# ViKER User Manual

## How to start the python server (back-end)

Using the terminal:

```
$ cd Back-end/
$ source venv/bin/activate
$ python WebServer.py
```
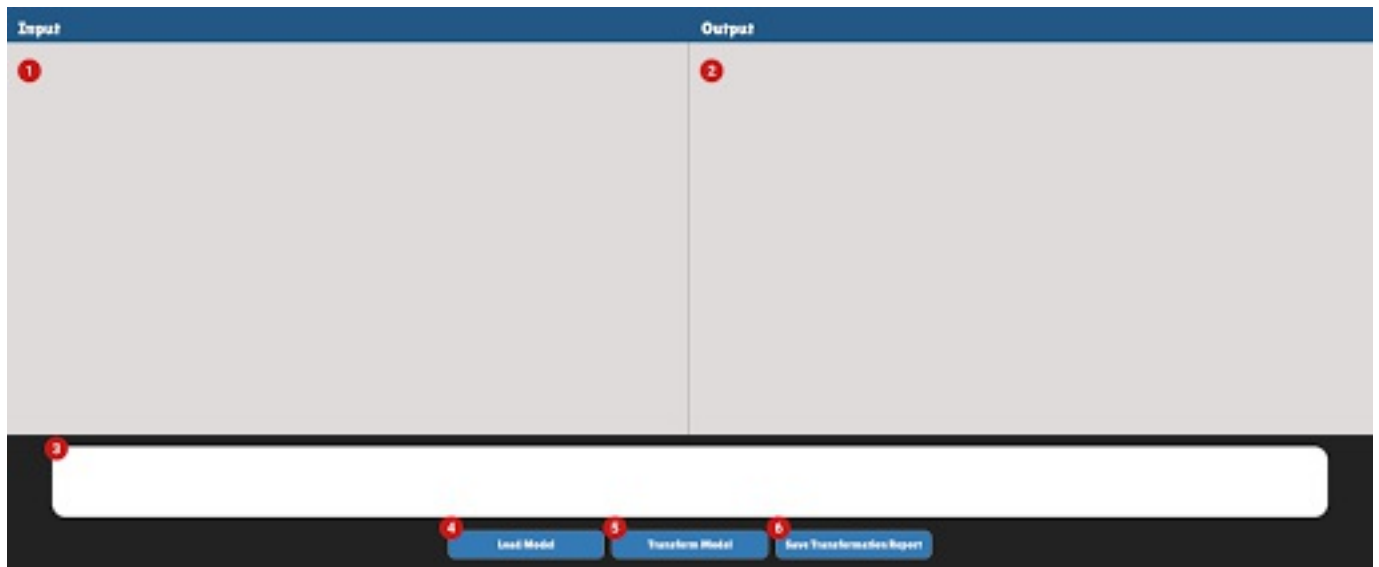
## How to start the node server (front-end)

Using the terminal:

```
$ cd Front-end/
$ npm install
$ npm start
```

Your browser will then open on http://localhost:3000/

## User Interface

1. Where the input model is rendered
2. Where the output model is rendered
3. Where the error log is printed
4. Button to load model as a JSON file
5. Button to send input model to server and transform to output model
6. Saves the output model as a JSON file along with the error log

# What the models look like in JSON

## ARM representation

```
{
"relations": [
    {
        "attributes": [
            {
```

```json
            "AttributeName": "self",
            "dataType": "OID",
            "isConcrete": false,
            "isFK": false,
            "isPathFunctionalDependancy": false
        },
        {
            "AttributeName": "CustomerID",
            "dataType": "Integer",
            "isConcrete": true,
            "isFK": false,
            "isPathFunctionalDependancy": true
        },
        {
            "AttributeName": "CustomerName",
            "dataType": "String",
            "isConcrete": true,
            "isFK": false,
            "isPathFunctionalDependancy": false
        },
        {
            "AttributeName": "CustomerAddress",
            "dataType": "String",
            "isConcrete": true,
            "isFK": false,
            "isPathFunctionalDependancy": false
        },
        {
```

```
                "AttributeName": "CustomerPostalCode",
                "dataType": "Integer",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            }
        ],
        "coveredBy": [],
        "disjointWith": [],
        "inheritsFrom": "none",
        "name": "Customer"
    }
  ]
}
```

