

Back to basics!

Android C3L



Vad är UI-utveckling?

User interface (UI) design

- Hur produkten ser ut och uppfattas
- T.ex. färger, ljus, kontrast, format på grafiska element, bilder, typsnitt och textformattering

User experience (UX) design

- Handlar om interaktion mellan människa och produkt
- Fokus på funktion och upplevelse

Varför är det viktigt?

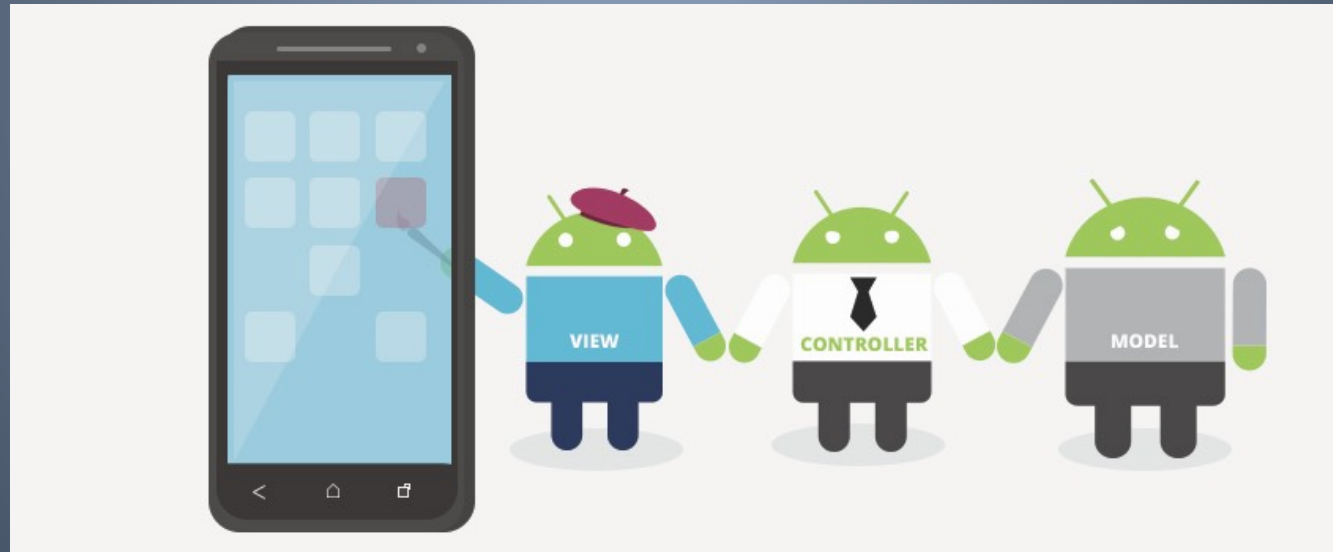
Diskutera två och två:

- Försök komma på exempel på en bra och en dålig app som ni använt.
- Varför var de bra resp. dåliga?



MVC (Model, View, Controller)

- Ett arkitekturpattern som används ofta inom UI-utveckling
- Är ett sätt att separera vår vylogik från vår affärslogik



MVC (Model, View, Controller)

- Controllern är den del som ligger mellan vyn och modellen, och hanterar kommunikationen mellan dessa. Ansvarig för interaktion
- Senaste dagarna
 - vår controller = vår activitykod
 - Vår vy = xml-layouts
 - Vår modell = pojos och det user repository som vi skrev igår
- Modellagret är ofta uppbackat av någon typ av datakälla

Synkron vs. asynkron programming

Synkron

“My boss is a busy man. He tells me to write the code. I tell him: Fine. I get started and he's watching me like a vulture, standing behind me, off my shoulder. I'm like "Dude, WTF, why don't you go and do something while I finish this?" He's like: "No, I'm *waiting right here* until you finish."”

Asynkron

“The boss tells me to do it, and rather than waiting right there for my work, the boss goes off and does other tasks. When I finish my job I simply report to my boss and say: "I'm DONE!"”

Callback

- En funktion som körs när den asynkrona operationen är färdig
- Implementeras i Java genom någonting som kallas SAM (Single Abstract Method) interfaces
- SAM är interfaces med en void-metod som tar in resultatet som en parameter

Threads/trådar

Ett exekveringskontext som innehåller all information som processorn behöver för att exekvera en ström av instruktioner.

ThreadLocalStorage

- Processorn gör det vi säger åt den att göra och kan bara göra en sak i taget. Eftersom vi har fler program och processer igång samtidigt så måste operativsystemet dela upp processor-tiden på de olika processerna.
- X antal instruktioner från din process körs och efter det körs någon annan process instruktioner. Hur vet processorn om vart den ska börja exekvera din process igen senare?
- Detta görs via ThreadLocalStorage, där information lagras om vad som har körts och vart vi ska börja igen.

ThreadLocalStorage

- Exempel:

Om vi läser en bok och behöver pausa, men vill kunna börja läsa på exakt samma ställe som vi slutade så kan vi skriva ner vilket kapitel, vilken sida, rad och ord vi var på när vi stannade. Med hjälp av den informationen så kan vi sen fortsätta på exakt samma ställe som vi slutade.

Vår vän som också vill läsa samma bok kan använda sig av samma teknik och vi kan läsa boken samtidigt, så länge vi inte göra det bokstavligen vid samma millisekund.

Looper och meddelandekö

- Någon som kan gissa/förklara?
- Det är en tråd som körs under obestämd tid. Den kommer inte att avslutas som en vanlig tråd gör, när den inte har några instruktioner kvar att köra. Den ligger istället och väntar på nya meddelanden i meddelandekön. Den arbetar endast med ett meddelande i taget.

UI-thread

Exempel på en tråd med looper och meddelandekö är UI-tråden.

Här sker all rendering och användarinteraktion med våra vyer.

Det är den tråd som vi under de senaste dagarna har kört all vår kod på.

En sak som ofta händer vid UI-utveckling är att UI-tråden fryser och därför inte går att interagera med. Renderingen kan bli konstig och vårt UX blir lidande, vad tror ni att utvecklaren har gjort för att förtjäna det?

Diskutera!

Hur kan vi lösa det?

ASYNKRON PROGRAMMERING!

AsyncTask

Den här klassen möjliggör för dig att utföra operationer i bakgrunden och publicera resultat i UI-tråden utan att behöva manipulera trådar

Handlerthread

Använda oss av många mindre nedladdningar från nätverket, t.ex. av små thumbnails, eller visning av subtitles där det krävs lite beräkning hela tiden för att visa upp rätt text.

Slut på the basics

- Grunden ytligt
- Ingen koddemo nu, kommer att använda vissa delar senare
- Läs gärna på mer själva, länkar kommer finnas på Wikin.