

Rymdkollen – Processanalys

Namn: Malin Widberg

Kurs: Javascriptramverk JSR200

Termin: HT2025/2026

<https://webbkurs.ei.hv.se/~mawi0122/examJSR200/examensarbete/#/>

1. Inledning

Det här projektet gick ut på att skapa en applikation som baseras på något JavaScriptramverk. Jag valde att göra en om rymden och använder NASA:s API *Astronomy Picture of the Day* (APOD) för att visa rymdbilder och information (NASA, u.å.-a). Jag valde att bygga applikationen i Vue 3 tillsammans med Vue Router (Vue.js, u.å.; Vue Router, u.å.), eftersom jag ville testa ett ramverk som både är lätt att komma i gång med och samtidigt ger en tydlig struktur när man bygger flera sidor.

Mitt mål var att skapa en applikation som känns enkel att använda, ser sammanhängande ut och fungerar direkt i webbläsaren utan någon backend. Jag ville också att den skulle kännas lite inspirerande — som en liten rymdportal där man kan utforska bilder, slumpa fram nya datum, spara favoriter och till och med se var ISS befinner sig just nu.

Applikationen riktar sig främst till rymdintresserade användare och personer som vill utforska visuellt material från NASA på ett enkelt och inspirerande sätt. Målgruppen är bred, men fokus ligger på nyfikna användare som uppskattar tydlig navigering, visuellt innehåll och lättillgänglig information.

Bildtexterna i appen visas på engelska eftersom NASA:s API inte erbjuder översättningar. Jag försökte faktiskt lösa det genom att testa olika översättningstjänster, men CORS-begränsningar satte stopp för det. Därför fick jag acceptera att NASA:s originaltexter får stå kvar, medan resten av appen är på svenska.

2. Genomförande

2.1 Struktur och teknik

Jag byggde applikationen som en Single Page Application (SPA). Det innebär att allt laddas en gång, och sedan byts innehållet ut med hjälp av Vue Router utan att sidan laddas om. Det gör att appen känns snabb och smidig att använda. Varje sida är en egen komponent, vilket gör det lättare att hålla ordning på koden och utveckla funktionerna steg för steg.

2.2 APOD-komponenten

Den viktigaste delen av appen är *ApodView*, som hämtar data från NASA:s API (NASA, u.å.-a). Den hanterar tre olika lägen: dagens bild, slumpad bild och bild baserad på ett specifikt datum. Jag lade till en loader, felmeddelanden och stöd för både bilder och videor, eftersom API:t ibland returnerar YouTube-länkar. Jag byggde också in ett favoritsystem som sparar data i `localStorage`, så att man kan samla sina favoritbilder. Efter genomgång av min app så hittades de några buggar där en av dessa var att när man sparade en video i favoriter så syntes inte videon. Jag löste detta genom att bygga en

funktion som konverterar vanliga YouTube-URL:er till embed-format. Jag lade även till fallback-logik för äldre favoriter som saknade `media_type`, så att de fortfarande visas korrekt.

2.3 Favoriter

Favoritsystemet är helt klientbaserat och sparar data i webbläsarens `localStorage`. Det var en smidig lösning eftersom uppgiften inte krävde någon backend. Det gjorde också att jag kunde hålla applikationen enkel och lätt att publicera. Jag valde även att skapa möjligheten att ladda ner bilderna i favoriter. Detta gjorde jag då jag själv kände att vissa bilder var så vackra att jag själv ville behålla dem.

2.4 Live-sidan

Live-sidan var den del som krävde mest problemlösning. Min första tanke var att bädda in en livevideo från ISS, men det visade sig snabbt att det inte var så enkelt. YouTube blockerar inbäddning på vissa domäner, NASA Worldview tillåter inte iframe-inbäddning (NASA, u.å.-b), och många ISS-strömmar tas bort eller byts ut.

