

IDATT2506 Prosjektøving

Oppgaven

Denne prosjektøvingen skal gjøres av alle. Merk at spørreundersøkelsen nedenfor også er en del av øvingen (gjøres til slutt).

Oppgaven går ut på å lage en enkel app for handle/todoliste. "Enkel" i det henseendet at det skal være raskt å føre opp nye innslag, slik at appen gjør jobben sin fortest mulig og lar deg fortsette hva enn du holdt på med.

De følgende kravene settes til oppgaven:

- Man skal kunne opprette nye lister (som vil typisk være for type lister, som "dagligvarer", "hytteutstyr", "handle på nett" osv).
- Man skal kunne slette eksisterende lister
- Øverst skal man ha en oversikt over listene, f.eks. ved bruk av tabs eller annen måte du finner hensiktsmessig.
- Plassert under oversikten over listene, skal man ha ett input-felt for å kunne legge ny innslag inn i den stående lista. Dette skal bare være en tekststreng.
- Når setter fokus på feltet, skal keyboardet dukke opp. Man skal kunne avslutte inntasting og legge innslaget i lista ved å trykke på "enter". Keyboardet skal bli stående oppe og fokus skal gå tilbake til tekstfeltet, slik at man kan legge inn flere innslag på rad.
- Under input-feltet skal man ha listeinnslagene. Disse skal være organisert slik at øverst har man ugjorte ("ukjøpte") innslag øverst, mens nederst er innslagene som er markert som ferdig ("kjøpt").
- Man skal kunne markere et innslag ferdig/"kjøpt" ved å tappe (korttrykk) på det. Det skal da flyttes fra uferdig til ferdig ("ukjøpt" til "kjøpt")
- Listene skal lagres til fil på JSON-format, fortrinnsvis ei fil pr. liste, men det er ikke et absolutt krav. Hver gang man legger inn et innslag, skal fila oppdateres.
- Om du rekker det, implementer å ha mulighet til å omorganisere innslag i en liste vha. long press. Dvs. man har long press på et innslag, som skal visuelt indikere at det er registrert etter en liten stund, og når det har skjedd, skal man kunne dra innslaget opp eller ned for å flytte det over/under de andre innslagene. Når man så løfter fingeren vekk, skal innslaget bli liggende hvor enn det er plassert. JSON-fila må oppdateres i henhold til dette.

Implementasjonen skal lages i sin helhet av deg som student. Kopiering eller avskrivning av kode regnes som plagiat og kan føre til utestengelse av studiet.

Løsningen skal fungere på nyere versjoner av mobilemulator (Android). Angi hvilken versjon som er brukt ved utvikling/testing.

Før du begynner implementasjonen så bør du undersøke ulike kryssplattformløsninger og gjøre deg opp en mening om hvilken som er mest hensiktsmessig for deg i dette tilfellet. Naturlige kandidater er:

```
Ionic (https://ionicframework.com/)  
Flutter (https://flutter.dev/)  
React Native (https://reactnative.dev/)  
Cordova (https://cordova.apache.org/)  
Kirigami (https://develop.kde.org/docs/use/kirigami/)
```

Xamarin ansees som lite aktuell pga. endringer som skjer med denne. Det finnes også flere "mindre" alternativer.

Rapport og problemstilling

Det du kom fram til skal resultere i et lite dokument 2-4 sider, hvor du beskriver forskjellene mellom de ulike kryssplattformløsningene beskrevet over (med unntak av Xamarin). Ta med andre muligheter også, om det har interesse for deg. Beskriv hvilke programmeringsspråk som brukes/kan brukes, og hva som kan være fordeler og ulemper i forhold til de andre løsningene. Til slutt skal du velge en av teknologiene for din egen implementasjon. Begrunn valget du gjør i dokumentet.

Innleveringen skal inneholde dokumentet, programkode/prosjekt, og en beskrivelse av hvordan man kan få testet løsningen på en nyere Androidemulator.

Begrunnelsen for å kreve en rapport er som følger:

- Man kan lære en teknologi fort, men som utdanningsinstitusjon så ser vi de lengre linjene. Mao. er det minst like viktig å vurdere hvilken teknologi man skal bruke. Begrunnelsen "jeg velger det for jeg kan det" eller "fordi jeg vil" er ikke god nok. Mao. om man kommer med begrunnelsen "jeg velger React" fordi jeg kan det - så er det ikke godt nok. Hvis man kjører en slik begrunnelse for teknolgivalg i Bachelor-prosjektet - så er man i seriøst problemer.

Så for dere som velger React Native, hvorfor ikke heller velge React på Ionic-plattformen?

Hvis du føler behov for en mer konkret problemstilling i forhold til rapporten så kan du tenke slik:

- Vurder de ulike plattformene i forhold til enkle applikasjoner med internasjonaliseringsbehov, kontra applikasjoner med behov for hurtig skjermoppdatering (ala action/skytespill).
- Det kan også være verdt å vurdere "språk" mot hverandre. Er det en fordel med typede spåk vs. ikke typede.
- Hva med mulighet for kompilering?.. osv.

Forventingen er at man bruker en dag (give or take) på denne delen av prosjektet.

Faget er laget for ingeniørstudenter og da skal vi forvente at man kan gjøre kloke valg basert på fakta. Man skal ikke velge teknologi fordi man vil eller "kan den", men fordi det er den beste for formålet....

Gjennomføring

Implementasjonen skal lages i sin helhet av deg som student (individuelt). Kopiering eller avskrivning av kode regnes som plagiat og kan føre til utestengelse av studiet.

Løsningen skal fungere på nyere versjoner av mobilemulator (Android). Angi hvilken versjon som er brukt ved utvikling/testing.

Siste frist for innlevering: Se Informasjon om emnet/Viktige datoer. Denne fristen er absolutt.

Spørreundersøkelsen og lenke for innlevering finner du under.

Lykke til :)