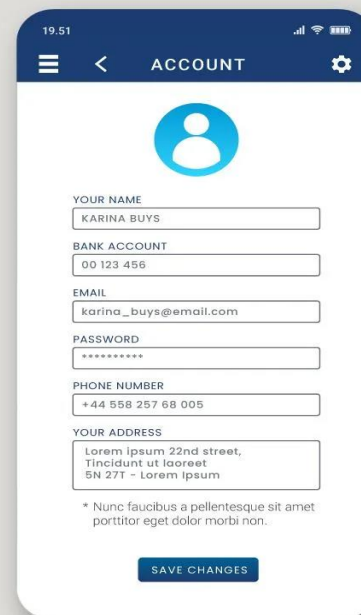
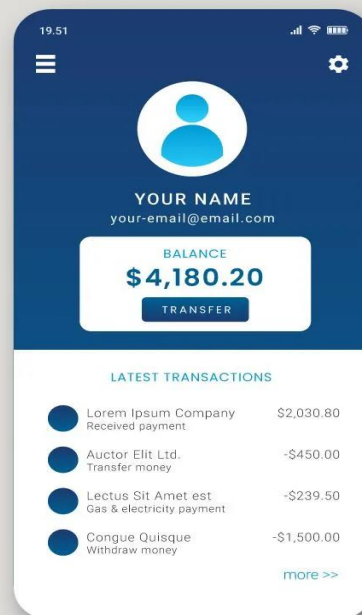
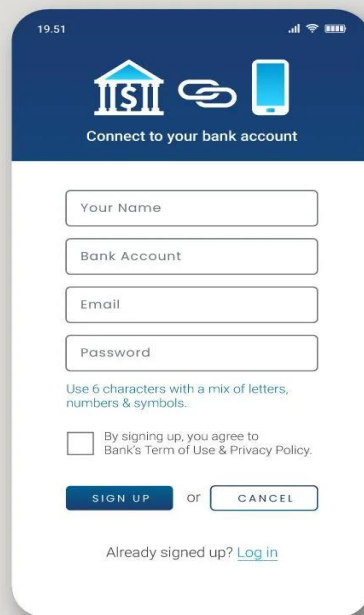
FORMGIVNING

FORMGIVNING / DESIGN

1. Målgrupp – vem ska använda appen?
2. Skissa och planera gränssnittet.
3. Enhetlighet
4. Minimera antalet steg
5. Testa designen
6. Anpassa gränssnittet för olika enheter och skärmstorlekar.
7. Responsiv design
8. Tillgänglighet
9. Fortsätt att förbättra designen baserat på feedback och användardata.

