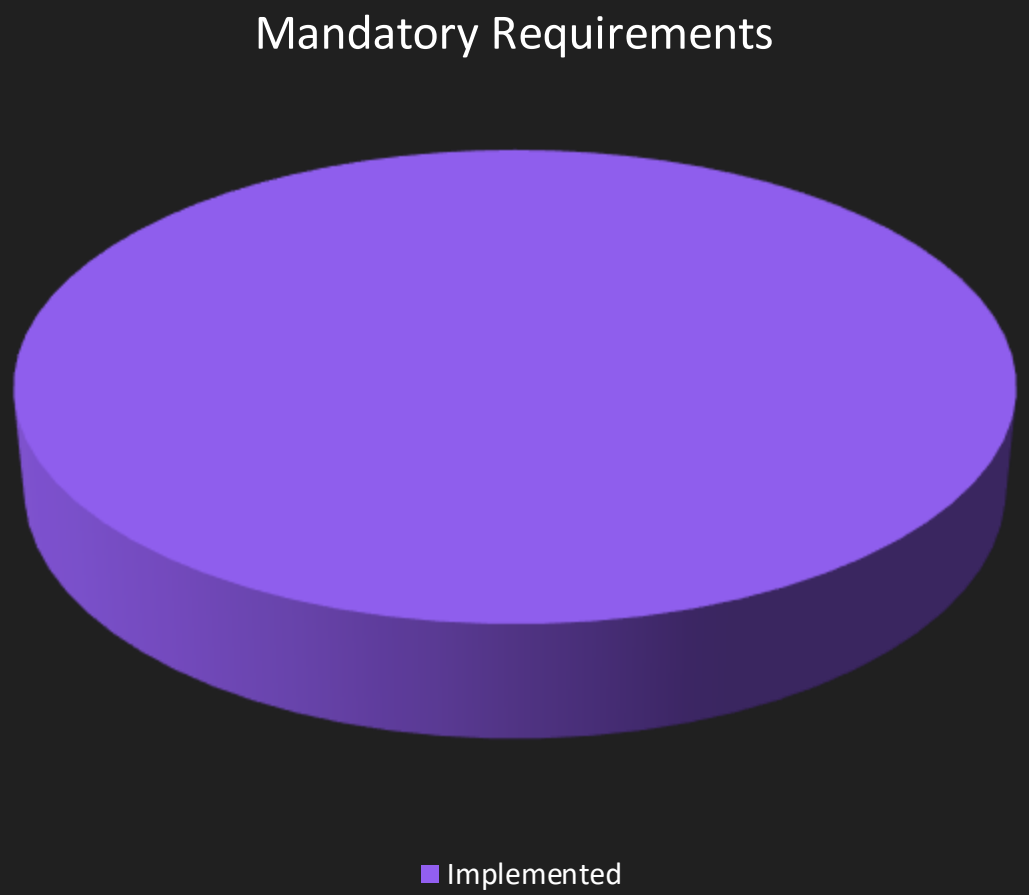
Karlsruhe Institute of Technology

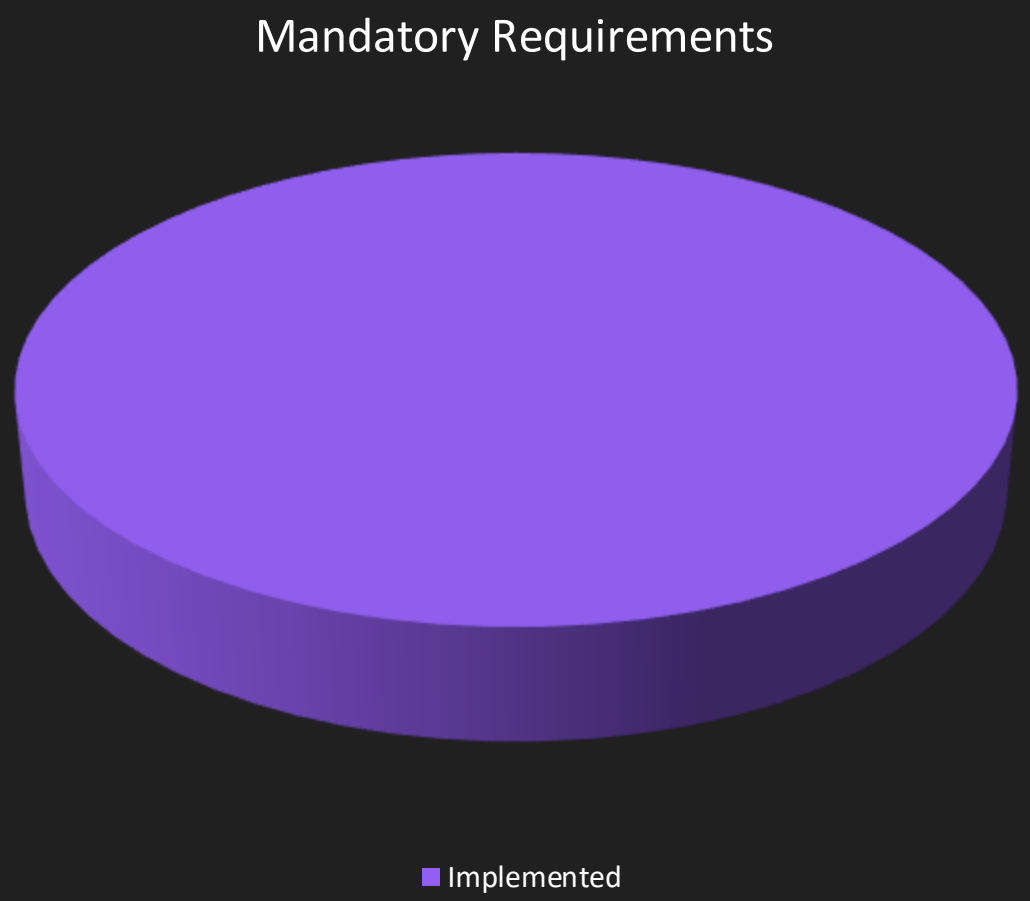
Implementation Presentation

Bennet Hörmann, Salomo Hummel, Simeon Schrape, Erik Wu, Udo Zucker

What we implemented

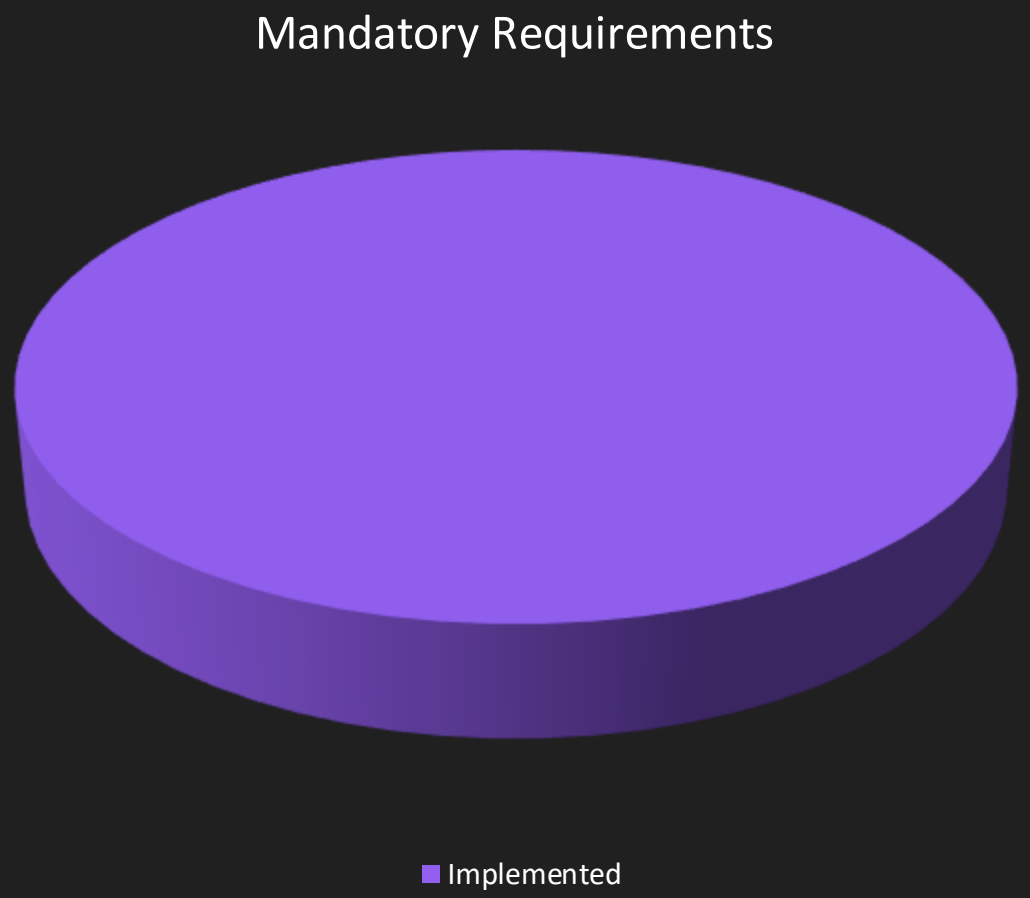


What we implemented



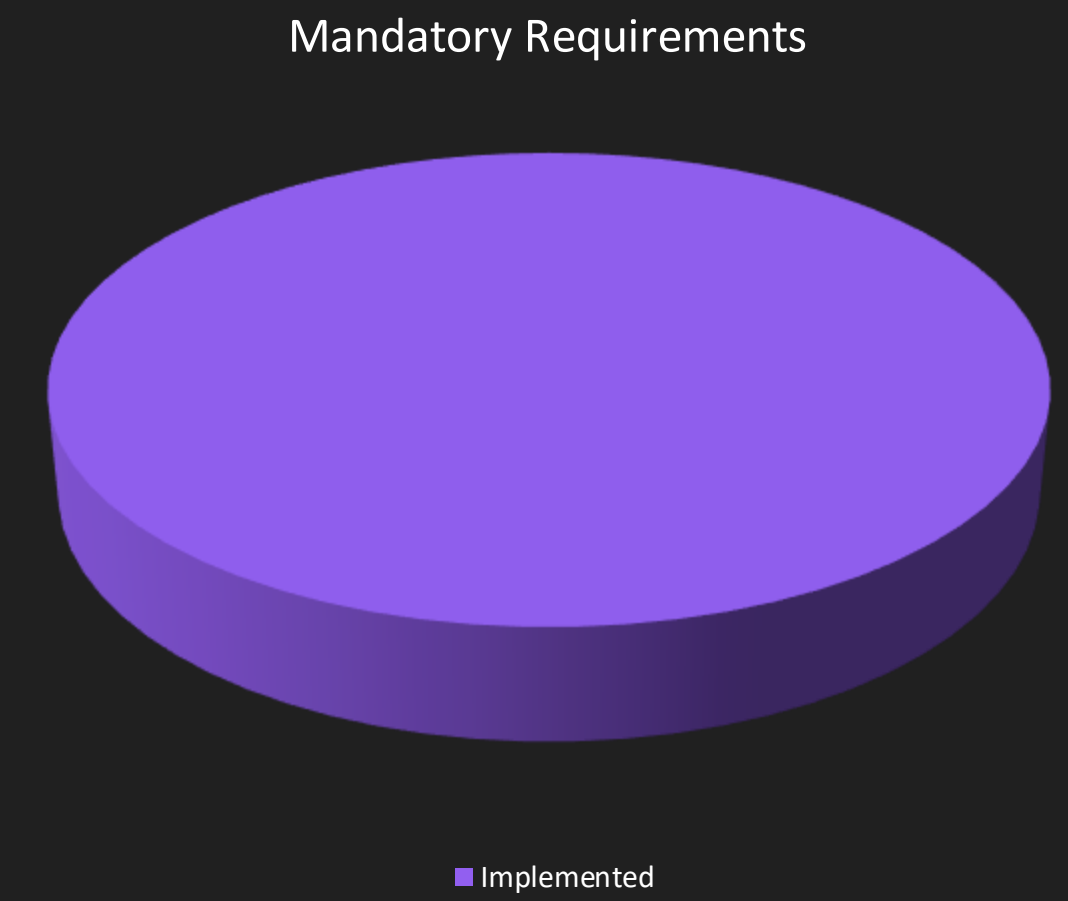
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