# EER representation

```
{
"entities": [
    {
        "attributes": [
            {
                "AttributeName": "CustomerID",
                "composedOf": [],
                "isIdentifier": true,
                "isMultiValued": false
            },
```

```
            {
                "AttributeName": "CustomerName",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "CustomerAddress",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "CustomerPostalCode",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": true,
        "name": "Customer",
        "relationships": []
    }
  ]
}
```
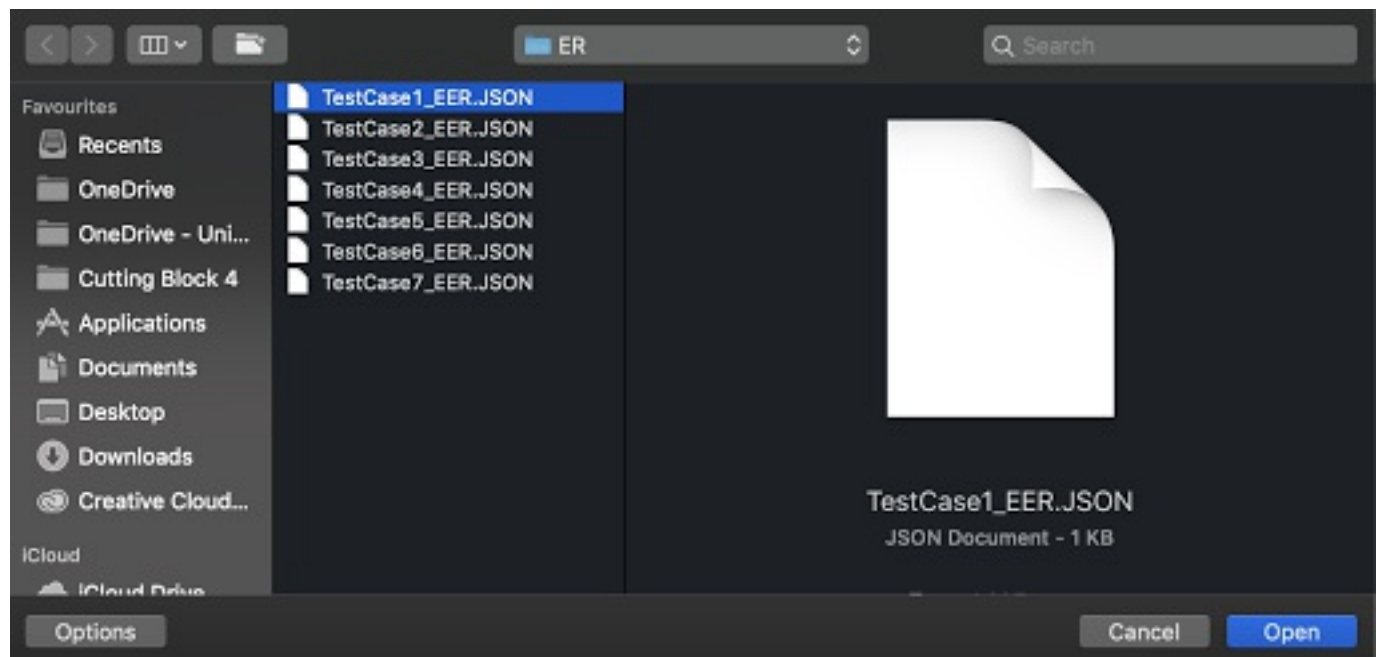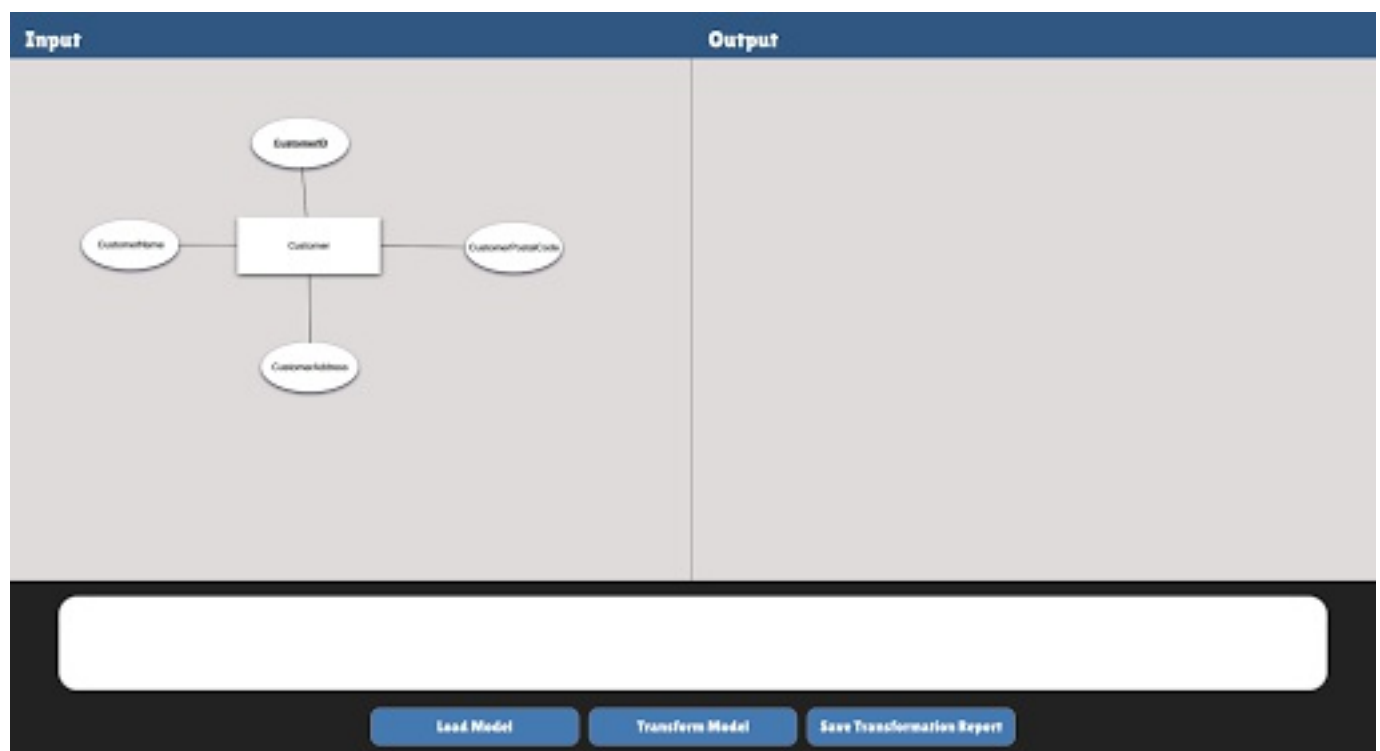
# Loading a model

