

Studie av Cross Platform teknik för mobil applikationsutveckling

Emil Kristiansson

Sammanfattning

Idag domineras smartphone-marknaden av operativsystemen *iOS* och *Android*.

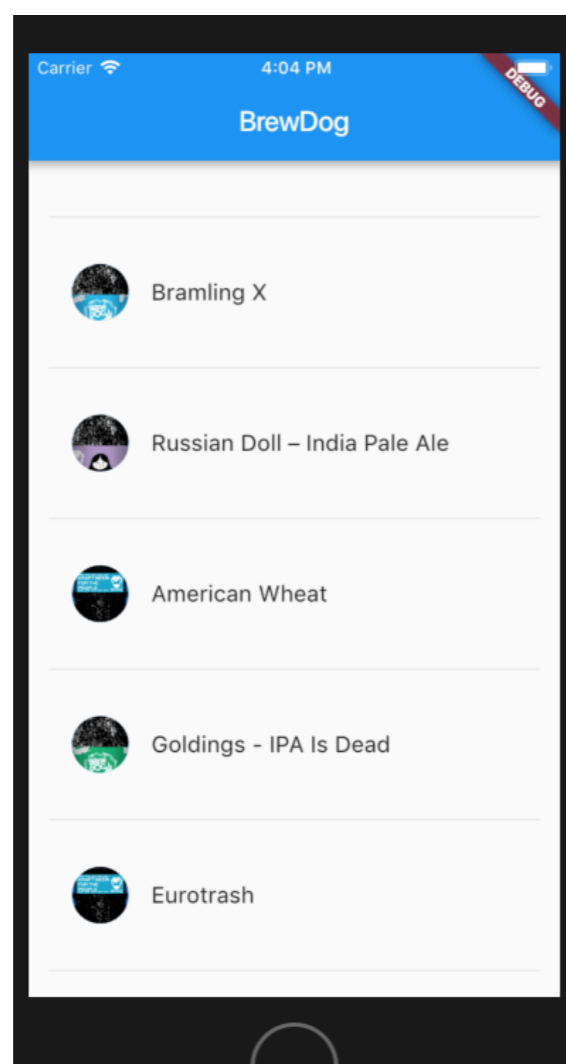
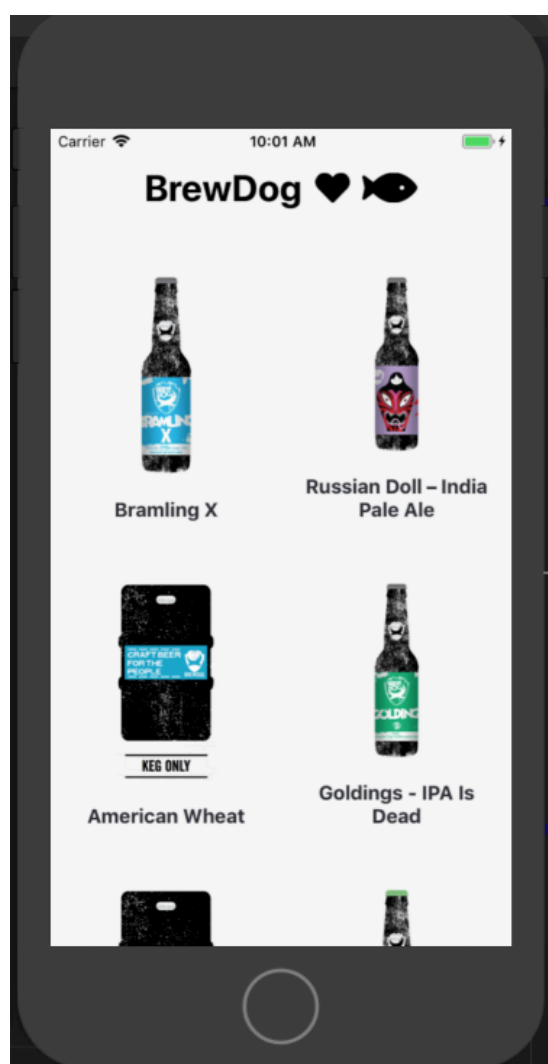
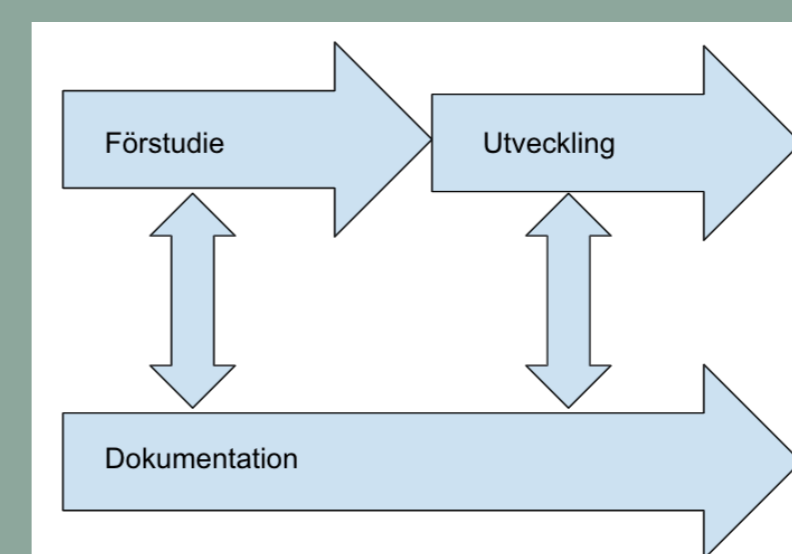
För att utveckla en mobilapplikation som fungerar på båda operativsystemen kan man använda sig av *Cross Platform* ramverk. Ett ramverk är i det här fallet är en samling färdiga verktyg som används för systemutveckling. Två av dessa är *React Native* och *Flutter*. Examensarbetet har tittat på vad det finns för för- respektive nackdelar med *React Native* och *Flutter*. *Smart Refill AB* är ett företag som utvecklar mobilapplikationer. De är nyfikna på att få veta mer om *React Native* och *Flutter* samt vilket ramverk de eventuellt ska införa i sin nuvarande utvecklingsprocess.