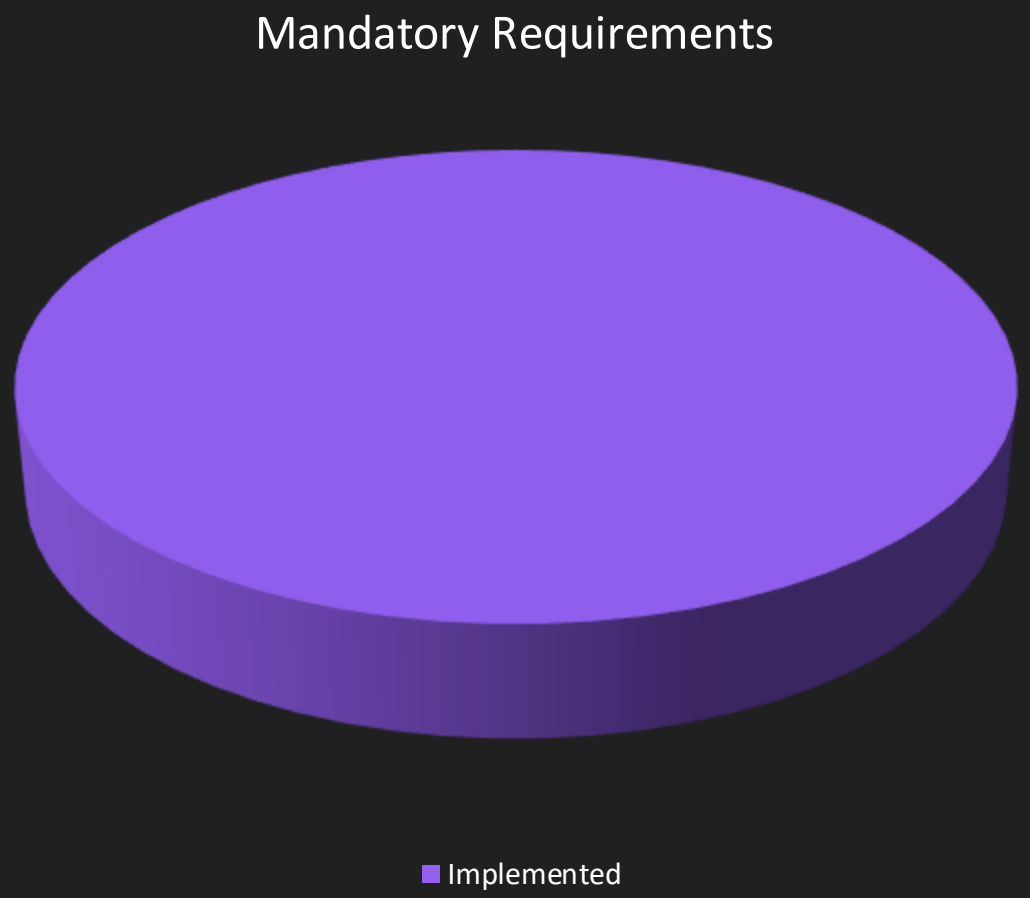
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



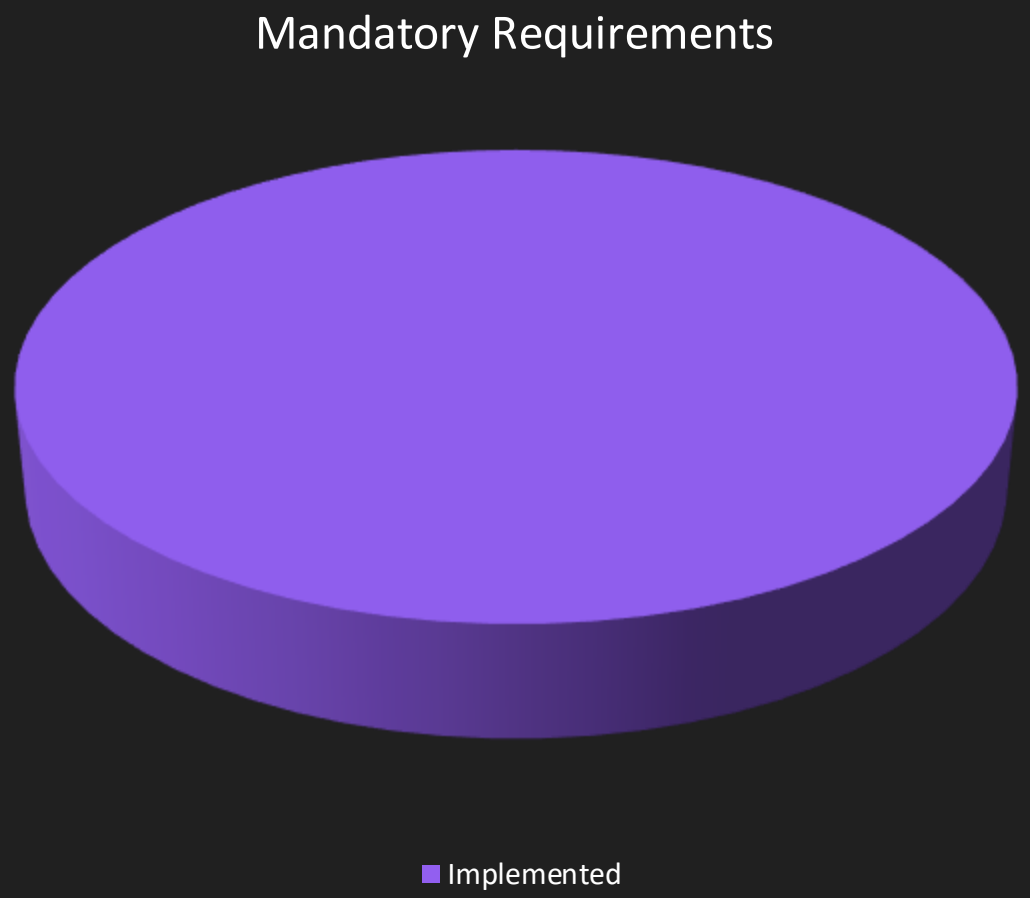
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



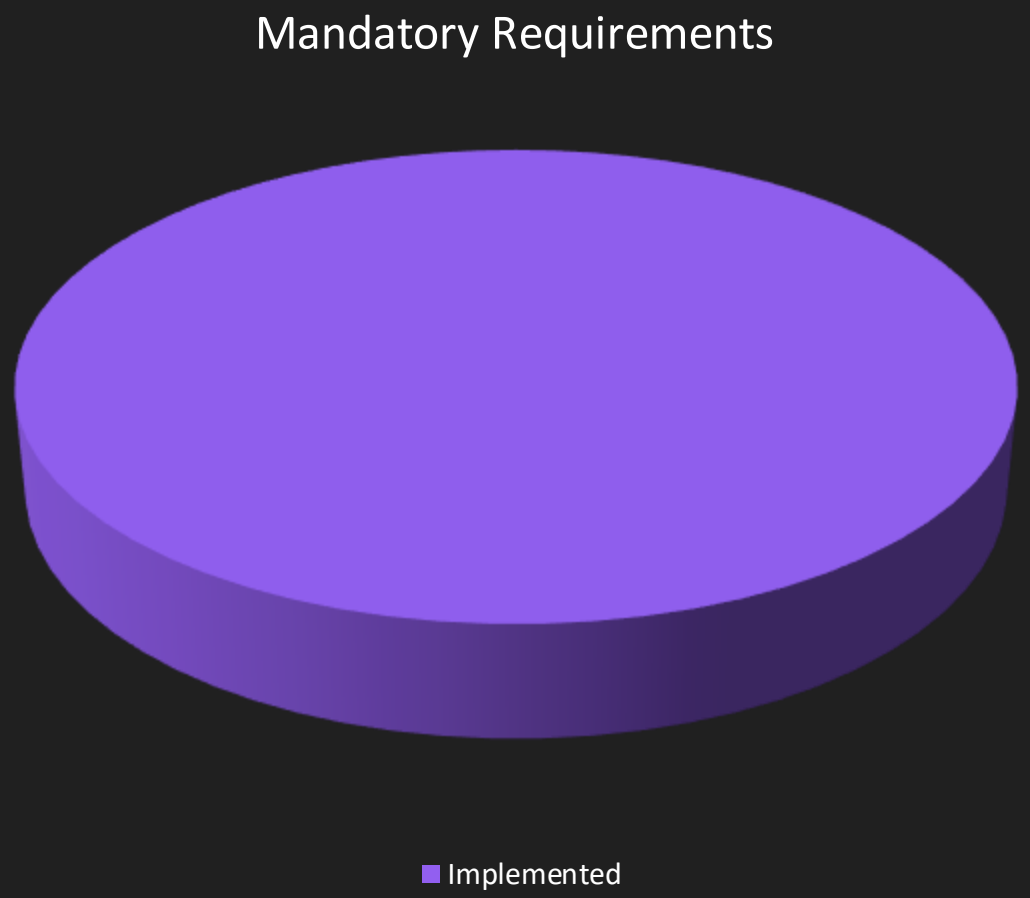
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



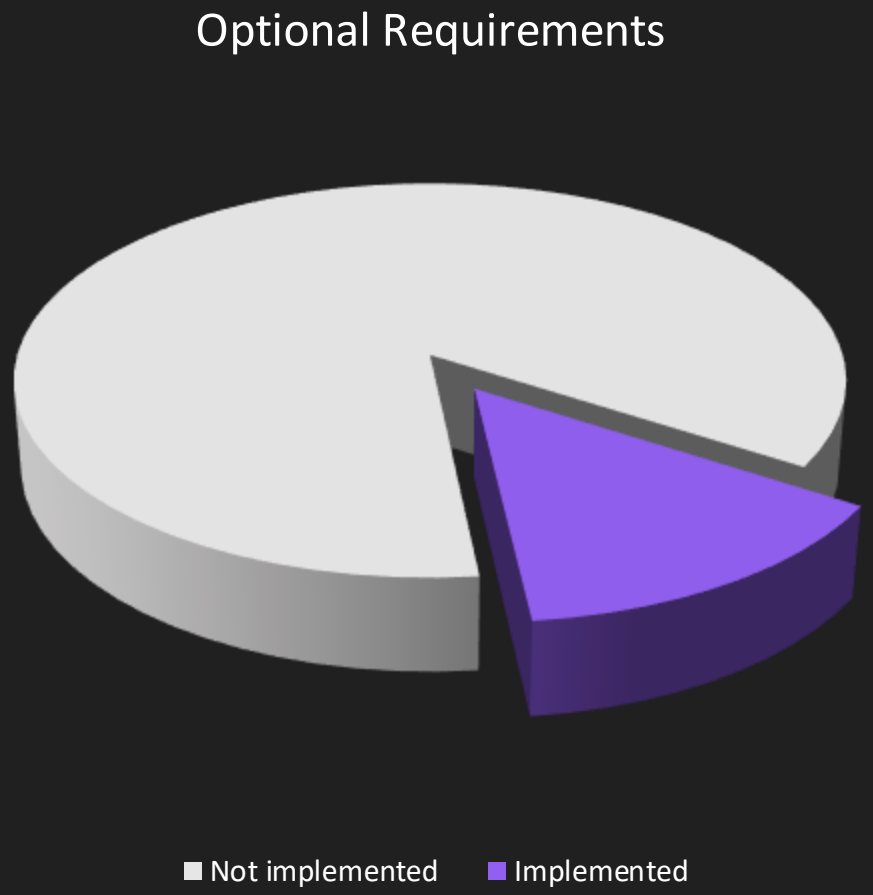
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



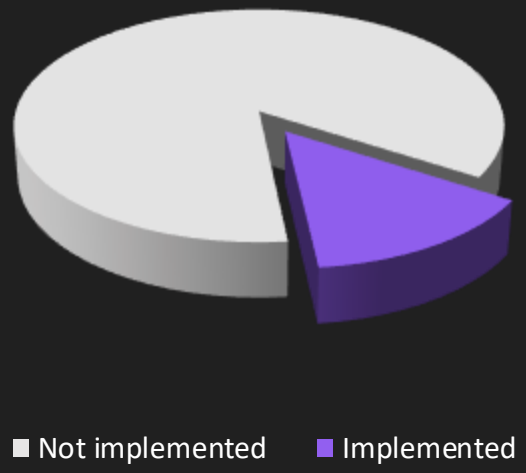
1. User Management
2. Dashboard
3. Create experiment
4. Run experiment
5. Get experiment result

What we implemented



Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

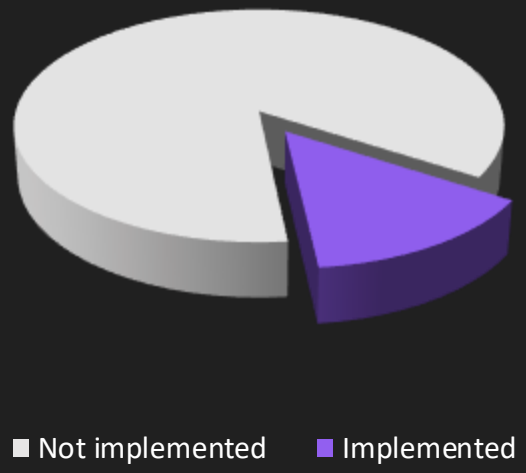
- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

