



ML und KI im Agenturalltag

Inhalt

- Wer ist Apps with love?
- KI-Generierte **Inhalte** in Apps: Stefan Spieler
- KI im **Coding**-Alltag: Olivier Oswald
- Generelle KI-**Anwendung** in anderen Teams: Valentin Naegeli
- Fragen & Diskussion

Über Apps with love

Apps with love ist darauf spezialisiert, schöne und intuitive **digitale Produkte** zu entwickeln, die das tägliche Leben der Menschen vereinfachen.

Von der ersten **Idee** über **Beratung, Konzept** und **Design**, der **technischen Entwicklung** bis hin zur **Vermarktung** von digitalen Produkten arbeiten wir mit unseren Kund*innen zusammen.

Mit viel **Liebe zum Detail** feilen wir am Zusammenspiel von **Design, Usability** und **Technik**. Dabei stehen die Bedürfnisse der User*innen stets im Zentrum.

appswithlove.com

Facts & Figures

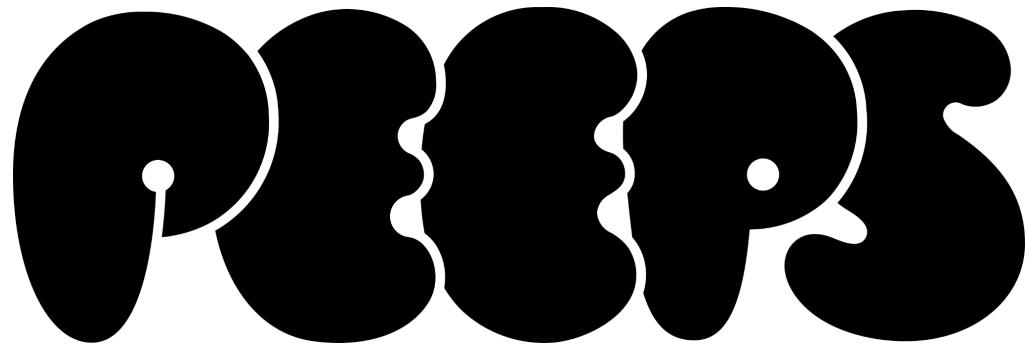
- Gegründet: 2010
- Mitarbeitende: knapp 50
- Nearshore & Freelance: 20+
- Aktive Projekte: 100+
- App und Web-Awards: 60+



Services & Angebot

- Beratung rund um alles Digitale
- User Research & Usability Testings
- Agiles Projektmanagement
- Userzentrierte Konzepte & Designs
- UX & Interaction Design
- Native Apps & Webapplikationen
- Backends, CMS & API Entwicklung
- Software Testing & Continuous Integration
- IoT Hard & Software Entwicklung
- Blockchain & Crypto Technology
- App Promotion & App Store Optimierung
- Software Support, Hosting & Betrieb

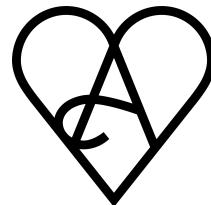


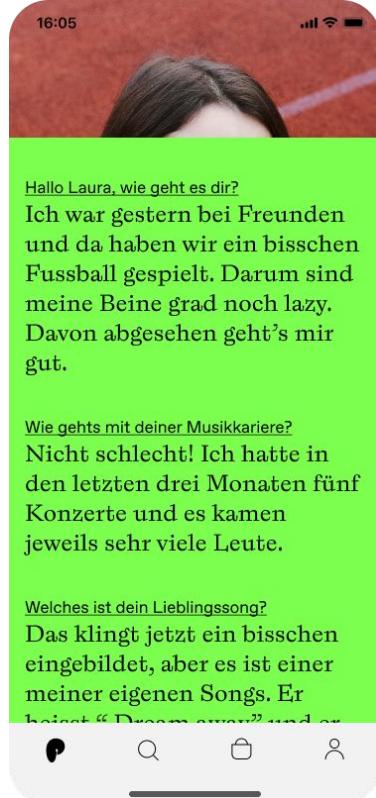
The word "Apps" is written in a bold, sans-serif font. Each letter is formed by a series of overlapping black circles, creating a layered effect. The letters are outlined in white, which provides a strong contrast against the black background.

KI-Generierte Inhalte in Apps



INTR





- eine porträtierte Person pro Tag inkl. Interview
- weitere Artikel zu den Interessen der Person
- ein Shop Produkt passend zur Person
- alternierend eine Verlosung passend zur Person

