From PoCs to Large-Scale ML Operationalization

Covering the End-to-End Pipeline

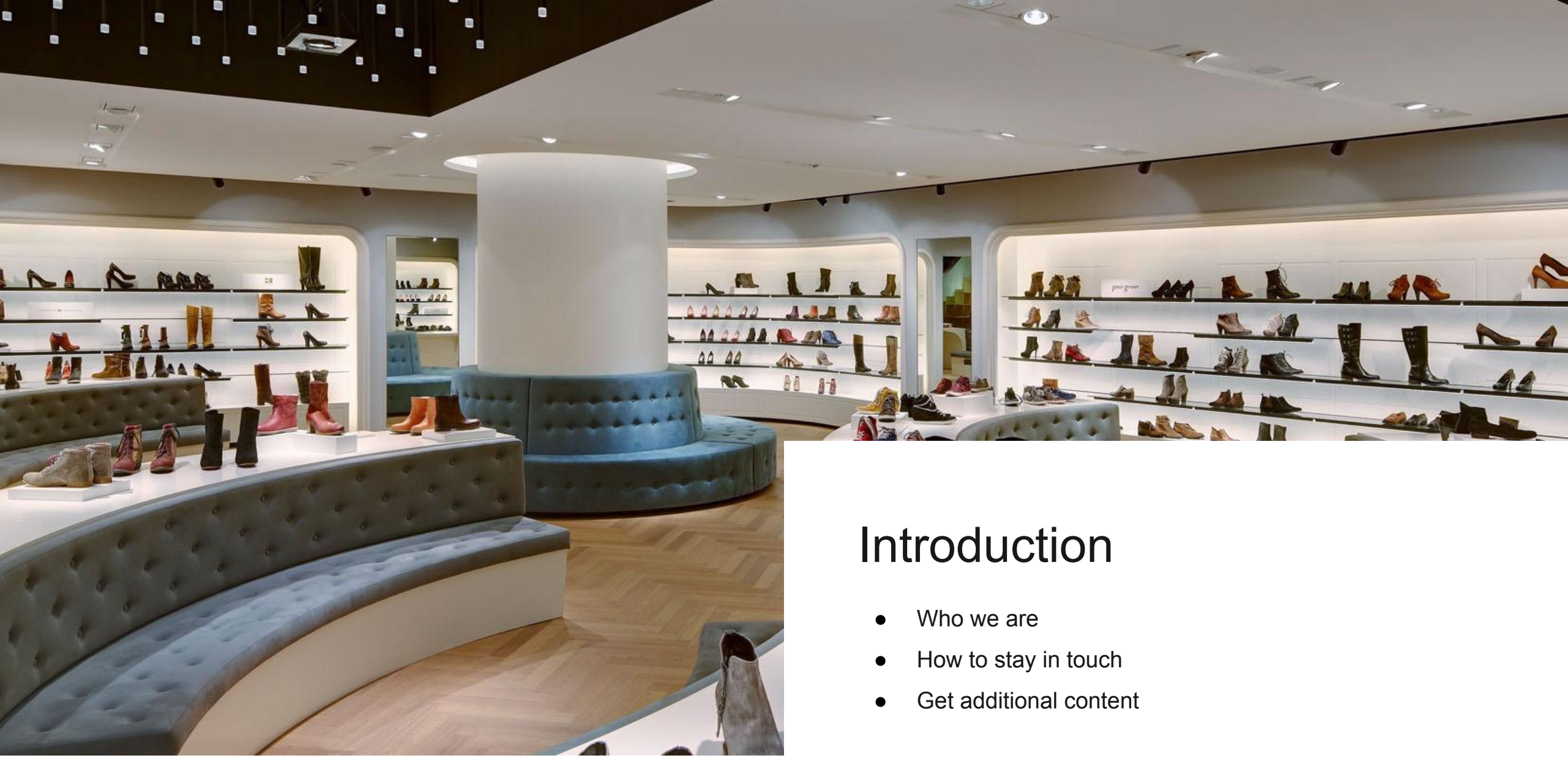
data2day, 21/09/2022

Alec Sproten · Head of Data Science
Olivier Bénard · Data Software Engineer

Agenda

What is our roadmap?

1. Introduction
2. Objectives
3. What is Breuninger
4. Workshop
    1. Getting new Data Science ideas
    2. Building a typical ML Business-Case
    3. Orchestrating Machine Learning Ops and Data Governance
5. Conclusion
6. Q&A



Introduction

- Who we are
- How to stay in touch
- Get additional content

Introduction

The Speakers

Dr. Alec Sproten

Head of Data Science (2020)

Cognitive Architectures · Belgian Beer · Anime

Olivier Bénard

Data Software Engineer (2021)

Technical blog · Marathons · Dance

Introduction

The Speakers

Dr. Alec Sproten
Head of Data Science (2020)

john.doe@breuninger.de

Olivier Bénard
Data Software Engineer (2021)

