Projektplan

* **Programbeskrivning**. Beskriv detaljerat vad programmet gör.

**A**

Vi ska göra en notepad-app för android. Appen ska kunna anteckningar som skrivs in med systemets tangentbord. Man ska kunna välja storlek och färg på texten. Anteckningen sparas enbart i appen (med hjälp av en SQLite-databas), äldre anteckningar flyttas längre ner på listan när nya läggs till. Man ska kunna få påminnelser på ett angivet datum och tid. Man ska även kunna skicka anteckningar till andra via sms. Ev kommer fler funktioner tillkomma längre fram i projektet.

**B**

De grundläggande funktioner till vårt projekt har inte ändrats så mycket. Man får lägga till en not, redigera, och ta bort not.

Under redigeringen får man ändra färg, stil och storlek på texten, men det är bara en tillfärlig ändring, färg och stil kommer inte sparas. Man får även tar bort alla noter om man vill.

Den största ändringen är att vi använder nu XML-databas istället för androids SQLite-databas.

**C**

Anleding till vi övergav SQLite var att vi behöver SQL för att kunna använda den, men vi tycker det är bättre att har enbar Java kod i vårt program.

* **Användarbeskrivning**. Vem kommer att använda ert program? Vilka antaganden gör ni om användarna? Är de vana datoranvändare, är de specialister, nybörjare, små barn, etc.

**A**

Den enda begränsningarna vi kan se är att användaren är en person som kan läsa och skriva på engelska samt har en android-telefon eller en android-emulator installerad på sin dator.

**B**

TODO

**C**

TODO

* **Användarscenarier**.  Ge minst två exempel på scenarier där en av era tänkta användare använder programmet. Beskriv **i detalj** vad de ser, vilken typ av input de måste ge, hur de ger sin input och hur programmets output ser ut.

**A**När appen startas ser man en lista med de anteckningar som finns sparade, de senaste ligger överst på listan. Om man vill lägga till eller ta bort en anteckning så trycker man på telefonens menyknapp, dessa två val dyker då upp på knappar i displayens underdel. Vill man skapa en ny anteckning öppnas ett nytt fönster , vill man ta bort en anteckning måste man välja den och därefter bekräfta borttagningen. Om man trycker på en anteckningarna I listan så öppnas även då ett nytt fönster (samma som visas om man ska skapa en ny anteckning) där man har möjlighet att redigera anteckningen, dela den via sms, lägga in en påminnelse. Går man tillbaka till huvudmenyn sparas anteckningen automatiskt.Appens output är den lista med anteckningar som visas när man startar den.

**B**

V

**C**

V

* **Testplan**. Beskriv hur ni tänker testa programmet. I den här uppgiften ska ni lägga extra vikt vid **användartestning**. Beskriv vilka uppgifter som testanvändaren ska utföra. De två användarscenarierna ska ingå i användartestningen.

**A**

Vi kommer att lägga stor vikt vid negativ testning. När vi säkerställt att alla de grundläggande funktionerna fungerar så kommer vi därför att försöka täcka in alla möjliga typer av användarbeteenden för att försöka få appen att krascha.

**B**

TODO

**C**

TODO

* **Programdesign**. Beskriv de grundläggande klasserna som ni avser att implementera och ge en beskrivning av de viktigaste metoderna i varje klass.

**A**

Android-appar är uppbyggda av fyra komponenter; Activities, Services, Content providers, och Broadcast receivers.Activities är den komponent som appen huvudsakligen använder sig av, varje fönster i appen är en klass som ärver från Activitiy eller en av dess subklasser. Vårt startfönster kommer att att ärva från ListActivity. Fönstret där man behandlar en enskild anteckning kommer att ärva från Activity. Service är en komponent som möjliggör att appen körs i bakgrunden när appen inte används. Eftersom vi vill ha möjlighet att ge påminnelser kommer vi att behöva implementera en Service komponent.Content providers hanterar data på telefonen. Eftersom vi inte tänkt använda oss av sådana data kommer vi inte använda oss av en sådan. Möjligtvis skulle vi om möjligt villa implementera en funktion där man kan starta telefonens ”Kontakter”-app för att kunna returnera ett telefonnummer som en anteckning ska skickas till.Broadcast receivers används för att avisera meddelanden över hela systemet. Eftersom vi planerar att implementera påminnelser behöver vi ha en broadcast receiver-komponent så att vi kan avisera dessa till systemet.Alla android-appar behöver dessutom en android-manifest fil i xml-format. Den specificerar vad appen innehåller och vilka rättigheter den kräver av systemet.Det grafiska gränssnittet i android definieras med hjälp av layoutfiler. Varje klass har en layoutfil, även den i xml-format, som med hjälp av klassen View och dess subklasser bygger det grafiska gränssnittet.