Till slut hittade jag en lösning som fungerade bra: en realtidskarta från ESA som går att bädda in (European Space Agency, u.å.), och sedan länkar till både livevideo och NASA Worldview i nya flikar (YouTube, u.å.). Det blev faktiskt bättre än min ursprungliga idé, eftersom det är mer stabilt. Jag valde även att lägga till information att det kommer öppnas i ett nytt fönster och vart man hamnar då det är något jag uppskattar.

3. Tekniska val

3.1 Val av ramverk

Jag valde Vue eftersom det är lätt att förstå och ger en bra struktur när man bygger flera sidor. Jag gillar också att det är komponentbaserat — det gör att man kan jobba i små delar och bygga ihop dem till en helhet vilket är ett sätt jag uppskattar att arbeta på.

3.2 Val av API

NASA:s APOD-API var jag först lite nervös att använda och var orolig att jag skulle använda eller göra något fel då det är NASA vi pratar om. Jag valde efter att googla en del, även fråga Copilot om vad som gäller. När detta orosmoln var borta så var det ett självklart val eftersom det är lätt att använda och ger tillbaka både bilder, videor och förklarande texter (NASA, u.å.-a). Det gav mig också möjlighet att jobba med olika mediatyper och felhantering.

3.3 Hantering av begränsningar

Under projektet stötte jag på flera begränsningar som påverkade designen. CORS-problem gjorde att jag inte kunde översätta bildtexterna. YouTube och NASA

Worldview hade inbäddningsrestriktioner. Det gjorde att jag fick tänka om och hitta lösningar som fungerade i praktiken, inte bara i teorin. Jag valde att kombinera bild, video och text eftersom dessa medier kompletterar varandra och skapar en rikare användarupplevelse. Bilderna ger den visuella upplevelsen, videorna förstärker känslan av närvaro och rörelse, och texterna ger sammanhang och förklaringar. Detta följer grundläggande multimodalitetsteori där flera uttrycksformer tillsammans stärker budskapet och gör informationen mer engagerande.

Det visade sig även att den klassiska bilden jag har på startsidan inte bara var att "ta" och inte heller liknande bilder på just månlandningen. Till slut hittade jag den jag har nu på Buzz Aldrin som kommer från Wikimedia Commons (u.å.). en wikipediasida där jag fick ladda ner bilden och använda den.

4. Problem och lösningar

4.1 YouTube-blockeringar

ISS-strömmar kunde inte bäddas in på grund av YouTubes restriktioner (YouTube, u.å.). Lösningen blev att länka till en YouTube-sökning som alltid visar aktuella strömmar.

4.2 NASA Worldview

NASA Worldview använder X-Frame-Options: SAMEORIGIN, vilket gör inbäddning omöjlig (NASA, u.å.-b). Jag löste det genom att öppna tjänsten i en ny flik.

4.3 CORS-problem vid översättning

Externa översättningstjänster blockerade förfrågningar från HV:s server. Jag fick därför behålla NASA:s originaltexter om jag inte skulle begränsa bilderna och översätta varje bildtext själv.

4.4 API-fel

Vissa datum saknar bilder eller returnerar fel. Jag löste det genom att lägga till felmeddelanden och en loader, så att man alltid får en tydlig återkoppling.

4.5 Buggar

Vid genomgången av min app på handledningen den 15/1–26 så hittade Patrik några buggar. Det var toppen så jag kunde åtgärda dessa då jag ansåg att det verkligen behövdes. När man tryckte på knappen: lägg till i favoriter så syntes det inte att man tryckt och man kunde trycka flera gånger och då lades samma bild till i favoriter lika många gånger. När man sedan ville ta bort dubbletterna så togs alla bort av samma bild. Det gick även inte att se vilken video man sparat på favoriter då det inte visade själva

videon vilket gav ett ofullständigt intryck ut och funktionen att få spara den var då helt oanvändbar. Detta löste jag genom att gå igenom favoritsystemet steg för steg och identifiera vad som saknades i logiken.

Först lade jag till en kontroll som ser om en bild redan finns sparad i favoriter innan den läggs till. NASA:s APOD-bilder har alltid ett unikt datum, så jag använde datumet som identifierare. Om datumet redan finns i listan sparas inte bilden igen. På så sätt förhindras dubletter helt, och det går inte längre att fylla listan med samma bild flera gånger.