PROTOTYPER

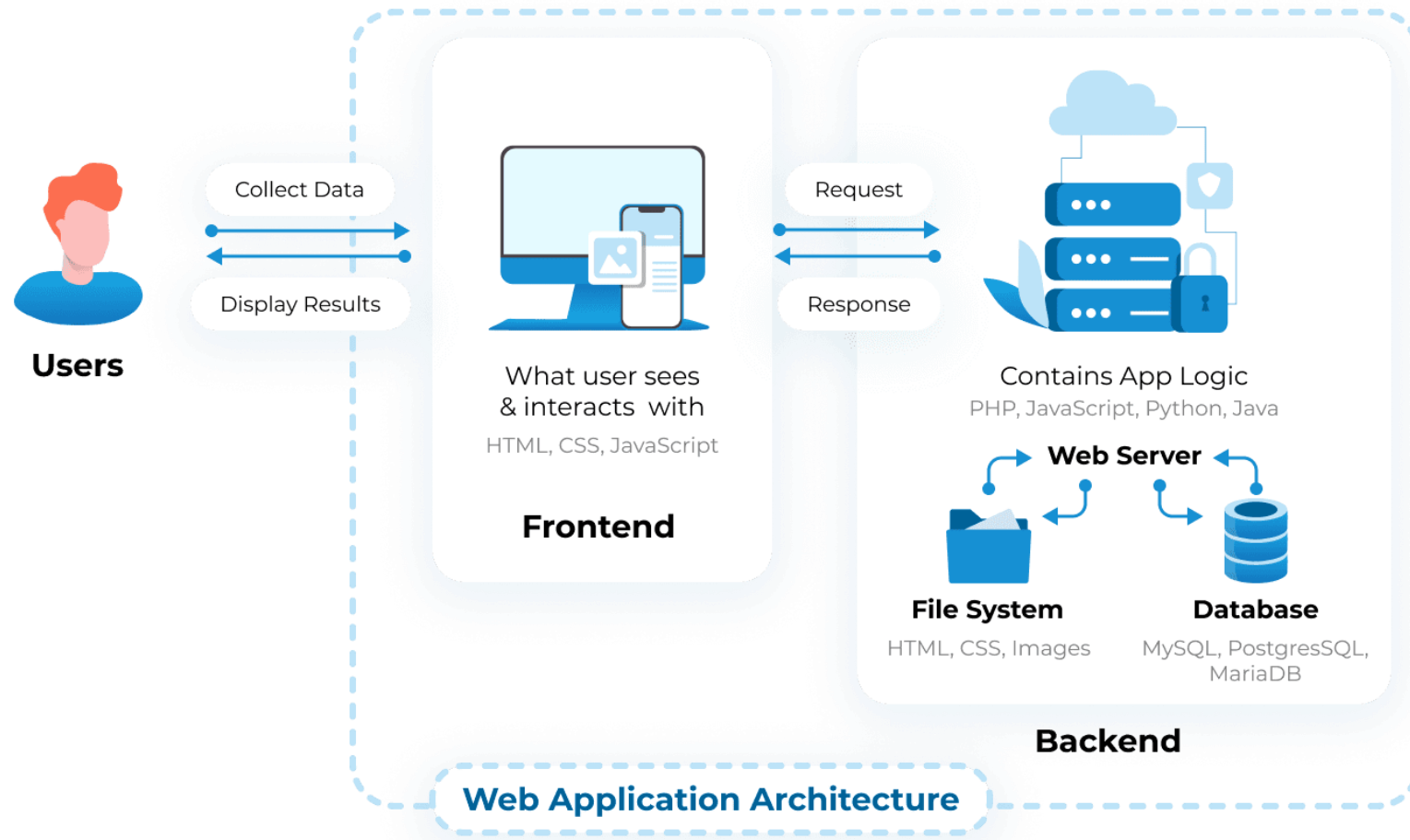
- 1. Sketch: Sketch är ett vektorbaserat designverktyg som är populärt bland UI/UX-designers.
- 2. Adobe XD: Adobe XD är en annan populär plattform för att designa och prototypa appar.
- 3. Figma: Figma är ett kollaborativt designverktyg som gör det möjligt för flera användare att samarbeta i realtid.
- 4. InVision: InVision är en plattform för digital produktutveckling som inkluderar verktyg för att skapa interaktiva prototyper.