**B**

Det blev till 5 publika klasser i slutet, några av de publika klasserna innehåller innre klass och anonym klass.

Klassen **NoteList** ärver från ListActivity och visa upp alla noter som finns lagrad i databas. Där får man lägga till, tar bort eller redigera genom att öppna redigerings Activity EditNote.

**EditNote** redigerar noten, man får ändra färg och stil till texten, och även skicka sms om man vill genom att starta androids inbyggda sms funktion, texten i sms är hämtat från noten.

**Note** klassen skapa objekt som lagra information för en not, den har instansvariabler titel, content, date(då noten sparas får den en long värde av datum) och alarm. Varje not har också ett unikt id, den är en statisk variabel och anleding till det är att vi behöver någoting som kan identifierar en not på ett enkelt sätt. Klassen har implementerat interfacen Comparable, så att vi kan använda TreeSet för att sortera noter i datum ording, där den nyaste not kommer först.

**Notes** klassen hanterar databas, både inskrivning och uthämtning av data, den använder en TreeSet att spara noter hämtat från databas.Den har också en variabel som heter ”draft” som lagra information för en not om man vill forsätta med den nästa gång eller om programmet plötsligt avbryts och man vill inte information förloras.

Alarm......

Det finns också flera layout-filer som beskriver layout på de 2 Activity. Vi har också layout-filer för dialoger i EditNote,

Både alarmdialog och fontdialog använder layout som vi implementerat själv.

**C**

TODO

* **Tekniska frågor**. En lista av tekniska frågor som ni måste hantera när ni bygger ert system. Var så detaljerad som möjligt. Ett viktigt steg mot en god design är att få ner så många frågor som möjligt på papper på ett organiserat sätt med så många förslag till lösningar som möjligt.

**A**

Vi har börjat sätta oss in i hur android-appar är uppbyggda, men det återstår givetvis mycket att lära sig. Vi kommer att använda oss mycket av Android API:n så vi kommer behöva lägga mycket tid på att se vad som finns att tillgå där.Tekniska frågor som vi funderar på är:

-  Hur många funktioner är det rimligt att hinna implementera? Det är svårt att säga nu eftersom det är svårt att uppskatta hur pass tidskrävande det är.

**B**

Vi har stöt på många problem under utveckling.

Vi visste inte var XML-filen för databasen skulle ligga. Vi har prövat många mappar, men där kunde man bara läsa, inte skriva, eftersom alla de mapparna komprimeras till en .apk fil i en android applikation, som är en binärtil.Men i slutet hittade vi något som heter ”Internal storage”......

**C**

TODO

* **Arbetsplan**. Beskriv hur arbetet kommer att delas upp mellan personerna i projektet. Gör en tidsplan som visar när olika delmoment i projektet ska vara klara.

**A**

- Än så länge är det oklart hur vi kommer att dela upp projektet. Vi kan ännu inte se hur vi på ett enkelt sätt ska kunna dela upp projektet i två delar. Den här veckan (16/4-22/4) räknar vi med att först skissa upp vilka klasser vi kommer att behöva. Sen ska vi börja koda.

**B**

TODO

**C**

TODO

SammanFattning

Vi har lärt oss mycket under utveckling av Notepad projekt. Android applikationer är uppbyggda på ett mycket annorlunda sätt än J2SE:s GUI, om man har erfarenhet inom GUI tidigare blir det lättare att lära sig Android.

Vårt projekt kan blir utvecklad ytterligare. Man kan t.ex. lagra information för färg och stil för texter i databas, funktionen har en mindre mening om man inte fick lagra färg och stil permenant.

Alla texter på knappar och dialoger är sparad i en xml fil, om vi vill programmet ska kunna köras på flera språk, kan vi använda flera sådana xml filer, vi behöver inte ändra någonting i koden, utan bara ändra lite konfigureringar, det ger mycket mer flexibilitet.