1.) Click the 'Load Model' button

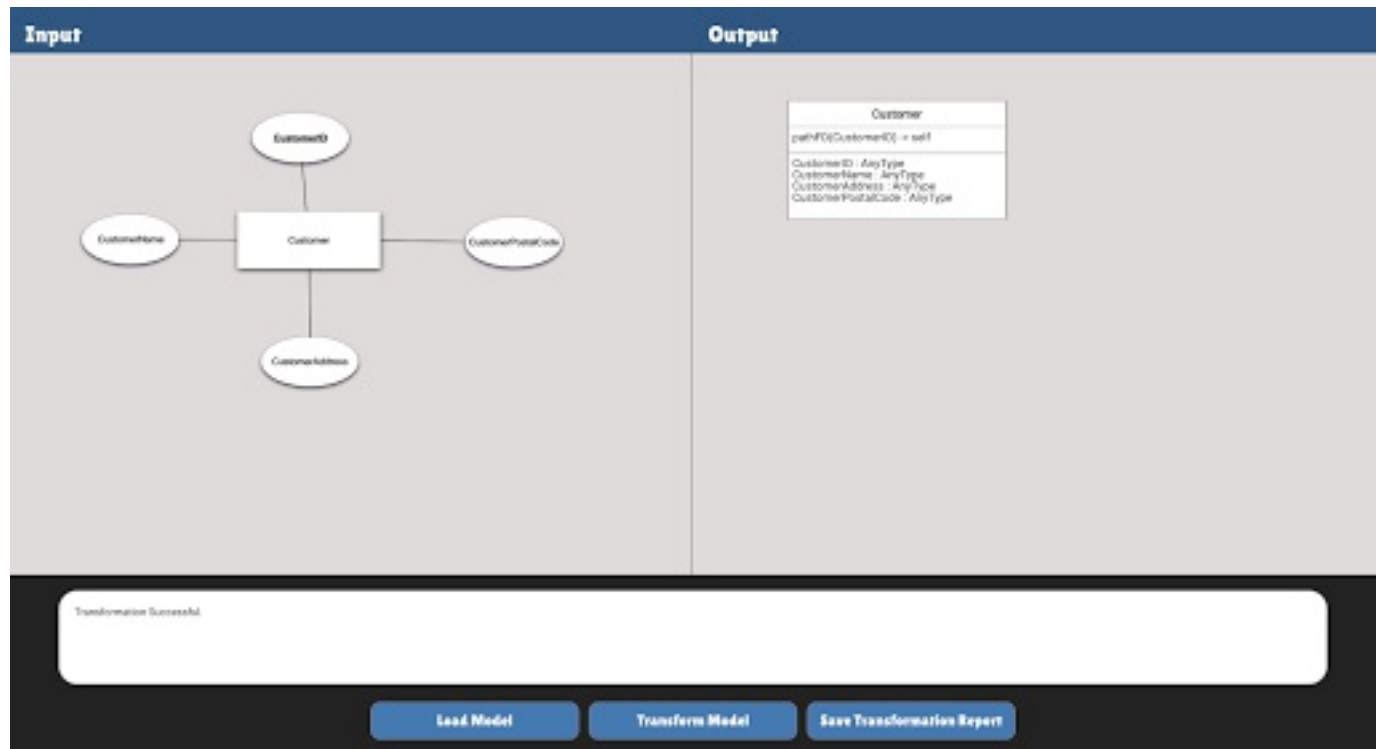2.) The file chooser will open such as below:



