From Idea to Prototype #2

Compute Specs DB

in @ndsz

Introducing a new series: From Problem to Prototype

Execution over perfection.

What

In this series, I take a real problem, design **the smallest useful solution**, and ship it. Fast, intentionally scoped, and shared early. Using my technical skills and knowledge combined with AI tools.

Why

Because I genuinely enjoy bringing ideas to life. I learn the most when I build, and I believe good ideas become better once they exist in the real world, even the prototypes.

How

By starting simple and being intentional:

- one clear problem
- a realistic scope
- rapid execution (max 12h of work)
- early sharing and iteration

Compute Specs DB

This project started from a simple, practical need: **servers specifications for datacenter hardware are scattered and time-consuming to find and compare**. I needed a single place to search, filter, visualize, and compare specs without jumping between vendor pages, wikis, and spreadsheets. I also needed a fast way to get relevant information such as Turbo Frequency, L3 Cache, Launch Year or TDP - all in one place.

So I built a website with an API that would:

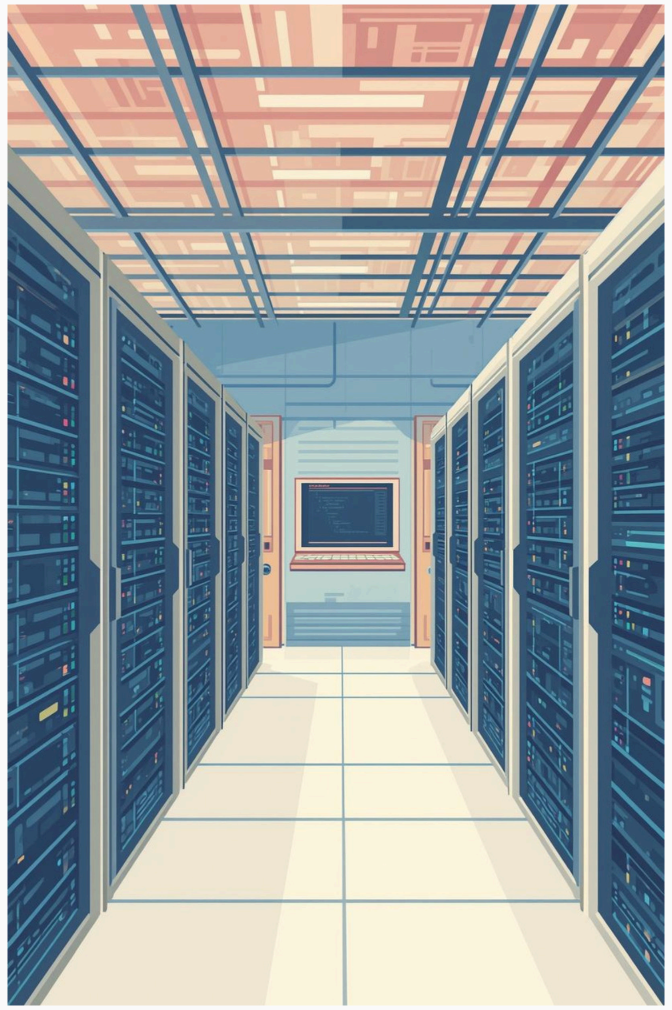
- **Solve my problem:** one searchable, filterable source for HPC/datacenter CPU/GPU specs.
- **Support end-to-end use:** search in the UI, explore in charts, compare two CPUs/GPUs, and use the same data via API or export.
- **Result in something concrete**: API I can share, link in docs, and, if it proves reliable, reuse at work (e.g. as an internal API or fork).

Important: I personally validated **every entry** against the most reliable sources available, official Intel and AMD specification pages, TechPowerUp's CPU database, and PassMark benchmarks. That said, with 170 entries and 10+ fields each, mistake can happen.
If you find one, I'd appreciate you reporting it on GitHub, <u>it helps make the dataset better for everyone.</u>

Contribution model: Public access is **read-only**. Users can request additions or report errors via GitHub issue templates,
but all changes go through manual review and validation against official sources before being applied. No unverified data enters the database.
<u>That is a deliberate choice for data quality over openness.</u>

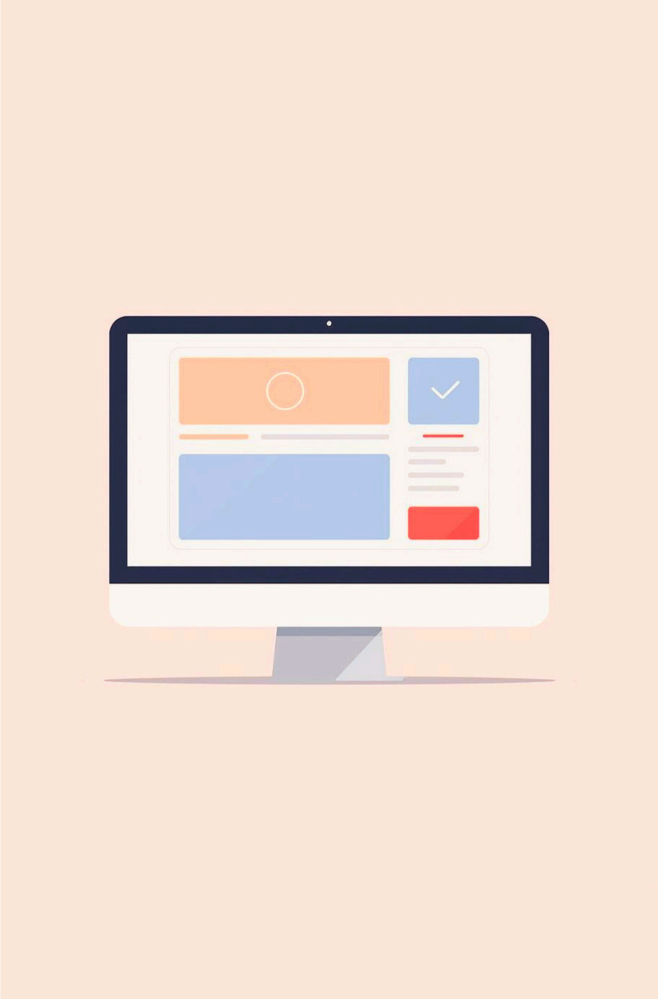
Even at this stage, the project is useful for me, and hopefully for others who need to search and compare CPU specs quickly. The goal is to make it reliable enough that it could be forked or used as an API in real projects (e.g. at work) later, once everything missing is in place.