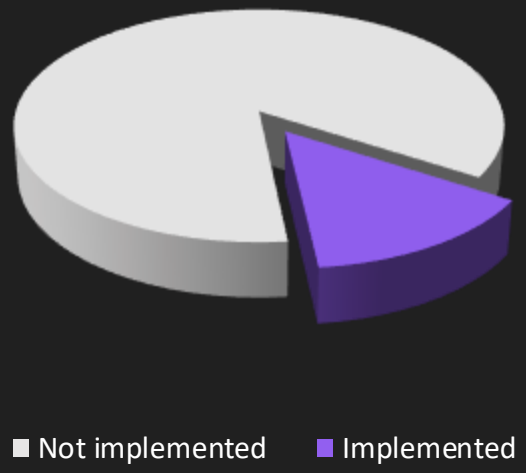
- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

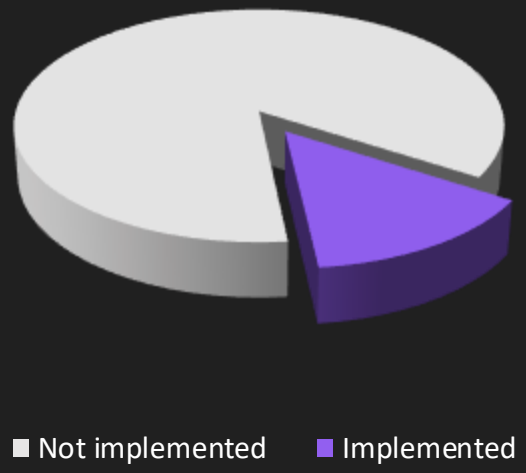
- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

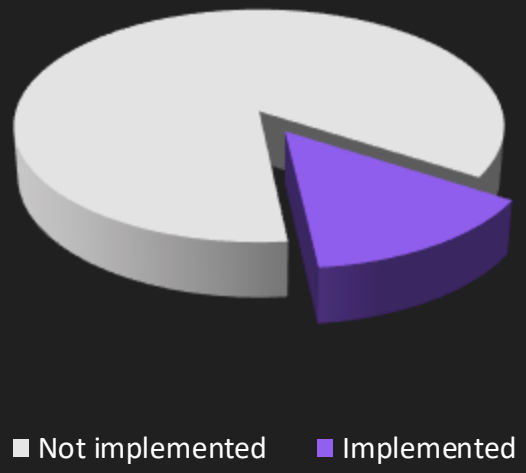
- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

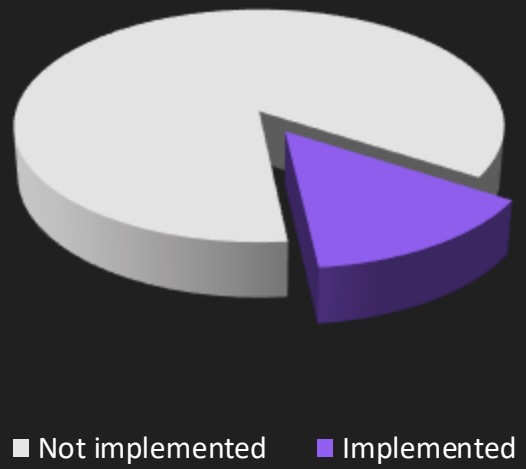
- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Implemented

- RO7: Filter/Search on dashboard
- RO8: Sort on dashboard
- RO13: Nested subspace logic
- RO18: No ground-truth file needed
- RO22: Run experiments concurrently



Not implemented

- RO1: Delete account
- RO2/3: Register/Login with email
- RO4: Share experiment result
- RO6: Forgot password feature
- RO9: Loading animations

…

- RO24: Show ROC curves for experiment

…

- R027: Compare experiments
- RO28: Light theme

Effective subspace outlier analysis

Sign Up

Try it out

Effective subspace outlier analysis

Sign up Try it out


Outlr.
Create experiment
U
Signed in as
Underleaver
Account
Log out
Outlr.
Effective subspace outlier analysis
Navigate to dashboard

Effective subspace outlier analysis

Navigate to dashboard

Outlr.

Dashboard

Create experiment

U

Click one experiment to select it and drag to select multiple experiments.

Refresh | Delete | Compare | All Time

Download CSV

Search | Clear

Start time

Experiment Name	Dataset	ODM	Hyperparameter	Start time	Accuracy
Experiment 1	Exampleset	Example ODM	Parameters	5 seconds ago	87,67%
Experiment 2	Exampleset	Example ODM		20 seconds ago	87,67%
Experiment 3	Exampleset	Example ODM		5 minutes ago	87,67%
Experiment 4	Exampleset	Example ODM		20 minutes ago	87,67%
Experiment 5	Exampleset	Example ODM		1 hour ago	87,67%
Experiment 6	Exampleset	Example ODM		2 hours ago	87,67%
Experiment 7	Exampleset	Example ODM		1 day ago	87,67%
Experiment 8	Exampleset	Example ODM		3 days ago	87,67%

Search...

Clear

Name	Dataset	ODM	Hyperparameters	Date	Accuracy
new Experiment	dataset.csv	ABOD		13.2.2023, 18:56:10	96.5%
new Experiment	dataset.csv	ABOD		13.2.2023, 18:55:04	96.5%
new Exzjrjhgperiment	dataset.csv	ABOD	method: fast,	13.2.2023, 18:52:13	98%

Experiment #378

Dataset Optional name

dataset1.csv

Ground-truth

ground-truth.csv

Preview | Summary

Identifier	Column 1	Column 2	Column 3	Column 3	Column 3
0	0	5017	198746	436566	344
1	1	2745	987234	3403	66135
2	1	358683	13024	304	3761
3	1	25627	43022	44024	7
4	0	8432	344211	6788	13416
5	0	6048	8774547	340568	1234
6	0	5950	56354	23	136778
7	0	29092	554888	1	6597
8	1	15010	99987	266777	46874

Outlier Detection Method

LUNAR ›

ODM Parameters

model type | neighbours | sampling | val size | scaler

epsilon | proportion | epochs | lr | wd | verbose

Subspace logic

```
[0 .. 4] and [4 .. 7]
```

Run experiment

new Experiment

Upload Files

Dataset

Durchsuchen... Keine Datei ausgewählt.

Groundtruth

Durchsuchen... Keine Datei ausgewählt.

Create Experiment

Outlier Detection Method

GMM

ODM Parameters

n_components	covariance_typ	tol	reg_covar
max_iter	n_init	init_params	weights_init
means_init	precisions_init	random_state	warm_start
contamination			

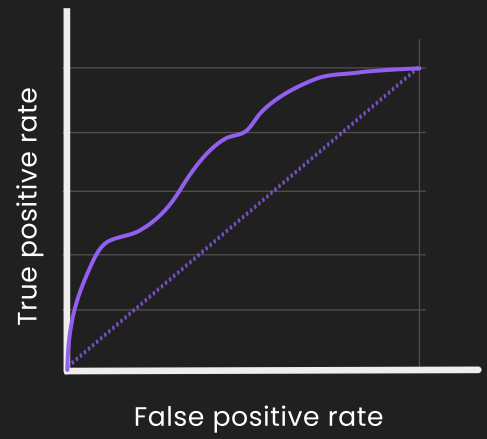
Subspace Logic

Enter Subspace Logic

23/70

Summary

ODM:	LUNAR
Accuracy:	98.643 %
Execution date:	20:36:42 23/11/2022
Execution time:	3d 4m 11.25s
Number of detected outliers:	735



Download CSV

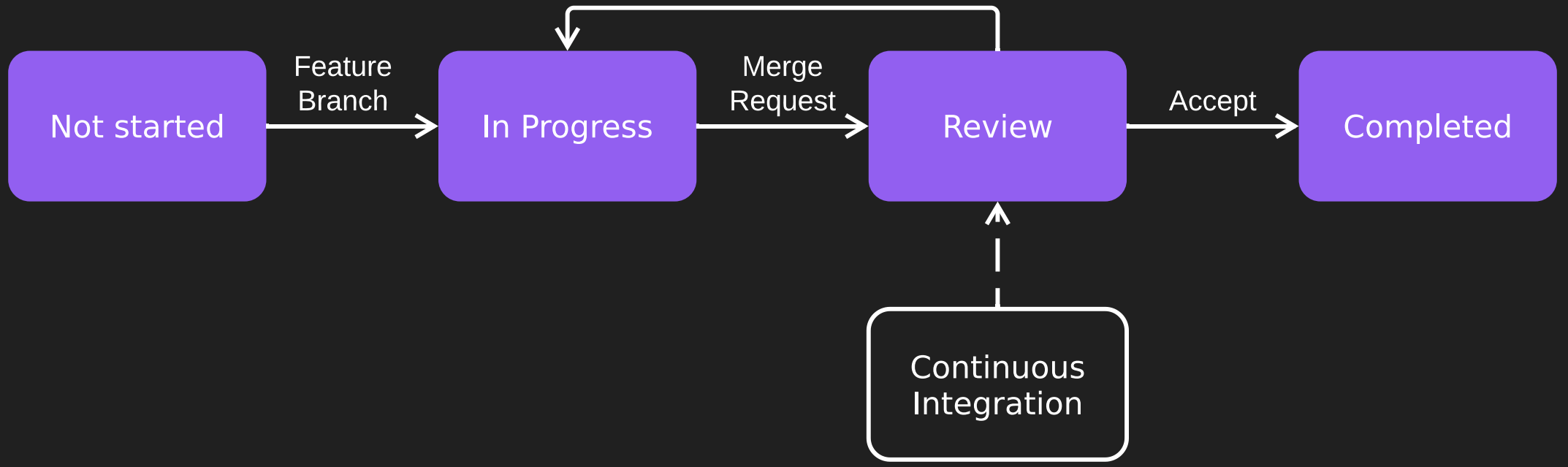
Identifier	Results	Supspace 2	Subspace 3	Subspace 4	Subspace 5	Subspace 6
32	x	x	x	x		
66	x	x	x		x	
76	x	x	x		x	
267			x		x	
420	x	x	x			
999	x		x			
1234		x	x			x
2048			x			x
2222	x	x	x			x