3.) The model will then render such as below:



4.) Click the 'Transform Model' button to transform the model and

render the output model, such as below:



5.) You can then save the output model and the error report as JSON, by clicking the 'Save Transformation Report' button. You will see the file being downloaded, such as below:



6.) You can also transform from ARM to EER, the same way!
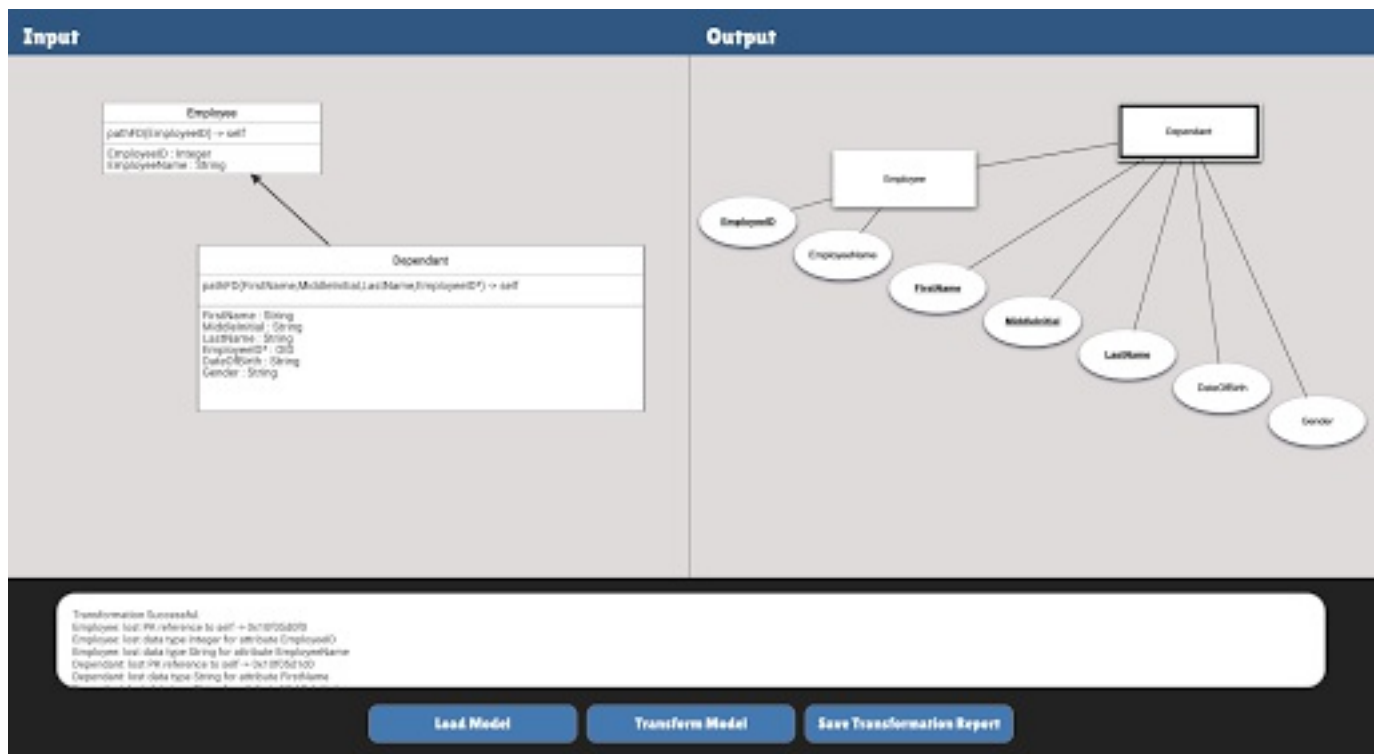
*Note: All diagrams can be moved around and interacted with!