Viele

Musikerinnen Influencerinnen

Gamer Tänzer Künstler

Studentinnen Filme Macher

Rapper

Wenige

Zoowärter

Mütter

Jäger

Wirte

Flugbegleiter

Ritter

Schneider

Architektinnen

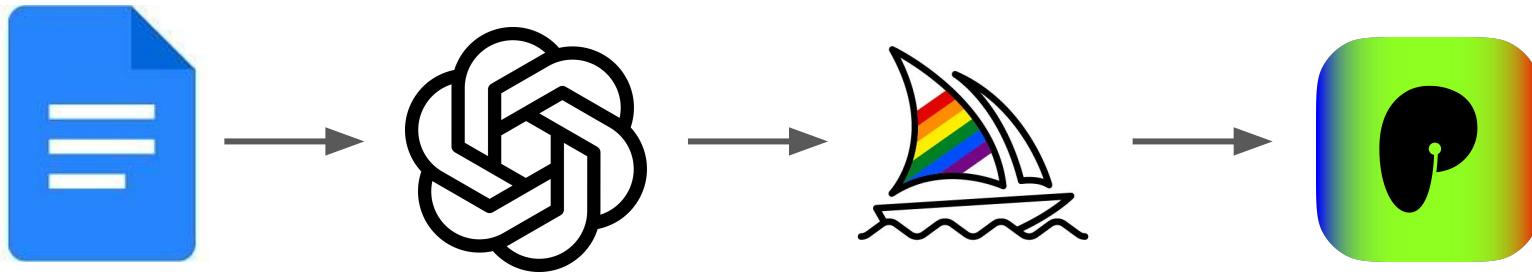
Astronautinnen

Römer

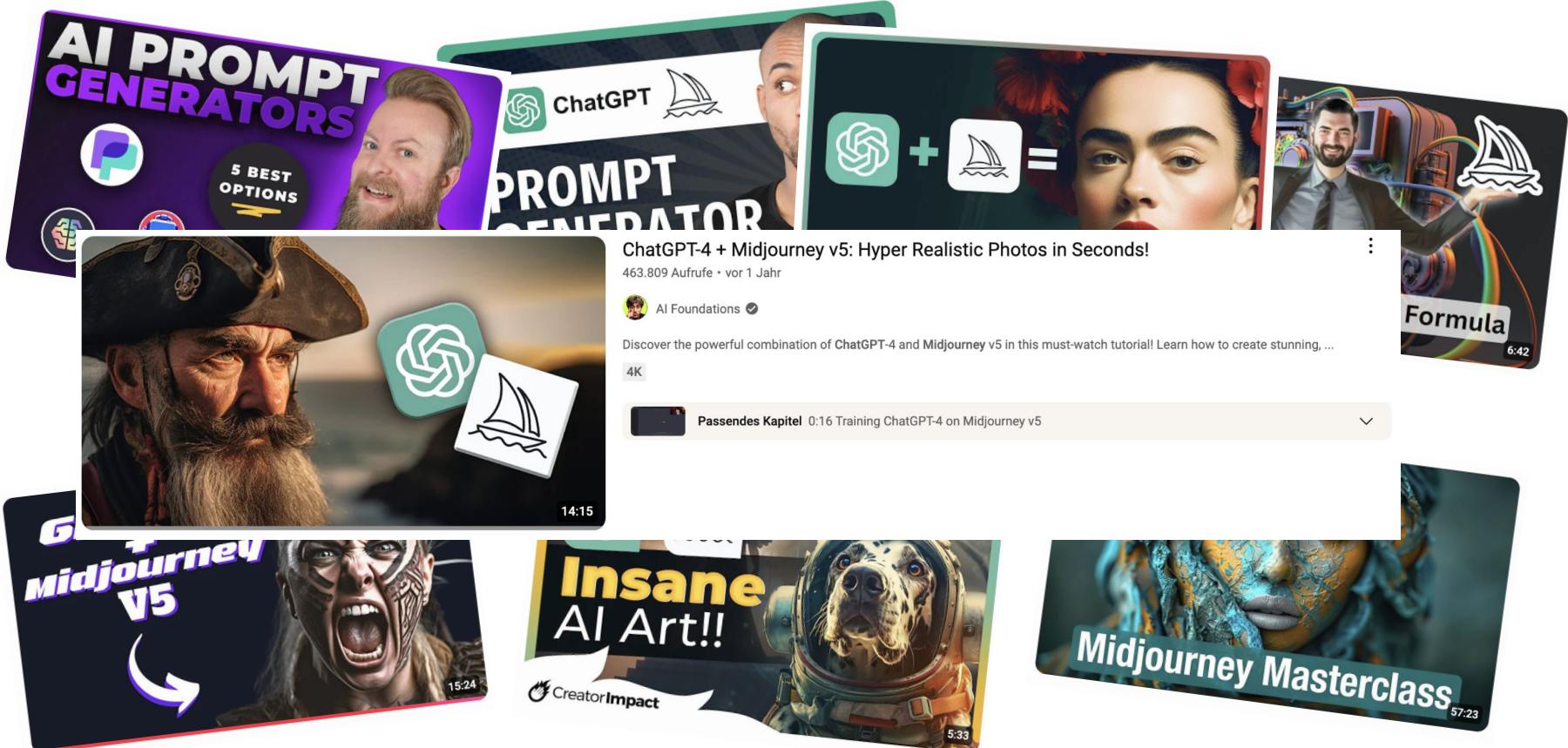
- 1. Personas erfinden**
- 2. Interview schreiben**
- 3. Portrait machen ???**