Tech Stack Overview



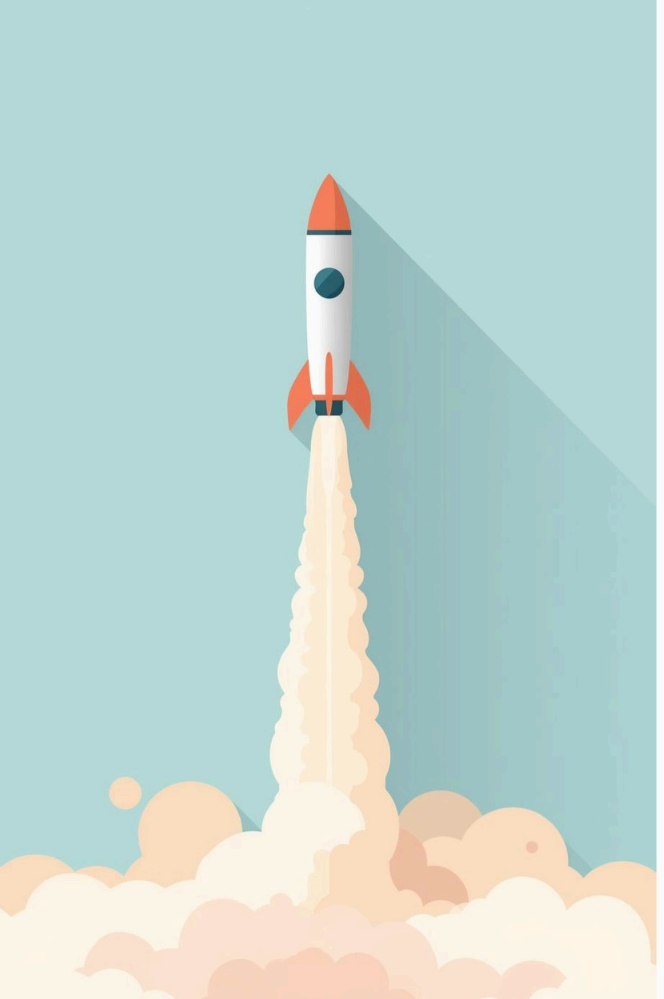
Backend

- Python
- FastAPI
- SQLite
- JWT
- passlib/bcrypt



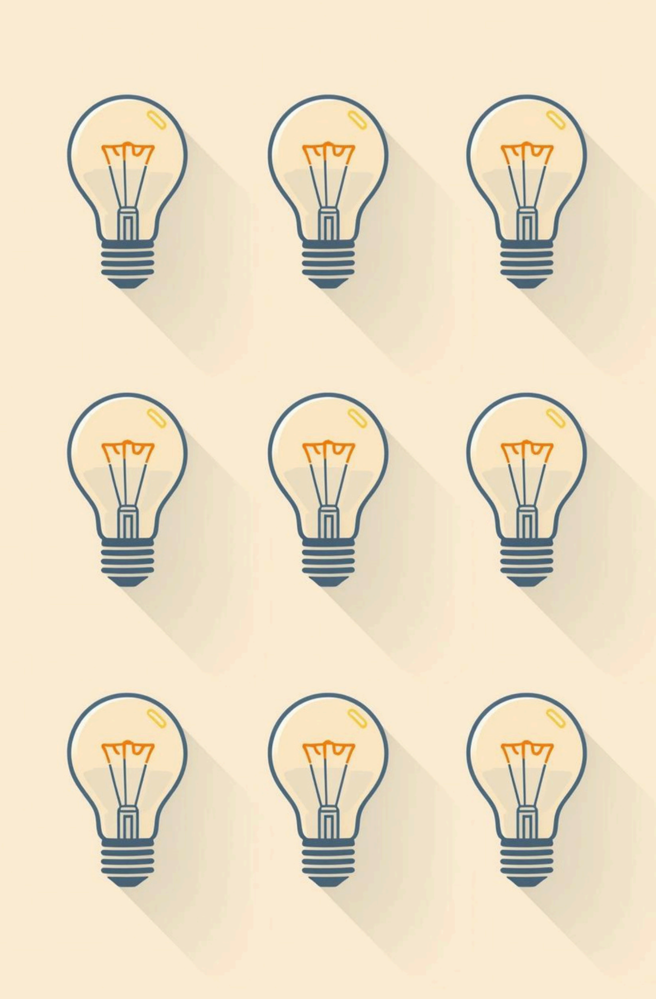
Frontend

- Vanilla
- HTML/CSS/JavaScript
- Plotly.js for interactive charts



Deployment

- Render.com
- CloudFlare
- Github



Features

- Searchable/sortable table
- Summary stats
- Interactive visualizations
- REST over FastAPI

Step 1

Search or browse the database. From the main page, use the search bar to filter by model, family, codename, or CPU model. Results are displayed in a sortable, paginated table. Summary cards at the top show key stats like total CPUs, families, codenames, max cores, and year range. For raw data, click "All CPUs" to view the full dataset in JSON format.

Step 2

Explore visualizations.
Open "*Visualizations*," pick X-axis and Y-axis parameters (e.g. TDP, Max Turbo Frequency, Launch Year) and chart type (scatter, bar, box). Use "Filter by Family" and "Filter by Codename"

Step 3

Compare two CPUs. On the same Visualizations page, scroll to the CPU Comparison Tool. Select CPU 1 and CPU 2 from the dropdowns, click "Compare CPUs," then review the side-by-side table and difference column (with % where applicable). Use "Download Comparison" to export as CSV.

Step 4

Use the API or export the dataset.
Open "API Docs" to try endpoints (e.g. `GET /api/cpus/search?q=intel`) or use "API Info" for a quick reference. Export full data via `/api/export/csv` or `/api/export/excel` for offline analysis.

Step 1

🔧 Compute Specs DB

HPC and Datacenter Compute Specifications

Search by model, family, codename, or CPU model (e.g., Mila

Search

📡 Access JSON API:

📋 All CPUs | 📊 Visualizations | ℹ️ API Info | 📚 API Docs

169 Total CPUs | **10** Families | **25** Codenames | **128** Max Cores | **2009–2024** Year Range

ID	CPU MODEL NAME	CODENAME	FAMILY	CPU MODEL	CORES	THREADS	TURBO GHZ	L3 MB	TDP (W)	YEAR	MEM TB
1	**AMD EPYC 4564P 16-Core Processor**	Genoa	AMD EPYC	EPYC 4564P	16	32	5.7	64	170	2024	1
2	**AMD EPYC 7262 8-Core Processor**	Rome	AMD EPYC	EPYC 7262	8	16	3.4	128	155	2019	4
3	**AMD EPYC 7282 16-Core Processor**	Rome	AMD EPYC	EPYC 7282	16	32	3.2	64	120	2019	4
4	**AMD EPYC 7301 16-Core Processor**	Naples	AMD EPYC	EPYC 7301	16	32	2.7	64	170	2017	4
5	**AMD EPYC 7302 16-Core Processor**	Rome	AMD EPYC	EPYC 7302	16	32	3.3	128	155	2019	4
6	**AMD EPYC 7313 16-Core Processor**	Milan	AMD EPYC	EPYC 7313	16	32	3.7	128	155	2021	4
7	**AMD EPYC 7351 16-Core Processor**	Naples	AMD EPYC	EPYC 7351	16	32	2.9	64	170	2017	2
8	**AMD EPYC 7351P 16-Core Processor**	Naples	AMD EPYC	EPYC 7351P	16	32	2.9	64	170	2017	2

Step 1

Search or browse the database. From the main page, use the search bar to filter by model, family, codename, or CPU model. Results are displayed in a sortable, paginated table. Summary cards at the top show key stats like total CPUs, families, codenames, max cores, and year range. For raw data, click "All CPUs" to view the full dataset in JSON format.