# Examples

## EER

### ISA & Multivalued attribute

#### JSON

```
{
"entities": [
    {
        "attributes": [
            {
                "AttributeName": "EmployeeID",
                "composedOf": [],
```

```json
                "isIdentifier": true,
                "isMultiValued": false
            },
            {
                "AttributeName": "EmployeeName",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "EmployeeType",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "Skill",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": true
            }
        ],
        "isStrong": true,
        "name": "Employee",
        "relationships": [
            {
                "Entity": "HourlyEmployee",
                "RelationTypeForeign": "ISA",
```

```json
                "RelationTypeLocal": "ISA",
                "relationAttributes": []
            },
            {
                "Entity": "SalariedEmployee",
                "RelationTypeForeign": "ISA",
                "RelationTypeLocal": "ISA",
                "relationAttributes": []
            },
            {
                "Entity": "ContractEmployee",
                "RelationTypeForeign": "ISA",
                "RelationTypeLocal": "ISA",
                "relationAttributes": []
            }
        ]
    },
    {
        "attributes": [
            {
                "AttributeName": "HourlyRate",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": false,
        "name": "HourlyEmployee",
```

```json
        "relationships": [
            {
                "Entity": "Employee",
                "RelationTypeForeign": "ISA",
                "RelationTypeLocal": "ISA",
                "relationAttributes": []
            }
        ]
    },
    {
        "attributes": [
            {
                "AttributeName": "AnnualSalary",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": false,
        "name": "SalariedEmployee",
        "relationships": [
            {
                "Entity": "Employee",
                "RelationTypeForeign": "ISA",
                "RelationTypeLocal": "ISA",
                "relationAttributes": []
            }
        ]
```
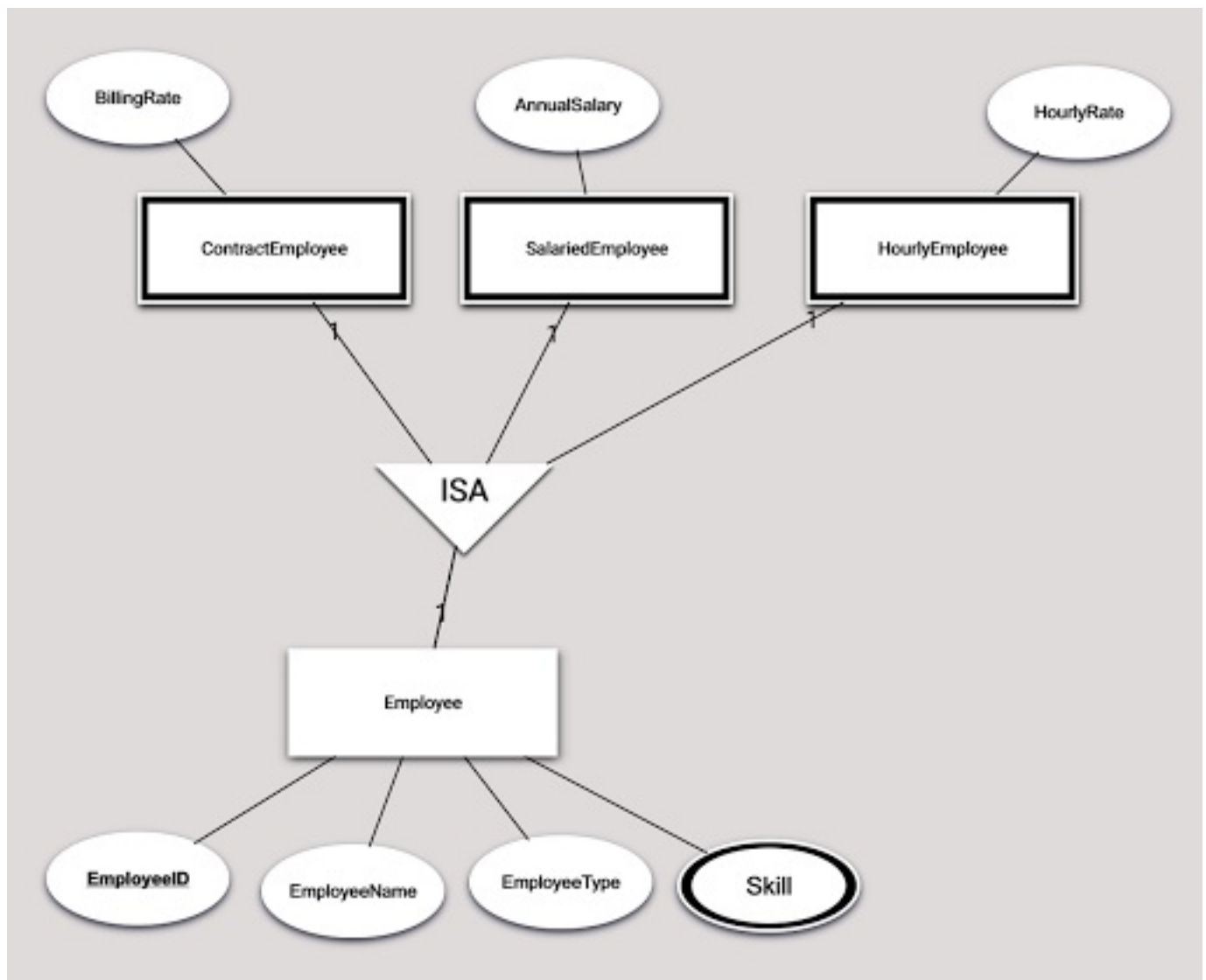
```
        },
        {
            "attributes": [
                {
                    "AttributeName": "BillingRate",
                    "composedOf": [],
                    "isIdentifier": false,
                    "isMultiValued": false
                }
            ],
            "isStrong": false,
            "name": "ContractEmployee",
            "relationships": [
                {
                    "Entity": "Employee",
                    "RelationTypeForeign": "ISA",
                    "RelationTypeLocal": "ISA",
                    "relationAttributes": []
                }
            ]
        }
    ]
}
```