johnny.doe@breuninger.de

Introduction

Stay in touch

Mara Sharma

Community Manager

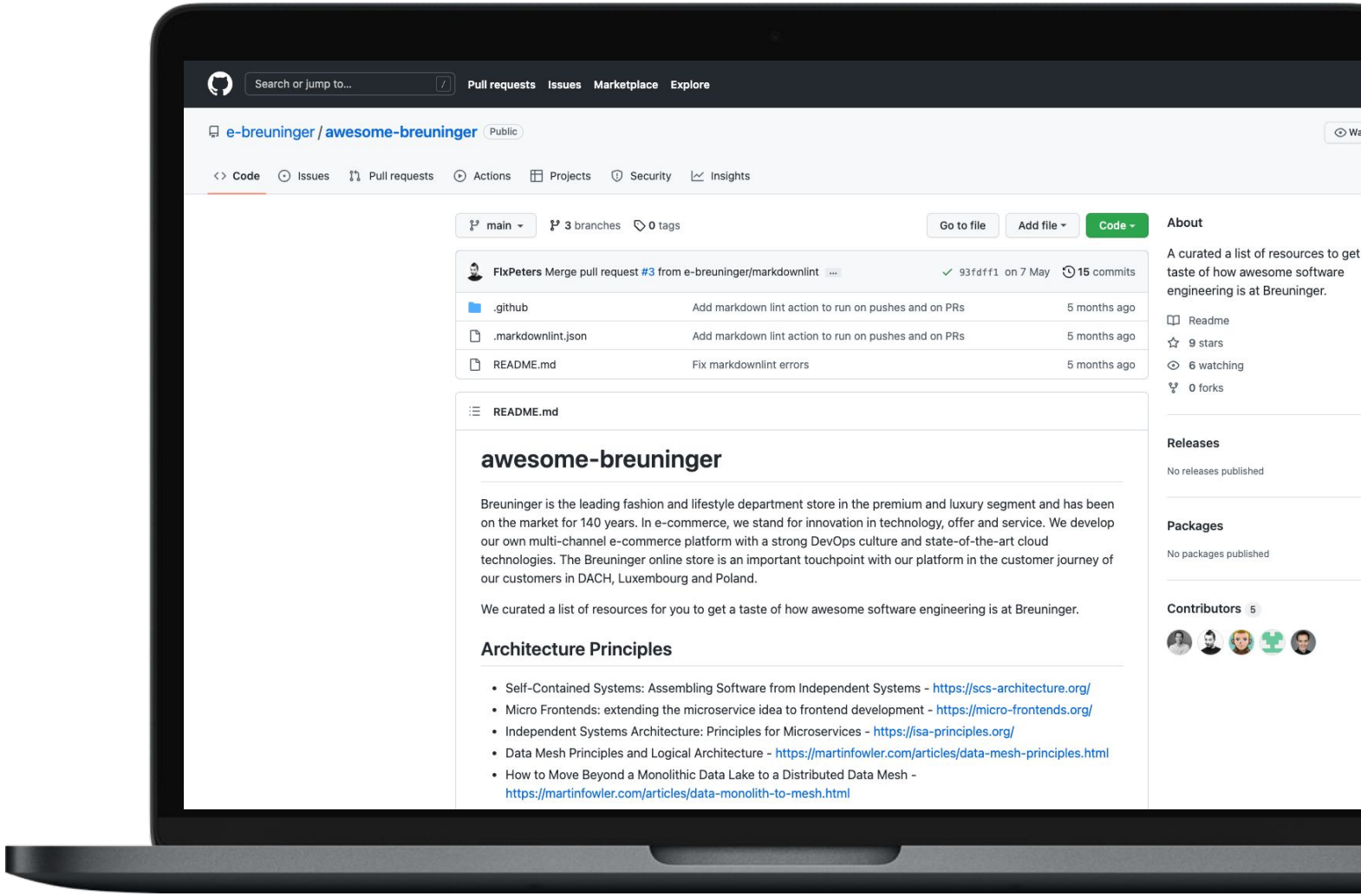
jane.doe@breuninger.de

Introduction

We ❤️ open-source

- The content is publicly available on **Github**

  `github.com/e-breuninger/data2day-2022`



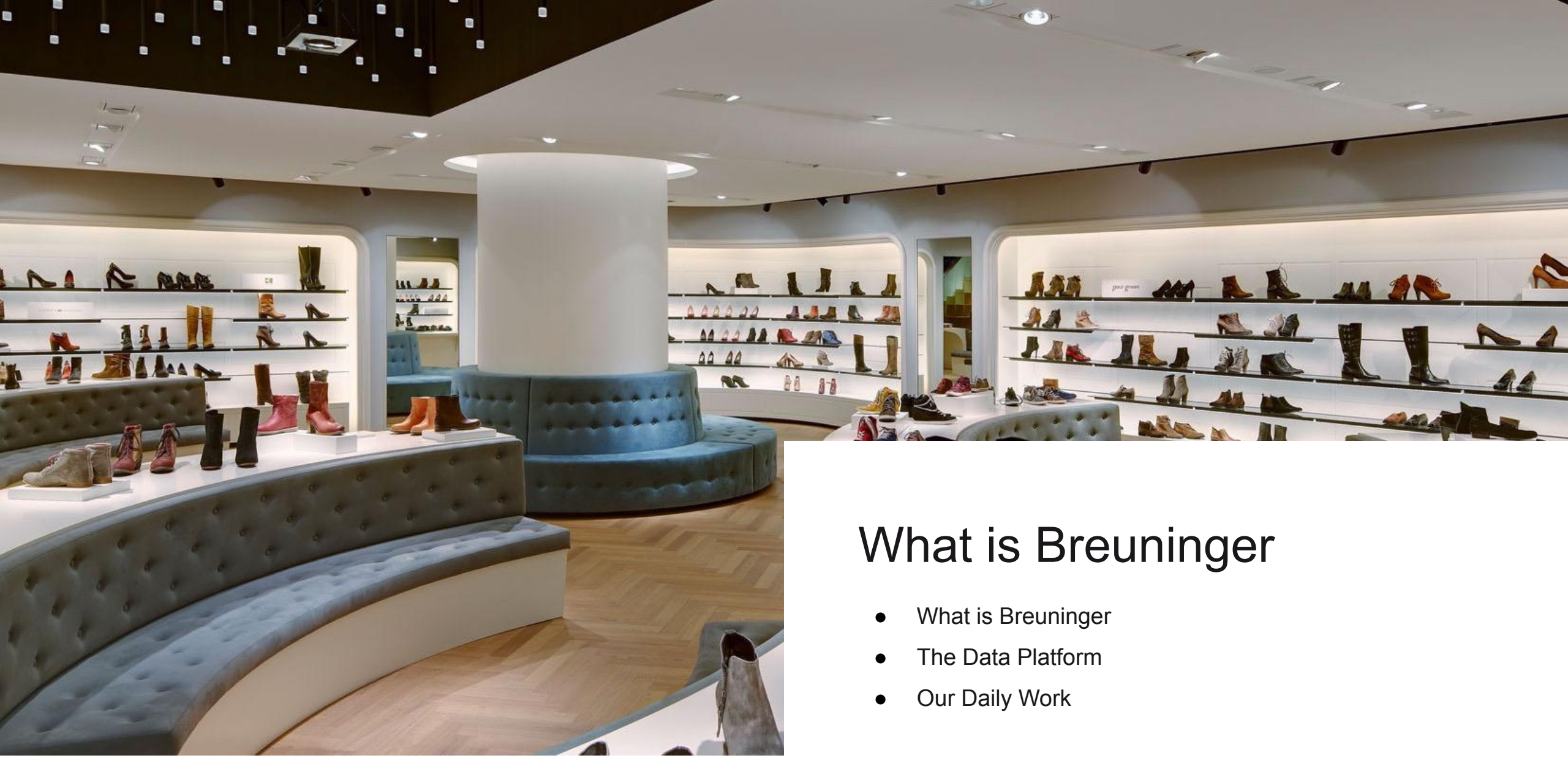
Objectives

- What are the objectives
- What is our journey to reach them

Objectives of the presentation

What are the objectives

- Giving you an **up-to-date** image of Breuninger and **interesting insights** on how Breuninger is working with data.
- Present a typical **Breuninger ML Business-Case**, from MVP to production
- Based on that example, open the discussion on how you can overcome **ML Operations** and **Governance** challenges.