Nästa problem var att användaren inte fick någon visuell bekräftelse när en bild lagts till i favoriter. Det gjorde att man lätt trodde att inget hade hänt och fortsatte klicka. Jag löste detta genom att lägga till en reaktiv variabel som håller reda på om den aktuella bilden redan är sparad. När användaren klickar på knappen ändras texten direkt till Sparad, vilket gör det tydligt att bilden redan finns i favoriter och att knappen inte behöver tryckas igen.

Det sista problemet gällde videor. NASA:s API returnerar ibland YouTube- eller Vimeo-länkar i format som inte går att bädda in direkt, vilket gjorde att videor i favoriter inte visades alls. För att lösa detta skrev jag en funktion som känner igen olika typer av videolänkar och gör om dem till korrekta embed-länkar. Funktionen plockar ut video-ID:t och bygger en stabil URL som fungerar i en iframe. Jag tog även bort onödiga URL-parametrar för att undvika att Vue renderar om iframen i onödan. Efter detta visas videor i favoriter precis som bilder.

5. Resultat

Den färdiga applikationen innehåller tydlig navigering, API-anrop till NASA, loader och felhantering, stöd för både bilder och videor, ett favoritsystem, en realtidskarta över ISS, länkar till livevideo och satellit vy samt en responsiv och rymdinspirerad design.

6. Reflektion

Det här projektet har lärt mig mycket — inte bara om Vue och API-anrop, utan också om hur webbläsare fungerar, hur CORS kan sätta käppar i hjulen och hur externa tjänster påverkar vad som faktiskt går att bygga. Jag fick tänka om flera gånger, särskilt när det gällde livevideo och översättning, men i slutändan blev lösningarna bättre än mina första idéer.

Jag är nöjd med resultatet. Appen känns både pedagogisk och användarvänlig, och jag tycker att designen och funktionerna hänger ihop på ett bra sätt. Det känns som ett projekt jag kan stå för och som jag känner är häftigt att jag kunnat bygga med så pass mycket data som hämtas från NASA.

Min ursprungliga planering fungerade bra som grund, men under arbetets gång behövde jag justera flera delar, särskilt kring live-sidan och hanteringen av videor. Jag följde min designidé genom hela projektet, men jag fick också tänka om när tekniska begränsningar uppstod. I efterhand ser jag att jag hade kunnat lägga mer tid på förarbete och skisser, men samtidigt gav den iterativa processen mig möjlighet att hitta bättre lösningar än de jag först tänkte mig.

7. Skärmdumpar

Figur 1



Startsida för Rymdkollen i mobilversion.

Figur 2


Rymdkollen

[Start](#)[Dagens](#)[Slumpad](#)[Datum](#)[Favoriter](#)[Fakta](#)[Live](#)