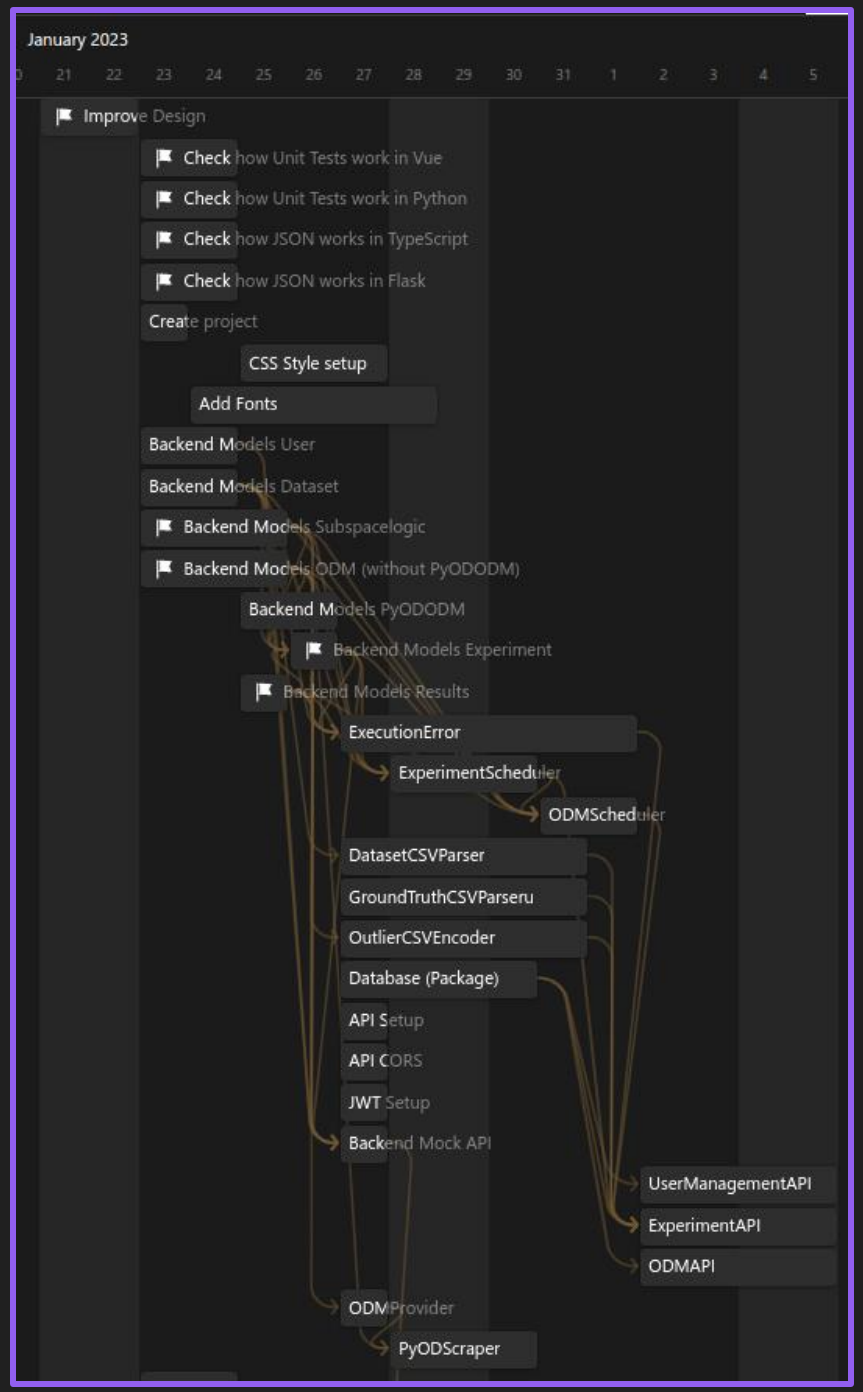
Summary

Download

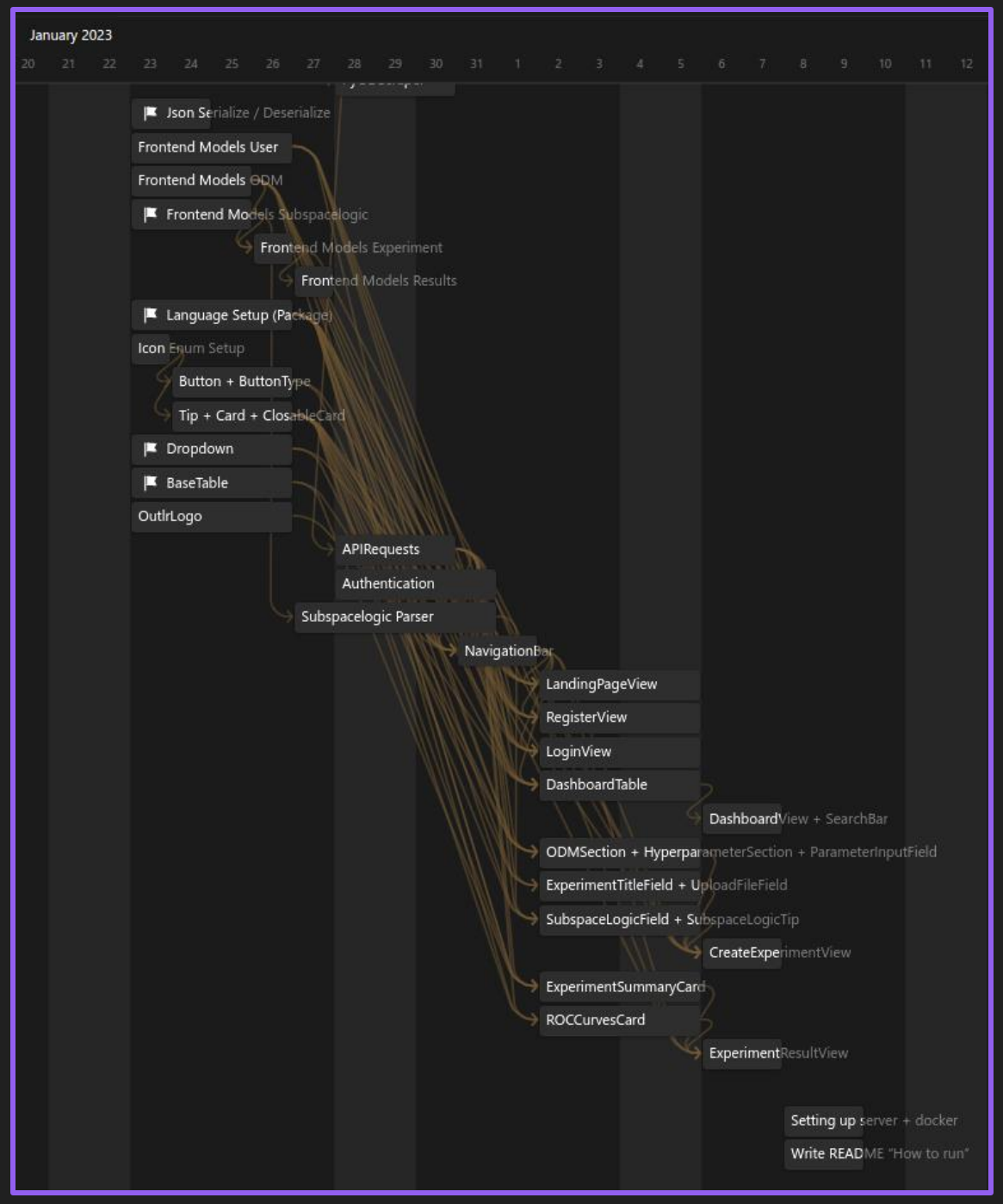
ODM:	ABOD
Accuracy:	98%
Execution Date:	13.2.2023, 18:52:13
Execution Time:	20s 82ms 234µs
Number of detected outliers:	20

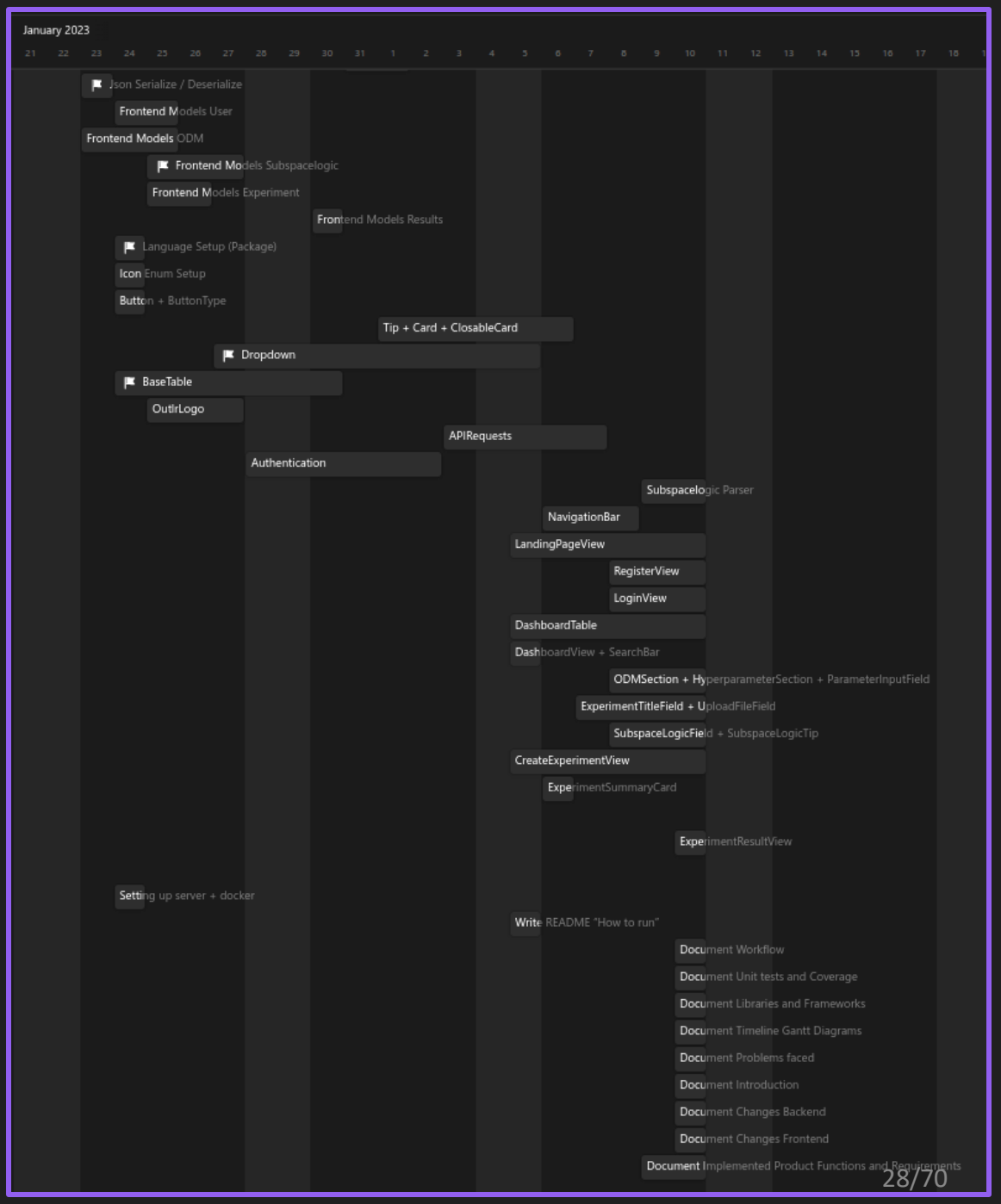
Workflow


Not started
Feature Branch
In Progress
Merge Request
Review
Accept
Completed
Continuous Integration


January 2023
Improve Design
Check how Unit Tests work in Vue
Check how Unit Tests work in Python
Check how JSON works in TypeScript
Check how JSON works in Flask
Create project
CSS Style setup
Add Fonts
Backend Models User
Backend Models Dataset
Backend Models Subspacelogic
Backend Models ODM (without PyODODM)
Backend Models PyODODM
Backend Models Experiment
Backend Models Results
ExecutionError
ExperimentScheduler
ODMScheduler
DatasetCSVParser
GroundTruthCSVParseru
OutlierCSVEncoder
Database (Package)
API Setup
API CORS
JWT Setup
Backend Mock API
UserManagementAPI
ExperimentAPI
ODMAPI
ODMProvider
PyODScraper


January 2023
Improve Design
Check how Unit Tests work in Vue
Check how Unit Tests work in Python
Check how JSON works in TypeScript
Check how JSON works in Flask
Create project
CSS Style setup
Add Fonts
Backend Models User
Backend Models Dataset
Backend Models Subspacelogic
Backend Models ODM (without PyODODM)
Backend Models PyODODM
Backend Models Experiment
Backend Models Results
ExecutionError
ExperimentScheduler
ODMScheduler
DatasetCSVParser
GroundTruthCSVParseru
OutlierCSVEncoder
Database (Package)
API Setup
JWT Setup
Backend Mock API
UserManagementAPI
ExperimentAPI
ODMAPI
PyODScraper

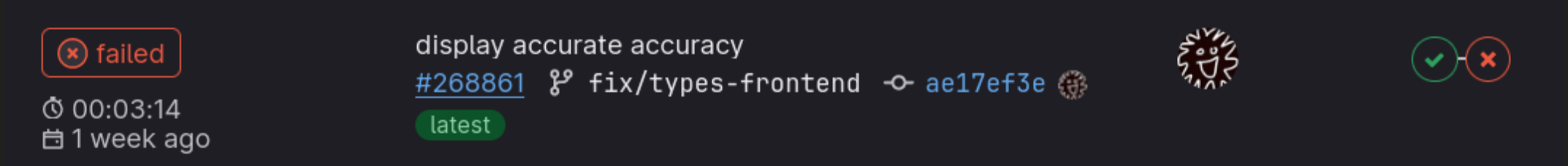

January 2023
20 21 22 23 24 25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12
Json Serialize / Deserialize
Frontend Models User
Frontend Models ODM
Frontend Models Subspacelogic
Frontend Models Experiment
Frontend Models Results
Language Setup (Package)
Icon Enum Setup
Button + ButtonType
Tip + Card + ClosableCard
Dropdown
BaseTable
OutlrLogo
APIRequests
Authentication
Subspacelogic Parser
NavigationBar
LandingPageView
RegisterView
LoginView
DashboardTable
DashboardView + SearchBar
ODMSection + HyperparameterSection + ParameterInputField
ExperimentTitleField + UploadFileField
SubspaceLogicField + SubspaceLogicTip
CreateExperimentView
ExperimentSummaryCard
ROCCurvesCard
ExperimentResultView
Setting up server + docker
Write README "How to run"


