

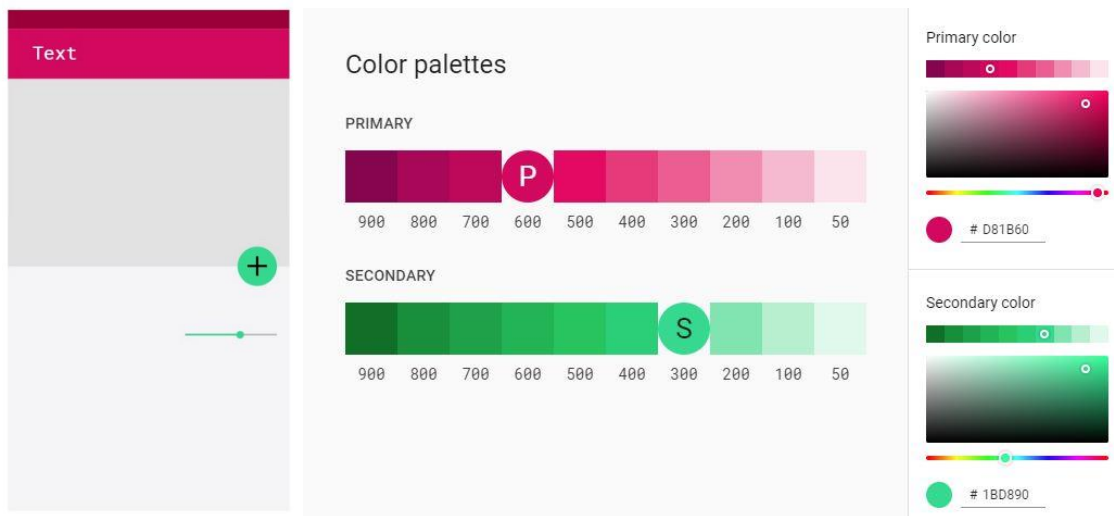
Kveld Med Venner



Studentnummer: s326318

Designvalg

For denne oppgaven har brukt googles ressurside Material.io som bakgrunn for designvalg. Dette for å følge anbefalte standarder for applikasjons utforming. Jeg valgte en primærfarge og deretter brukte jeg et fargehjul (<https://www.canva.com/colors/color-wheel/>) for å velge en passende komplementærfarge. I Material brukes ofte sekundærfarge på knapper og ulike toner av primærfargen for resten av designet og dette er et prinsipp jeg har fulgt. I overskriften har jeg derfor også benyttet meg av primærfargen til applikasjonen.



<https://material.io/design/color/#tools-for-picking-colors>

Ikoner

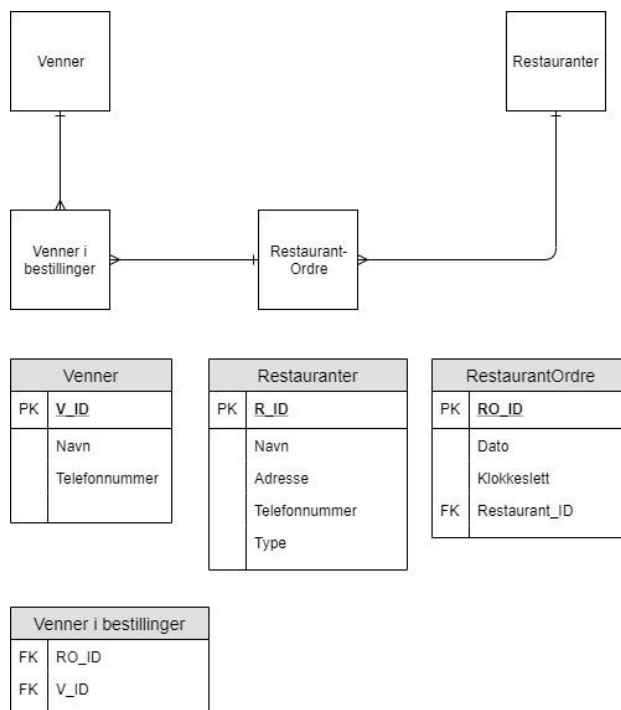
Ikoner brukt i denne applikasjonen er hentet fra android developers ressursider:

Henholdsvis kalender ikonet for å representere reservasjoner, personikonet for å representere venner, kniv og gaffel for å representere og hjulet, som er etablert ikon for innstillinger



