

LITTERATURLISTE

og sån...

Alexander Wiencken

Senest redigeret: May 28, 2025

Indholdsfortegnelse

Forord	1
Litteratur- & kildeliste	2
Vulkan	7
Vulkan videoer	10
Vulkan sider	10
Flutter	13
Flutter videoer	14
Flutter sider	15
Andre	17

Forord

Denne bibliografi fungerer som en samlet kildeoversigt for 4. semester forløbet og er struktureret i to hoveddele. Først præsenteres en fælles liste over alle kilder, hvorefter de er opdelt efter emneområderne **Vulkan** og **Flutter**. Under hvert emne er kilderne yderligere kategoriseret i undersektioner som *videoer*, *artikler/hjemmesider* og andre relevante typer. Til sidst er der en sektion kaldet “**Andre**”, der indeholder diverse kilder, der ikke falder direkte ind under de nævnte emner.

Det skal nævnes, at mange af kilderne ikke har en tilknyttet bedømmelse¹. Dette skyldes, at de blev tilført i en tidlig fase af semesteret, før jeg begyndte systematisk at vurdere og anmelde hver enkelt kilde. Jeg arbejder gradvist på at tilføje disse bedømmelser, efterhånden som tid og muligheder tillader det.

¹Nogle mangler også en specifik “sidst tilgået” dato

Litteratur- & kildeliste

`std::vector<T,Allocator>::reserve`

cppreference.com. *std::vector<T,Allocator>::reserve*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/reserve>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve reserve".

`std::vector<T,Allocator>::resize`

cppreference.com. *std::vector<T,Allocator>::resize*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/resize>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve resize".

Discogs API

Discogs. *Discogs API*.

URL: <https://www.discogs.com/developers/>. (sidst tilgået: 03/03-2025).

Fundet ved at søge på Google "Discogs API".

Dart in 100 Seconds

Fireship. *Dart in 100 Seconds*.

URL: <https://www.youtube.com/watch?v=Nr00CJCbYLA>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Dart" på Fireships kanal.

Flutter Basic Training - 12 Minute Bootcamp

Fireship. *Flutter Basic Training - 12 Minute Bootcamp*.

Bedømmelse: Kender allerede til Fireship, som plejer at lave rimelig gode korte og konkrete videoer om diverse IT emner.

URL: <https://www.youtube.com/watch?v=1xipg02Wu8s>. (sidst tilgået: 15/02-2025).

Fundet ved at søge på Google "Flutter basics".

Flutter in 100 seconds

Fireship. *Flutter in 100 seconds*.

URL: <https://www.youtube.com/watch?v=1HhRhPV--G0>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Flutter" på Fireships kanal.

Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02

Brendan Galea. *Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02*.

Bedømmelse: Rimelig god video, som forklarer Vulkans pipeline på en måde så selv jeg kan forstå det (nogenlunde).

URL: https://www.youtube.com/watch?v=_riranMmtvI. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan Graphics Pipeline explained" på YouTube.

A Quick Overview of Vulkan Components

GetIntoGameDev. *A Quick Overview of Vulkan Components*.

URL: <https://www.youtube.com/watch?v=2NV1G9TFT1c>. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan overview" på YouTube.

Constructs

Google. *Constructs*.

Bedømmelse: God forklaring af hvordan constructors fungerer i Flutter, samt hvornår

man vil bruge sprog specifikke keywords som *factory*, og spøjse ting som *Redirecting constructors*.

URL: <https://dart.dev/language/constructors>. (sidst tilgået: 09/04-2025).

Fundet ved at søge på Google "Flutter factory constructor".

Flutter Samples

Google. *Flutter Samples*.

URL: <https://github.com/flutter/samples>. (sidst tilgået: 16/02-2025).

Tilgået fra flutter.dev, i et link i bunden hvor der står samples, hvis man går ind på én af eksemplerne og trykker "source code" bliver man ført til deres Github.

Intro to Dart

Google. *Intro to Dart*.

URL: <https://docs.flutter.dev/get-started/fundamentals/dart>. (sidst tilgået: xx/02-2025).

Tilgået fra flutter.dev ved at gå til docs, under Learn Flutter/Learn the fundamentals.

Introduction to Dart

Google. *Introduction to Dart*.

URL: <https://dart.dev/language>. (sidst tilgået: 16/02-2025).