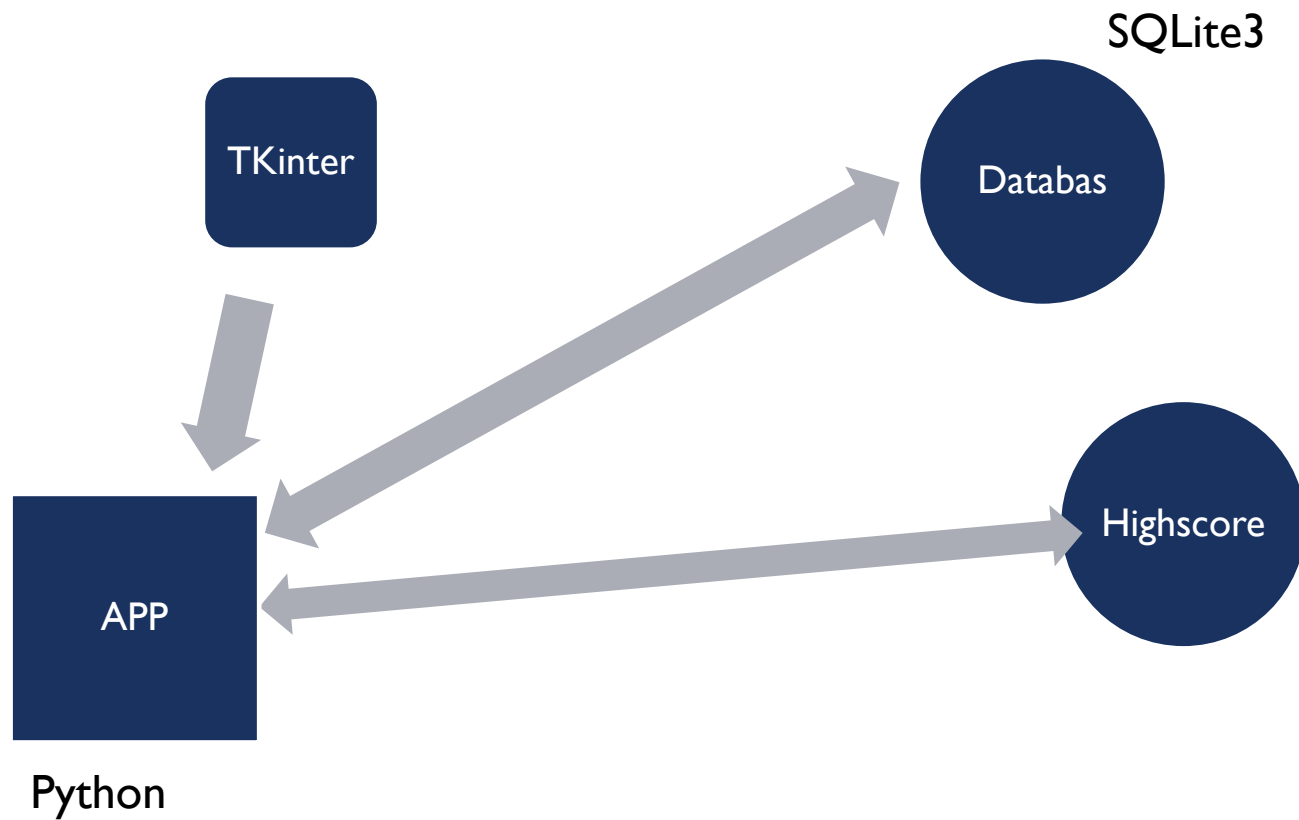


HUR SER EKOSYSTEMET UT?

EKOSYSTEMET



HUR DU KAN UTFORMA DITT EKOSYSTEM





DEFINIERA PROGRAMMETS DELAR

DELAR

Klasser

- Bok
 - Metoder
 - är_tillgänglig
 - låna
 - lämna_tillbaka
- Medlem`
 - Metoder

- Bibliotek
 - Metoder
 - lägg_till_bok
 - hitta_bok

Funktioner

- main

VILKA DELAR?

moduler	main	klasser	metoder	funktioner



PSEUDOKOD

VAD ÄR PSEUDOKOD?

Pseudokod är ett verktyg som används för att beskriva logiken och stegen i en algoritm på ett enkelt och förståeligt sätt. Till skillnad från riktig kod, är pseudokod inte bunden till syntaxen av ett specifikt programmeringsspråk. Istället fokuserar det på att använda enkel syntax och naturligt språk för att förklara hur en algoritm fungerar.

PLANERING OCH DESIGN

- Pseudokod används ofta i planerings- och designstadierna av programvaruutveckling för att bryta ner ett problem och utveckla lösningar innan den faktiska koden skrivs. Det ger utrymme att experimentera med olika tillvägagångssätt innan man binder sig till en specifik implementation.

KLARHET OCH FÖRSTÅELSE

- Pseudokod är lättare att förstå än komplex kod i ett specifikt programmeringsspråk, särskilt för personer som är nya inom området eller inte är bekanta med det specifika språket. Det gör det lättare att kommunicera och förklara algoritmer för andra programmerare.

ABSTRAKTION OCH GENERALISERING

- Pseudokod möjliggör abstraktion och generalisering av algoritmer, vilket innebär att man kan fokusera på de övergripande stegen i en algoritm utan att fastna i detaljerna i ett specifikt språk eller implementation.