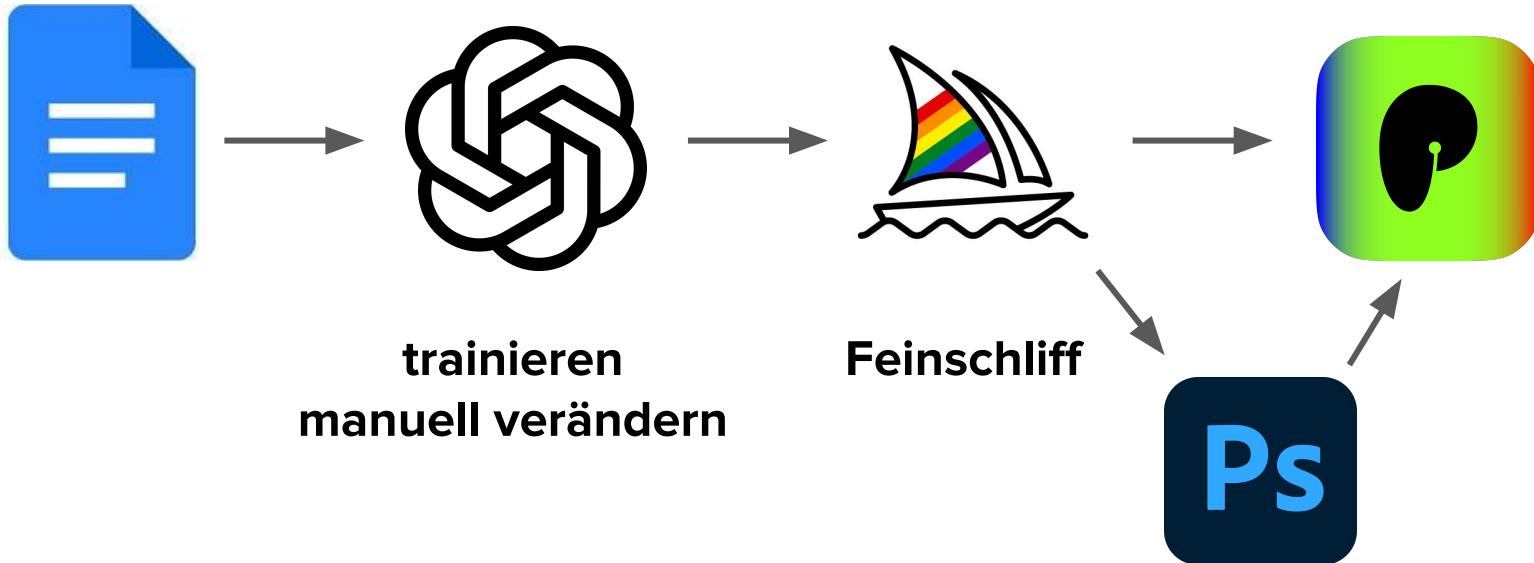
AI bedient AI



Chat GPT trainieren



Final Workflow



Resultate

TITLE

(Name, Age, Description, Domicil)

○ Margrit, 28, Flugbegleiterin, Winterthur im Jahr 1952

ALT TEXT FOR HERO PHOTO

(Image description for screen reader: What can be seen in the photo?)

XXX

INTERVIEW 3 QUESTIONS / 3 ANSWERS

650 - 1080 characters

(Wordcount: Shift+CMD+C)

Hi Margrit! Du bist Flugbegleiterin bei der [Swissair](#). Was sind deine Aufgaben an Bord?

Bevor das Flugzeug abhebt, helfe ich den Passagierinnen beim Verstauen ihres Gepäcks. Jenen, die Flugangst haben, widme ich besondere Aufmerksamkeit. Während normale Menschen ihrem Alltag dort unten auf der Erde nachgehen, serviere ich Hühnchen-Sandwiches mit Fruchtsalat mehrere tausend Meter über dem Meer – das ist unglaublich!

Was ist dein Geheimtipp gegen Flugangst?

Da hilft nur eines: Jodeln! Da verschwinden die Ängste im Nu. Das hat schon mein Idol [Nelly Diener](#) so gemacht: Sie war die erste Flugbegleiterin Europas. Bei allfälligen Turbulenzen, so wie letztens auf einem Flug nach New York, serviere ich dann auch ab und zu ein paar zusätzliche Zigaretten oder Schoggi-Pralinen. (lacht)

Gabs denn schon mal einen Zwischenfall?

Mit mir an Bord zum Glück noch nicht. Aber eine meiner Sky-Girl-Kolleginnen war bei der Bruchlandung mit der [Douglas DC-4](#) letztes Jahr in der Nähe von Amsterdam dabei. Gott sei Dank hat sich dabei niemand ernsthaft verletzt.



TITLE

(Name, Age, Description, Domicil)

◊ Nikki Alves, 22, Interstellar-Reisende:r, Gstaad 2.0 im Jahr 2794

ALT TEXT FOR HERO PHOTO

(Image description for screen reader: What can be seen in the photo?)

XXX

INTERVIEW 3 QUESTIONS / 3 ANSWERS

650 - 1080 characters

(Wordcount: Shift+CMD+C)

Hey Nikki, du wohnst gar nicht mehr auf der Erde, oder?

Nein, auf der Erde wohnen ist voll out (*lacht*). Meine Familie und Friends leben alle auf dem Planeten **Kepler-22b**. Hier ist es einfach besser! Wir haben Gstaad 2.0 gegründet, weil meine Vorfahr:innen aus Gstaad on Earth kamen.

Alles klar! Fühlst du dich denn stark mit deinen Vorfahr:innen auf der Erde verbunden?

Ja, mega! Meine Familie hat eine grosse Appreciation für sie. Sie waren im Forschungsteam, das das Raumschiff *Escape_Earth_3000* gebaut hat. Dank ihnen konnte die Menschheit die Erde überhaupt verlassen.

Wow, crazy! Und du führst ihre Tradition jetzt fort?

Genau, ich mache gerade eine Ausbildung für Interstellar-Travelling. Vorher hab ich das eher so casual gemacht und bin wie alle anderen am Wochenende oder nach der Schule ein paar Stunden rumgeflogen – zum Spass halt. Aber seit ich die Ausbildung begonnen habe, träume ich davon, mehr bewohnbare Planeten zu entdecken. Auf Kepler-22b wird's nämlich schon etwas eng.

Dieses Timetraveler-Interview wurde basierend auf [dieser Analyse](#) der NASA verfasst.



TITLE

(Name, Age, Description, Domicile)

◊ Aya, 22, Architektin, Biel im Jahr 1973

ALT TEXT FOR HERO PHOTO

(Image description for screen reader: What can be seen in the photo?)