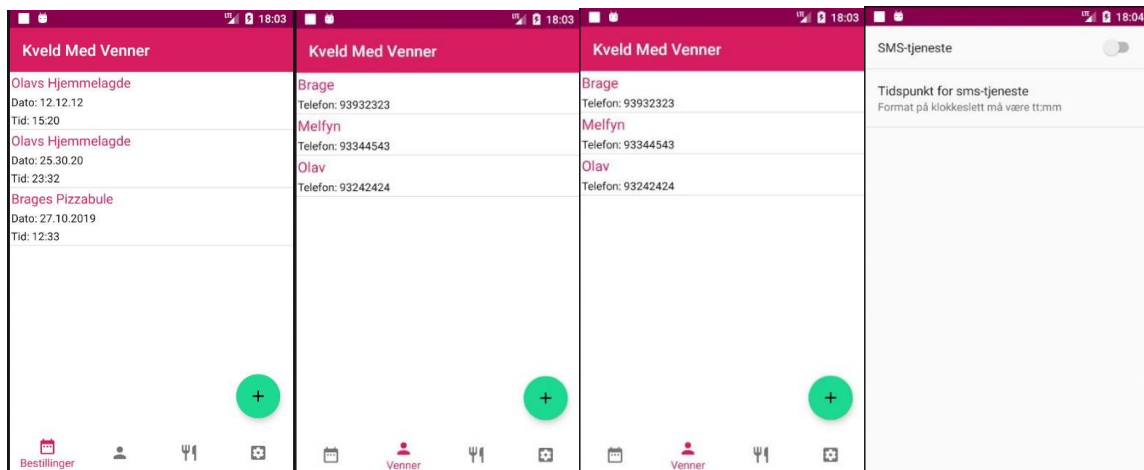
<https://material.io/resources/icons/?style=baseline>

Databasen

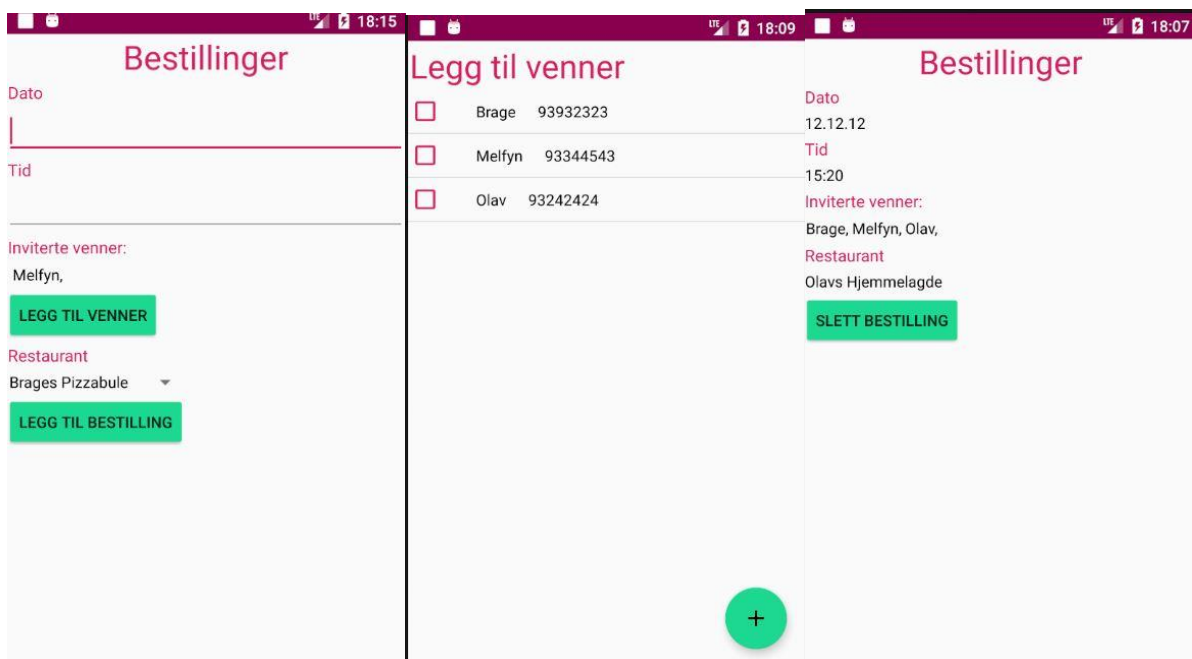


Over viser hvordan databasen er oppbygd og samsvarer med hvordan den faktiske databasen er bygd opp med SQLite. Attributtnavn er noe annerledes i selve databasen. Her har jeg forsøkt å få databasen på 3NF for å unngå redudans. Derfor har jeg skilt ut venner i bestilling til en egen hjelpetabell.

Gangen i applikasjonen



Det første brukeren møter er skjermen med bestillinger som er lagt inn. Her har jeg benyttet adaptore og listview for å vise riktig data når du navigerer med bunnmenyen. Det er lik utforming og knappene er knyttet til hva du har klikket på bunnmenyen.