January 2023
Json Serialize / Deserialize
Frontend Models User
Frontend Models ODM
Frontend Models Subspacelogic
Frontend Models Experiment
Frontend Models Results
Language Setup (Package)
Icon Enum Setup
Button + ButtonType
Tip + Card + ClosableCard
Dropdown
BaseTable
OutlrLogo
APIRequests
Authentication
Subspacelogic Parser
NavigationBar
LandingPageView
RegisterView
LoginView
DashboardTable
DashboardView + SearchBar
ODMSection + HyperparameterSection + ParameterInputField
ExperimentTitleField + UploadFileField
SubspaceLogicField + SubspaceLogicTip
CreateExperimentView
ExperimentSummaryCard
ExperimentResultView
Setting up server + docker
Write README "How to run"
Document Workflow
Document Unit tests and Coverage
Document Libraries and Frameworks
Document Timeline Gantt Diagrams
Document Problems faced
Document Introduction
Document Changes Backend
Document Changes Frontend
Document Implemented Product Functions and Requirements

Problems Encountered

CI/CD Pipeline

- Late CI setup



- No Continuous Deployment

Defined JSON too late

```
{
  "id": 3,
  "name": "exp1",
  "dataset_name": "name",
  "odm": {"name": "odm1", "id": 1},
  "param_values": {
    "name": "value",
    "name2": "value2"
  },
  "subspace_logic": {}, // all subspaces and their outliers
  "error_json": {...error as json...},
  "experiment_result": {
    "id": 0,
    "accuracy": 0.8,
    "execution_date": "2023-02-05T21:54:02.038308", // ISO 8601 format
    "execution_time": 5 // in µs (microseconds)
    "result_space": {
      "name": "result",
      "colums": [], // Empty columns identify the result space
      "outliers": [1,3,5,9,20]
    }
  }
}
```

Experiment JSON (response)

```
{
  "error": "error-name",
  "messsage": "Explain the error",
  "status": 400
}
```

Error JSON (response)

```
{
  "username": "username",
  "password": "plaintext_password"
}
```

User JSON (request)

```
{
  "username": "username",
  "access_token": "access_token"
}
```

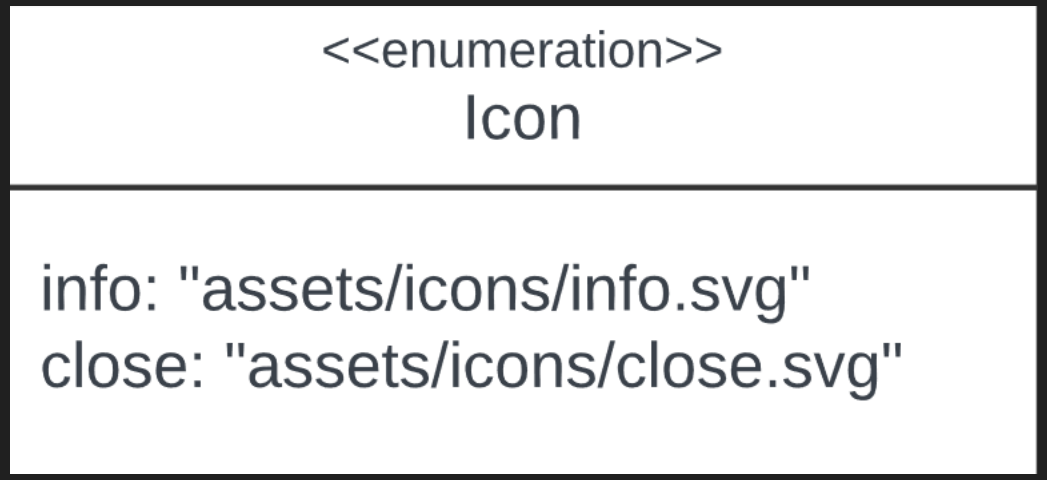
User JSON (response)

Tasks took longer than expected

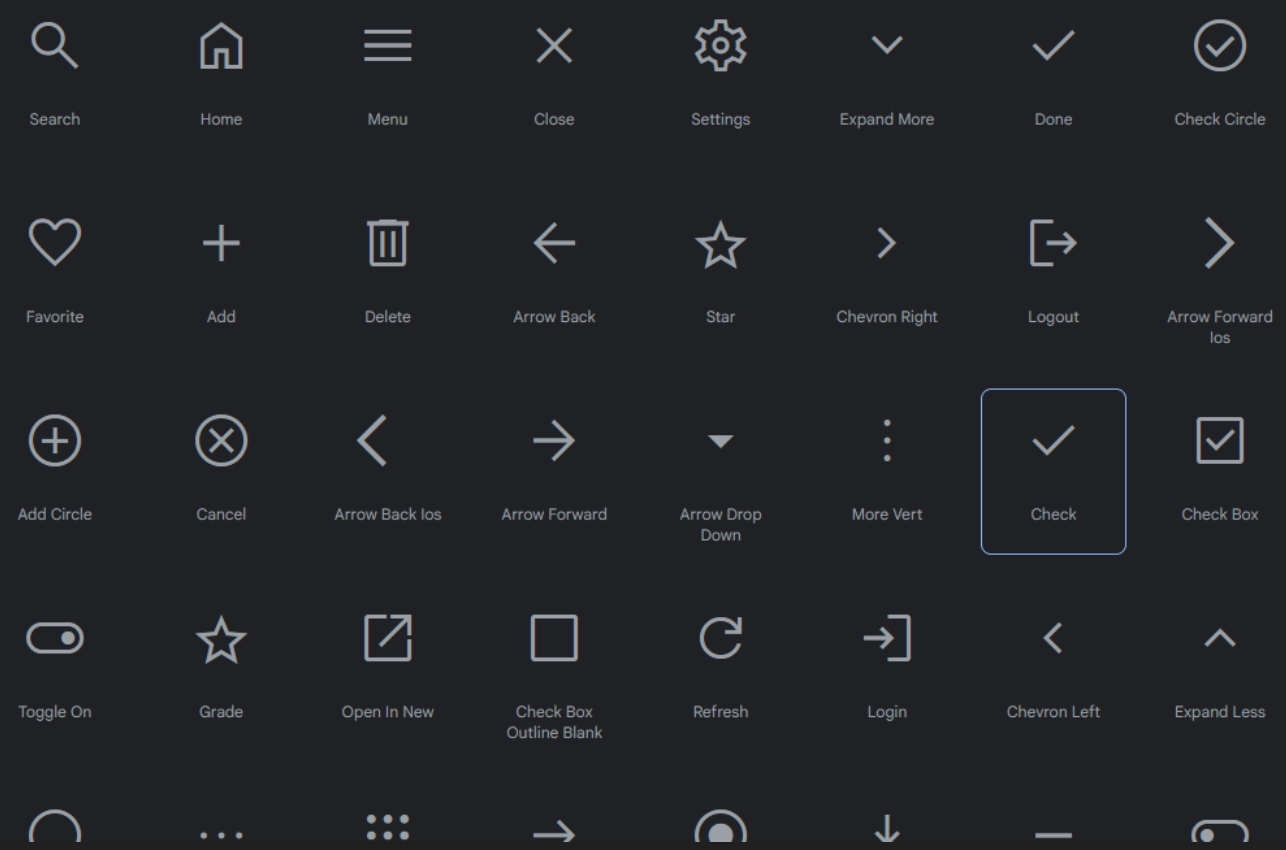
- Cause:
  - Learn Language and Libraries
  - Unexpected complexity

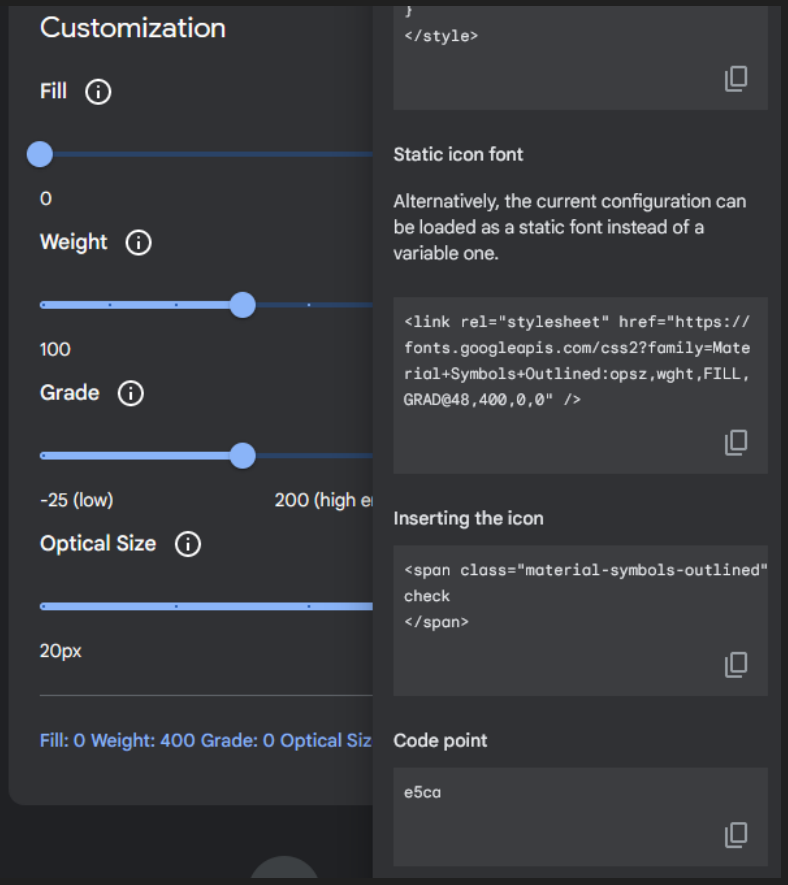
- Effect:
  - Frontend-Backend schedule didn't match
  - Unused Mock API