Step 2

Explore visualizations.
Open "*Visualizations*," pick X-axis and Y-axis parameters (e.g. TDP, Max Turbo Frequency, Launch Year) and chart type (scatter, bar, box). Use "Filter by Family" and "Filter by Codename"

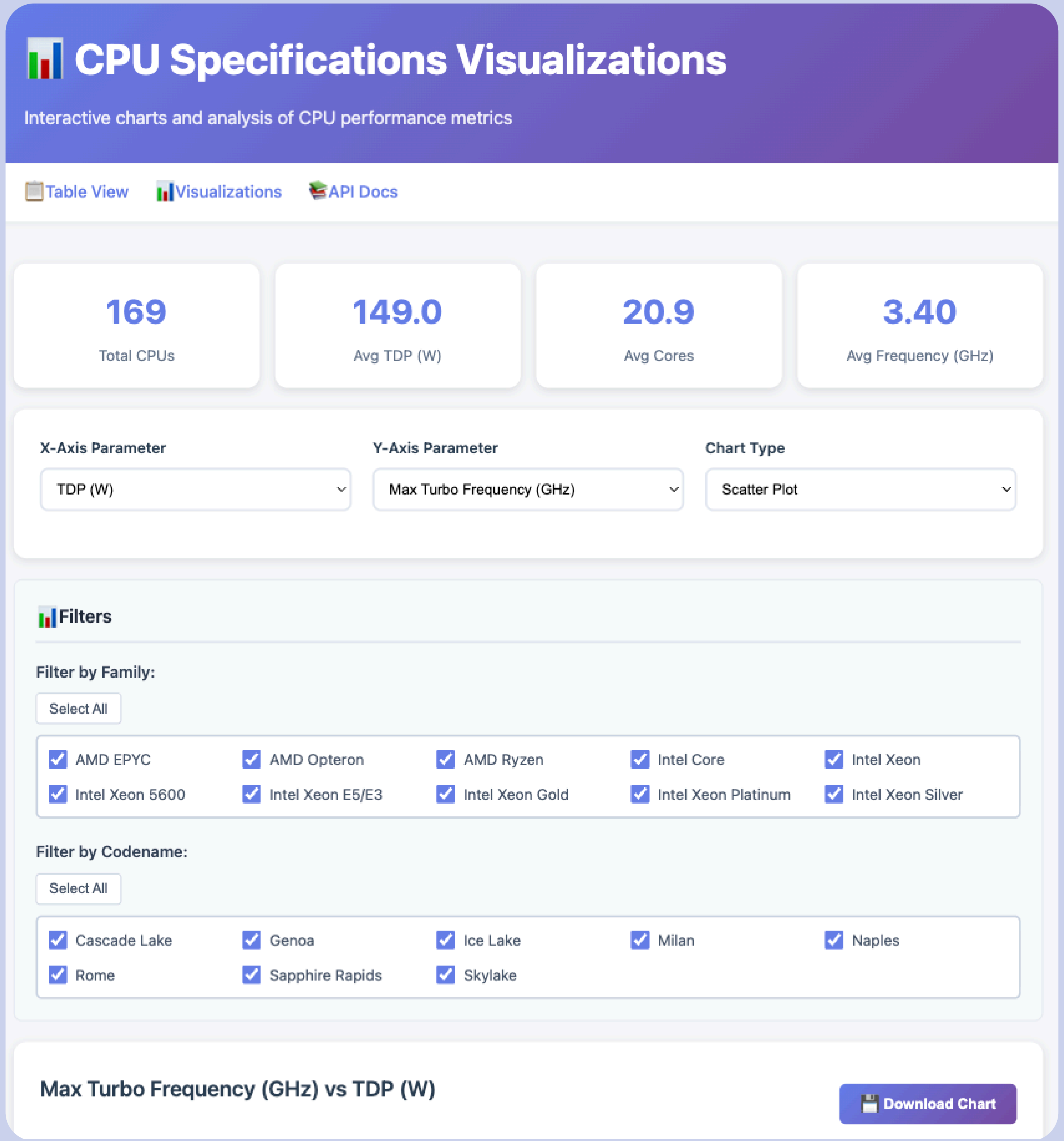
Step 3

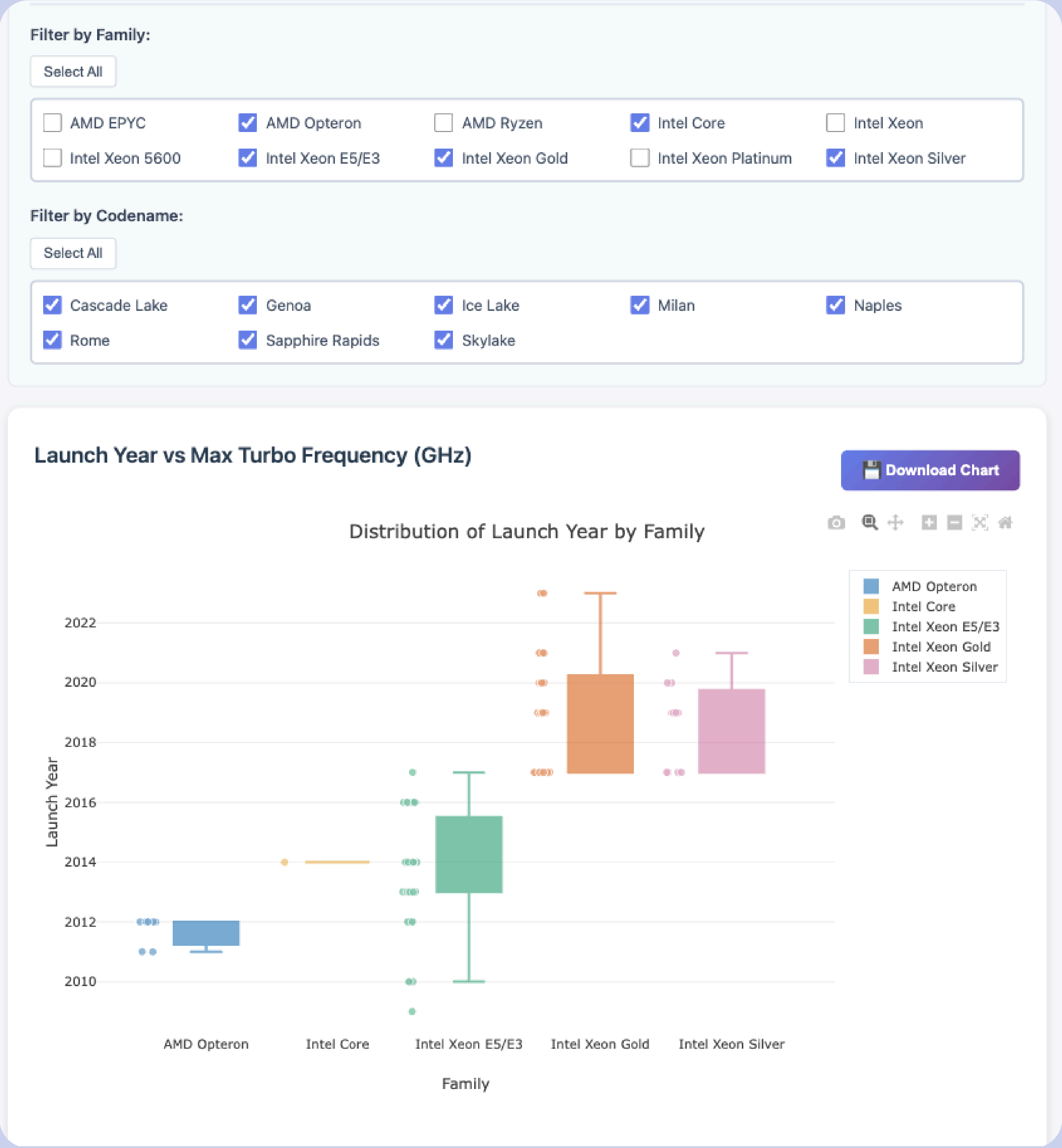
Compare two CPUs. On the same Visualizations page, scroll to the CPU Comparison Tool. Select CPU 1 and CPU 2 from the dropdowns, click "Compare CPUs," then review the side-by-side table and difference column (with % where applicable). Use "Download Comparison" to export as CSV.