XXX

INTERVIEW 3 QUESTIONS / 3 ANSWERS

650 - 1080 characters

(Wordcount: Shift+CMD+C)

Hey Aya! Wo steht dein Lieblingsgebäude?

Back in Tokyo bin ich jeden Morgen am [Nakagin Capsule Tower](#) vorbei ins Büro spaziert. Er besteht aus 140 aufeinander gestapelten, kleinen, quadratischen Wohnungen. Das Schönste daran finde ich die riesigen runden Fenster ins Weite!

Was hat es mit diesen runden Fenster auf sich?

Mich erinnern sie an ein Aquarium (*lacht*)! Aber der Architekt meinte in einem Interview, sie seien von den traditionellen Teehäusern Japans inspiriert. Die versteckten symbolischen Details sind ein Hinweis auf die Liebe zur Architektur. Gerade arbeite ich an einem Projekt hier in Biel, bei dem wir ähnliche Fenster verwenden wollen.

Was hat dich denn von Tokyo nach Biel verschlagen?

Für einmal war es nicht die Liebe zur Architektur, sondern meine Liebe zu Theo. (*lacht*) Er ist ebenfalls Architekt. Wir haben uns an einer Weiterbildung in Tokyo kennengelernt und lassen nun unsere gemeinsame Faszination für die beinahe utopischen Ideen der Metabolist:innen in die Schweizer Architektur einfließen. Ein Ausdruck unserer verrückten Liebe sozusagen. (*lacht*)



TITLE

(Name, Age, Description, Domicile)

◊ Jacques, 28, Stadtwächter, Genf im Jahr 1602

ALT TEXT FOR HERO PHOTO

(Image description for screen reader: What can be seen in the photo?)

XXX

INTERVIEW 3 QUESTIONS / 3 ANSWERS

650 - 1080 characters

(Wordcount: Shift+CMD+C)

Hallo Jacques! Wie fühlst du dich nach der Schlacht?

Auf jeden Fall etwas benommen! Aber hauptsächlich erleichtert. Alle haben geholfen, die Stadt zu verteidigen, sogar die Bürger:innen. Ich bin glücklich, dass meine Familie gesund und glücklich ist. Zur Feier unseres Sieges singen wir gleich einen Psalm vor der Kathedrale Saint-Pierre.

Wie habt ihr von dem Angriff auf die Stadt erfahren?

Savoyische Soldaten haben mitten in der Nacht die Stadtmauern erklimmen. Wir haben erst Alarm geschlagen, als sie oben angekommen sind. Zum Glück hatte ein Soldat schon das Fallgitter ausgelöst; so konnte verhindert werden, dass die Gegner durch die Stadttore eindringen konnten. Wer auch immer das war, ist ein:e Held:in!

Wie siehst du die Zukunft von Genf?

Dieser Sieg zeigt, dass unsere Stadt trotz ihrer kleinen Grösse mit dem Herzogtum Savoyen mithalten kann. Die Nacht vom 11. auf den 12. Dezember 1602 bleibt für immer ins Genfer Gedächtnis eingebrannt, da bin ich mir sicher.



TITLE

(Name, Age, Description, Domicil)

◊ Lucia Cornelia, 23, Hausfrau, Aventicum, 153
n.Chr.

ALT TEXT FOR HERO PHOTO

(Image description for screen reader: What can be seen in the photo?)

XXX

INTERVIEW 3 QUESTIONS / 3 ANSWERS

650 - 1080 characters

(Wordcount: Shift+CMD+C)

Hi Lucia. Wie sieht dein Alltag aus?

Vor Kurzem habe ich Gaius Julius Cornelius geheiratet, und zusammen sind wir in ein Haus nicht weit von meinen Eltern gezogen. Wir wünschen uns viele Kinder. Damit das klappt, habe ich eine Statue der Muttergöttin [Dea Nutrix](#) erworben. Momentan schaue ich nach unserem neuen Heim.

Hat sich Aventicum in den letzten Jahren stark verändert?

Ja. Ich selbst kann mich nicht an die Zeit erinnern, als Aventicum noch nicht Hauptstadt des römischen Helvetiens war, aber meine Grosseltern haben mir davon berichtet. Sie mussten damals römische Bräuche wie das Drei-Namen-Prinzip und die Anbetung von römischen Göttern und Göttinnen annehmen.

Was machst du in deiner freien Zeit?

Ich gehe gerne mit meinem Mann ins Amphitheater. So oft schaffen wir es nicht dorthin, aber letzte Woche haben wir eine [Venatio](#) gesehen. Das ist eine Veranstaltung, in der Tiere gegeneinander kämpfen: Diesmal waren es Luchse, Wildschweine, Hirsche... Es herrschte aufgeregte Stimmung; die ganze Stadt hat sich auf der Tribüne versammelt.



pops

Software-Development mit KI

iOS, Android, Web, .Net, DevOps

Welche KI nutzen unsere Developer?

ChatGPT:

- Weniger gut integriert in IDEs
- Eher copy/paste, aber...
- Vielseitiger Einsatz über Coden hinaus

Copilot:

- Sehr gute IDE Integration
- Gute Qualität
- Echtzeit-Vervollständigung, auch ganzer Codeblöcke

Claude:

- Super für Code-Generierung
- Gute UX mit File-Handling

Perplexity:

- Fragen statt Suchen
- Super für Research, wie Alternativen zu Produkt X
- Vorteil: ist aktuell
- hat Quellenangaben

Engineering Use Cases

Es gibt unzählige Use Cases, wo wir KI im Development einsetzen:

- Dokumentation verstehen, ausfragen
- Dokumentation und Texte schreiben
- Lernen von Programmiersprachen, Tools und Technologien
- Code schreiben, obviously.