What is Breuninger

- What is Breuninger
- The Data Platform
- Our Daily Work

What is Breuninger

Industry

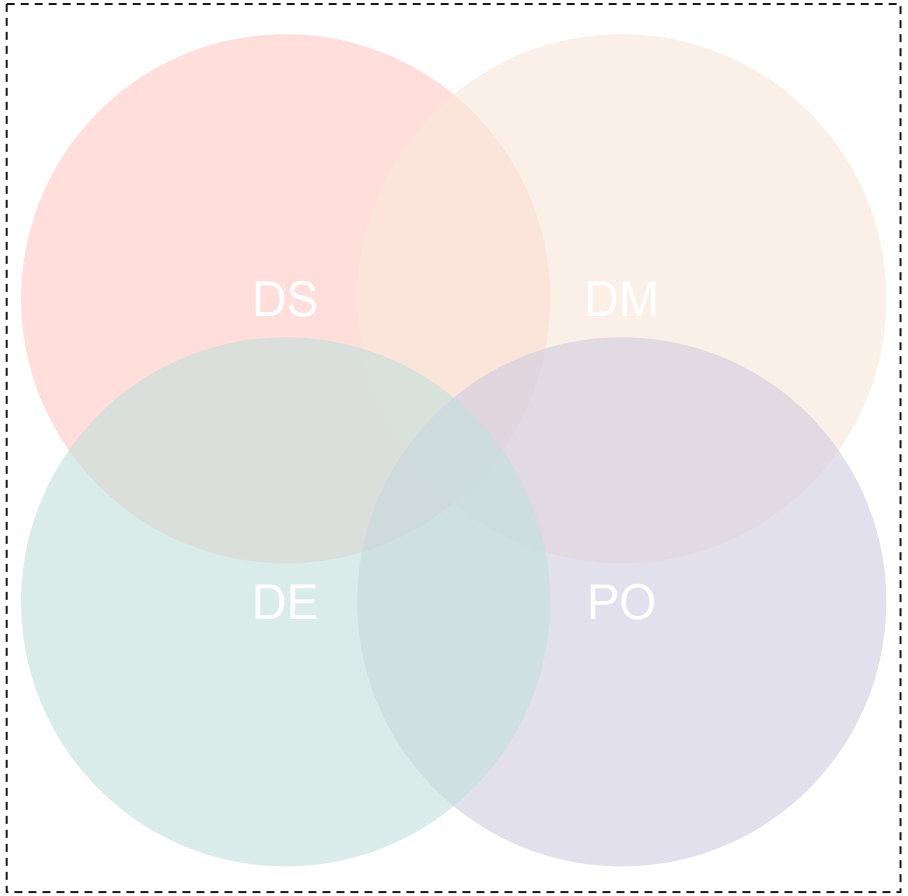
- Department Store Retailer with **140+ years** of experience
- **13 offline shops** in Germany
- Specialised in **fashion**, **clothing**, **premium household goods**

The Breuninger shop in Karlsruhe.

What is Breuninger

Data Platform Services · Data Driven Company

- **30+** employees within the team
- **4** sub-teams
- **Provide** and **distribute data** across the **whole organisation**.



Data Platform Services

What is Breuninger

Our daily work

1. **Ingest** TBs of data in an automatic and reliable way, no matter what the sources are (Extract)
2. **Collect** it at one central place, accessible for all trusted users in the Cloud (Load)
3. **Transform** the data to give it the shape you want to suit your business needs (Transform)

What is Breuninger

Our daily work

1. **Predict**, **correlate** and **segment** to help our stakeholders provide an outstanding shopping experience.
2. **Consult** and **evaluate** possibilities of ML solutions.
3. **Enjoy** the possibilities of our Data Platform thanks to our Data Engineering and Data Modeling work.

Workshop

- Getting new Data Science ideas
- Building a typical ML Business-Case
- Orchestrating Operationalization and Data Governance

Workshop

- Getting new Data Science ideas
- Building a typical ML Business-Case
- Orchestrating Operationalization and Data Governance

Workshop

Getting new business ideas

Source: Images created with Dall-E 2

Workshop

Getting new business ideas

Source: Images created with Dall-E 2

Workshop

- Getting new Data Science ideas
- Building a typical ML Business-Case
- Orchestrating Operationalization and Data Governance

Workshop

Building a typical Breuninger ML Business-Case

1. Requirements
2. Workflow
3. Implementation v 1.0

Workshop

Building a typical Breuninger ML Business-Case

1. Requirements
2. Workflow
3. Implementation v 1.0

Workshop

Building a typical Breuninger ML Business-Case · **Requirements**

- Predict the offline shop **footfall** (= customer frequentation)
- Based on **pre-covid** data (< 2019)
- For a **single location**

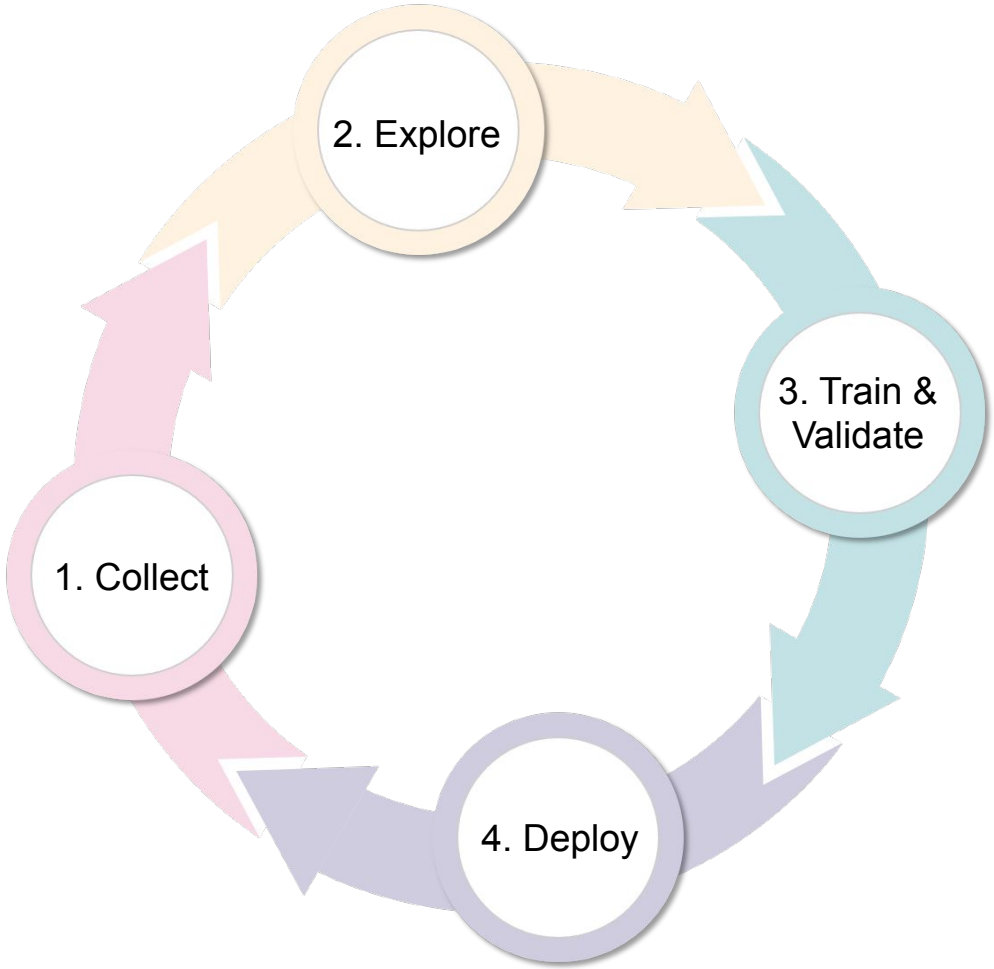
Workshop

Building a typical Breuninger ML Business-Case

1. Requirements
2. Workflow
3. Implementation v 1.0

Workshop

Building a typical Breuninger ML Business-Case · **Workflow**



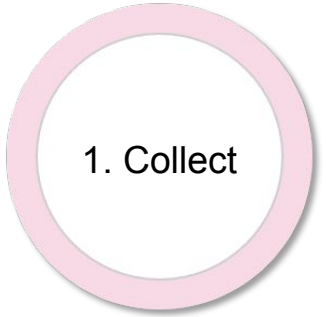
Workshop

Building a typical Breuninger ML Business-Case

1. Requirements
2. Workflow
3. Implementation v 1.0

Workshop

Building a simple forecast tool · **Implementation v 1.0**



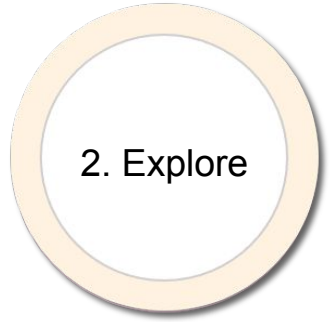
1. There are **11 entrances** to the shop in Stuttgart
2. From those entrances, we assume how many people are **coming in** and **going out**
3. We obtain the **total number of people** present in the shop