Bestillingskjermen har jeg input-felt for dato og tid, en spinner som henter lagrede restauranter, og en egen skjerm for å legge til venner til ordren. Venner som er lagt til legger seg under inviterte venner. Skjerm bilde to viser hvordan man legger til venner når man har trykket på «legg til venner»-knappen. Her har jeg benyttet meg av checkboxes hvor du kan huke av hvilke venner som skal med og deretter trykker man på knappen i hjørnet for å gå tilbake til bestillingsmenyen. I skjerm bilde til høyre vises informasjonen man får opp når man trykker på en eksisterende bestilling i hovedaktiviteten. Siden det ikke skulle være nødvendig å endre en bestilling har jeg valgt å skjule inputfeltet og erstattet dem med textview-felt. Her har jeg også skjult «legge til bestilling», spinneren, og «legg til venner»-knappen.

Derimot skal man kunne endre restauranter og venner, så her har jeg valgt å beholde edittext når du trykker på en eksisterende restaurant, slik at du enkelt kan slette eller oppdatere restauranten eller venn.

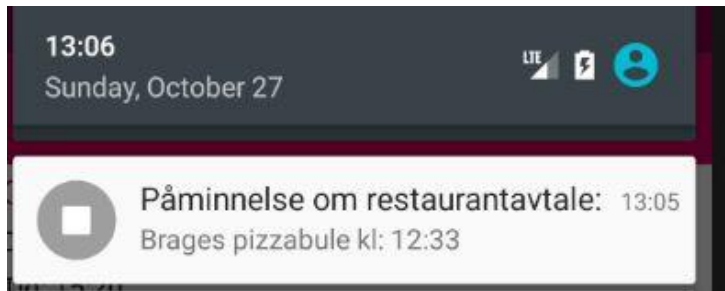
Restauranter	
Navn	
Adresse	
Telefon	
Type	
LEGG TIL	

Restauranter	
Navn	Olavs Hjemmelagde
Adresse	Olavsgata
Telefon	32323232
Type	Pizzeria
OPPDATER	
SLETT RESTAURANT	

Venner	
Navn	Melfyn
Telefon	93344543
OPPDATER	
SLETT VENN	

Venner	
Navn	
Telefon	
LEGG TIL	

Notifikasjoner



Notifikasjonen er bygd opp med en metode `generateNotificationText()`, som er basert på ordren den fant på dagen sjekken gjøres. Med denne bygges en streng med restauranten og klokkeslett.

```
// method to check if there are any orders at current date
public Boolean compareDates(){
    db = new DBHandler( context: this);

    int day = Calendar.getInstance().get(Calendar.DAY_OF_MONTH);
    int month = Calendar.getInstance().get(Calendar.MONTH)+1; // Month starts on zero.
    int year = Calendar.getInstance().get(Calendar.YEAR);
    String dato = day+"."+month+"."+year;

    Log.d( tag: "DatoStreng", dato);

    ArrayList<RestaurantOrder> orders = db.getRestaurantOrder();
    for (RestaurantOrder order : orders){
        Log.d( tag: "Datoer fra db", order.getDate());
        if(order.getDate().equals(dato)){
            return true;
        }
    }
    return false;
}
```

Denne metoden sammenligner datoen som er i databasen med dagens dato. Her har jeg brukt calendar for å hente ut dagens dato.

```
if(compareDates() == true) {
    // Set the alarm to start at approximately the time set in set HOUR_OF_DAY and MINUTE
    Calendar calendar = Calendar.getInstance();
    calendar.setTimeInMillis(System.currentTimeMillis());
    calendar.set(Calendar.HOUR_OF_DAY, 8);
    calendar.set(Calendar.MINUTE, 30);
    AlarmManager alarm =
        (AlarmManager) getSystemService(Context.ALARM_SERVICE);
    alarm.setInexactRepeating(AlarmManager.RTC_WAKEUP, calendar.getTimeInMillis(), AlarmManager.INTERVAL_DAY, pintent);
}

return super.onStartCommand(intent, flags, startId);
}
```

Sammenligningsmetoden brukes deretter til å returnere en boolean og om boolean = true, sendes det ut en notifikasjon til brukeren.

SMS-tjeneste

Denne funksjonaliteten ble dessverre ikke helt ferdig. Jeg fikk til å hente telefonnumre fra ordren og sende ut meldinger til alle venner som er med, men denne trigges samtidig som notifikasjonen, så fikk ikke helt til koblingen mellom innstillingene og når sms'en sendes ut.