Data Model Generation

"generate entities for **departments** and **people**, with view models and mapping code."

I'll create TypeScript models for both Person and Department, including POCO (entity) and ViewModel versions with mapping code.

</> **Department Models**
Click to open code

</> **Person Models**
Click to open code

I've created two TypeScript files with the following key features:

1. **department-models.ts** :
 - **Department** interface (POCO)
 - **DepartmentViewModel** interface
 - **mapDepartmentToViewModel()** mapping function
2. **person-models.ts** :
 - **Person** interface (POCO)

← Department Models X

```
// Department POCO (Entity Model)
export interface Department {
  department_id: number;
  parent_id?: number;
  name?: string;
}

// Department ViewModel
export interface DepartmentViewModel {
  id: number;
  parentId?: number;
  name?: string;
}

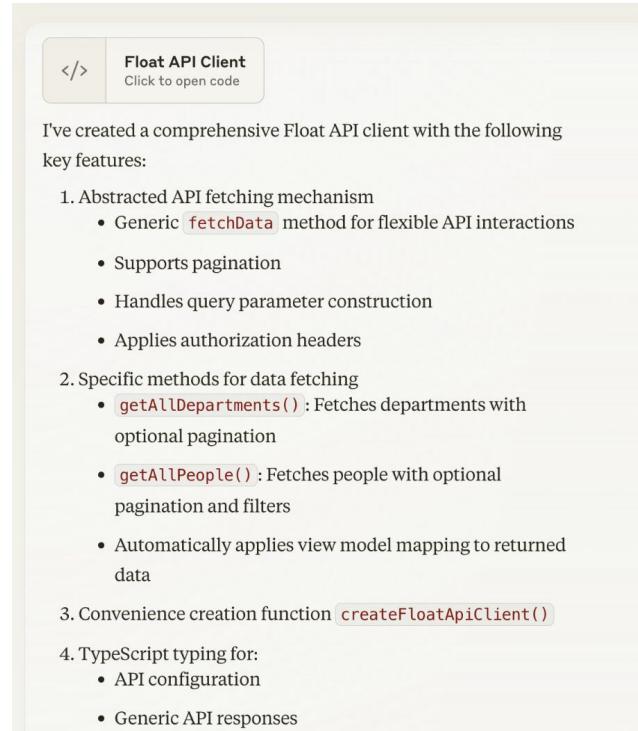
// Mapping function from POCO to ViewModel
export function mapDepartmentToViewModel(department: Department): DepartmentViewModel {
  return {
    id: department.department_id,
    name: department.name,
    parentId: department.parent_id
  }
}
```

Use Case: API Client Generation

"create an API client
for Float, with two
functions to **get**
departments and
people."

...

"handle pagination
and throttling and
error handling"



Float API Client
Click to open code

I've created a comprehensive Float API client with the following key features:

1. Abstracted API fetching mechanism
 - Generic `fetchData` method for flexible API interactions
 - Supports pagination
 - Handles query parameter construction
 - Applies authorization headers
2. Specific methods for data fetching
 - `getAllDepartments()` : Fetches departments with optional pagination
 - `getAllPeople()` : Fetches people with optional pagination and filters
 - Automatically applies view model mapping to returned data
3. Convenience creation function `createFloatApiClient()`
4. TypeScript typing for:
 - API configuration
 - Generic API responses

```
←  Float API Client

    data: result.data.map(mapDepartmentToViewModel)
};

/**
 * Fetch all people
 * @param page Page number for pagination
 * @param perPage Number of items per page
 * @param filters Optional filters to apply
 */
async getAllPeople(
  page = 1,
  perPage = 100,
  filters: Record<string, string | number> = {}
): Promise<ApiResponse<Person>> {
  const result = await this.fetchData<Person>('people', {
    page,
    per_page: perPage,
    ...filters
  });

  return {
    ...result,
    data: result.data.map(mapPersonToViewModel)
};
\
```

Use Case: Mocking Functionality

"create a mock of the API client."

- Uses view model mapping

Key Capabilities:

- Simulate network latency
- Filter departments and people
- Provide consistent, predictable mock data
- Easy to extend or modify mock data

Usage Example:

typescript

Copy

```
const mockClient = createMockFloatApiClient();

// Fetch all departments
const departments = await mockClient.getAllDepartments();

// Filter people by department
const devOpsPeople = await mockClient.getAllPeople(1, 100, {
  department: mockDepartments[1]
});
```

