# Slutrapport Skurtify

## Syfte och målsättning

Har du någonsin hört en låt på radio och tänkt att ”den här var jäkligt bra, den vill jag lyssna mer på”? Vårt mål är att skapa en mashuptjänst som anropar Sveriges radio API, skriver ut vilken låt som spelas på radio just nu och låter användaren hitta den här låten på Spotify. Eftersom det inte alltid spelas en låt ska tjänsten isåfall visa föregående låt som spelades. Vi vill även visa vilken tid låten startade så att användaren lättare kan veta att det är just den låten de är ute efter. För att sticka ut från potentiella konkurrenter profilerar vi oss med den folkkära grodan Skurt. Varumärkesskyddet har nog expirerat vid det här laget.

Backend kommer skrivas i Python med hjälp av modulerna Json och Urllib2. För kommunikation med Spotifys API har vi använt Pythonmodulen Spotipy. Funktionaliteten skickas till mashupapplikationen med hjälp av modulen Bottle.

## Användarmanual

Gränssnittet är väldigt enkelt. Användaren väljer kanal på startsidan och applikationen kontaktar då Sveriges Radio API för information om relevant kanal. Svaret tas emot i JSON och skickas vidare till templates som webbläsaren tolkar (i nuläget på http://localhost:8080).

Det första hindret vi stötte på var att anropet */playlists/rightnow* gav olika returdata vid varje förfrågan. Detta ledde till att vi ständigt fick KeyError. Ibland skickade SR:s API vilken låt som spelades, ibland bara vilken som var föregående låt och ibland vilken låt som var nästa på spellistan. Ibland en kombination av ovanstående. Vi bestämde oss därför för en hierarki vad som skulle visas:

* Om en låt spelas just nu visas den.
* Om ingen låt spelas just nu visas föregående.
* Om ingen låt spelas och ingen föregående låt returneras från SR visas nästkommande låt på kanalen.

Applikationen skriver ut låtnamn, artist och länkar sedan vidare till låten på Spotify genom en logotyp. Låten spelas i Spotifys öppna webbspelare, eller om användaren har Spotify installerat och igång kommer låten helt enkelt att spelas där. Skulle låten inte hittas på Spotify skriver applikationen ut detta. Skulle användaren vilja söka på annan kanal går de bara tillbaka i webbläsaren. Vi tänker att detta är ett gränssnitt folk är vana vid och applikationen innehåller bara två olika tillstånd så det borde inte leda till komplikationer för användaren.

## Backendstruktur

Funktionaliteten är uppbyggd av en huvudfunktion (skurtify.py) som implementerar bottle och läser templates från mappen ”views” och hämtar CSS samt bilder från mappen ”static”. Det är denna funktionen som tar datan från Spotify och Sveriges radio för att skicka ut dem i templates.

Från denna körs sr\_api.py där data från Sveriges Radio tas emot och returneras tillbaka till skurtify.py. Datan tas emot i JSON då vi tycker det är enklare att behandla än XML. Sr\_api kollar genom datan för att se om föregående, nuvarande eller nästkommande låt har returnerats. Härifrån körs också en tidsfunktion (song\_time.py) som översätter timestamps till ett mer lättläst format och en liten funktion som översätter SR:s id-nummer för kanal till deras faktiska namn (channel.py).

Data om låtinformationen tas emot i strängformat av skurtify.py och skickas sedan direkt vidare till spotip.py där vi med hjälp av modulen Spotipy kontaktar Spotifys API och kollar om låten finns i deras databas. Om sökningen lyckas hämtar den tillhörande bild och länk till låten.

Till slut returnerar antingen skurtify.py informationen på templates så att användaren kan använda det i sin webbläsare eller genom JSON så att API-konsumenter av tjänsten kan använda detta för sina egna applikationer. För instruktioner till API:t, se APIdokumenation.