Changes made - Frontend

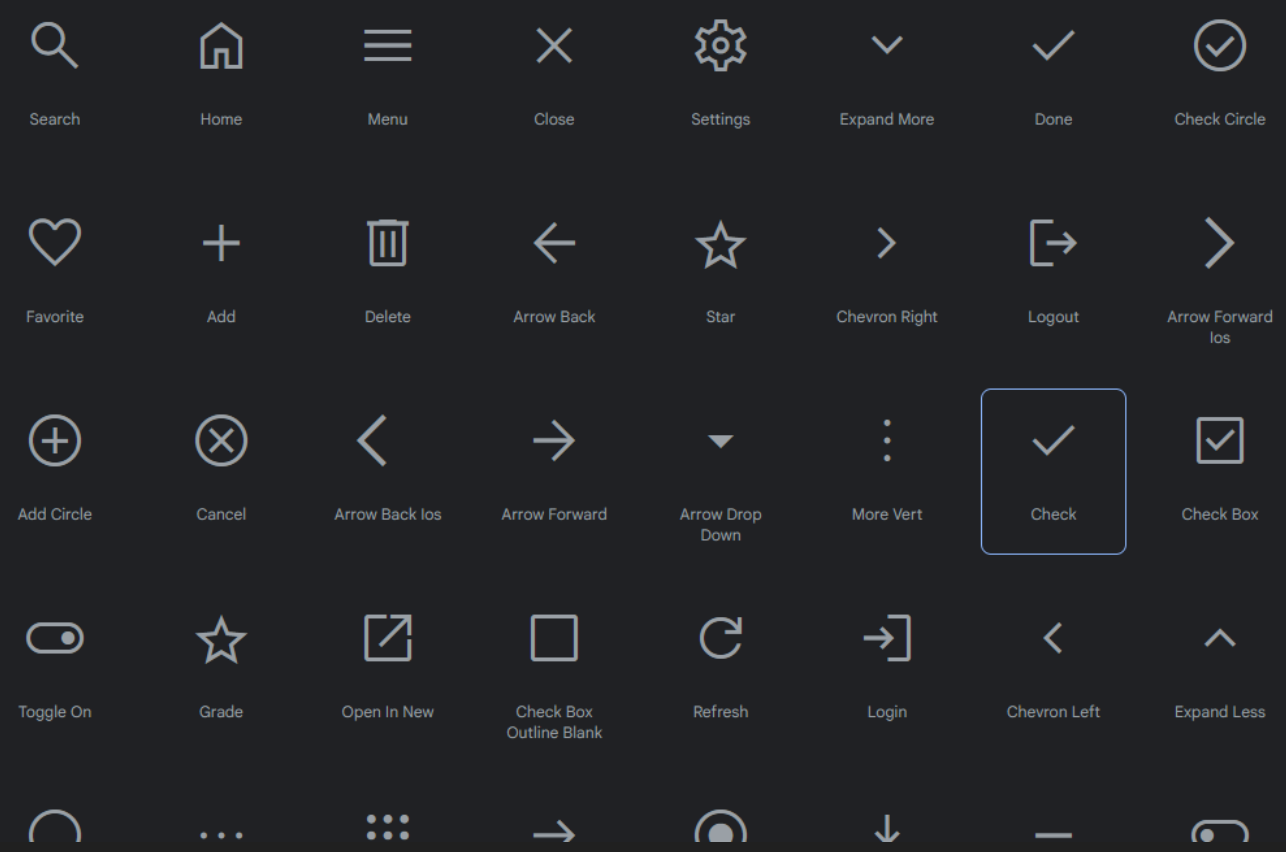


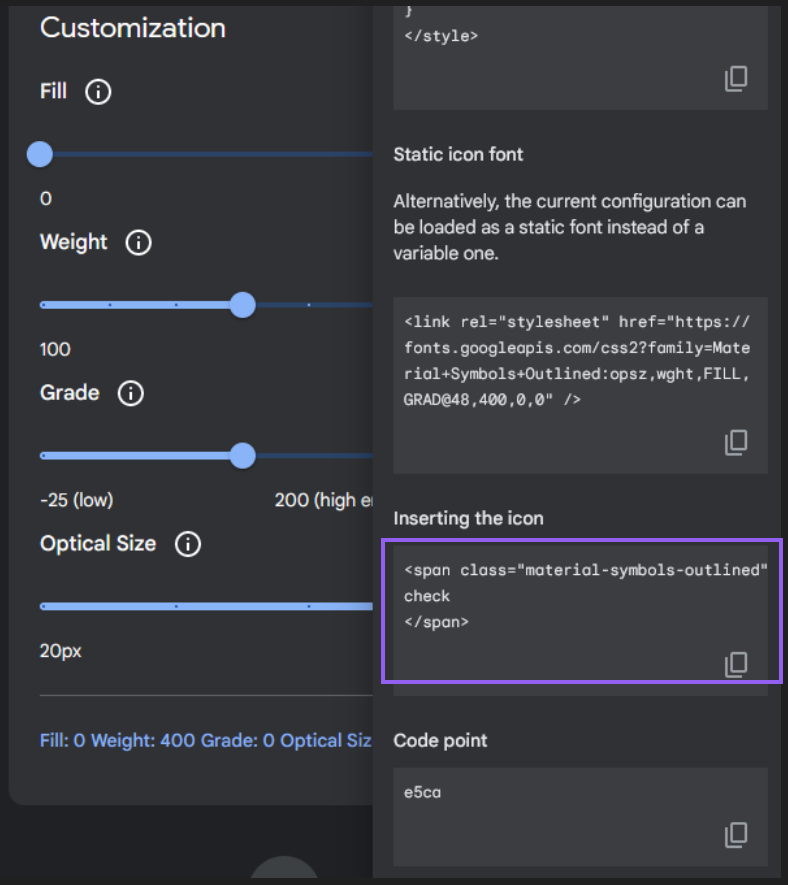
Changes made - Frontend



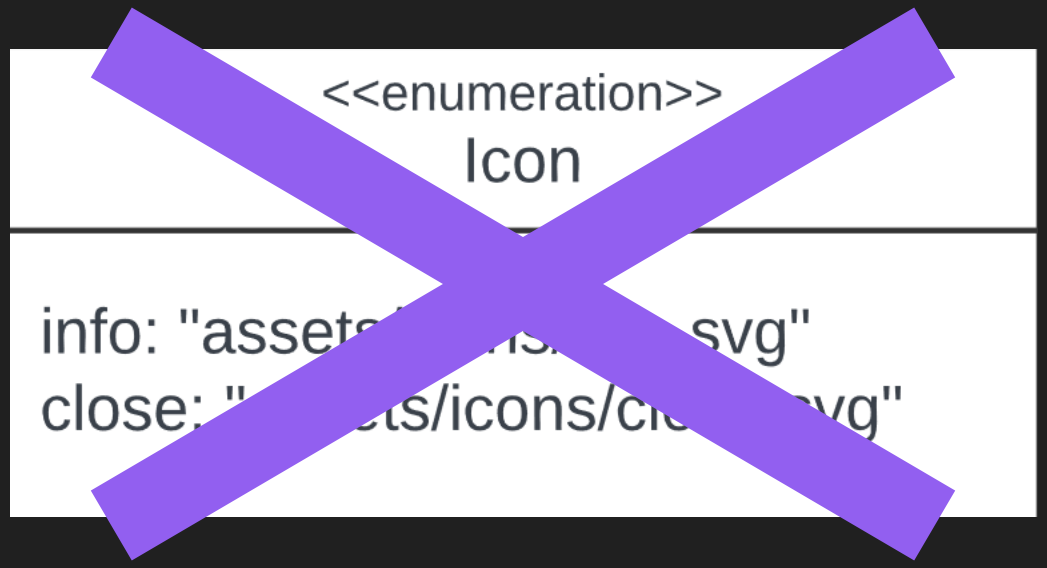


Changes made - Frontend



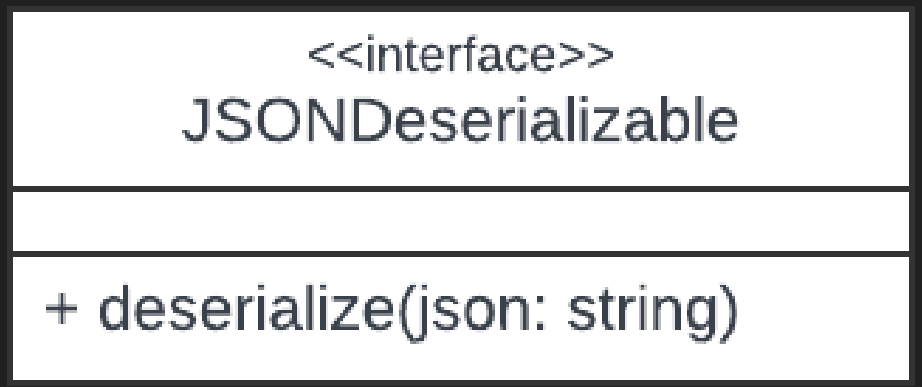


Changes made - Frontend




```vue
<template>
  <button :class="buttonType" :style="style" @click="onClick">
    <div v-if="IconProvided">
      <div style="...">
        <div> {{ text }}</div>
        <div>
          <span class="material-icons md-dark icon" style="font-size: 30px;
          font-weight: 100; color: var(--color-stroke)" > {{startIcon}} </span>
        </div>
    </div>
    </div>
    <div v-else>
      {{ text }}
    </div>
  </button>
</template>
```

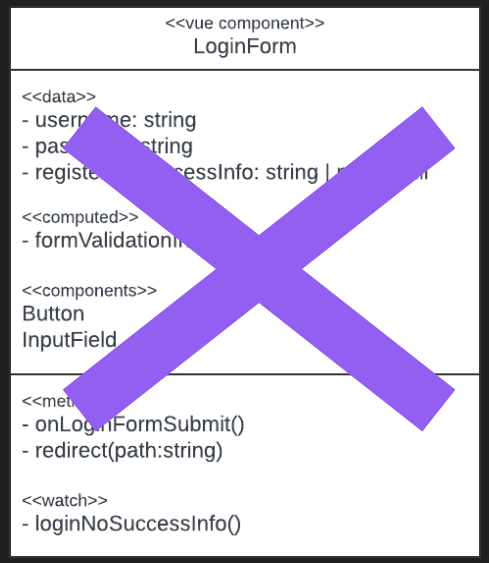
Changes - Frontend

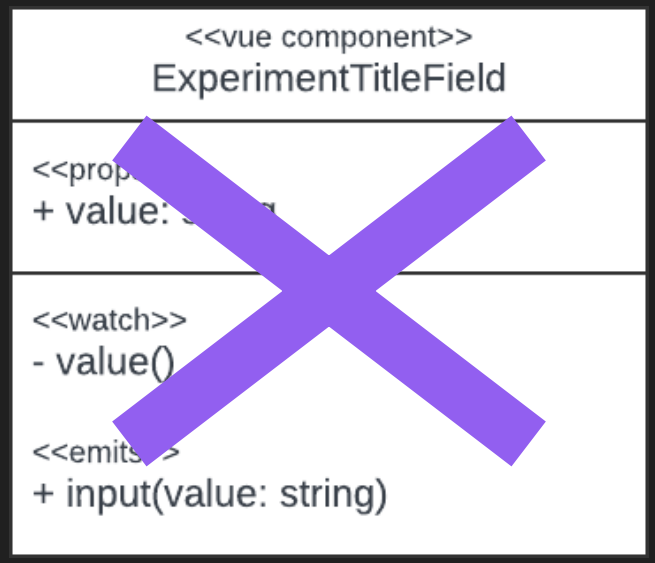



```
 */
static fromJSONObject(jsonObject: any, subspaceMap: Map<number, Subspace>, outlierMap: Map<number, Outlier>): ExperimentResult {
```

```
static fromJSONObject(jsonObject: any, outlierMap: Map<number, Outlier>): Subspace {
```

Changes - Frontend





Changes - Frontend

- Enum for dashboard sorting

```
export enum DashboardSortColumn {
    NAME = "Name",
    DATASET = "Dataset",
    ODM = "ODM",
    HYPERPARAMETER = "Hyperparameter",
    DATE = "Date",
    ACCURACY = "Accuracy",

}

export function getDashboardSortColumnLabel(type: string): DashboardSortColumn {
    switch (type) {
        case "Name":
            return DashboardSortColumn.NAME;
        case "Dataset":
            return DashboardSortColumn.DATASET;
        case "ODM":
            return DashboardSortColumn.ODM;
        case "Hyperparameter":
            return DashboardSortColumn.HYPERPARAMETER;
        case "Date":
            return DashboardSortColumn.DATE;
        case "Accuracy":
            return DashboardSortColumn.ACCURACY;
        default:
            return DashboardSortColumn.NAME;
    }
}
```

Changes – Frontend – API

- Renamed and added some methods

```
export async function initialValidityCheck() : Promise<void> {
```

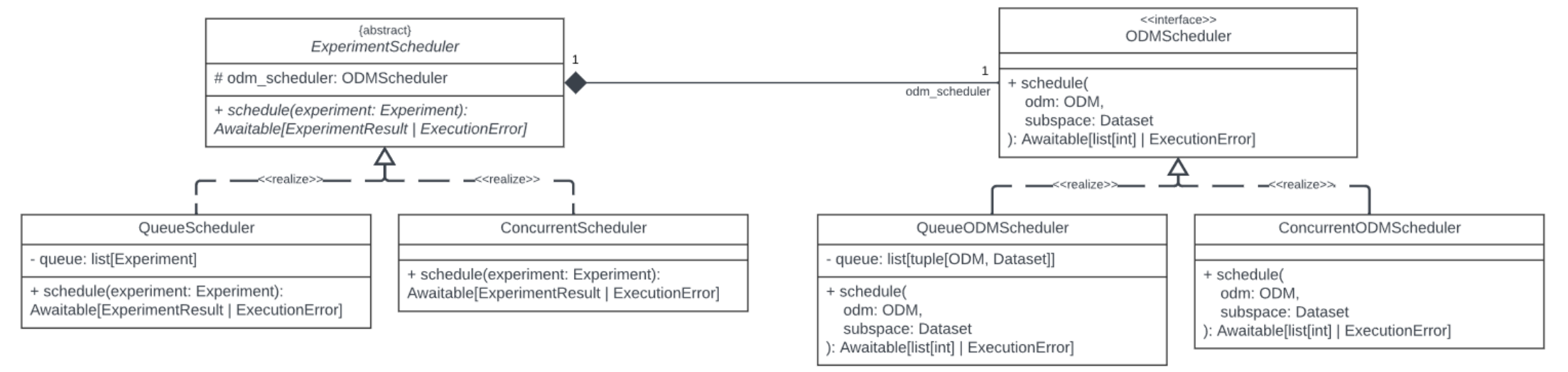
```
export function validatePassword(password:string) {
```

```
export function validateUsername(username : string) : boolean {
```