Problem

- Hur kan man utveckla en prototyp av en mobilapplikation med *React Native* som hämtar data från ett öppet *API* (data som hämtas från en extern databas) och visar datan i prototypen på ett överskådligt sätt i form av en lista?
- Hur kan man utveckla en prototyp av en mobilapplikation med *Flutter* som hämtar data från ett öppet *API* och visar datan i prototypen på ett överskådligt sätt i form av en lista?
- Hur lång tid tar det för respektive prototyp att hämta data från *API*:et och skiljer sig tiden väsentligt mellan prototyperna?
- Hur skiljer sig prototyperna ur ett utvecklarperspektiv?
 - Behöver man några särskilda verktyg för att kunna utveckla?
 - Vilket programspråk använder man?
 - Hur lång tid tar respektive prototyp att byggas med ett utvecklingsverktyg?
 - Är där någon skillnad i filstorlek på prototyperna?
 - Vad har andra utvecklare för erfarenhet av respektive ramverk?
 - Hur populärt verkar respektive ramverk vara bland utvecklare i allmänhet på olika forum?

Metod

Examensarbetet genomfördes med en metod som bestod av två parallella steg. Det ena steget bestod av två olika faser, en förstudie och en utvecklingsprocess. Det andra steget bestod av en dokumentationsfas. För utvecklingen tillämpades en enkel projektmodell som var inspirerad av *SCRUM*. *SCRUM* är ett projektramverk som ofta används för systemutvecklingsprojek. Den utgick från en enkel *Scrum Board* med de tre kategorierna: *Att göra*, *Gör* och *Gjort*.



Lösning

Lösningen blev två prototyper som hämtar billdata från ett *API* som tillhandahölls av öltillverkaren *Brewdog*. På dessa applikationer gjordes sedan tester utifrån problemställningen, exempelvis så analyserades tiden för att hämta data med *API*:et för respektive prototyp. Utöver detta så undersöktes även de andra delarna av frågeställningen genom att titta på hur prototyperna skilde sig åt. För att besvara frågorna om erfarenhet och community så undersöktes olika forum och bloggar för utvecklare som arbetar med ramverken.

Diskussion

Utifrån undersökningen så konstaterades det att *React Native* är det ramverk som bör användas idag. Det som talade för *React Native* var främst att det var i snitt **438** gånger snabbare på att hämta data från *API*:et. Detta i kombination med att ramverket baseras på *JavaScript* som är ett av de vanligaste programmeringsspråken idag samt att det finns ett stort intresse kring just *React Native*, gör ramverket till ett bättre val än *Flutter*. *Flutter* använder sig av programmeringsspråket *Dart* som inte alls är lika stort och det finns ett mindre intresse kring *Flutter* i allmänhet, vilket gör att det bör vara svårare för ett företag att attrahera utvecklare som vill jobba med *Flutter* jämfört med *React Native*.

