

Project Umbra

Max Nordlund
maxno@kth.se

Oskar Segersvärd
oskarseg@kth.se

29 april 2011

1 Programbeskrivning

Projektplan

Basversion

Programmet skall:

- redigera text skriven vertikalt i kolumner från vänster till höger.
- skapa nytt, spara och ladda dokument i rent textformat.

I praktiken skall man alltså kunna skriva mycket enkla stycken text i programmet och spara dessa.

Avancerad version

Den avancerad version av programmet skall även:

- kalla på pdfL^AT_EX, om sådant finns installerat på datorn, för att kunna göra en pdf av den aktiva filen.
- anpassas till språket mhasira, med en egengjord font.
- anpassas för mongoliska skriftspråket Hudam.
- bättre användning av fokus. Det vill säga, texten skall vara korrekt markerad före och efter sparning.
- ha möjlighet att välja mellan mongoliska och mhasira som input, med översatta tooltips för mongoliska.

Som användare av den avancerade versionen skall man således kunna se texten man skriver med rätt font samt kunna göra en pdf av sitt dokument för att kunna skriva ut detta.

Resultat

Då det var lätt att implementera sparing och laddning av filer, har vi dock märkt att att kunna redigera text i kolumner var mycket svårare att implementera än vi uppskattat. Vi har inte ens hunnit börja med den avancerade versionen.

Slutsats

Basversionen var ett lämpligt mål med tanke på tidsramen för projektet. Man skulle kunna säga att vi var lite för optimistiska när vi satt och skrev ner programbeskrivningen för den avancerade versionen.

2 Användarbeskrivning

Projektplan

Basversion

Programmet kommer användas av linguister och lajvare. Användaren förväntas ha viss datorvana, som till exempel att skriva dokument i Word/Notepad. Vi antar även att de har kännedom om språket i fråga. De bör även ha grundläggande förståelse för det arabiska skriftsystemet.

Avancerad version

Användaren kommer istället för grundläggande förståelse för arabisk skrift ha förståelse för det alfabet som språket mhasira använder sig av.

Programmet kommer dessutom kunna användas av mongoler då man skall kunna skriva mongoliska tecken.

Resultat

Då vi inte känner några linguister eller lajvare i Sverige för tillfället har användertester ej varit genomförbara.

Slutsats

Även om vår målgrupp är korrekt, borde vi ha haft en större test grupp. Vi sköt oss själva i foten genom att ha begränsat oss så mycket.

3 Användarscenarier

Projektplan

Basversion

Då användaren startar programmet kommer den mötas av en knapprad, och en tom skrivyta. Om en sparad fil öppnades i samband med start, kommer filens innehåll att visas i skrivytan.

När användaren trycker på *nytt*-knappen kommer ett nytt fönster med en tom textyta att öppnas.

När användaren trycker på *öppna*-knappen kommer en dialog upp där användaren kan välja vilken fil som skall öppnas. Denna kommer att visas i textytan om denna är full, annars kommer ett nytt fönster att öppnas med den nya filen.

När användaren trycker på *spara*-knappen kommer en dialog där användaren kan välja till vilken fil texten skall sparas. Textytans innehåll kommer sedan att sparas till den filen. Om användaren redan valt en fil att spara till, kommer texten att sparas till den istället, utan visad dialog.

När användaren trycker in en serie knappar kommer de tecken dessa är bundna till att uppenbara sig där markören är, varpå markören förflyttar sig nedåt. Notera att markören är av arabiskt ursprung. När textremsan nåt textytans nedrekanter kommer det senastskrivna ordet att flytta till kolumnen till höger.

Då användaren trycker på *enter*-tangenter kommer all text som står nedanför markören att flyttas till en ny kolumn till höger. Om ingen text nedanför markören finns kommer helt enkelt en ny kolumn att bildas.

Avancerad version

I den avancerade versionen kommer ytterligare två knappar finnas tillgängliga; *exportera* och *inställningar*. Då användaren trycker på *exportera*-knappen och filen inte är sparad, kommer spar-dialogen först att dyka upp. Om filen är sparad kommer en ny pdf-fil att placeras i samma mapp som text filen.