date	offline_customer_quantity
2018-01-01	3605
2018-01-02	2996
2018-01-03	13232
2018-01-04	17747
2018-01-05	17748

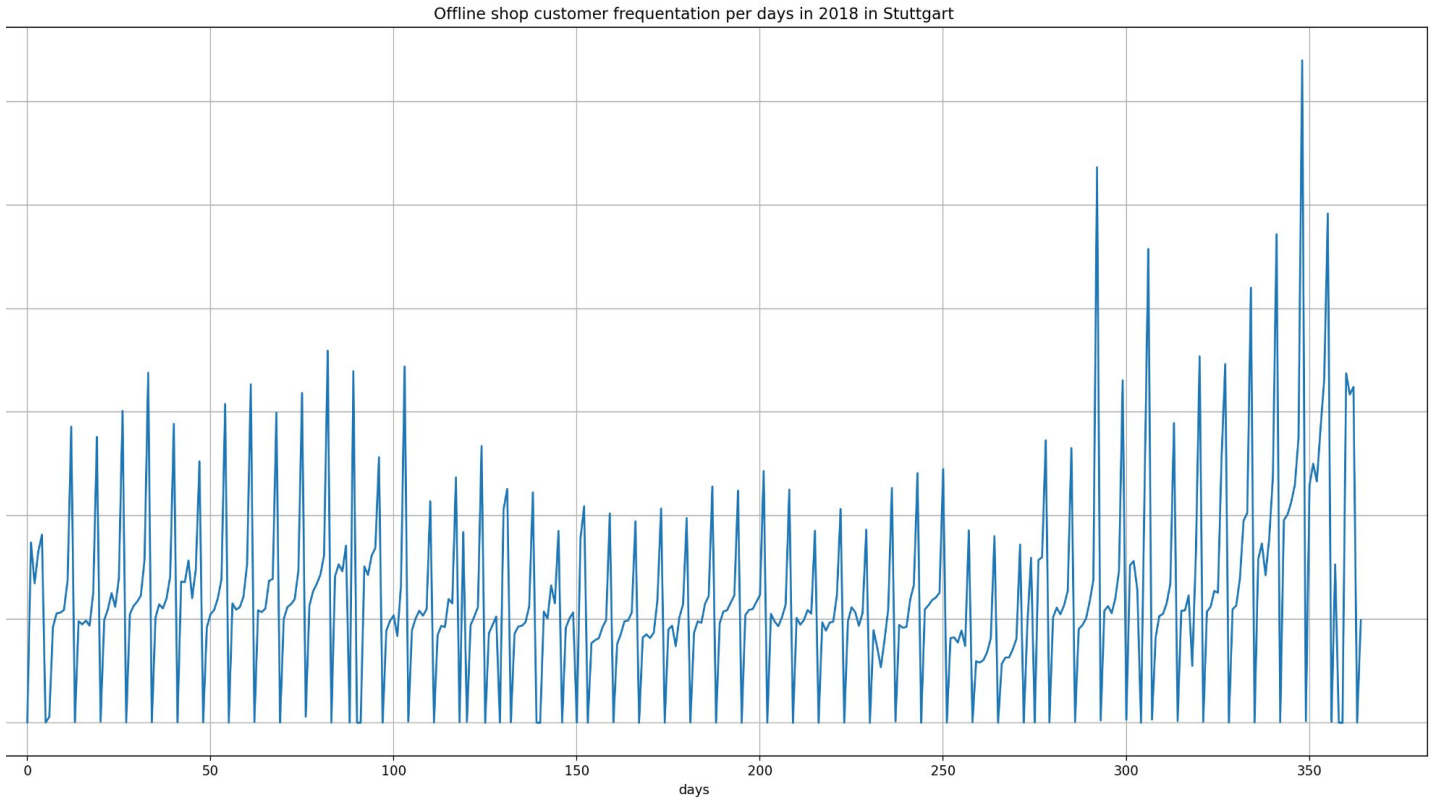
** The displayed values are not the real values.*

Workshop

Building a simple forecast tool · **Implementation v 1.0**



1. Frequentations **double** on Saturdays.
2. Strong **seasonal pattern** (Black Week, Xmas).
3. Frequentations **lower but steady mid-year**.



** The real quantity is not displayed for confidentiality issues. However, you still get a fair order of magnitude.*

Workshop

Building a simple forecast tool · **Implementation v 1.0**

Comparing **4 *univariate* models**:

a. Simple Exponential Smoothing (Holt-Winters)
b. Holt-Winters with additive seasonality
c. Holt-Winters with multiplicative seasonality
d. Arima

Notes:

- As these models are univariate, no additional features are considered (e.g. weather,...)
- Hyperparameters: seasonality is set to **7 days**

Workshop

Building a simple forecast tool · **Implementation v 1.0**

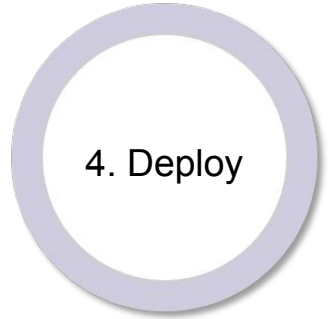
Using the **Mean Absolute Percentage Error (MAPE)** to select the best performing model:

```
> from sklearn.metrics import mean_absolute_percentage_error
> mean_absolute_percentage_error(quantity, model3_fit.predict(start=0, end=364)))
MAPE model_3: 29.55%
```

Holt-Winters with multiplicative seasonality is the winning model.