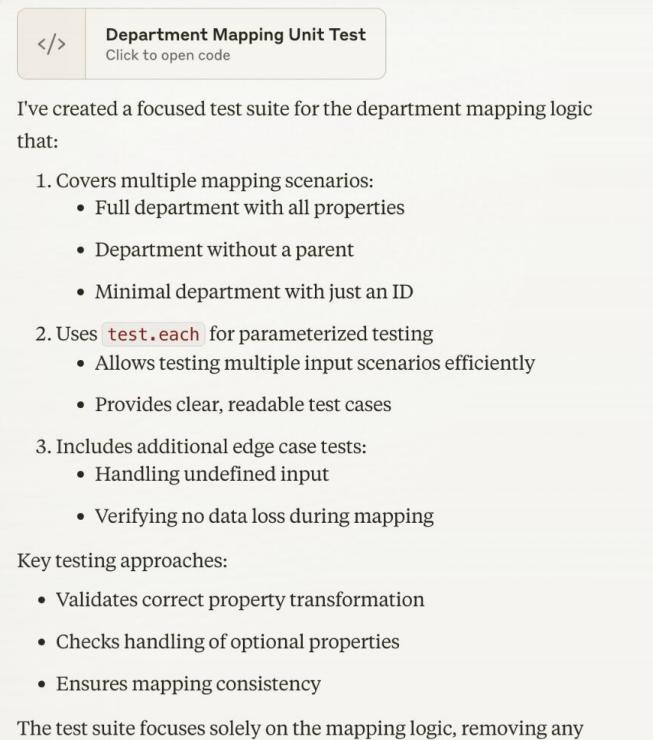
← Mock Float API Client

```
{
  department_id: 8,
  name: 'HR',
  parent_id: null
}
];

/** 
 * Mock data for people
 */
const mockPeople: Person[] = [
{
  people_id: 1,
  name: 'John Doe',
  email: 'john.doe@company.com',
  job_title: 'Senior Software Engineer',
  role_id: 1,
  department: mockDepartments[1], // Dev
  active: 1,
  people_type_id: 1,
  ...
```

Use Case: Generating Test Cases

"write tests to verify
the mapping of the
models you provided"



The screenshot shows a browser window with a light gray background. At the top left is a button labeled '</>'. To its right is a title bar with the text 'Department Mapping Unit Test' and a link 'Click to open code'. Below the title bar is a text area containing the following content:

I've created a focused test suite for the department mapping logic that:

1. Covers multiple mapping scenarios:
 - Full department with all properties
 - Department without a parent
 - Minimal department with just an ID
2. Uses `test.each` for parameterized testing
 - Allows testing multiple input scenarios efficiently
 - Provides clear, readable test cases
3. Includes additional edge case tests:
 - Handling undefined input
 - Verifying no data loss during mapping

Key testing approaches:

- Validates correct property transformation
- Checks handling of optional properties
- Ensures mapping consistency

The test suite focuses solely on the mapping logic, removing any

```
← Department Mapping Unit Test

    name: undefined,
    parentId: undefined
  }
}
];

// Mapping test for each scenario
test.each(testCases)(
  'should correctly map department with different',
  ({ input, expectedOutput }) => {
    const viewModel = mapDepartmentToViewModel(
      input
    );

    // Validate mapped properties
    expect(viewModel).toEqual(expect.objectContaining({
      id: expectedOutput.id,
      name: expectedOutput.name,
      parentId: expectedOutput.parentId
    }));
  }
);

// Additional specific tests for edge cases
it('should handle undefined input', () => {
  // @ts-ignore - intentionally passing undefined
  expect(() => mapDepartmentToViewModel(undefined)).not.toThrow();
});
```

Generating Views and Components

"create a responsive component that lists all departments sorted by name"

</>

Departments List Svelte Component
Click to open code

I've created a comprehensive Svelte component for listing departments with the following features:

1. Responsive Design
 - Uses Tailwind's responsive grid (`grid-cols-1 md:grid-cols-2 lg:grid-cols-3`)
 - Adapts layout for mobile and desktop
 - Consistent padding and spacing
2. Data Fetching
 - Uses `onMount` to fetch departments
 - Integrates with the previously created Float API client
 - Uses environment variable for API key
3. State Management
 - Loading state with spinner
 - Error handling
 - Empty state handling
4. Department Display
 - Shows department ID and name

```
← Departments List Svelte Component

    font-bold
  ">
  {department.id}

```

```
  <!-- Department Name -->
  <div class="flex-grow">
    <h2 class="text-lg font-semibold text-gray-800">
      {department.name || 'Unnamed Department'}
    </h2>
  </div>
</div>
</each>
</div>
{/if}
```

```
<!-- Empty State -->
{#if !isLoading && departments.length === 0}
  <div class="text-center text-gray-500 py-12">
    <p class="text-xl">No departments found</p>
  </div>
{/if}
</div>
```

Refactoring

"never ever access float directly from the frontend!"

You're absolutely right. In a production environment, API calls to external services should always be made through a backend API route to:

- Keep API keys secure
- Prevent exposing credentials
- Add server-side validation and rate limiting
- Manage authentication and authorization

I'll modify the component to work with a SvelteKit backend route that will proxy the Float API request.

</>

Departments List Svelte Component with Backend Proxy

Click to open code

</>

SvelteKit Backend API Route for Departments

Click to open code

← Departments List Svelte Component

```
<script lang="ts">
    import { onMount } from 'svelte';
    import type { DepartmentViewModel } from '$lib/models/dep
    // Type for the API response from our backend
    type ApiResponse = {
        data: DepartmentViewModel[];
        error?: string;
    };

    let departments: DepartmentViewModel[] = [];
    let isLoading = true;
    let error: string | null = null;

    // Fetch departments via our backend API route
    onMount(async () => {
        try {
            const response = await fetch('https://api.example.com/departments');
            if (!response.ok) {
                error = `Error fetching departments: ${response.statusText}`;
                return;
            }
            const data = await response.json();
            departments = data;
            isLoading = false;
        } catch (error) {
            error = `Error fetching departments: ${error.message}`;
        }
    });
}
```