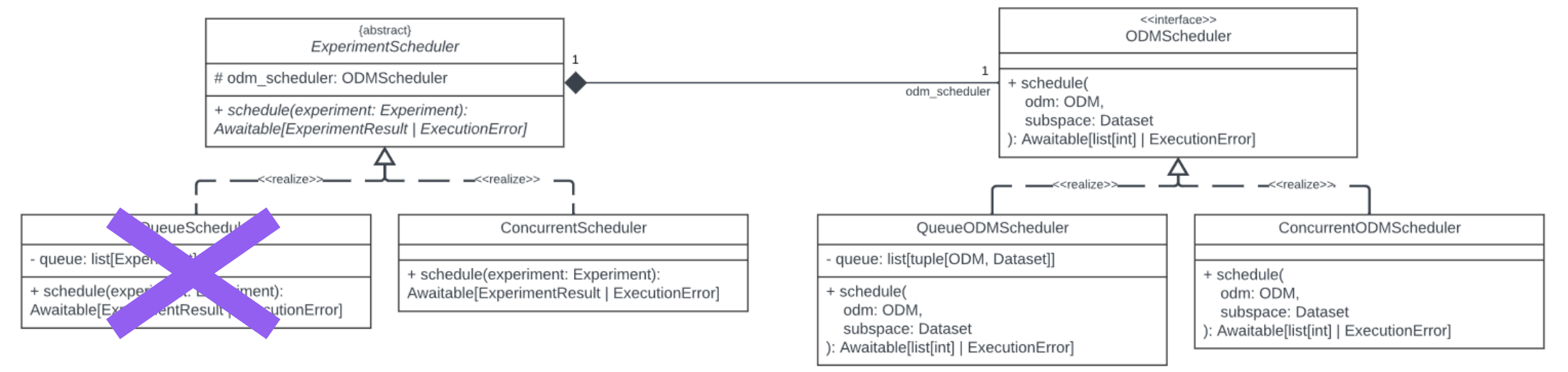
Changes in general - Backend

- np.NDArray were used in place of python lists
- static classes, variables not implemented as classes