Step 4

Use the API or export the dataset.
Open "API Docs" to try endpoints (e.g. `GET /api/cpus/search?q=intel`) or use "API Info" for a quick reference. Export full data via `/api/export/csv` or `/api/export/excel` for offline analysis.

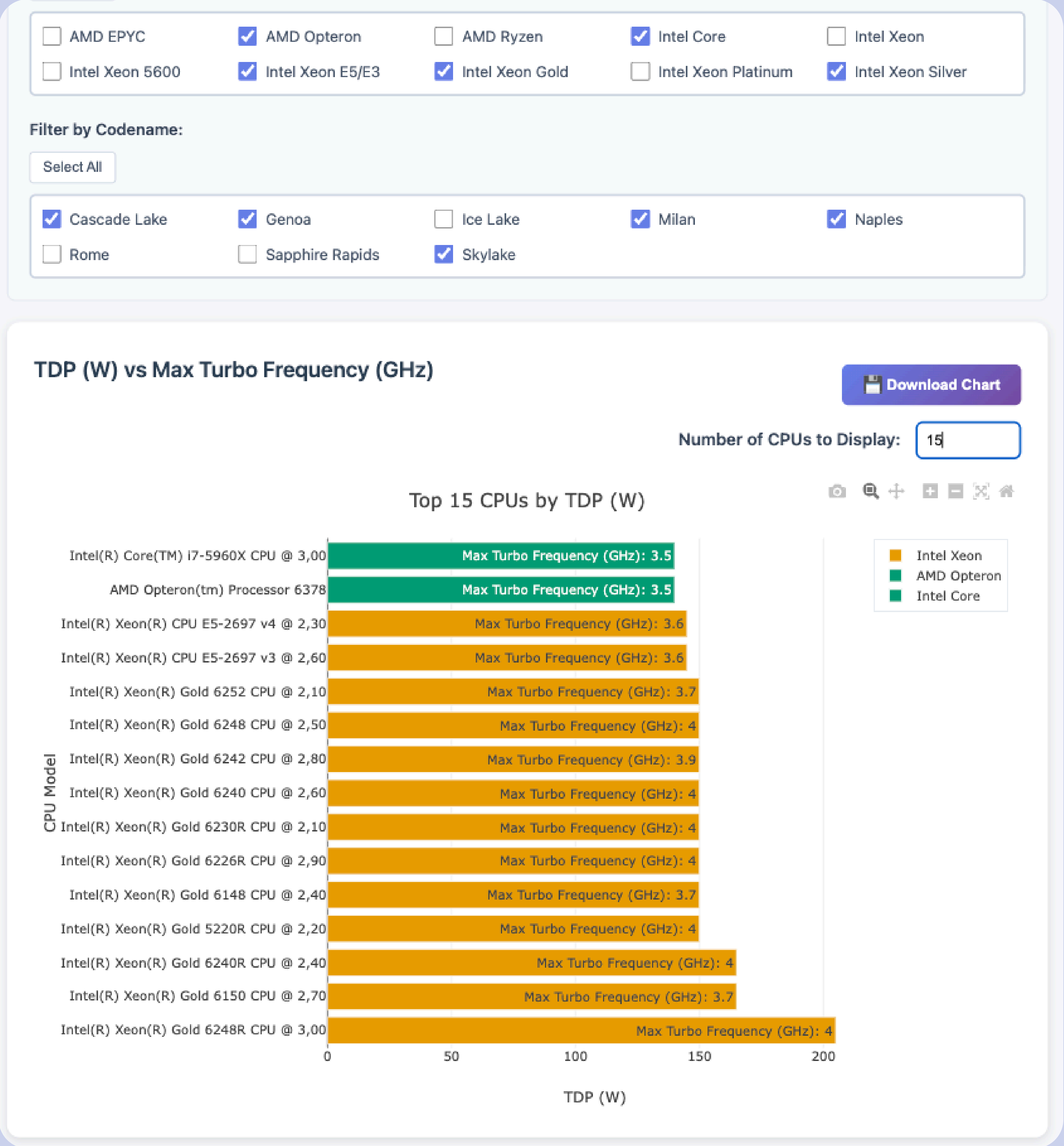
Step 2


CPU Specifications Visualizations
Interactive charts and analysis of CPU performance metrics
Table View
Visualizations
API Docs
169
Total CPUs
149.0
Avg TDP (W)
20.9
Avg Cores
3.40
Avg Frequency (GHz)
X-Axis Parameter
TDP (W)
Y-Axis Parameter
Max Turbo Frequency (GHz)
Chart Type
Scatter Plot
Filters
Filter by Family:
Select All
AMD EPYC
AMD Opteron
AMD Ryzen
Intel Core
Intel Xeon
Intel Xeon 5600
Intel Xeon E5/E3
Intel Xeon Gold
Intel Xeon Platinum
Intel Xeon Silver
Filter by Codename:
Select All
Cascade Lake
Genoa
Ice Lake
Milan
Naples
Rome
Sapphire Rapids
Skylake
Max Turbo Frequency (GHz) vs TDP (W)
Download Chart


Filter by Family:
Select All
AMD EPYC
AMD Opteron
AMD Ryzen
Intel Core
Intel Xeon
Intel Xeon 5600
Intel Xeon E5/E3
Intel Xeon Gold
Intel Xeon Platinum
Intel Xeon Silver
Filter by Codename:
Select All
Cascade Lake
Genoa
Ice Lake
Milan
Naples
Rome
Sapphire Rapids
Skylake
Launch Year vs Max Turbo Frequency (GHz)
Download Chart
Distribution of Launch Year by Family
AMD Opteron
Intel Core
Intel Xeon E5/E3
Intel Xeon Gold
Intel Xeon Silver
2022
2020
2018
2016
2014
2012
2010
Launch Year
AMD Opteron
Intel Core
Intel Xeon E5/E3
Intel Xeon Gold
Intel Xeon Silver
Family