Fundet ved at søge på Google "Dart".

Your first Flutter app

Google. *Your first Flutter app*.

URL: <https://codelabs.developers.google.com/codelabs/flutter-codelab-first>. (sidst tilgået: 17/03-2025).

Tilgået via flutter.dev docs under Learn Flutter/Write your first app, hvorefter den kan findes ved at trykke "Start codelabs".

Face Culling

Khronos. *Face Culling*.

Bedømmelse: Endnu en side der egentlig er “beregnet” til OpenGL, men det er ikke det sværeste at overføre til Vulkan da grund idéerne om winding og face culling er det samme, det sættes blot anderledes op.

URL: https://www.khronos.org/opengl/wiki/Face_Culling. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google ”Face culling”.

VkPhysicalDeviceFeatures(3) Manual Page

LunarG & Khronos. *VkPhysicalDeviceFeatures(3) Manual Page*.

URL: <https://registry.khronos.org/vulkan/specs/latest/man/html/VkPhysicalDeviceFeatures.html>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google ”Vulkan Physical Device”.

What is actually a Queue family in Vulkan?

Jesse Hall & Michael M. *What is actually a Queue family in Vulkan?*

Bedømmelse: Meget i dybde.

URL: <https://stackoverflow.com/questions/55272626/what-is-actually-a-queue-family-in-vulkan>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google ”Vulkan queue family”.

Vulkan Tutorial

Alexander Overvoorde. *Vulkan Tutorial*.

Bedømmelse: Super fed tutorial, som går meget i dybde med, at få lavet noget basalt i Vulkan, hvor man samtidigt bliver præsenteret for diverse ekstra ting til fx debugging. Den kommer bredt omkring hvordan man bl.a. også sætter sit dev. env. op, og sørger også for, at man lære nogle gode kode etiketter til når man sidder med Vulkan. Personligt, synes jeg ikke han går nok i dybde med hvad de forskellige ting er, meget af det er beskrevet meget overfladisk.

URL: <https://vulkan-tutorial.com>. (sidst tilgået: xx/04-2025).

Fundet ved at søge på Google "Vulkan tutorial", hvor det var den første side der blev foreslået.

Getting started with glsl

RIP Tutorial. *Getting started with glsl*.

Bedømmelse: Ikke så meget en 'getting started' da næsten hele siden bare er glsl eksempler med lidt til ingen forklaring. Det virker også til at siden mere er til at give et overblik over hvad man overhovedet kan med GLSL.

URL: <https://riptutorial.com/glsl>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

Shaders

Joey de Vries. *Shaders*.

Bedømmelse: Selvom hjemmesiden overall er egnet til at lære OpenGL, så er denne specifikke side/afsnit meget lærerig, da alt forklaringen om hvordan GLSL fungerer ikke bliver påvirket af om man bruger OpenGL eller Vulkan, og så er der masser af gode kode eksempler, ofte med tilhørende OpenGL output så man kan se resultatet.

URL: <https://learnopengl.com/getting-started/shaders>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

The new Vulkan Coordinate System

Matthew Wellings. *The new Vulkan Coordinate System*.

Bedømmelse: Kort, kontant og lige til med hvordan Vulkans NDC system er anderledes fra fx OpenGL.

URL: <https://matthewwellings.com/blog/the-new-vulkan-coordinate-system/>. (sidst tilgået: 28/05-2025).

Fundet ved at søge på Google "Vulkan Normalized Device Coordinates".

Vulkan

`std::vector<T,Allocator>::reserve`

cppreference.com. *std::vector<T,Allocator>::reserve*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/reserve>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve reserve".

`std::vector<T,Allocator>::resize`

cppreference.com. *std::vector<T,Allocator>::resize*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/resize>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve resize".

Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02

Brendan Galea. *Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02*.

Bedømmelse: Rimelig god video, som forklarer Vulkans pipeline på en måde så selv jeg kan forstå det (nogenlunde).

URL: https://www.youtube.com/watch?v=_riranMmtvI. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan Graphics Pipeline explained" på YouTube.

A Quick Overview of Vulkan Components

GetIntoGameDev. *A Quick Overview of Vulkan Components*.

URL: <https://www.youtube.com/watch?v=2NV1G9TFT1c>. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan overview" på YouTube.

Face Culling

Khronos. *Face Culling*.