När användaren trycker på *inställningar*-knappen kommer en dialog där användaren kan välja mellan alla inställningar programmet har att erbjuda. Där kommer användaren med en enkel knapptryckning att kunna välja mellan mongoliska och mhasira. Även andra inställningar för programmet finns här.

Markören skall befinna sig på samma plats i textytan före som efter eventuella knapptryckningar. Med andra ord, varje instans av programmet skall hålla koll på var markören finns i textstycket.

Resultat

Då vi inte hittat testare är det svårt att utvärdera våra användarscenarion. Vi kan dock med tryggt säga att vi under utvecklingen använt oss av dessa funktioner och kunnat säkerställa att de fungerar enligt förväntan.

Slutsats

Att vi själva känner oss trygga med de funktioner vi implementerat är av föga betydelse, då vi inte genomfört användartestning som efterfrågades för att otvivelaktigt kunna redovisa korrekt funktionalitet.

4 Testplan

Projektplan

Vi skall skriva JUnit tester för allt som går att testa med JUnit, för att säkerställa att allt fungerar korrekt. Genom att simulera mus och tangentbords input kan vi undersöka att programmet, under förbestämda förhållanden, ger det förväntade resultatet. Testerna som skall utföras inkluderar att redigera text, spara och ladda filer samt att för den avancerade versionen exportera till pdf. Vi skall också se till att alla tre varianterna att starta applikationen på fungerar. Via Kommandotolken, med eller utan en *path* till en fil, samt genom att direkt dubbel-klicka på *JAR*-filens ikon.

Våra användartester kommer att inkludera alla funktioner som beskrevs under rubriken användarscenarier. Vi kommer att ge några få enkla instruktioner och observera användarens handlingar och därifrån justera programmet för att vara mer användarvänligt.

Resultat

Testplanen, om än välformulerad och genomtänkt, blev i slutändan oanvändbar. Då vi begränsat vår målgrupp så kraftigt blev det omöjligt för oss att användartesta vårt program.

Slutsats

Vi borde ha haft en bredare målgrupp för programmet så att det skulle finnas fler möjliga testare.

5 Programdesign

Projektplan

Basversion

Istället för att vi lägger textarean direkt på skärmen, så sätter vi oss emellan. Vi läser från skärmen och skickar till textarean bakom. Det gör att vi kan över-sätta saker, som till exempel piltangenterna för att förflytta markören i texten. Vi kommer att lyssna på input och sedan skicka vidare justerad input. Up-påtpilen kommer att *mappas om* till högerpil etc, då detta kommer bli mycket mer naturligt i vår önskade kontext. Detsamma måste göras med musens input, annars kommer alla koordinater att bli helt åt skogen. Eftersom vi har vridit bilden vi visar på skärmen 90° åt vänster, så måste all input vridas 90° åt höger.

För att lösa dessa problem skall vi använda oss av två *BufferedImages*. Metoden *createImage()* kommer att rita textytan till en imaginär bild i minnet som sedan kommer att ritas till skärmen medans vi genererar följande bild i minnet. Vi använder med andra ord en dubbel buffer. Planen är att skapa en subklass av *JScrollPane*, och *Override* dess *paint()*-metod, så att den vrider bilden innan den ritas den.

Avancerad version

Vi skall även lyssna på alla andra tangenter på tangentbordet så att man inte behöver ha arabiska installerat på sin dator för att kunna skriva korrekt mhasira i programmet. Inte nog med att programmet måste byta ut varje tangentbordsstryckning till rätt tecken, så måste programmet även ha koll på vilka tecken som finns runt omkring det tecken man just skrev in och justera dem utefter det system mhasira kräver.

Resultat

I princip följde vi vår projektplan när det gällde programdesign. Vi använder oss av två *BufferedImages* för att rotera en *JScrollPane* som vi önskat göra. Efter långa informationssökningar kom vi dock fram till att det var lämpligast att inte röra om i *paint()*-metoden. Istället tar vi en *BufferedImages Graphics2D*-objekt och ritas vår *JScrollPane* till denna för att slutligen rotera bilden till en annan *BufferedImage* som är den som visas i programfönstret. Det största problemet visade sig dock vara att det inte bara var så enkelt som att skicka all input vidare från en *JComponent* till en gömd *JTextArea*.