@ndsz

Step 2


AMD EPYC
AMD Opteron
AMD Ryzen
Intel Core
Intel Xeon
Intel Xeon 5600
Intel Xeon E5/E3
Intel Xeon Gold
Intel Xeon Platinum
Intel Xeon Silver
Filter by Codename:
Select All
Cascade Lake
Genoa
Ice Lake
Milan
Naples
Rome
Sapphire Rapids
Skylake
TDP (W) vs Max Turbo Frequency (GHz)
Download Chart
Number of CPUs to Display: 15
Top 15 CPUs by TDP (W)
Intel(R) Core(TM) i7-5960X CPU @ 3,00
AMD Opteron(tm) Processor 6378
Intel(R) Xeon(R) CPU E5-2697 v4 @ 2,30
Intel(R) Xeon(R) CPU E5-2697 v3 @ 2,60
Intel(R) Xeon(R) Gold 6252 CPU @ 2,10
Intel(R) Xeon(R) Gold 6248 CPU @ 2,50
Intel(R) Xeon(R) Gold 6242 CPU @ 2,80
Intel(R) Xeon(R) Gold 6240 CPU @ 2,60
Intel(R) Xeon(R) Gold 6230R CPU @ 2,10
Intel(R) Xeon(R) Gold 6226R CPU @ 2,90
Intel(R) Xeon(R) Gold 6148 CPU @ 2,40
Intel(R) Xeon(R) Gold 5220R CPU @ 2,20
Intel(R) Xeon(R) Gold 6240R CPU @ 2,40
Intel(R) Xeon(R) Gold 6150 CPU @ 2,70
Intel(R) Xeon(R) Gold 6248R CPU @ 3,00
Max Turbo Frequency (GHz): 3.5
Max Turbo Frequency (GHz): 3.5
Max Turbo Frequency (GHz): 3.6
Max Turbo Frequency (GHz): 3.6
Max Turbo Frequency (GHz): 3.7
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 3.9
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 3.7
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 4
Max Turbo Frequency (GHz): 3.7
Max Turbo Frequency (GHz): 4
CPU Model
0
50
100
150
200
TDP (W)
Intel Xeon
AMD Opteron
Intel Core


Filter by Codename:
Select All
Cascade Lake
Genoa
Ice Lake
Milan
Naples
Rome
Sapphire Rapids
Skylake
Max Turbo Frequency (GHz) vs Launch Year
Download Chart
Max Turbo Frequency (GHz) vs Launch Year
Max Turbo Frequency (GHz)
4.5
4
3.5
3
2.5
2008
2010
2012
2014
2016
2018
2020
2022
2024
Launch Year
AMD EPYC
AMD Opteron
AMD Ryzen
Intel Core
Intel Xeon
Intel Xeon E5/E3
Intel Xeon 5600
Intel Xeon Gold
Intel Xeon Platinum
Intel Xeon Silver
CPU Comparison Tool
CPU 1
CPU 2

@ndsz

Step 1

Search or browse the database. From the main page, use the search bar to filter by model, family, codename, or CPU model. Results are displayed in a sortable, paginated table. Summary cards at the top show key stats like total CPUs, families, codenames, max cores, and year range. For raw data, click "All CPUs" to view the full dataset in JSON format.

Step 2

Explore visualizations.
Open "*Visualizations*," pick X-axis and Y-axis parameters (e.g. TDP, Max Turbo Frequency, Launch Year) and chart type (scatter, bar, box). Use "Filter by Family" and "Filter by Codename"

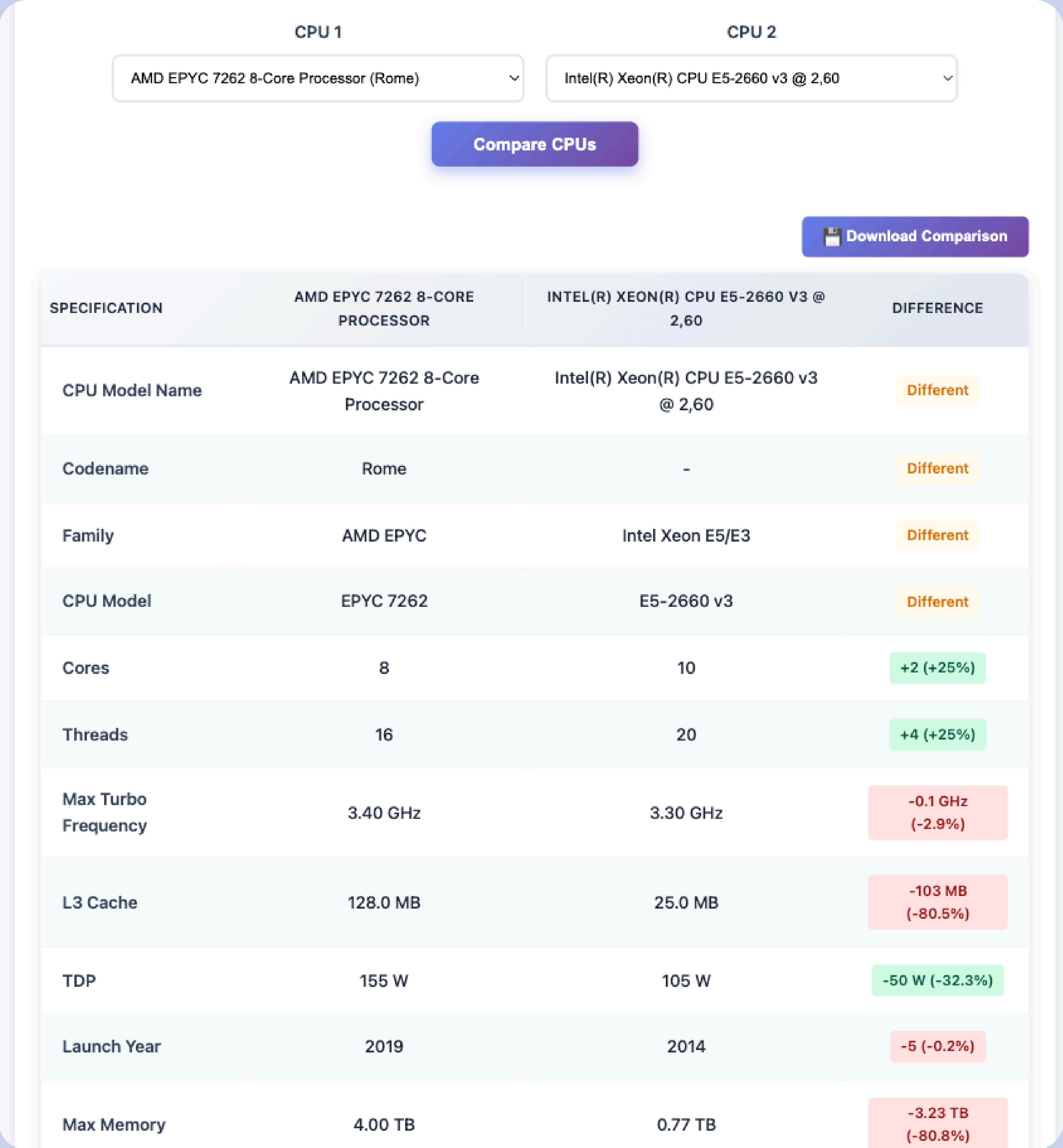
Step 3

Compare two CPUs. On the same *Visualizations* page, scroll to the CPU Comparison Tool. Select CPU 1 and CPU 2 from the dropdowns, click "Compare CPUs," then review the side-by-side table and difference column (with % where applicable). Use "Download Comparison" to export as CSV.

