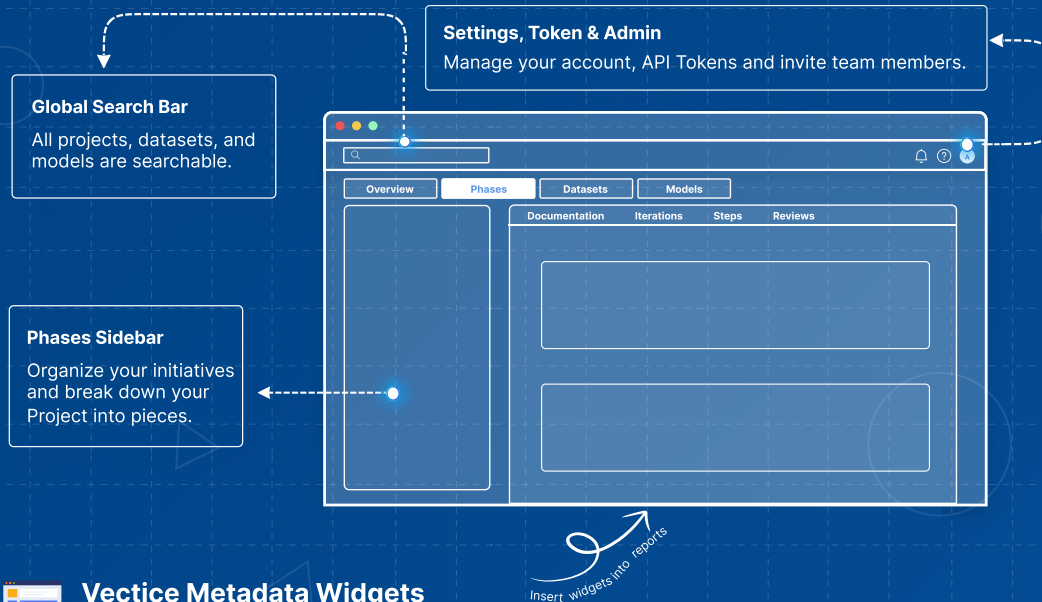




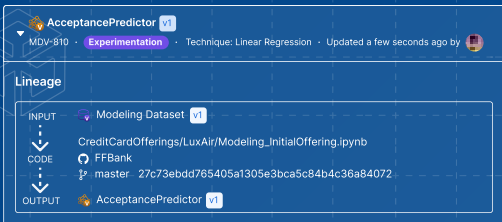
Vectice Blueprint

You can find each section in the areas pointed below



Vectice Metadata Widgets

You can use every asset as widgets to conveniently add them to reports.



Asset Metadata Widget for Datasets and Models

View its lineage, status, technique, and which team member worked on it.

Metadata Comparison Widget for Datasets and Models

Dynamically compare Model Versions' status, technique, and metrics.

Logistic Regression						
MDL-414 · ML Regression · 1 in Production 1 in Staging 2 in Experimentation · Updated 5 minutes ago by [user]						
Search... Filters						
Version	Status	Technique	Lineage	Created Date	Creator	AUC
Version 4	Experimentation	Logistic Regression	[icon]	2 hours ago	[user]	0.5881
Version 3	Experimentation	Logistic Regression	[icon]	6 hours ago	[user]	0.6129
Version 2	Staging	Logistic Regression	[icon]	5 minutes ago	[user]	0.7590
Version 1	Production	Logistic Regression	[icon]	2 months ago	[user]	0.7028



Structure

How vectice keeps your work neat

Workspace (incl. dashboard)

Safe environment for enterprises

Project (incl. overview)

Machine learning initiative, cross-team, cross-function

Phase

Customizable structure for your project

Steps list

Guidance on what needs to be completed

Iteration

Work auto-documented from the notebook/IDE. Iterations includes steps!

Documentation

Audience-specific reports leveraging auto-documented assets

Review

Gain insight into phase progress and data science initiatives

Dataset

Auto-versioned repository based on dataset-metadata

Model

Auto-versioned repository based on model-metadata



Code Snippets

Auto-document your work from your Notebook, IDE or Pipeline

Install the latest Vectice Python client library

```
%pip install vectice
import vectice
```

Connect to Vectice

```
con = vectice.connect(api_token="MY-TOKEN")
```

Retrieve Phase ID & create iteration

```
phase = con.phase("PHA-XXXX")
iter = phase.create_or_get_current_iteration() or iter = phase.create_iteration()
```

List the steps of an iteration

```
iter.list_steps()
```

Document a Dataset

```
clean_dataset = Dataset.clean(name, resource=FileResource(paths, dataframes))
iter.step_quickstart = clean_dataset
```

Document a Model

```
model = Model(name, library, technique, metrics, predictor, attachments)
iter.step_quickstart = model
```

Document context from notebook/IDE

```
iter.step_quickstart += "Context about model or dataset"
```

Also available as:

```
FileResource(paths, dataframes)
GCSResource(uris, dataframes, gcs_client)
S3Resource(uris, dataframes, s3_client)
BigQueryResource(paths, dataframes, bq_client)
```