Slutsats

Det var mycket svårare än väntat att utföra vad som försk kan verka vara en simpel rotation av en *JTextArea* 90° åt vänster. I slutändan var det nästan som om vi skapade en ny *JComponent* än att vi roterade en annan. Resultatet blev att mycket mer arbete var nödvändigt för ett visuellt sett litet resultat.

6 Tekniska frågor

Projektplan

Basversion

För att kunna visa texten i vertikala kolumner från vänster till höger kommer vi använda oss av en kombination av något exotiska verktyg. Vi skall använda det arabiska skrivsättet, som är skrivet från höger till vänster för, och vrida texten ett kvarts varv åt vänster. Arabiska är ett utmärkt val för denna process eftersom vi då får tre problem lösta på en gång. Vi kommer att kunna skriva i vertikala kolumner med alla bokstäverna i rätt ordning, någonting som inte skulle kunna åstadkommas lika lätt med det latinska alfabetet. En annan fördel är att det arabiska skrivsättet är att den, likt mhasira, har flera olika former av varje tecken. Mhasira har tre former, initial, medial och final. Alla dessa återfinns i arabiskan vilket gör att den lämpar sig utmärkt.

För att göra detta skall vi sätta oss mellan skärmen och textarean, och agera tolk. Först ritar vi textarean till bufferten, vrider den för att slutligen visa den färdiga bilden på skärmen. Det innebär att man också måste översätta all musinteraktion mellan skärmen och textarean då koordinaterna annars blir fel.

Avancerad version

För att kunna visa text i en Mhasira font kommer en sådan att skapas, och då det arabiska alfabetet används för att kunna skriva texten, kommer denna nya font att behöva vara baserad på just detta. Att göra en arabisk font kan utgöra en tröskel. Oavsett måste alla tangenttryckningar översättas till exempel från *a* till *ا*.

Då det vi vill åstadkomma liknar mongoliska så mycket som det gör, torde det inte vara så svårt att anpassa programmet för mongoliska.

Resultat

Att använda det arabiska alfabetet (abjad) för vårt projekt visade sig inte utgöra något större problem, och inte heller rotationen av text genom användning av bilder.

Mus- och tangentbords-input var det som i slutändan utgjorde den största tankenöten. Det var inte så bara som vi först uppskattat det att vara. Att skicka vidare *KeyEvents* visade sig otillräckligt för vissa oundgängliga funktioner.

Slutsats

När man skall utföra ett projekt som detta är det ytterst viktigt att snabbt sammanställa en tidig *alfa*-version för att så tidigt som möjligt upptäcka de aspekter som kan utgöra svårigheter senare i utvecklingen. Om vi skulle gjort det skulle vi möjligen kunna upptäckt svårigheterna tidigt och kunnat anpassa vårt arbete för att åtgärda dessa problem snabbare.

7 Arbetsplan

Projektplan

Basversion

- Efter den första veckan skall texten vara korrekt roterad och filer skall kunna skapas, sparas och laddas.
- Vid slutet av den andra vecka skall pil tangenterna och musen fungera. Basversionen skall vara fullt funktionsduglig. En prototyp.
- Efter tredje veckan skall basversionen vara helt färdig, välputsad och fin.

Avancerad version

- Efter den fjärde veckan skall exporten till pdf fungera.
- Till den sista veckan skall vi ha dubbelt språk stöd, samt en knapp för att växla mellan språken. I och med det skall även den slutgiltiga versionen av mhasira fonten vara färdigställd.

Resultat

På grund av en feluppskattning av tiden vi haft till vårt förfogande blev vår plan felbalanserad. Vi antog att ytterligare en vecka skulle finnas till förfogande, därav är blev arbetet på de avancerade versionen av programmet opåbörjat.

Slutsats

Att säga att vi borde lagt ner mer tid på projektet är lätt i efterhand, men det visade sig vara svårt att uppskatta hur mycket tid saker och ting faktiskt skulle ta. Detta är en av de största lärdomarna av detta projekt, en bättre tidsuppfattning för utvecklingen av mjukvara.