Math the easy way

"Write the web mercator projection formula in swift so that I can use it to convert a GPS coordinate"

statt Stack Overflow

swift

 Code kopieren

```
import Foundation

struct WebMercatorProjection {
    // Earth radius in meters for Web Mercator projection (sphere approximation)
    static let radius: Double = 6378137.0

    // Converts latitude and longitude to Web Mercator x, y coordinates
    static func toWebMercator(latitude: Double, longitude: Double) -> (x: Double, y: Double) {
        let x = radius * longitude * .pi / 180.0
        let y = radius * log(tan(.pi / 4.0 + (latitude * .pi / 180.0) / 2.0))
        return (x, y)
    }
}

// Example usage
let latitude = 40.7128      // Example: Latitude for New York City
let longitude = -74.0060     // Example: Longitude for New York City
let mercatorCoords = WebMercatorProjection.toWebMercator(latitude: latitude, longitude: longitude)
```

Getting help in Python with DataFrames

"I only want rows from **float_people_workhours** where there is no matching row in **full_day_timeoffs**"



To fix the issue, we need to adjust the condition in the `.query()` method to correctly identify rows where there is no match. In Pandas, NaN values should be checked using `.isnull()` rather than a direct comparison with NaN. Here's how to update the code:

```
python
Copy

# Calculate `full_day_timeoffs` as before
full_day_timeoffs = (
    float_people_timeoffs
    .query('days >= 1')
    [['people_id', 'calendar_day', 'hours']]
    .groupby(['people_id', 'calendar_day'])
    .agg({'hours': 'sum'})
    .rename(columns={'hours': 'timeoff_hours'})
)

# Merge and filter for rows where there is no match in `full_day_timeoffs`
df = (
    float_people_workhours
    .merge(full_day_timeoffs, on=['people_id', 'calendar_day'], how='left', indicator=True)
    .query('_merge == "left_only"')
    .drop(columns='_merge')
)
```

Properly represent that Gitlab Project

"generate a gitlab profile
picture for my data
warehousing project."

...

"192x192 pixels please"



Here's a profile picture designed for your GitLab data warehousing project. It has a clean, tech-focused style with data visualization elements, fitting well for a professional environment. Let me know if you need any further adjustments!

Naming things

"name a function that fetches projects by id but also filters them by active projects"



A clear name for such a function could be `fetchActiveProjectById` or `getActiveProjectById`. This suggests it retrieves projects by ID with the additional constraint that they are active.

Alternatively, you could consider:

- `fetchProjectByIdIfActive`
- `getActiveProjectsById`
- `retrieveActiveProjectById`

Each option clearly implies the function's purpose and condition. Let me know if you'd like further refinements!

Decrypt Error Messages

"tell me what's wrong"

```
ings, AssetKeys, AssetsDefinitions, or SourceAssets. Target: (UnresolvedAssetJobDefinition(name='google_storage_import_pdf_job', selection=KeysAssetSelection(selected_keys=[AssetKey(['ubs_creditcard_scans_in_bucket'])]), config=None, description=None, tags={}, run_tags={}, metadata={}, partitions_def=None, executor_def=None, hooks=None, op_retry_policy=None),)

Stack Trace:
File "/Users/ooswald/Sources/awl/data-warehouse/dw-dagster/.venv/lib/python3.11/site-packages/dagster/_core/errors.py", line 287, in user_code_error_boundary
    yield
File "/Users/ooswald/Sources/awl/data-warehouse/dw-dagster/.venv/lib/python3.11/site-packages/dagster/_grpc/server.py", line 255, in __init__
    loadable_targets = get_loadable_targets(
                        ^^^^^^^^^^
File "/Users/ooswald/Sources/awl/data-warehouse/dw-dagster/.venv/lib/python3.11/site-packages/dagster/_grpc/utils.py", line 50, in get_loadable_targets
    else loadable_targets_from_python_module(module_name, working_directory)
                        ^^^^^^^^^^
File "/Users/ooswald/Sources/awl/data-warehouse/dw-dagster/.venv/lib/python3.11/site-packages/dagster/_core/workspace/autodiscovery.py", line 31, in loadable_targets_from_python_module
    module = load_python_module(
                        ^^^^^^
File "/Users/ooswald/Sources/awl/data-warehouse/dw-dagster/.venv/lib/python3.11/site-packages/dagster/_core/code_pointer.py", line 134, in load_python_module
    return importlib.import_module(module_name)
                        ^^^^^^
File "/Library/Frameworks/Python.framework/Versions/3.11/lib/python3.11/importlib/_init_.py", line 126, in import_module
```

Likely Cause

In the `ubs_creditcard_sensor.py` file, where the sensor is defined, the decorator `@sensor` appears to have an incorrect target parameter. The target likely should be specified as one or more valid asset keys or asset definitions, but it instead contains a job definition (`UnresolvedAssetJobDefinition`).

Understanding existing code

"explain that UI component"

2. Component State

- **data:** Passed as a prop, representing time-off records in a yearly format.
- **loading:** Indicates whether data is still loading.
- **error:** Holds any error messages if data fetching fails.
- **availableYears:** An array of years available in the data, allowing users to select a year.
- **selectedYear:** Defaults to the current year, and users can choose a different year.

Landscape View (Desktop)

- Presents a table showing monthly time-off hours with a "Total" column for each entry.
- Hovering over a month cell displays a tooltip with detailed time-off information.

Mobile View

- Uses a card layout for each month's data, displaying hours and tooltips.
- Each card shows monthly hours, detailed tooltips, and a "Total" for the year.