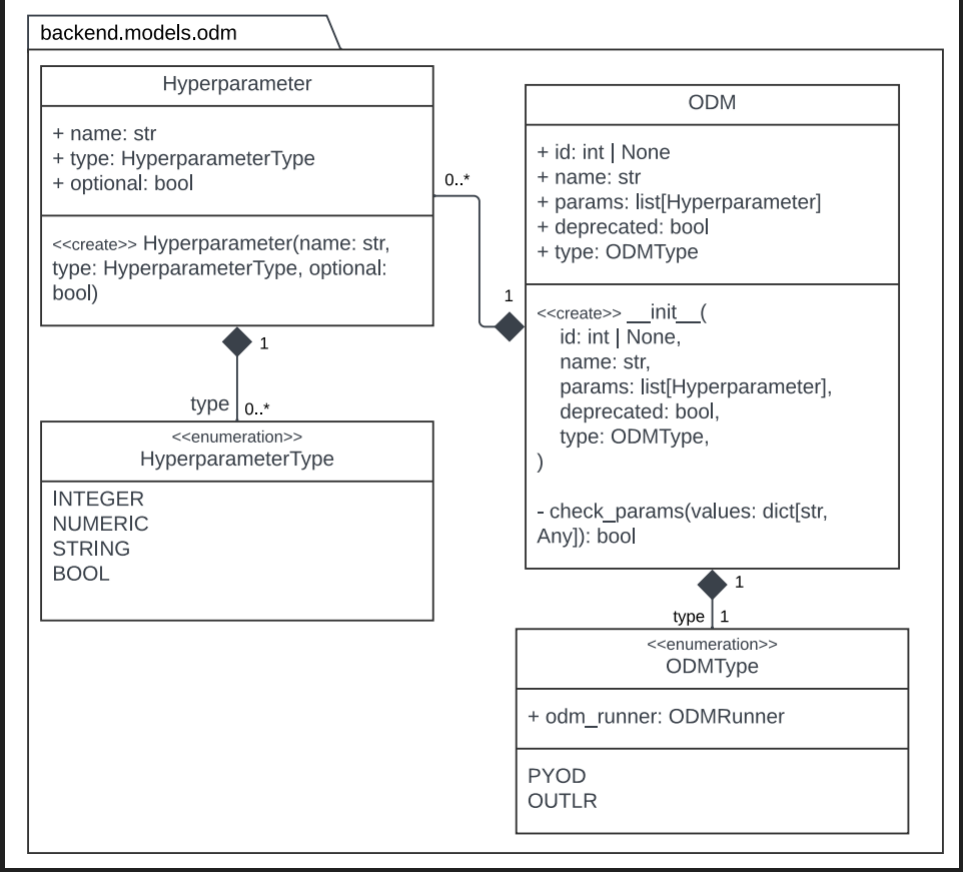
Changes - Backend

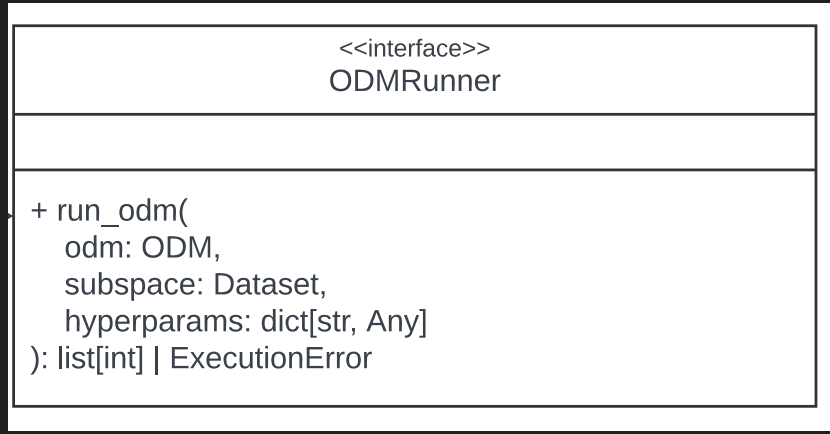


Changes - Backend

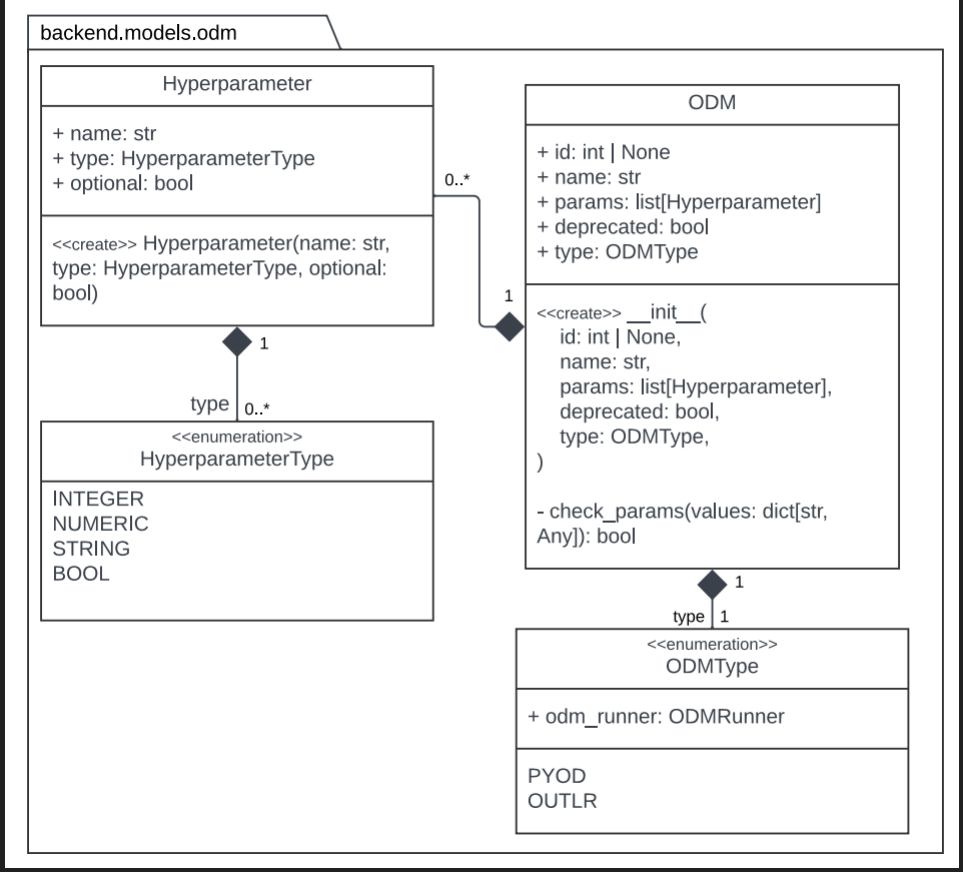


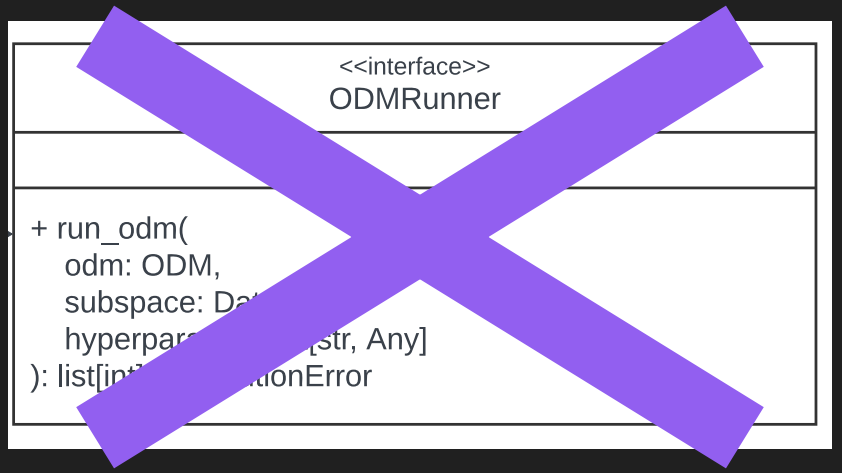
Changes - Backend





Changes - Backend

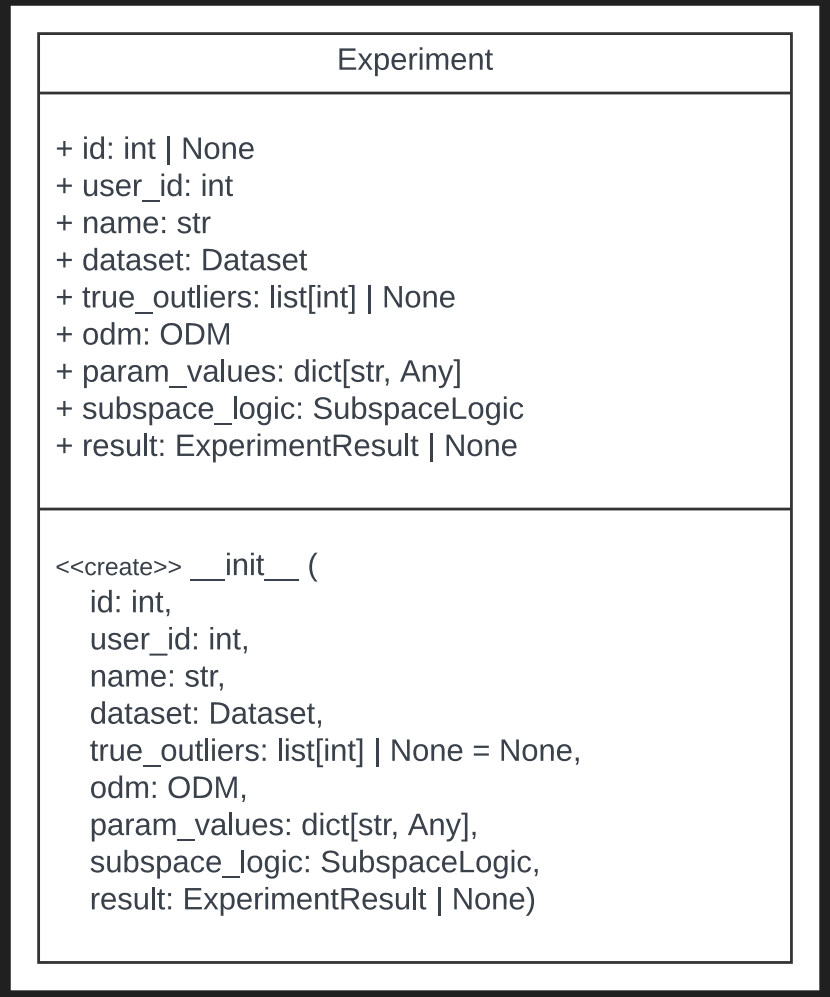




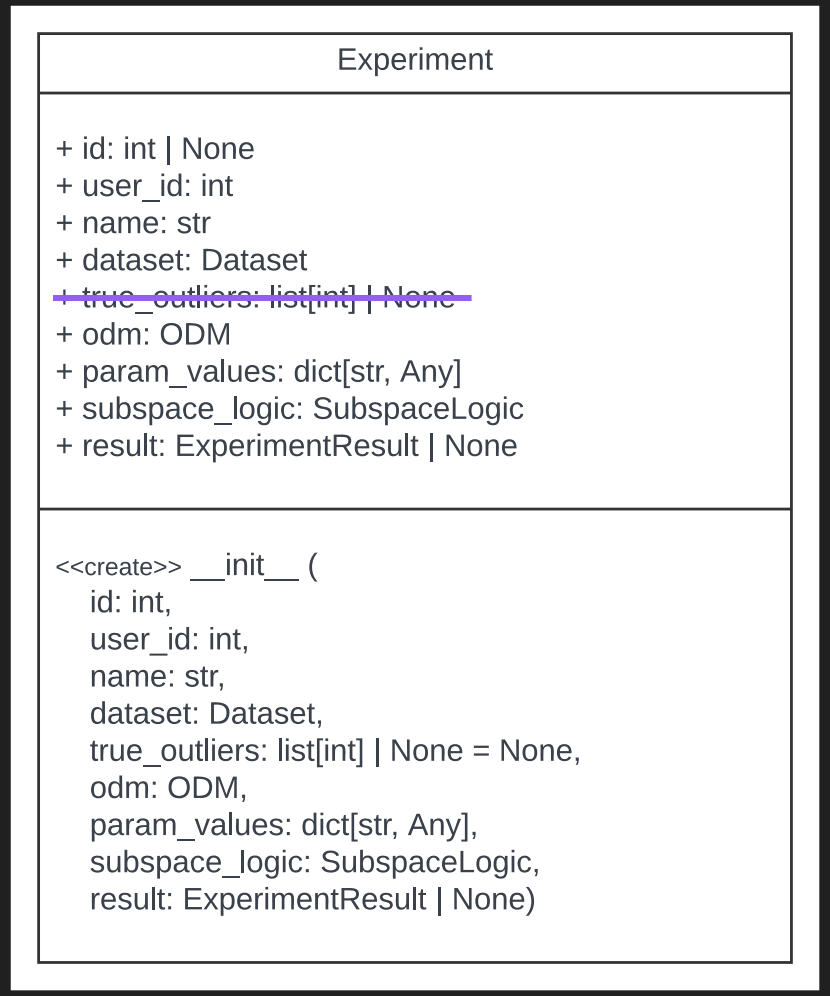
Changes - Backend

```python
class PyODM(ODM):

    def run_odm(self, subspace: pd.DataFrame, hyper_params: dict[str, Any]) -> NDArray:
        """Runs the ODM on the given subspace
        Args:
            subspace (DataFrame): The subspace to run the ODM on
            hyper_params (dict[str, Any]): The hyper parameters for the ODM
        Returns:
            list[int]: The labels for the outliers on this subspace
        """
        pyOD_module_name, cls_name = self.name.split('.')
        module = importlib.import_module(f'pyod.models.{pyOD_module_name}')
        pyODM_cls = getattr(module, cls_name)
        pyODM = pyODM_cls(**hyper_params)
        pyODM.fit(subspace)
        return pyODM.labels_
```

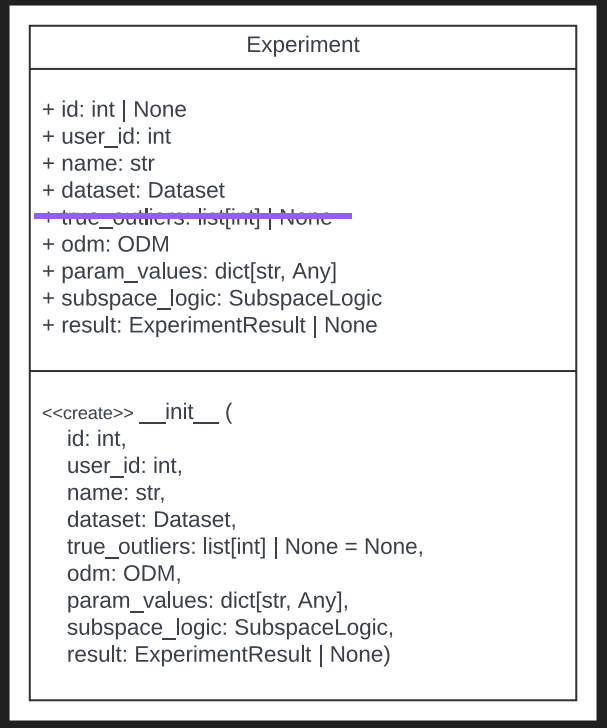
Changes - Backend



Changes - Backend

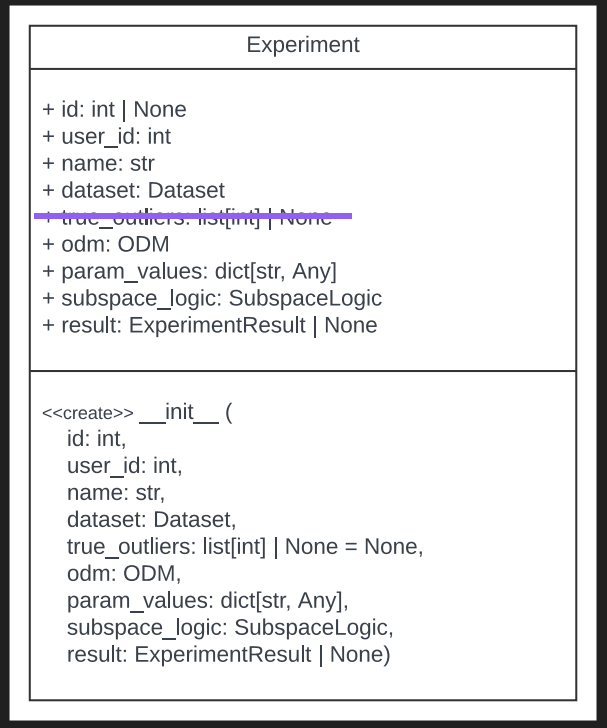


Changes - Backend




```python
class Experiment(Base):

    __tablename__ = EXPERIMENT_TABLE_NAME

    id: Mapped[int] = mapped_column(primary_key=True, autoincrement=True)
    user_id: Mapped[int] = mapped_column(ForeignKey("user.id"), primary_key=True)
    name: Mapped[str]
    param_values = mapped_column(JSON)
    _subspace_logic_json = mapped_column(JSON, nullable=True)
    dataset_name: Mapped[Optional[str]]
    error_json: Mapped[Optional[dict]] = mapped_column(JSON)

    odm_id: Mapped[int] = mapped_column(ForeignKey(ODM.id))
    odm: Mapped['ODM'] = relationship()

    subspaces: Mapped[list['Subspace']] = relationship(
        back_populates="experiment",
        foreign_keys=[Subspace.experiment_id, Subspace.user_id],
        primaryjoin="and_(Experiment.id == Subspace.experiment_id, Experiment.user_id == Subspace.user_id)"
    )

    experiment_result: Mapped["ExperimentResult"] = relationship(
        back_populates="experiment",
        foreign_keys=[ExperimentResult.experiment_id, ExperimentResult.user_id],
        primaryjoin="and_(Experiment.id==ExperimentResult.experiment_id, Experiment.user_id==ExperimentResult.user_id)")

    dataset = None
    _subspace_logic = None
    ground_truth = None
```

Changes - Backend




```python
class Experiment(Base):

    __tablename__ = EXPERIMENT_TABLE_NAME

    id: Mapped[int] = mapped_column(primary_key=True, autoincrement=True)
    user_id: Mapped[int] = mapped_column(ForeignKey("user.id"), primary_key=True)
    name: Mapped[str]
    param_values = mapped_column(JSON)
    _subspace_logic_json = mapped_column(JSON, nullable=True)
    dataset_name: Mapped[Optional[str]]
    error_json: Mapped[Optional[dict]] = mapped_column(JSON)

    odm_id: Mapped[int] = mapped_column(ForeignKey(ODM.id))
    odm: Mapped['ODM'] = relationship()

    subspaces: Mapped[list['Subspace']] = relationship(
        back_populates="experiment",
        foreign_keys=[Subspace.experiment_id, Subspace.user_id],
        primaryjoin="and_(Experiment.id == Subspace.experiment_id, Experiment.user_id == Subspace.user_id)"
    )

    experiment_result: Mapped["ExperimentResult"] = relationship(
        back_populates="experiment",
        foreign_keys=[ExperimentResult.experiment_id, ExperimentResult.user_id],
        primaryjoin="and_(Experiment.id==ExperimentResult.experiment_id, Experiment.user_id==ExperimentResult.user_id)")

    dataset = None
    _subspace_logic = None
    ground_truth = None
```

Changes – Backend

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Changes – Backend

Experiment creation

- Dataset and potentially ground truth (Form data)
- Experiment information passed by user (JSON)

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Changes – Backend

Experiment creation

- Dataset and potentially ground truth (Form data)
- Experiment information passed by user (JSON)

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Changes – Backend

Experiment creation

- Dataset and potentially ground truth (Form data)
- Experiment information passed by user (JSON)

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Changes – Backend

Experiment creation

- Dataset and potentially ground truth (Form data)
- Experiment information passed by user (JSON)

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Changes – Backend

Experiment creation

- Dataset and potentially ground truth (Form data)
- Experiment information passed by user (JSON)

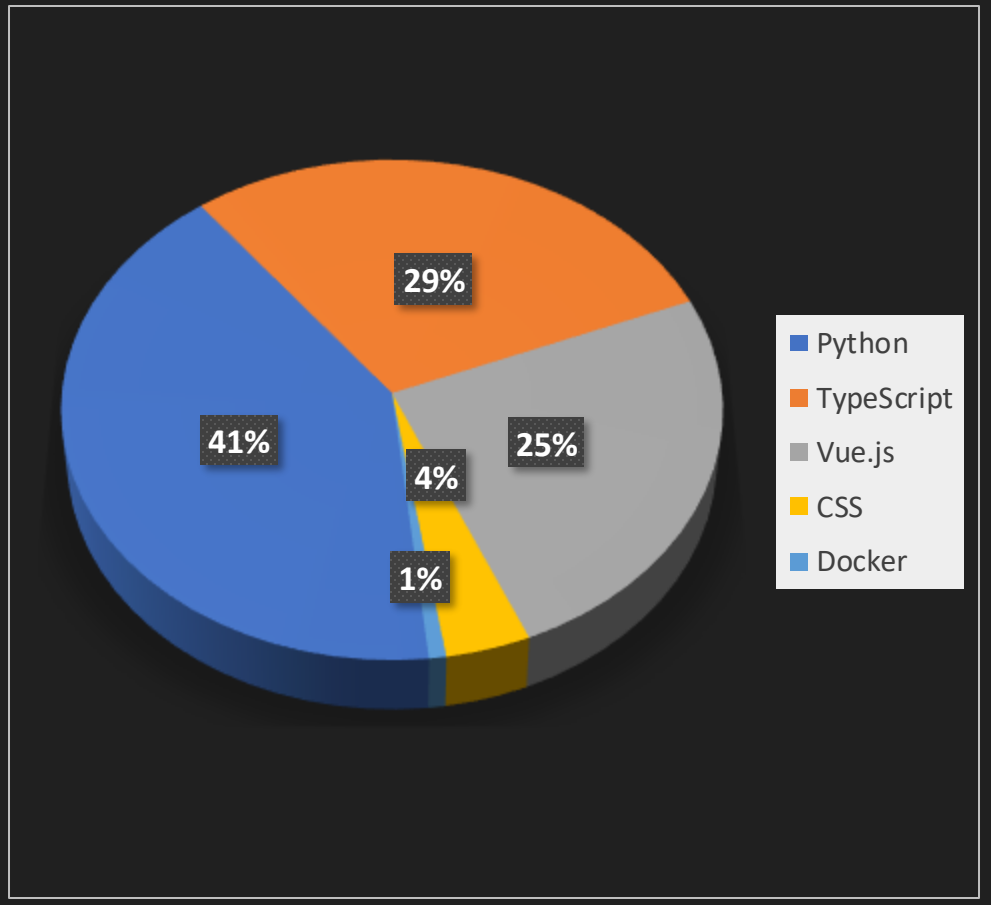
⇒Added /experiment/upload-files to API to upload the .csv files for an experiment

Experiment API:

- /experiment/validate-dataset: Checks whether the specified dataset is valid. Requires an access token.
- /experiment/validate-ground-truth: Checks whether the ground truth file is valid. Requires an access token.
- /experiment/get-result/<int:exp-id>: Returns the result of an experiment if an experiment with id <exp-id> exists and has finished executing. Requires an access token.
- /experiment/get-all: Returns a list of all the experiments the user has run. Requires an access token.
- /experiment/create: Creates and runs a new experiment. Requires an access token.
- /experiment/download-result/<int:exp-id>: Checks if the user has an experiment with the specified <exp-id>. Downloads the outliers with the applied subspace logic as a CSV file. Requires an access token.

Code-Statistics

- Ca. 6400 Lines of Code



DEMO

- Now to the Demon-stration

- Now to the Demon-stration