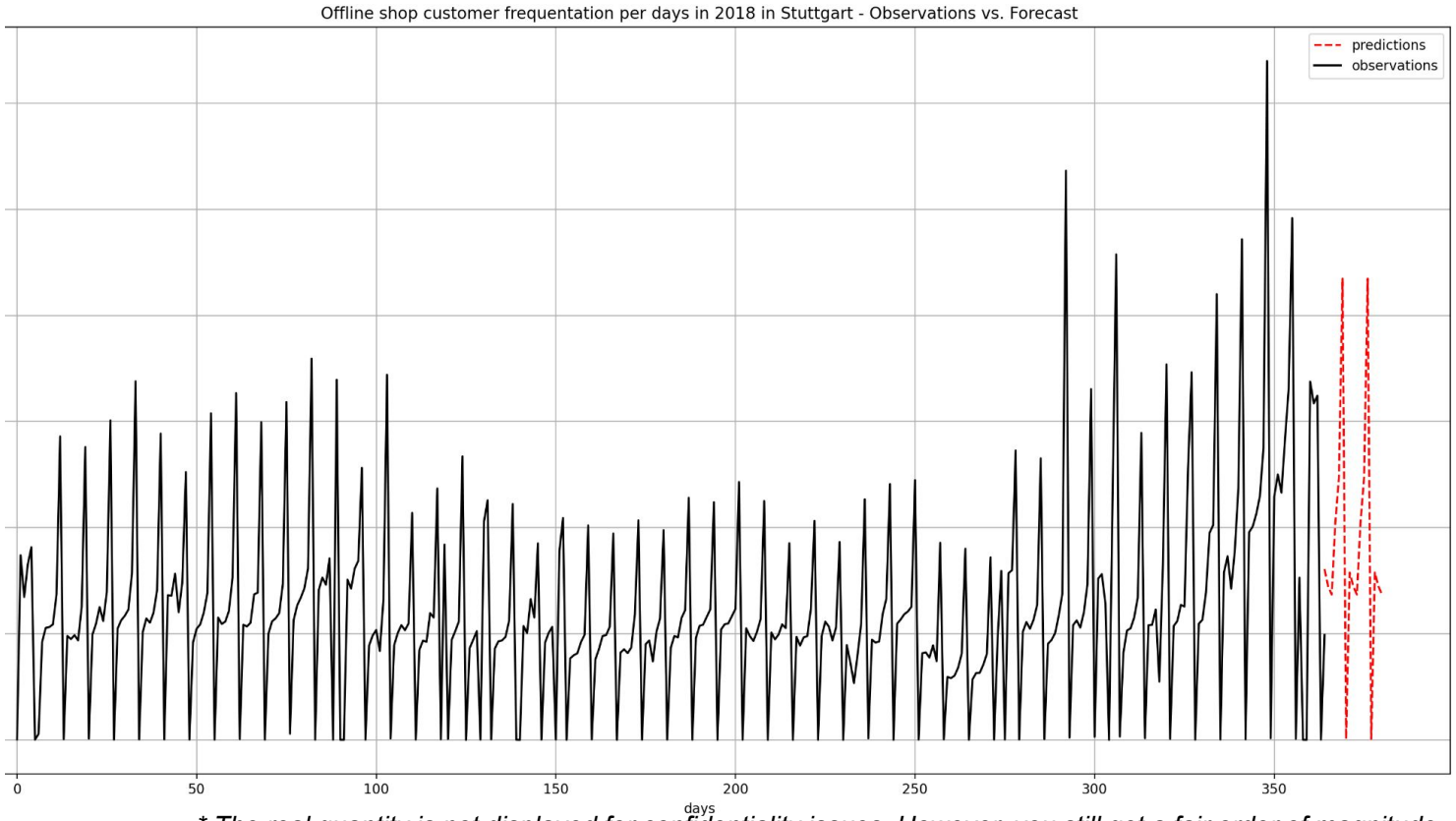
Workshop

Building a simple forecast tool · **Implementation v 1.0**



However:

- The forecast only captures the overall **pattern**.
- It cannot react to **external drivers** yet.
- A forecast alone is not sufficient, it has to be **operationalized** and **integrated**.



** The real quantity is not displayed for confidentiality issues. However, you still get a fair order of magnitude.*

Workshop

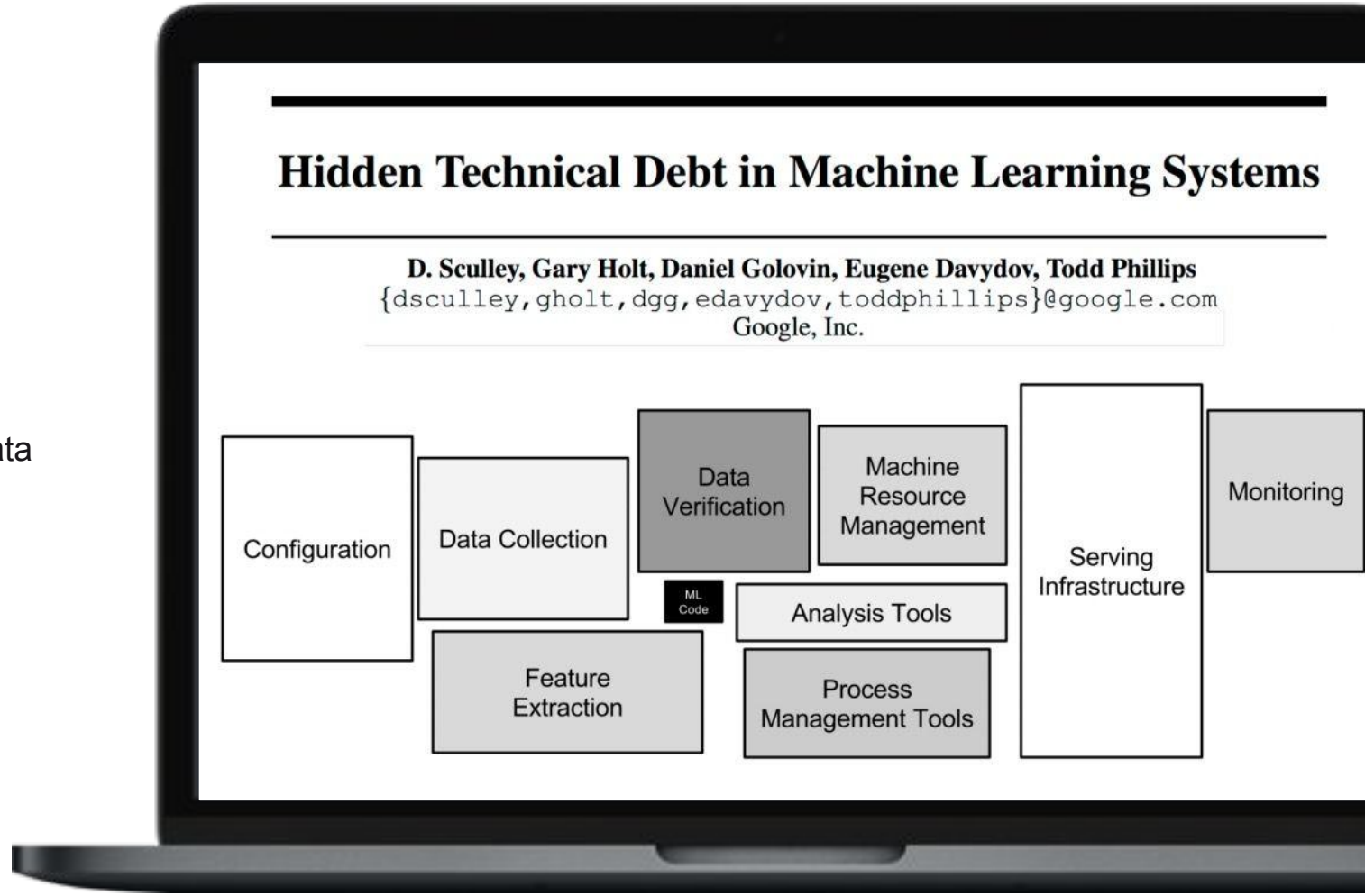
- Getting new Data Science ideas
- Building a typical ML Business-Case
- Orchestrating Operationalization and Data Governance