Bedømmelse: Endnu en side der egentlig er "beregnet" til OpenGL, men det er ikke det sværeste at overføre til Vulkan da grund idéerne om winding og face culling er det samme, det sættes blot anderledes op.

URL: https://www.khronos.org/opengl/wiki/Face_Culling. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Face culling".

VkPhysicalDeviceFeatures(3) Manual Page

LunarG & Khronos. *VkPhysicalDeviceFeatures(3) Manual Page*.

URL: <https://registry.khronos.org/vulkan/specs/latest/man/html/VkPhysicalDeviceFeatures.html>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google "Vulkan Physical Device".

What is actually a Queue family in Vulkan?

Jesse Hall & Michael M. *What is actually a Queue family in Vulkan?*

Bedømmelse: Meget i dybde.

URL: <https://stackoverflow.com/questions/55272626/what-is-actually-a-queue-family-in-vulkan>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google "Vulkan queue family".

Vulkan Tutorial

Alexander Overvoorde. *Vulkan Tutorial*.

Bedømmelse: Super fed tutorial, som går meget i dybde med, at få lavet noget basalt i Vulkan, hvor man samtidigt bliver præsenteret for diverse ekstra ting til fx debugging. Den kommer bredt omkring hvordan man bl.a. også sætter sit dev. env. op, og sørger også for, at man lære nogle gode kode etiketter til når man sidder med Vulkan. Personligt, synes jeg ikke han går nok i dybde med hvad de forskellige ting

er, meget af det er beskrevet meget overfladisk.

URL: <https://vulkan-tutorial.com>. (sidst tilgået: xx/04-2025).

Fundet ved at søge på Google "Vulkan tutorial", hvor det var den første side der blev foreslået.

Getting started with glsl

RIP Tutorial. *Getting started with glsl*.

Bedømmelse: Ikke så meget en 'getting started' da næsten hele siden bare er glsl eksempler med lidt til ingen forklaring. Det virker også til at siden mere er til at give et overblik over hvad man overhovedet kan med GLSL.

URL: <https://riptutorial.com/glsl>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

Shaders

Joey de Vries. *Shaders*.

Bedømmelse: Selvom hjemmesiden overall er egnet til at lære OpenGL, så er denne specifikke side/afsnit meget lærerig, da alt forklaringen om hvordan GLSL fungerer ikke bliver påvirket af om man bruger OpenGL eller Vulkan, og så er der masser af gode kode eksempler, ofte med tilhørende OpenGL output så man kan se resultatet.

URL: <https://learnopengl.com/getting-started/shaders>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

The new Vulkan Coordinate System

Matthew Wellings. *The new Vulkan Coordinate System*.

Bedømmelse: Kort, kontant og lige til med hvordan Vulkans NDC system er anderledes fra fx OpenGL.

URL: <https://matthewwellings.com/blog/the-new-vulkan-coordinate-system/>. (sidst tilgået: 28/05-2025).

Fundet ved at søge på Google "Vulkan Normalized Device Coordinates".

Vulkan videoer

Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02

Brendan Galea. *Graphics Pipeline Overview - Vulkan Game Engine Tutorial 02*.

Bedømmelse: Rimelig god video, som forklarer Vulkans pipeline på en måde så selv jeg kan forstå det (nogenlunde).

URL: https://www.youtube.com/watch?v=_riranMmtvI. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan Graphics Pipeline explained" på YouTube.

A Quick Overview of Vulkan Components

GetIntoGameDev. *A Quick Overview of Vulkan Components*.

URL: <https://www.youtube.com/watch?v=2NVlG9TFT1c>. (sidst tilgået: 14/02-2025).

Fundet ved at søge "Vulkan overview" på YouTube.

Vulkan sider

`std::vector<T,Allocator>::reserve`

cppreference.com. *std::vector<T,Allocator>::reserve*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/reserve>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve reserve".

`std::vector<T,Allocator>::resize`

cppreference.com. *std::vector<T,Allocator>::resize*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/resize>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve resize".

Face Culling

Khronos. *Face Culling*.

Bedømmelse: Endnu en side der egentlig er "beregnet" til OpenGL, men det er ikke det sværeste at overføre til Vulkan da grund idéerne om winding og face culling er det samme, det sættes blot anderledes op.