Quick Translations

"translate to schweizerdeutsch"



Here's the JSON converted to Schweizerdeutsch:

```
json

{
  "departments": {
    "title": "Firmenabteilige",
    "unnamed": "Ohni Name Abteilig",
    "empty": "Kei Abteilige gfunde"
  },
  "errors": {
    "fetch": "Abteilige chönd nid abgolt werde: {{message}}"
  },
  "language": {
    "en": "Englisch",
    "es": "Schpanisch",
    "fr": "Französisch"
  }
}
```

Summarize the Changelog

"outline all **breaking changes** between 1.7.8 and 1.9.0, see here for the changelog:
<https://docs.dagster.io/changelog>"



Between versions 1.7.8 and 1.9.0 of Dagster, several breaking changes were introduced:

Version 1.8.0:

- **AutoMaterializeSensorDefinition Renamed:** AutoMaterializeSensorDefinition has been renamed to AutomationConditionSensorDefinition.
- **Removal of Deprecated Features:**
 - The experimental and deprecated "Op job versioning and memoization" feature has been removed.
 - The MemoizableIOManager, VersionStrategy, SourceHashVersionStrategy, OpVersionContext, ResourceVersionContext, and MEMOIZED_RUN_TAG have been removed.
 - The experimental and deprecated build_asset_with_blocking_check has been removed; use the blocking argument on @asset_check instead.

Fix breaking changes

"now find those breaking changes in my code, update my solution with dagster v1.9.0, run tests and commit that while I grab a coffee thanks."

...coming soon

Developer Learnings

Learnings

KI unterstützt sehr gut bei Fleissarbeiten, Mocks, Boilerplate, uvm.

KI spart Zeit. Meistens. Also manchmal. Es kommt drauf an.

Auch mal die offizielle Doku konsultieren, statt sich im Kreis zu drehen.

KI hat Stackoverflow ersetzt.

Code muss überprüft werden, funktioniert nicht immer gut, ist outdated.

Knowledge cut-off.

Kontext ist wichtig.

Developer Kontext

I am a senior **PHP/Laravel** developer, a senior software **architect**, and a senior software **developer**. I always use **TypeScript** for JavaScript projects.

Prefers **vim** over nano on the command line. Changes to the **root** user instead of prefixing commands with sudo.

Prefers using **Docker** containers over installing tools on their MacBook.

Follows **Ansible** lint recommendations in their Ansible configurations.

Do not explain unless instructed, **assume I know the context already**.

When writing text in german, never use the character ß. NEVER!!!

Use Cases in anderen Teams

CRM

The screenshot shows a CRM application interface with the following details:

Contact Information:

- Nexplore AG** (marked with a purple star)
- Kundenliste Postversand Partner*in**
- C.F.L. Lohnerstrasse 24, Gwatt (Thun), 3645**
- Account Manager Stephan Klaus**
- No information recorded** (button to "Add background info")
- Owner & Team**
- Valentin Naegeli** (owner)
- People at Nexplore AG**

Activity Stream: A large central area displays a blurred timeline of activities.

Right-hand Side Panel:

- Tasks:** No tasks to complete. (button to "Add Task")
- Latest Comment:** No comments for this contact. To add a comment, click the Add Comment icon on any logged activity.
- Enrichment:** Sub Industry Internet Software & Services, Employee Range 51-250, Estimated Annual Revenue \$10M-\$50M. (button to "View more")

Top Bar: Includes icons for Home, Person, Calendar, Signal, etc., a search bar ("Type / for search and recent items..."), an "Add +" button, and the user profile "Valentin Naegeli Apps with love".

Texting

Anderes Wort für "gesund" im Zusammenhang mit betriebswirtschaftlichen Zahlen wie "gesunde Eigenkapitalquote" oder "gesunder Kapazitätsauslastungsgrad"



Ein anderes Wort für "gesund" in diesem betriebswirtschaftlichen Kontext könnte "solide" sein.

Dieses Wort vermittelt eine ähnliche Bedeutung von Stabilität und Zuverlässigkeit. Man könnte also beispielsweise von einer "soliden Eigenkapitalquote" oder einer "Kapazitätsauslastungsgrad" sprechen. Weitere Synonyme können je nachdem, welcher Aspekt besonders betont werden soll.



Natürlich! Hier sind einige kreative und wetterbezogene Anredenvorschläge für Ihren Newsletter:

1. Strahlende Sonnengrüße
2. Sommerliche Grüße
3. Wärmende Sonnenstrahlen
4. Frische Sommerbrisen
5. Sonnige Grüße aus der Sommerhitze
6. Hitzefreie Grüße

Anrede für einen Newsletter die kreativer ist als Hallo. Etwas in die Richtung sonnige sommer grüsse, kann Wetterbezogen sein



Formatting & Story-Writing

If there are bullets (i.e. -) in the original, please add new line for this.
also if there are acceptance criteria: in this case create a new line and
some space between the requirement/story and the acceptance
criteria



Here is the table updated as per your request, with new lines for bullets and acceptance criteria:

Category	Details
OS Compatibility	<p>As a PO, I want the application running on the latest two released OS versions, so that most users are able to install the application.</p> <p>Acceptance Criteria:</p> <ul style="list-style-type: none">- Application is compatible with the latest two released OS versions when publishing the application the first time.- The application is compatible with the latest API level when publishing the application the first time.- The code base is set up for the defined OS versions.
Device Compatibility	<p>As a PO, I want the application running on specific devices, so that the use of the specific device can be guaranteed.</p> <p>Acceptance Criteria:</p> <ul style="list-style-type: none">- Application is running on [Zebra Tablet XYZ].- Application is running on [iPhone 13 mini].

Fazit / Fragen / Diskussion

- (generative) KI hat sich in allen Bereichen etabliert
- Integrationen in Apps & Websites und “echte” Use-Cases sind (noch?) rar

Wie erlebt ihr das?