Workshop

Operationalization and Data Governance

Breuninger comes from:

- Poor **infrastructure** set-up
- Complex and **error-prone operations**
- Inadequate or absent **monitoring** of models and data
- Long **dev-cycles**
- No real-time **model serving**



Workshop

Operationalization and Data Governance

Therefore, we decided to select and use one **unified ML-Ops platform** solution:

- Cross-functional **team**
- Derivation of **required features**
- **Demos** by solution providers
- Finalization of **criteria catalogue**
- Hands-on **evaluation**

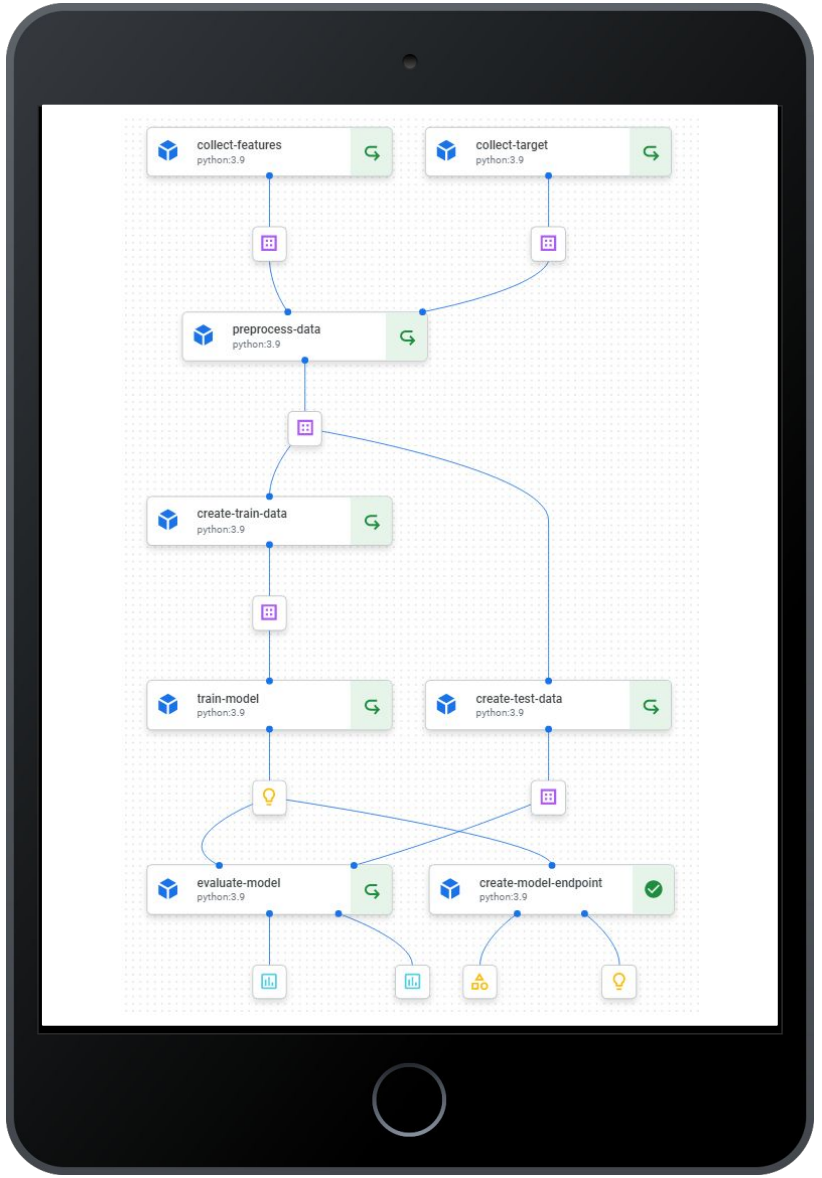
Categories		Weight of Feature in %
integration	big query integration	100
	gitlab integration	100
	installation/updating in cloud (Fokus auf Update und Maintainance)	50
	sso	20
dasc	notebooks (prototyping, development, ...)	100
	feature store	40
	automl	40
	computing parallelization / Big Data support	100
	A/B testing and canary releases	70
	model versioning	100
	experiment tracking	100
	preprocessing functionality (use-case specific after dps cleansing)	100
operations	workflow support	100
	monitoring model drift	90
	reproducability/data versioning	30
	model serving	100
	model extraction	100
	scheduling of jobs	100
	controlling via trigger by DPS (AirFlow)	50
	API / SDK	40
	Alerting: Fehlerhafte Ausführung von Jobs, Model drift, Metriken unter Threshold	40
usability	understandable ui	100
	documentation	80
	usable coding environment e.g. code completion, easy integration of non-standard libraries	90
	IDE integration (Intelllij/VS Code) (u.a. remote Kernel ausführen, debugging,)	50
	external libs	100
security	access control (prefered: RBAC)	100
	compliance data governance (ko)	-
Trust in product quality and support	Competence of contact persons	10
	Stability during testing-phase (ko)	-
	Promising Development Roadmap of Product	20
Price / Licence restrictions		
weighted SUM		

Workshop

Operationalization and Data Governance

For that, we decided to use **Vertex AI**, but choosing a software usually is the smallest part of the solution…

…welcome to our **ML Ops Expert Group**!



A typical ML workflow in Vertex AI

Workshop

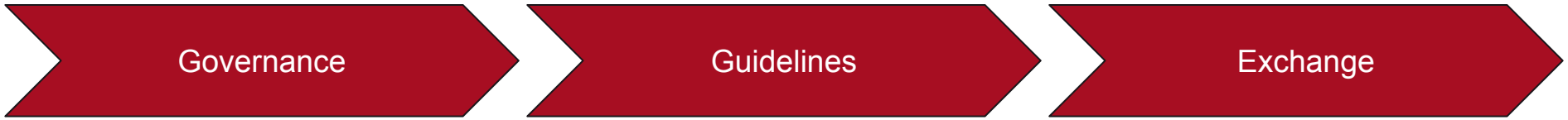
Operationalization and Data Governance

The **ML Ops Expert Group** is a **cross-functional group** including the following functions:

- Business
- Data Science
- Data Engineering
- Online Store
- Infrastructure
- Architecture

Workshop

Tasks of the Expert Group



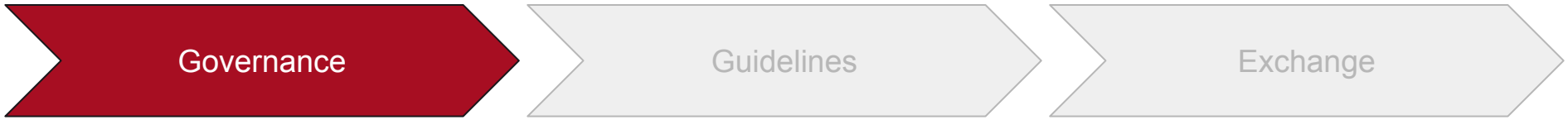
- Security
- Operative Model
- Documentation
- Data Governance
- Reproducibility and traceability

