**Youtubeify**

**Syfte**  
Vi vill skapa en tjänst som gör det möjligt för användaren att med ett enda knapptryck söka efter en låt och att få träffar från både Spotify och Youtube samtidigt. På så sätt får användaren lära känna en låt på flera olika sätt då Youtube även bl.a. ger live-framträdanden och covers som sökresultat. Tjänsten kan hjälpa både Spotify och Youtube då användandet av våran tjänst kan öka trafiken till bägge tjänster och kan resultera i t.ex en prenumeration av Spotify.

**Målgrupp**  
Vår målgrupp är musikintresserade som lyssnar på musik via webbläsaren. Användaren är då en person som är intresserad av att få en mångfald i sökresultaten med låtar, musikvideos, covers och live-framträdanden samtidigt.  
  
**Krav**  
Slutanvändaren kräver en fungerande tjänst som de känner sig trygga med att använda. Sökresultaten ska försöka motsvara de resultat användaren hade fått på både Spotify och Youtube individuellt.

**Tjänsten**Vi använder oss utav två datakällor vilket är Spotify och Youtube. Med hjälp av deras API så hämtar vi ut träffar ifrån Spotify, i form av artister, låtar eller både och. Från Youtube matchas sökningen med samma resultat man hade fått på deras hemsida.  
  
Vi har lagt till ramverket Flask till Python för att kunna skriva ut resultaten på en hemsida i form av .tpl-dokument.

När en sökning har skett så kontrolleras om det är både artist och låt eller bara en av alternativen som har skrivits in. Efter det redirectas man till nästa funktion för att få en bättre url, om ett alternativ inte är inskrivet ersätts det med ”none” i url:en. Är inget inskrivet i något av fälten skickas man tillbaka till startsidan. Beroende på vad användaren har sökt så bestäms Youtubes sökterm (där vi även gör om mellanslag till +), om både artist och låt är inskickat lägger vi till ett bindestreck mellan artist och låt, och vad användaren är ute efter vad det gäller Spotify med variabeln ”spotifystate”.

I Spotify-funktionen körs en HTTP-request beroende på variabeln ”spotifystate” och returnerar en lista av Uri-er. Om det finns ett mellanslag i någon sökvariabel så ändras det till %20 innan HTTP-requesten körs.

I Youtube-funktionen körs en vanlig sökning och en lista av video-id returneras.

När sökningarna är klara returneras index.tpl och listorna, är någon lista tom ändras den till en sträng som innehåller ”empty”. Innan första sökning kontrolleras det om det finns någon lista, finns ingen lista skriver vi ut ett element. Är en lista tom kontrolleras det genom att variabeln då är en sträng som innehåller ”empty”. Innehåller en lista värden loopas dem ut i iframe-taggar.

**Användarmanual**

1.Skriv in låten eller artisten du vill söka på i sökfältet som presenteras uppe i mitten under Youtubeify.

2.Tryck på sök-knappen som är lokaliserad till höger om sökfältet.

3.Resultaten presenteras i två spalter under sökfältet där Spotify-träffarna radas upp till höger och Youtubes till vänster.  
  
4.Klicka på den låten du vill lyssna på.  
  
5.När du vill söka på andra låtar så är det bara att söka igen.