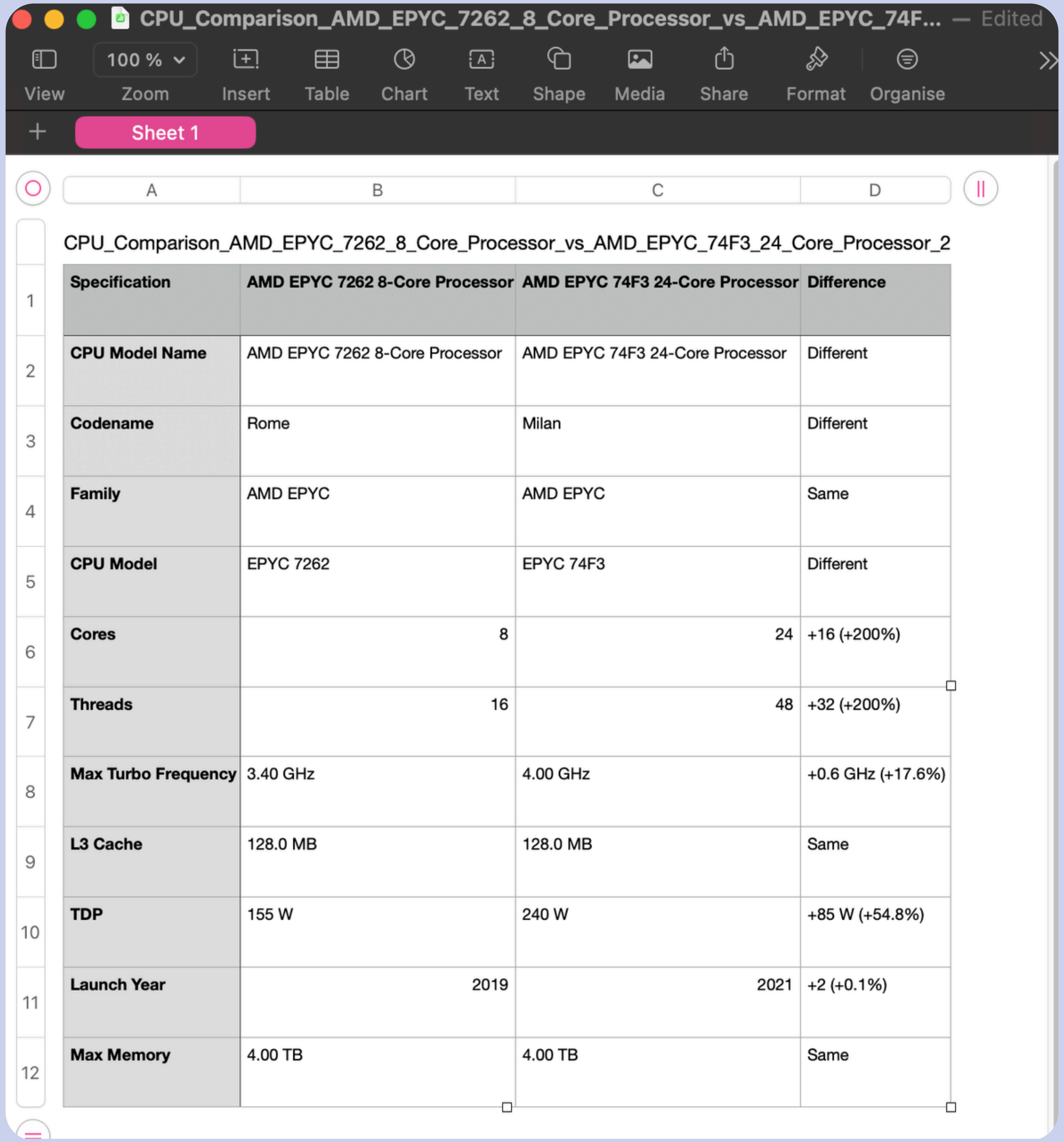
Step 4

Use the API or export the dataset.
Open "API Docs" to try endpoints (e.g. `GET /api/cpus/search?q=intel`) or use "API Info" for a quick reference. Export full data via `/api/export/csv` or `/api/export/excel` for offline analysis.

Step 3



SPECIFICATION	AMD EPYC 7262 8-CORE PROCESSOR	INTEL(R) XEON(R) CPU E5-2660 V3 @ 2,60	DIFFERENCE
CPU Model Name	AMD EPYC 7262 8-Core Processor	Intel(R) Xeon(R) CPU E5-2660 v3 @ 2,60	Different
Codename	Rome	-	Different
Family	AMD EPYC	Intel Xeon E5/E3	Different
CPU Model	EPYC 7262	E5-2660 v3	Different
Cores	8	10	+2 (+25%)
Threads	16	20	+4 (+25%)
Max Turbo Frequency	3.40 GHz	3.30 GHz	-0.1 GHz (-2.9%)
L3 Cache	128.0 MB	25.0 MB	-103 MB (-80.5%)
TDP	155 W	105 W	-50 W (-32.3%)
Launch Year	2019	2014	-5 (-0.2%)
Max Memory	4.00 TB	0.77 TB	-3.23 TB (-80.8%)



Specification	AMD EPYC 7262 8-Core Processor	AMD EPYC 74F3 24-Core Processor	Difference
CPU Model Name	AMD EPYC 7262 8-Core Processor	AMD EPYC 74F3 24-Core Processor	Different
Codename	Rome	Milan	Different
Family	AMD EPYC	AMD EPYC	Same
CPU Model	EPYC 7262	EPYC 74F3	Different
Cores	8	24	+16 (+200%)
Threads	16	48	+32 (+200%)
Max Turbo Frequency	3.40 GHz	4.00 GHz	+0.6 GHz (+17.6%)
L3 Cache	128.0 MB	128.0 MB	Same
TDP	155 W	240 W	+85 W (+54.8%)
Launch Year	2019	2021	+2 (+0.1%)
Max Memory	4.00 TB	4.00 TB	Same

@ndsz

Step 1

Search or browse the database. From the main page, use the search bar to filter by model, family, codename, or CPU model. Results are displayed in a sortable, paginated table. Summary cards at the top show key stats like total CPUs, families, codenames, max cores, and year range. For raw data, click "All CPUs" to view the full dataset in JSON format.

Step 2

Explore visualizations.
Open "*Visualizations*," pick X-axis and Y-axis parameters (e.g. TDP, Max Turbo Frequency, Launch Year) and chart type (scatter, bar, box). Use "Filter by Family" and "Filter by Codename"