- Clean Code
- Data Versioning
- Go-live criteria

- Transparency
- Challenging

Workshop

Tasks of the Expert Group



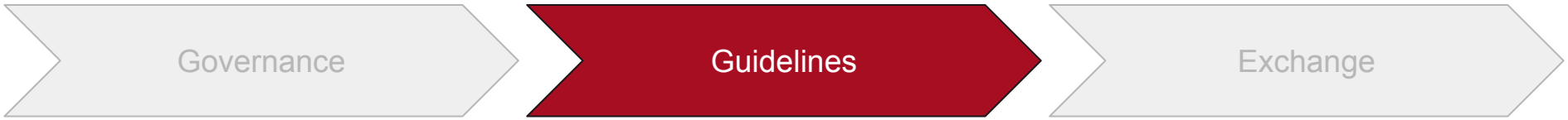
- Security
- Operative Model
- Documentation
- Data Governance
- Reproducibility and traceability

- Clean Code
- Data Versioning
- Go-live criteria

- Transparency
- Challenging

Workshop

Tasks of the Expert Group



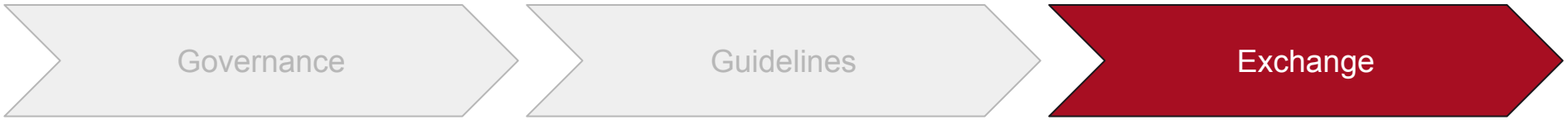
- Security
- Operative Model
- Documentation
- Data Governance
- Reproducibility and traceability

- Clean Code
- Data Versioning
- Go-live criteria

- Transparency
- Challenging

Workshop

Tasks of the Expert Group



- Security
- Operative Model
- Documentation
- Data Governance
- Reproducibility and traceability

- Clean Code
- Data Versioning
- Go-live criteria

- Transparency
- Challenging

Conclusion

- What could you learn from our mistakes
- Opening the discussion

Conclusion

In three key points

1. **Data Science** and **good software principles** have to go hand in hand
2. You need more than just a platform, you also need a **governance**
3. Don’t rely solely on your data scientists, **get your IT organization involved**

Q&A

Any questions?

Thank you

breuninger
breuninger.com

Additional Information

What is Breuninger

4 key numbers

- Created in **1881** (141 years old)
- **13** stationary stores in Germany (HQ in Stuttgart)
- **6,000+** employees with **30+** nationalities
- Online service in **8** countries (de, at, be, es, lu, nl, pl, pt)

Also in Luxembourg and Munich through acquisition of Bram and Konen.
New shopping center in Hamburg (Überseequartier) planned for 2023.

What is Breuninger

Additional facts

- First German outlets to install **elevators** (1950)
- First German retailer to introduce a **loyalty card** (1959)
- First German retailer to provide **multi-storey car park**
- Employee dress code sticks to **white**, **black** or **grey** (1920)
- Member of the *Fur Free Retailer Program* (2006)

Workshop

Software Development · **Best Practices**

Ensure excellent codebase quality, easy maintenance and speed-up development life cycles through the use of:

- **poetry** (python package manager)
- **pre-commit** hooks (the code must meets certain standards to be pushed)
  - `mypy`, `pylint`, `black`
- **Unittests** and **Doctests** in docstrings
- **DevOps principles**
  - `Cookiecutter` (too quickly scaffold new projects based on the same template)
  - Automated CI/CD pipelines (via .yaml files)
  - Standardised commit messages and branch naming conventions e.g. `dev/VERS-5463_adding_new_columns`
  - Clear MR rules and protected branches
  - `Makefile` (to store actionable commands)
  - `.gitignore` and logic separation (`dynaconf` to keep the code dry), most atomic functions as possible
  - Log File tracking system using `logging`

Workshop

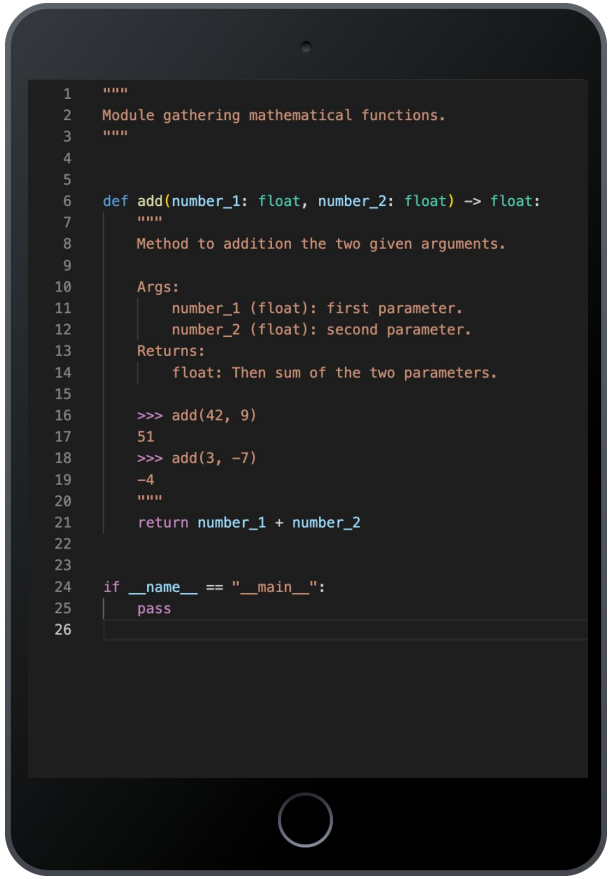
Software Development · **Best Practices**

```python
import pandas as pd


def add(a, b):
    return a+b


```

Version 1.0 - Unused import, no docstrings…



```python
"""
Module gathering mathematical functions.
"""


def add(number_1: float, number_2: float) -> float:
    """
    Method to addition the two given arguments.

    Args:
        number_1 (float): first parameter.
        number_2 (float): second parameter.
    Returns:
        float: Then sum of the two parameters.

    >>> add(42, 9)
    51
    >>> add(3, -7)
    -4
    """
    return number_1 + number_2


if __name__ == "__main__":
    pass

```

Version 2.0 - Black, mypy and black formatting

```
error: [mypy] Error 1 Function is missing a type annotation
poetry run pylint my_package
************* Module helpers.utils
my_package/helpers/utils.py:1:0: C0114: Missing module
docstring (missing-module-docstring)
my_package/helpers/utils.py:1:0: C0116: Missing function or
method docstring (missing-function-docstring)
my_package/helpers/utils.py:1:6: C0103: Argument name "a"
doesn't conform to snake_case naming style (invalid-name)
my_package/helpers/utils.py:1:18: C0103: Argument name "b"
doesn't conform to snake_case naming style (invalid-name)
my_package/helpers/utils.py:1:0: W0611: Unused pandas
imported as pd (unused-import)

-------------------------------------------------------------
Your code has been rated at 8.46/10 (previous run:
10.00/10, -1.54)
```