ETT LITET EXEMPEL

1. Läs in det första talet (num1)
2. Läs in det andra talet (num2)
3. Beräkna summan av num1 och num2 och spara resultatet i en variabel (summa)
4. Skriv ut summan

ETT MER AVANCERAT EXEMPEL

funktion bubblesort(lista):

 n = längden av lista

 bytplats = sant

 medan bytplats är sant:

 bytplats = falskt

 för varje i från 0 till n-2:

 om lista[i] > lista[i+1]:

 temp = lista[i]

 lista[i] = lista[i+1]

 lista[i+1] = temp

 bytplats = sant

 returnera lista

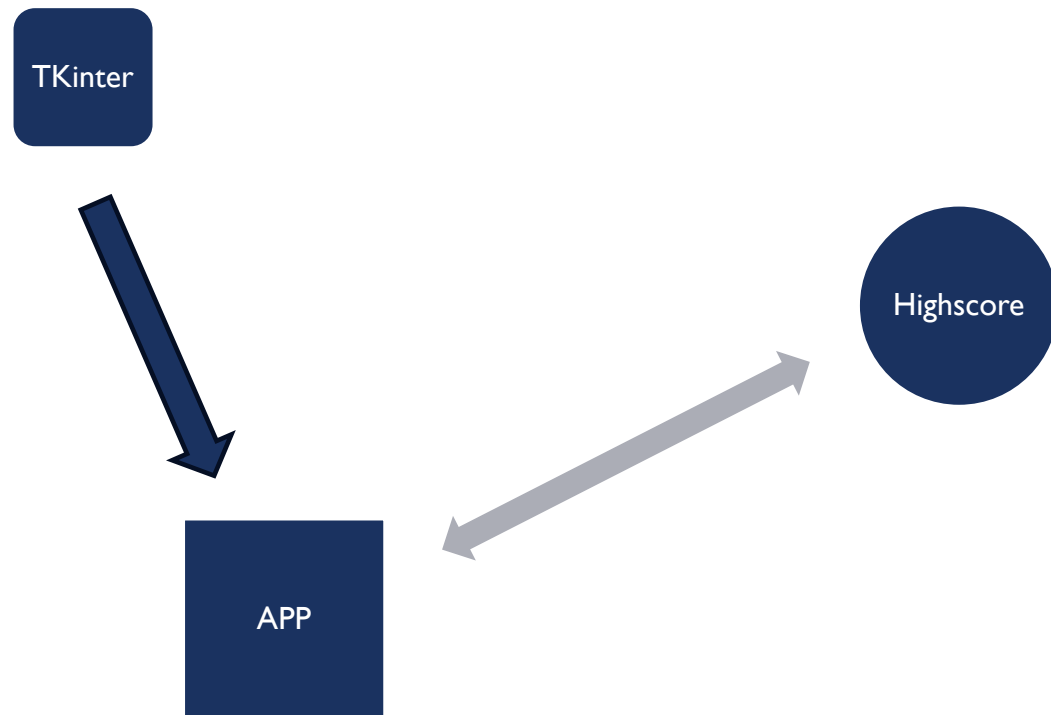


STEG FÖR STEG

ETT ENKELT POINT AND CLICK-SPEL

- Ett enkelt fönster
- En röd boll som studsar runt i fönstret
- Klickar man på en boll händer något
 - Klickar man på röd boll slumpas det upp en ny boll av slumpmässig storlek och färg
 - (Färger är röd, grön, blå, gul)
 - Klickar man på en grön försvinner en boll (finns det bara en röd boll kvar försvinner inte den röda bollen)
 - Klickar man på en gul boll blir den grön
 - Klickar man på en blå boll så byter den håll
- För varje klick får man 1 poäng
- En knapp i fönstret avslutar spelet men frågar först om du vill spara ditt resultat
- Om du vill spara får du ange ditt användarnamn
- Resultatet sparas i en textfil "Highscore"
- En knapp i fönstret säger Highscore.
- Om man klickar på den öppnas ett nytt fönster med information från textfilen Highscore
- Man sorterar listan från textfilen och visar de tre högsta highscoren i fönstret

HUR DU KAN UTFORMA DITT EKOSYSTEM



MVP

moduler	main	klasser	metoder	funktioner
TkInter	Skapa fönstret Skapa boll Flytta boll	Boll	Skapa boll Flytta boll	Flytta boll
Random				