

Varianthantering och gemensam kodbas för webbaserad apputveckling

POPULÄRVETENSKAPLIG SAMMANFATTNING
HENRIK GYLLENSVÄRD, NIKLAS WELANDER

Handledare: Thomas Hermansson (ÅF)
Examinator: Lars Bendix (LTH)

En inblick i applikationer

När man bygger ihop mobila applikationer idag så är det vanligt att man gör det till flera olika system och telefoner, idag är Android och iPhone de mest kända exemplen. Man vill ofta bygga samma applikation, den ska alltså både se ut och bete sig likadant. Problemet för utvecklarna är att plattformarna har sina helt egna språk och strukturer. Tänk dig att en applikation är som styrdonet på ett fordon och du tillverkar dessa. Problemet är att mobilplattformerna nu är så olika att de kan ses som helt olika fordon: Android - flygplan och iPhone - motorcykel. Uppenbarligen kan vi inte använda samma "ratt" på alla fordon.

Problem med dagens utveckling

Tillverkningen av styrdon skiljer sig så mycket att tillverkningen tvingas delas upp i helt skilda grupper spritt över världen och dessa grupper har ingen anledning till att kontakta varandra, de vet hur "applikationen" ska se ut. För ditt företag gäller då att göra investeringar i specialistkunskaper och grupper världen runt och hålla koll på dessa. Och om en ändring ska göras på "applikationen" så måste alla grupperna planera och införa den separat från varandra.

Nytt fordon, flera modeller

Just nu introduceras nya mobilplattformar på bilmarknaden som stödjer appar skrivna i web-format. I och med att det finns flera plattformar som stödjer detta så kan vi se dessa som olika bilsorter och applikationen blir en bilratt. Man vill dessutom ofta när man utvecklar appar komma åt hårdvaran på mobilen, vi kan se detta som knapparna på ratten (blinkers, torkare, stereo etc.). Även om nu alla har möjlighet att göra bilrattar på exakt samma sätt uppkommer ett snarlikt problem från tidigare. Olika bilmärken gör olika fästningar för rattarna så vi har fortfarande problem om vi vill göra en ratt till alla bilmärken.

Exakt samma ratt, med varianter

Det vi vill undersöka är möjligheterna att kombinera ihop byggandet av rattarna. Kan vi lyckas komma runt problemet med att ha olika lag som jobbar på samma sak? Skillnaderna i fästningsanordningen kommer kvarstå, men man kan kanske bygga en lösning för själva ratten och bygga ihop olika fästen på den? Man kommer fortfarande ha flera modeller för fästena, men man har ett gemensamt sätt att bygga övriga delar av ratten på. På så sätt kan man minimera det till att ha en grupp som utvecklar ratten för flera modeller samtidigt.

Kommunicera med hårdvaran

Att slutanvändaren, bilisten, ska ha en behaglig användarupplevelse har varit givet under en längre tid. I ratt-tillverkningen har vi velat underlätta interaktionen, men inte med bilisten. I varje bilmodell finns en fästningsanordning olik den nästa och det finns ett kluster med sladdar och kontakter som skiljer sig i varenda en. Som utvecklare behöver man nu, beroende på ratten man ska tillverka, lära sig vart varje sladd ska gå och hur varje kontakt ska kopplas in. På mobilsidan motsvarar detta ett bibliotek av anrop till hårdvaran. Även om biblioteken är skrivna på samma språk så är de så annorlunda att utvecklarna måste lägga tid på att lära sig alla (om man inte vill dela upp tillverkningen, och det ville vi ju inte). Så vad vi har gjort är att skapa ett bibliotek som länkar ihop de olika plattformarnas bibliotek. På bilen motsvarar detta en adapter som passar alla bilar men har samma utseende för ratt-tillverkaren.

Målsättning

Lösningen kan tyckas banal för exemplet med ratten. Tyvärr är de mobila plattformarna betydligt mer komplexa och det finns många faktorer att ta hänsyn till. Poängen är däremot densamma. Det vi alltså vill undersöka är om det går att kombinera ihop utvecklingen till de olika plattformarna. Detta skulle då lösa problemet med det dubbla underhållet. Dessvärre är problemet fortfarande att man behöver ha olika versioner för varje plattform, fästet skiljer sig ju fortfarande! Det man åstadkommit är att underlätta det dubbla underhållet. Däremot introducerar man olika varianter av ratten. De skillnader som finns behöver man hantera på ett bra sätt för att minimera mängden extra arbete. Så vi kommer även undersöka vilka problem som finns angående hanteringen av dessa varianter.

Slutsats

I slutändan lyckades vi att minimera det dubbla underhållet när man skapar applikationer. Resultatet blev ett verktyg som utvecklare kan använda sig av för att uppnå detta. Det är fortfarande bara en prototyp och det finns fortfarande vissa problem kvar, men vi har visat att konceptet fungerar och det går att jobba vidare med. Sammanfattningsvis kan man säga att man nu bara behöver en grupp som löser konstruktionen av styrdonet till flera olika typer av fordon, långt mycket bättre än det vi började med!