URL: https://www.khronos.org/opengl/wiki/Face_Culling. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Face culling".

VkPhysicalDeviceFeatures(3) Manual Page

LunarG & Khronos. *VkPhysicalDeviceFeatures(3) Manual Page*.

URL: <https://registry.khronos.org/vulkan/specs/latest/man/html/VkPhysicalDeviceFeatures.html>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google "Vulkan Physical Device".

What is actually a Queue family in Vulkan?

Jesse Hall & Michael M. *What is actually a Queue family in Vulkan?*

Bedømmelse: Meget i dybde.

URL: <https://stackoverflow.com/questions/55272626/what-is-actually-a-queue-family-in-vulkan>. (sidst tilgået: xx/02-2025).

Fundet ved at søge på Google "Vulkan queue family".

Vulkan Tutorial

Alexander Overvoorde. *Vulkan Tutorial*.

Bedømmelse: Super fed tutorial, som går meget i dybde med, at få lavet noget basalt i Vulkan, hvor man samtidigt bliver præsenteret for diverse ekstra ting til fx debugging. Den kommer bredt omkring hvordan man bl.a. også sætter sit dev. env. op, og sørger også for, at man lære nogle gode kode etiketter til når man sidder med Vulkan. Personligt, synes jeg ikke han går nok i dybde med hvad de forskellige ting

er, meget af det er beskrevet meget overfladisk.

URL: <https://vulkan-tutorial.com>. (sidst tilgået: xx/04-2025).

Fundet ved at søge på Google "Vulkan tutorial", hvor det var den første side der blev foreslået.

Getting started with glsl

RIP Tutorial. *Getting started with glsl*.

Bedømmelse: Ikke så meget en 'getting started' da næsten hele siden bare er glsl eksempler med lidt til ingen forklaring. Det virker også til at siden mere er til at give et overblik over hvad man overhovedet kan med GLSL.

URL: <https://riptutorial.com/glsl>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

Shaders

Joey de Vries. *Shaders*.

Bedømmelse: Selvom hjemmesiden overall er egnet til at lære OpenGL, så er denne specifikke side/afsnit meget lærerig, da alt forklaringen om hvordan GLSL fungerer ikke bliver påvirket af om man bruger OpenGL eller Vulkan, og så er der masser af gode kode eksempler, ofte med tilhørende OpenGL output så man kan se resultatet.

URL: <https://learnopengl.com/getting-started/shaders>. (sidst tilgået: 27/05-2025).

Fundet ved at søge på Google "Basic GLSL tutorial".

The new Vulkan Coordinate System

Matthew Wellings. *The new Vulkan Coordinate System*.

Bedømmelse: Kort, kontant og lige til med hvordan Vulkans NDC system er anderledes fra fx OpenGL.

URL: <https://matthewwellings.com/blog/the-new-vulkan-coordinate-system/>. (sidst tilgået: 28/05-2025).

Fundet ved at søge på Google "Vulkan Normalized Device Coordinates".

Flutter

Dart in 100 Seconds

Fireship. *Dart in 100 Seconds*.

URL: <https://www.youtube.com/watch?v=Nr00CJCbyLA>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Dart" på Fireships kanal.

Flutter Basic Training - 12 Minute Bootcamp

Fireship. *Flutter Basic Training - 12 Minute Bootcamp*.

Bedømmelse: Kender allerede til Fireship, som plejer at lave rimelig gode korte og konkrete videoer om diverse IT emner.

URL: <https://www.youtube.com/watch?v=1xipg02Wu8s>. (sidst tilgået: 15/02-2025).

Fundet ved at søge på Google "Flutter basics".

Flutter in 100 seconds

Fireship. *Flutter in 100 seconds*.

URL: <https://www.youtube.com/watch?v=1HhRhPV--G0>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Flutter" på Fireships kanal.

Constructs

Google. *Constructs*.

Bedømmelse: God forklaring af hvordan constructors fungerer i Flutter, samt hvornår man vil bruge sprog specifikke keywords som *factory*, og spøjse ting som *Redirecting constructors*.

URL: <https://dart.dev/language/constructors>. (sidst tilgået: 09/04-2025).

Fundet ved at søge på Google "Flutter factory constructor".

Flutter Samples

Google. *Flutter Samples*.

URL: <https://github.com/flutter/samples>. (sidst tilgået: 16/02-2025).