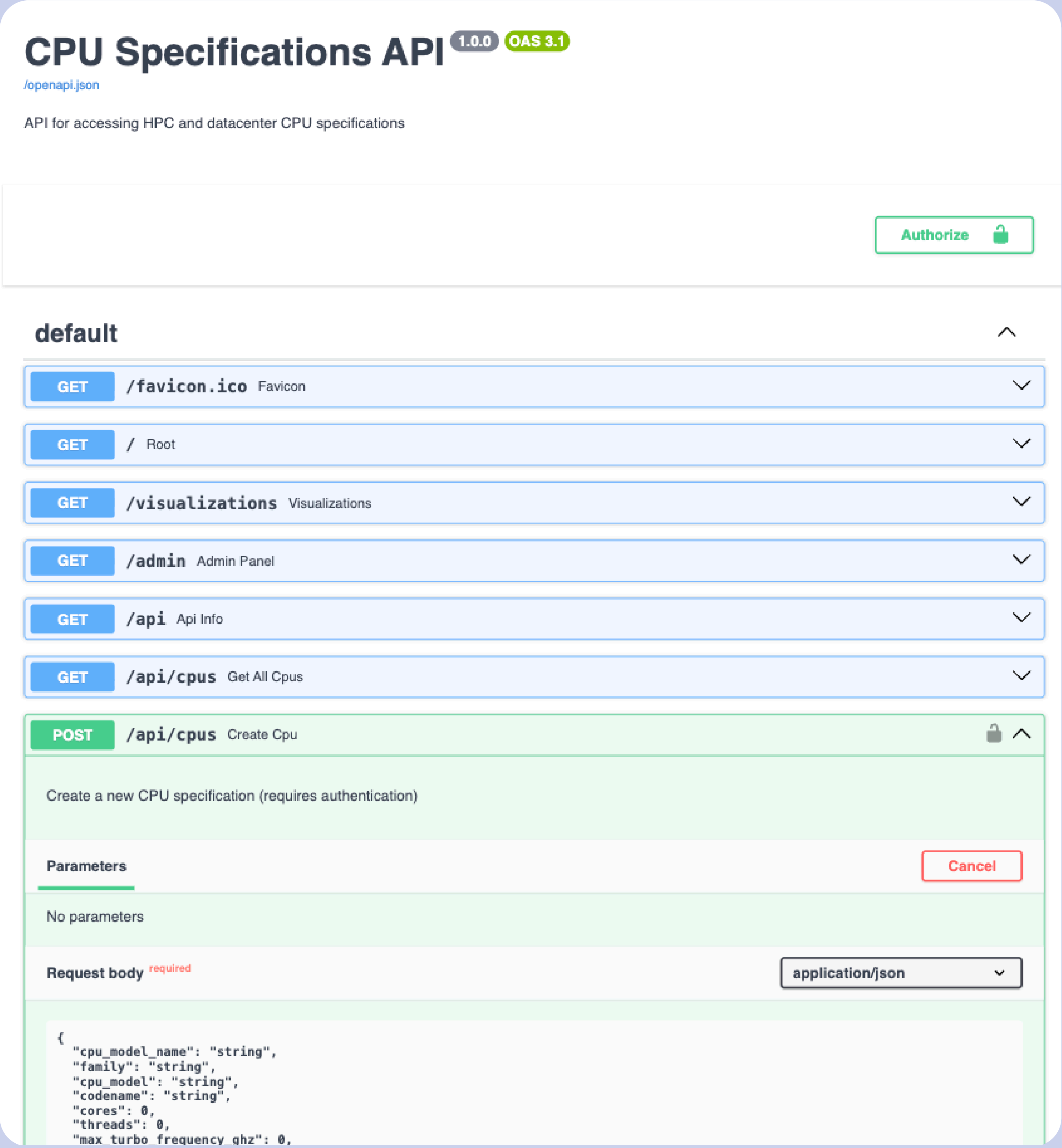
Step 3

Compare two CPUs. On the same Visualizations page, scroll to the CPU Comparison Tool. Select CPU 1 and CPU 2 from the dropdowns, click "Compare CPUs," then review the side-by-side table and difference column (with % where applicable). Use "Download Comparison" to export as CSV.

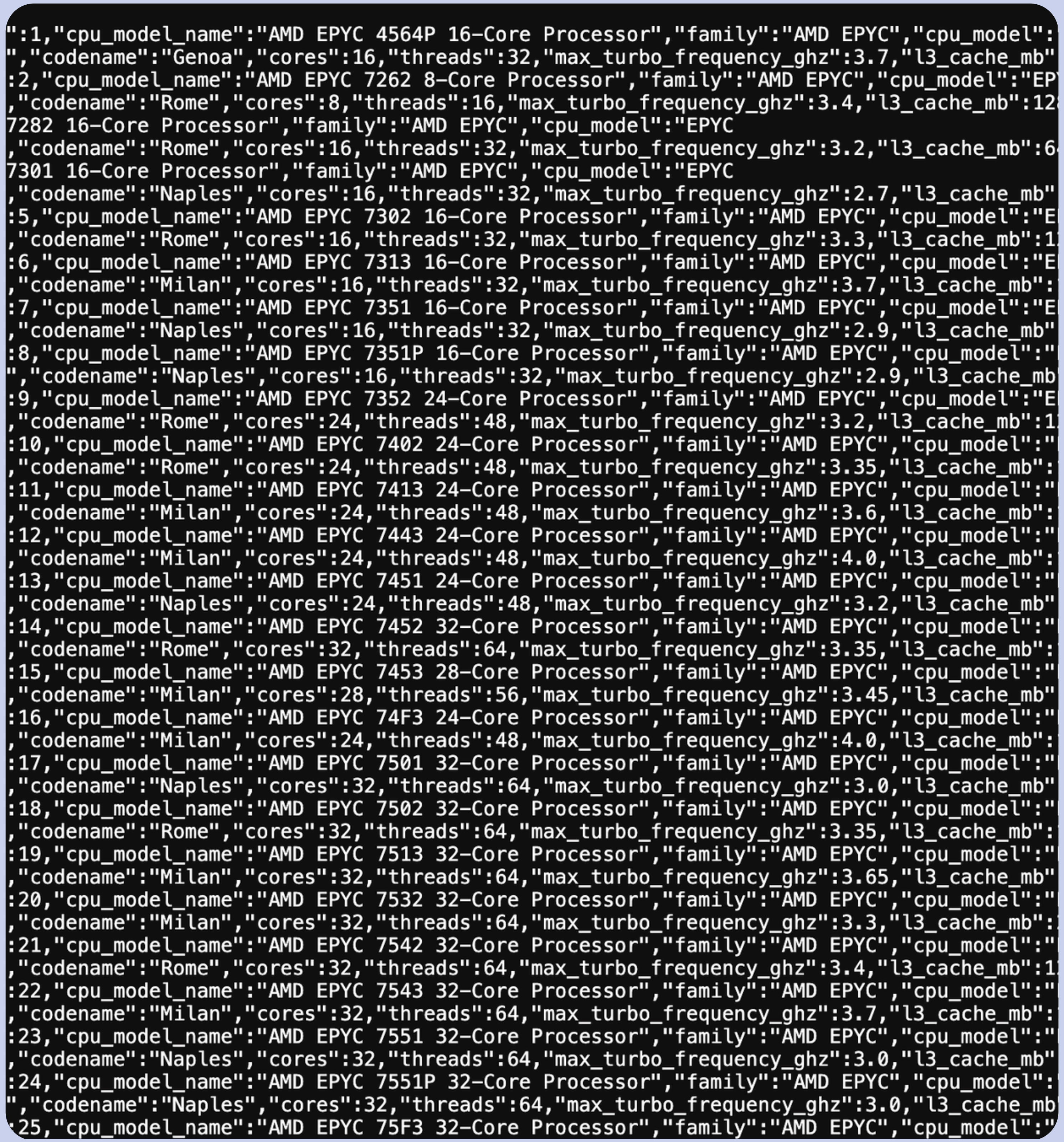
Step 4

Use the API or export the dataset.
Open "API Docs" to try endpoints (e.g. `GET /api/cpus/search?q=intel`) or use "API Info" for a quick reference. Export full data via `/api/export/csv` or `/api/export/excel` for offline analysis.