Workshop

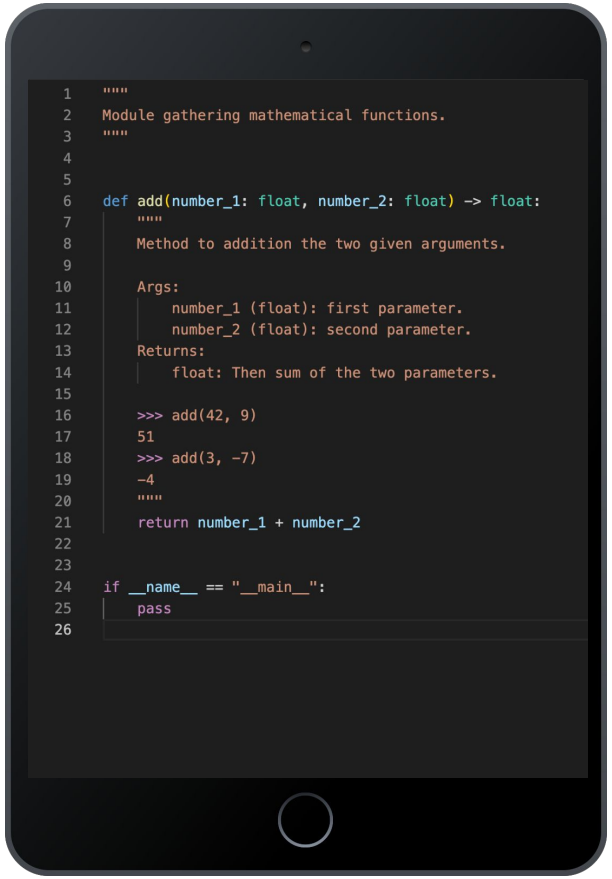
Software Development · **Best Practices**

```python
import pandas as pd


def add(a, b):
    return a+b


```

Version 1.0 - Unused import, no docstrings…



```python
"""
Module gathering mathematical functions.
"""


def add(number_1: float, number_2: float) -> float:
    """
    Method to addition the two given arguments.

    Args:
        number_1 (float): first parameter.
        number_2 (float): second parameter.
    Returns:
        float: Then sum of the two parameters.

    >>> add(42, 9)
    51
    >>> add(3, -7)
    -4
    """
    return number_1 + number_2


if __name__ == "__main__":
    pass

```

Version 2.0 - Black, mypy and black formatting

```
> pytest -s --doctest-modules -vv .
3.9.9 Py  15:30:37
================= test session starts ================
cachedir: .pytest_cache
rootdir: /my_package/helpers
collected 1 item

utils.py::utils.add PASSED

================= 1 passed in 0.03s ==================
```

Workshop

Extend the Forecast · **Implementation v 2.0**

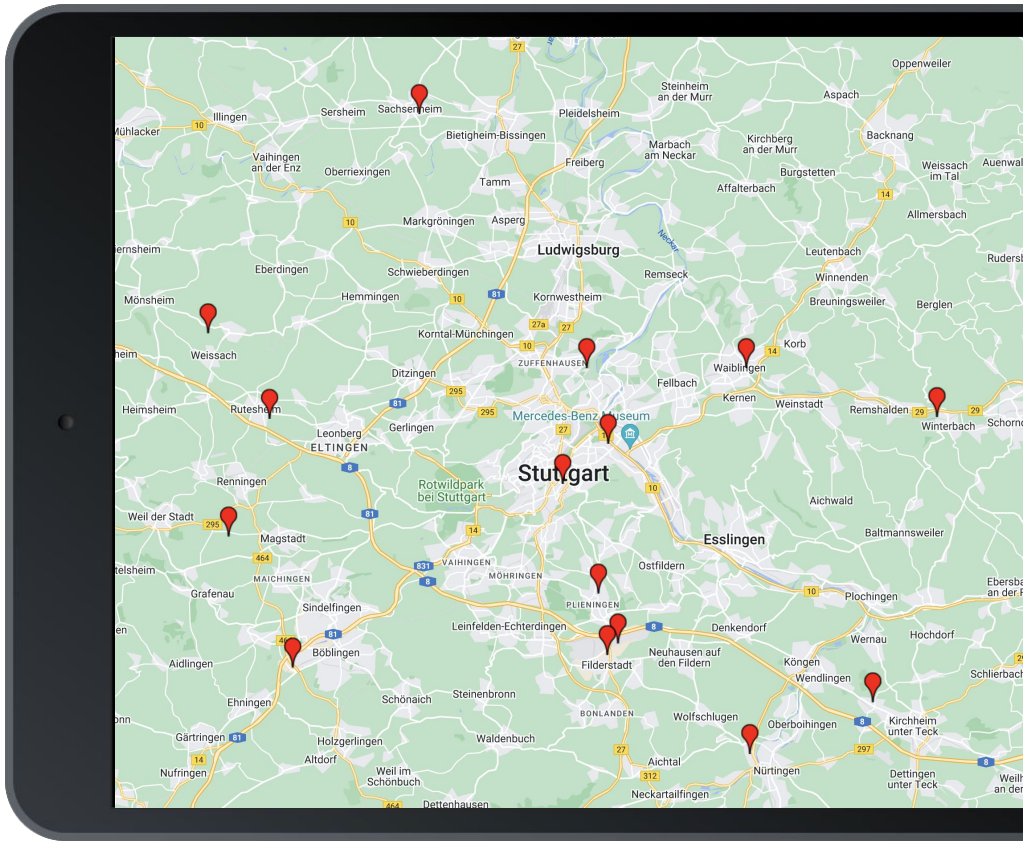
1. Requirements
2. Workflow
3. Implementation v 1.0
4. Implementation v 2.0
5. Deployment

Workshop

Extend the Forecast · **Implementation v 2.0**

The model can be extended and complexified with:

1. **Weather Data** (e.g. *Global Historical Climate Network*)
2. **Time-related information** e.g. weekend, weekday, vacations
3. **Media & Socials** e.g. demonstrations, events and venues



Weather stations located around Stuttgart in a perimeter < 30 km from the store.

```
SELECT *
FROM `bigquery-public-data:ghcn_d.ghcnd_stations`
```

Workshop

Extend the Forecast · **Implementation v 2.0**

date	offline_customer_quantity	weekday	vacation	sunny	rainy	precipitation	warm	cold	fashion_week	ongoing_strike
2018-01-01	3605	yes	no	no	yes	3.2	no	yes	no	no
2018-01-02	2996	yes	no	no	yes	7.2	no	yes	no	no
2018-01-03	13232	yes	no	no	yes	12.6	no	yes	no	no
2018-01-04	17747	no	no	yes	no	0.6	yes	no	no	no
2018-01-05	17748	no	no	yes	no	0.8	yes	no	no	no

** The displayed values are not the real values.*