Tilgået fra flutter.dev, i et link i bunden hvor der står samples, hvis man går ind på én af eksemplerne og trykker "source code" bliver man ført til deres Github.

Intro to Dart

Google. *Intro to Dart*.

URL: <https://docs.flutter.dev/get-started/fundamentals/dart>. (sidst tilgået: xx/02-2025).

Tilgået fra flutter.dev ved at gå til docs, under Learn Flutter/Learn the fundamentals.

Introduction to Dart

Google. *Introduction to Dart*.

URL: <https://dart.dev/language>. (sidst tilgået: 16/02-2025).

Fundet ved at søge på Google "Dart".

Your first Flutter app

Google. *Your first Flutter app*.

URL: <https://codelabs.developers.google.com/codelabs/flutter-codelab-first>. (sidst tilgået: 17/03-2025).

Tilgået via flutter.dev docs under Learn Flutter/Write your first app, hvorefter den kan findes ved at trykke "Start codelabs".

Flutter videoer

Dart in 100 Seconds

Fireship. *Dart in 100 Seconds*.

URL: <https://www.youtube.com/watch?v=Nr00CJCbYLA>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Dart" på Fireships kanal.

Flutter Basic Training - 12 Minute Bootcamp

Fireship. *Flutter Basic Training - 12 Minute Bootcamp*.

Bedømmelse: Kender allerede til Fireship, som plejer at lave rimelig gode korte og konkrete videoer om diverse IT emner.

URL: <https://www.youtube.com/watch?v=1xipg02Wu8s>. (sidst tilgået: 15/02-2025).

Fundet ved at søge på Google "Flutter basics".

Flutter in 100 seconds

Fireship. *Flutter in 100 seconds*.

URL: <https://www.youtube.com/watch?v=1HhRhPV--G0>. (sidst tilgået: xx/02-2025).

Fundet ved at søge efter "Flutter" på Fireships kanal.

Flutter sider

Constructs

Google. *Constructs*.

Bedømmelse: God forklaring af hvordan constructors fungerer i Flutter, samt hvornår man vil bruge sprog specifikke keywords som *factory*, og spøjse ting som *Redirecting constructors*.

URL: <https://dart.dev/language/constructors>. (sidst tilgået: 09/04-2025).

Fundet ved at søge på Google "Flutter factory constructor".

Flutter Samples

Google. *Flutter Samples*.

URL: <https://github.com/flutter/samples>. (sidst tilgået: 16/02-2025).

Tilgået fra flutter.dev, i et link i bunden hvor der står samples, hvis man går ind på én af eksemplerne og trykker "source code" bliver man ført til deres Github.

Intro to Dart

Google. *Intro to Dart*.

URL: `https://docs.flutter.dev/get-started/fundamentals/dart`. (sidst tilgået: xx/02-2025).

Tilgået fra flutter.dev ved at gå til docs, under Learn Flutter/Learn the fundamentals.

Introduction to Dart

Google. *Introduction to Dart*.

URL: `https://dart.dev/language`. (sidst tilgået: 16/02-2025).

Fundet ved at søge på Google "Dart".

Your first Flutter app

Google. *Your first Flutter app*.

URL: `https://codelabs.developers.google.com/codelabs/flutter-codelab-first`. (sidst tilgået: 17/03-2025).

Tilgået via flutter.dev docs under Learn Flutter/Write your first app, hvorefter den kan findes ved at trykke "Start codelabs".

Andre

`std::vector<T,Allocator>::reserve`

cppreference.com. *std::vector<T,Allocator>::reserve*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/reserve>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve reserve".

`std::vector<T,Allocator>::resize`

cppreference.com. *std::vector<T,Allocator>::resize*.

Bedømmelse: Jeg har brugt cppreference mange gange i mintid som C++ interesseret. Det er en ældre side, og den kan være skrevet ret spøjst til tider, men den har en masse information på meget lidt 'plads', endda nogle gange med eksempler.

URL: <https://en.cppreference.com/w/cpp/container/vector/resize>. (sidst tilgået: 24/04-2025).

Fundet ved at søge på Google "cppreference c++ reserve resize".

Discogs API

Discogs. *Discogs API*.

URL: <https://www.discogs.com/developers/>. (sidst tilgået: 03/03-2025).

Fundet ved at søge på Google "Discogs API".