Step 4



CPU Specifications API
1.0.0
OAS 3.1
/openapi.json
API for accessing HPC and datacenter CPU specifications
Authorize
default
GET /favicon.ico Favicon
GET / Root
GET /visualizations Visualizations
GET /admin Admin Panel
GET /api Api Info
GET /api/cpus Get All Cpus
POST /api/cpus Create Cpu
Create a new CPU specification (requires authentication)
Parameters
Cancel
No parameters
Request body required
application/json
{
"cpu_model_name": "string",
"family": "string",
"cpu_model": "string",
"codename": "string",
"cores": 0,
"threads": 0,
"max_turbo_frequency_ghz": 0,



":1,"cpu_model_name":"AMD EPYC 4564P 16-Core Processor","family":"AMD EPYC","cpu_model":
","codename":"Genoa","cores":16,"threads":32,"max_turbo_frequency_ghz":3.7,"l3_cache_mb"
:2,"cpu_model_name":"AMD EPYC 7262 8-Core Processor","family":"AMD EPYC","cpu_model":"EP
,"codename":"Rome","cores":8,"threads":16,"max_turbo_frequency_ghz":3.4,"l3_cache_mb":12
7282 16-Core Processor","family":"AMD EPYC","cpu_model":"EPYC
,"codename":"Rome","cores":16,"threads":32,"max_turbo_frequency_ghz":3.2,"l3_cache_mb":6
7301 16-Core Processor","family":"AMD EPYC","cpu_model":"EPYC
,"codename":"Naples","cores":16,"threads":32,"max_turbo_frequency_ghz":2.7,"l3_cache_mb"
:5,"cpu_model_name":"AMD EPYC 7302 16-Core Processor","family":"AMD EPYC","cpu_model":"E
,"codename":"Rome","cores":16,"threads":32,"max_turbo_frequency_ghz":3.3,"l3_cache_mb":1
:6,"cpu_model_name":"AMD EPYC 7313 16-Core Processor","family":"AMD EPYC","cpu_model":"E
,"codename":"Milan","cores":16,"threads":32,"max_turbo_frequency_ghz":3.7,"l3_cache_mb":
:7,"cpu_model_name":"AMD EPYC 7351 16-Core Processor","family":"AMD EPYC","cpu_model":"E
,"codename":"Naples","cores":16,"threads":32,"max_turbo_frequency_ghz":2.9,"l3_cache_mb"
:8,"cpu_model_name":"AMD EPYC 7351P 16-Core Processor","family":"AMD EPYC","cpu_model":"
","codename":"Naples","cores":16,"threads":32,"max_turbo_frequency_ghz":2.9,"l3_cache_mb
:9,"cpu_model_name":"AMD EPYC 7352 24-Core Processor","family":"AMD EPYC","cpu_model":"E
,"codename":"Rome","cores":24,"threads":48,"max_turbo_frequency_ghz":3.2,"l3_cache_mb":1
:10,"cpu_model_name":"AMD EPYC 7402 24-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Rome","cores":24,"threads":48,"max_turbo_frequency_ghz":3.35,"l3_cache_mb":
:11,"cpu_model_name":"AMD EPYC 7413 24-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":24,"threads":48,"max_turbo_frequency_ghz":3.6,"l3_cache_mb":
:12,"cpu_model_name":"AMD EPYC 7443 24-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":24,"threads":48,"max_turbo_frequency_ghz":4.0,"l3_cache_mb":
:13,"cpu_model_name":"AMD EPYC 7451 24-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Naples","cores":24,"threads":48,"max_turbo_frequency_ghz":3.2,"l3_cache_mb"
:14,"cpu_model_name":"AMD EPYC 7452 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Rome","cores":32,"threads":64,"max_turbo_frequency_ghz":3.35,"l3_cache_mb":
:15,"cpu_model_name":"AMD EPYC 7453 28-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":28,"threads":56,"max_turbo_frequency_ghz":3.45,"l3_cache_mb"
:16,"cpu_model_name":"AMD EPYC 74F3 24-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":24,"threads":48,"max_turbo_frequency_ghz":4.0,"l3_cache_mb":
:17,"cpu_model_name":"AMD EPYC 7501 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Naples","cores":32,"threads":64,"max_turbo_frequency_ghz":3.0,"l3_cache_mb"
:18,"cpu_model_name":"AMD EPYC 7502 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Rome","cores":32,"threads":64,"max_turbo_frequency_ghz":3.35,"l3_cache_mb":
:19,"cpu_model_name":"AMD EPYC 7513 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":32,"threads":64,"max_turbo_frequency_ghz":3.65,"l3_cache_mb"
:20,"cpu_model_name":"AMD EPYC 7532 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":32,"threads":64,"max_turbo_frequency_ghz":3.3,"l3_cache_mb":
:21,"cpu_model_name":"AMD EPYC 7542 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Rome","cores":32,"threads":64,"max_turbo_frequency_ghz":3.4,"l3_cache_mb":1
:22,"cpu_model_name":"AMD EPYC 7543 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Milan","cores":32,"threads":64,"max_turbo_frequency_ghz":3.7,"l3_cache_mb":
:23,"cpu_model_name":"AMD EPYC 7551 32-Core Processor","family":"AMD EPYC","cpu_model":"
,"codename":"Naples","cores":32,"threads":64,"max_turbo_frequency_ghz":3.0,"l3_cache_mb"
:24,"cpu_model_name":"AMD EPYC 7551P 32-Core Processor","family":"AMD EPYC","cpu_model":
","codename":"Naples","cores":32,"threads":64,"max_turbo_frequency_ghz":3.0,"l3_cache_mb
:25,"cpu_model_name":"AMD EPYC 75F3 32-Core Processor","family":"AMD EPYC","cpu_model":"

@ndsz

Where Could This Go Next?

This project is a first working prototype, with room to grow:

- **Validation pipeline:** Introduce authorized reviewers and a structured review workflow so new entries can be verified against official sources and merged quickly, without bottlenecking on a single maintainer.
- **Broader coverage:** Expand beyond server CPUs to include GPUs, ARM processors, and additional vendors, while maintaining the same standard of data quality.
- **Community contributions with guardrails:** The GitHub issue templates are a starting point. Next steps could include a submission form on the website or a PR-based workflow, always gated behind moderator approval before anything enters the database.
- **Richer specifications:** Add fields that matter for real-world comparisons, like memory channels, socket type, process node, and pricing where available.
- **Persistent storage:** Move from ephemeral SQLite to a hosted database so that data persists across deployments and admin edits aren't lost on redeploy.

Summary

Every project in this series starts as an idea aka problem in my head. First, I handle the **planning, design, and choice of technologies** myself before starting building the first version. For me, turning an idea into a real product always begins with **thoughtful preparation and intentional decisions.**

This is prototype #2 of a planned ~20 for 2026. The target pace is one every two weeks, though some naturally take longer, this one took 3.5 weeks because of the time spent on manual data validation. The focus is on steady, quality output rather than hitting exact deadlines.

To accelerate my progress and improve my workflow, I support my ideas and design process with AI tools like CursorAI and ChatGPT.
I strongly believe this is how these technologies are meant to be used: **<u>not to replace learning and doing everything for us, but to help us learn more, move faster, and deliver better results.</u>**

Let's see what happens when we combine real intent,
thoughtful design, and the leverage of powerful tools. :))

Let's talk

If you're interested in more presentations like this one, practical experiments, honest lessons, and the useful value of AI in hands-on product work, follow me here for more projects and reflections every 2 weeks!

Check out the prototype here:
https://computespecsdb.com/

Check out the repository here:
https://github.com/elokwentnie/compute-specs-db

Questions for You:

- What stops you from turning a side project idea into something real?
- When you use AI to help you build, how do you make sure you're actually learning and not just copy-pasting?
- What's one dataset or tool you wish existed but doesn't?

@ndsz