Slumpad bild



M94: A New Perspective

2010-01-14



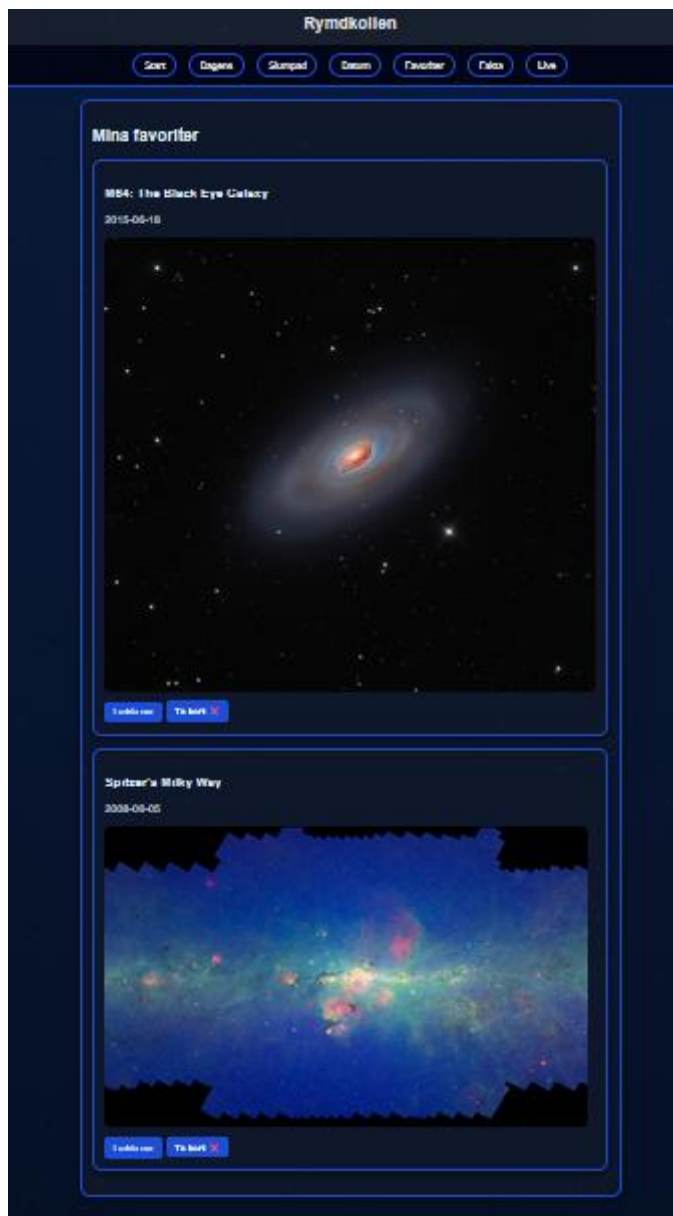
Bildtext från NASA

Beautiful island universe M94 lies a mere 15 million light-years distant in the northern constellation of the hunting dogs, Canes Venatici. A popular target for astronomers, the brighter inner part of the face-on spiral galaxy is about 30,000 light-years across. Traditionally, deep images have been interpreted as showing M94's inner spiral region surrounded by a faint, broad ring of stars. But a new multi-wavelength investigation has revealed previously undetected spiral arms sweeping across the outskirts of the galaxy's disk, an outer disk actively engaged in star formation. At optical wavelengths, M94's outer spiral arms are followed in this remarkable discovery image, processed to enhance the outer disk structure. Background galaxies are visible through the faint outer arms, while the three spiky foreground stars are in our own Milky Way galaxy. Note: An APOD editor will review great space images tonight in Houghton, Michigan.

[Lägg till i favoriter](#) [Slumpa igen](#) 

Exempel på APOD-visning med bild och bildtext.

Figur 3



Favoritsidan med sparade bilder.

Figur 4



Live-sidan med ISS-karta och länkar till livevideo och NASA Worldview.

8. Referenser

European Space Agency. (u.å.). *ISS Tracker*. <https://isstracker.spaceflight.esa.int/>

NASA. (u.å.-a). *Astronomy Picture of the Day API*. <https://api.nasa.gov/>

NASA. (u.å.-b). *Worldview*. NASA Earthdata. <https://worldview.earthdata.nasa.gov/>

Vue.js. (u.å.). *Vue 3 documentation*. <https://vuejs.org/>

Vue Router. (u.å.). *Vue Router documentation*. <https://router.vuejs.org/>

Wikimedia Commons. (u.å.). *Aldrin Apollo 11* [Fotografi].

https://en.wikipedia.org/wiki/File:Aldrin_Apollo_11.jpg

YouTube. (u.å.). *ISS live stream – sökresultat*. <https://www.youtube.com/>

Högskolan Väst. (u.å.-a). *Grunderna i Vue* [Videomaterial]. Kurs: JSR200.

<https://webbkurs.ei.hv.se/>

Högskolan Väst. (u.å.-b). *Webbapp i Vue* [Videomaterial]. Kurs: JSR200.

<https://webbkurs.ei.hv.se/>