## Model

# Composite attributes

## JSON

```json
{
"entities": [
    {
        "attributes": [
            {
                "AttributeName": "CustomerID",
                "composedOf": [],
                "isIdentifier": true,
```
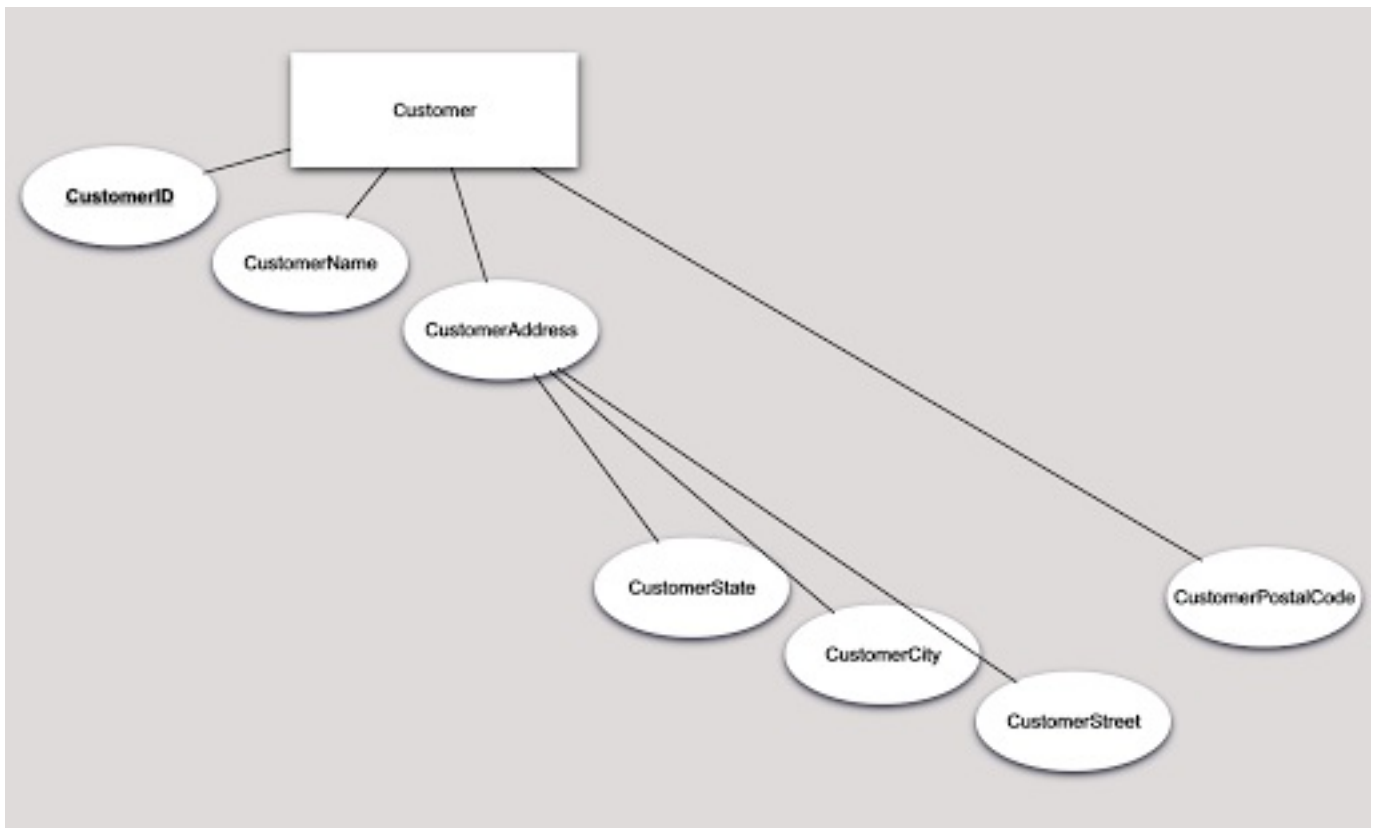
```json
                    "isMultiValued": false
        },
        {
                    "AttributeName": "CustomerName",
                    "composedOf": [],
                    "isIdentifier": false,
                    "isMultiValued": false
        },
        {
                    "AttributeName": "CustomerAddress",
                    "composedOf": [
                        "CustomerState",
                        "CustomerCity",
                        "CustomerStreet"
                    ],
                    "isIdentifier": false,
                    "isMultiValued": false
        },
        {
                    "AttributeName": "CustomerState",
                    "composedOf": [],
                    "isIdentifier": false,
                    "isMultiValued": false
        },
        {
                    "AttributeName": "CustomerCity",
                    "composedOf": [],
                    "isIdentifier": false,
```

```
                "isMultiValued": false
            },
            {
                "AttributeName": "CustomerStreet",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "CustomerPostalCode",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": true,
        "name": "Customer",
        "relationships": []
    }
  ]
}
```

## Model

# Weak entity

## JSON

```json
{
"entities": [
    {
        "attributes": [
            {
                "AttributeName": "DepartmentID",
                "composedOf": [],
                "isIdentifier": true,
                "isMultiValued": false
            },
            {
                "AttributeName": "DepartmentName",
```

```json
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": true,
        "name": "Department",
        "relationships": [
            {
                "Entity": "Course",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ExactlyOne",
                "relationAttributes": []
            },
            {
                "Entity": "Professor",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ExactlyOne",
                "relationAttributes": []
            }
        ]
    },
    {
        "attributes": [
            {
                "AttributeName": "ProfessorID",
                "composedOf": [],
                "isIdentifier": true,
```

```json
                "isMultiValued": false
            },
            {
                "AttributeName": "ProfessorName",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "OfficeNumber",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": true,
        "name": "Professor",
        "relationships": [
            {
                "Entity": "Department",
                "RelationTypeForeign": "ExactlyOne",
                "RelationTypeLocal": "ZeroOrMany",
                "relationAttributes": []
            },
            {
                "Entity": "Class",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ExactlyOne",
```
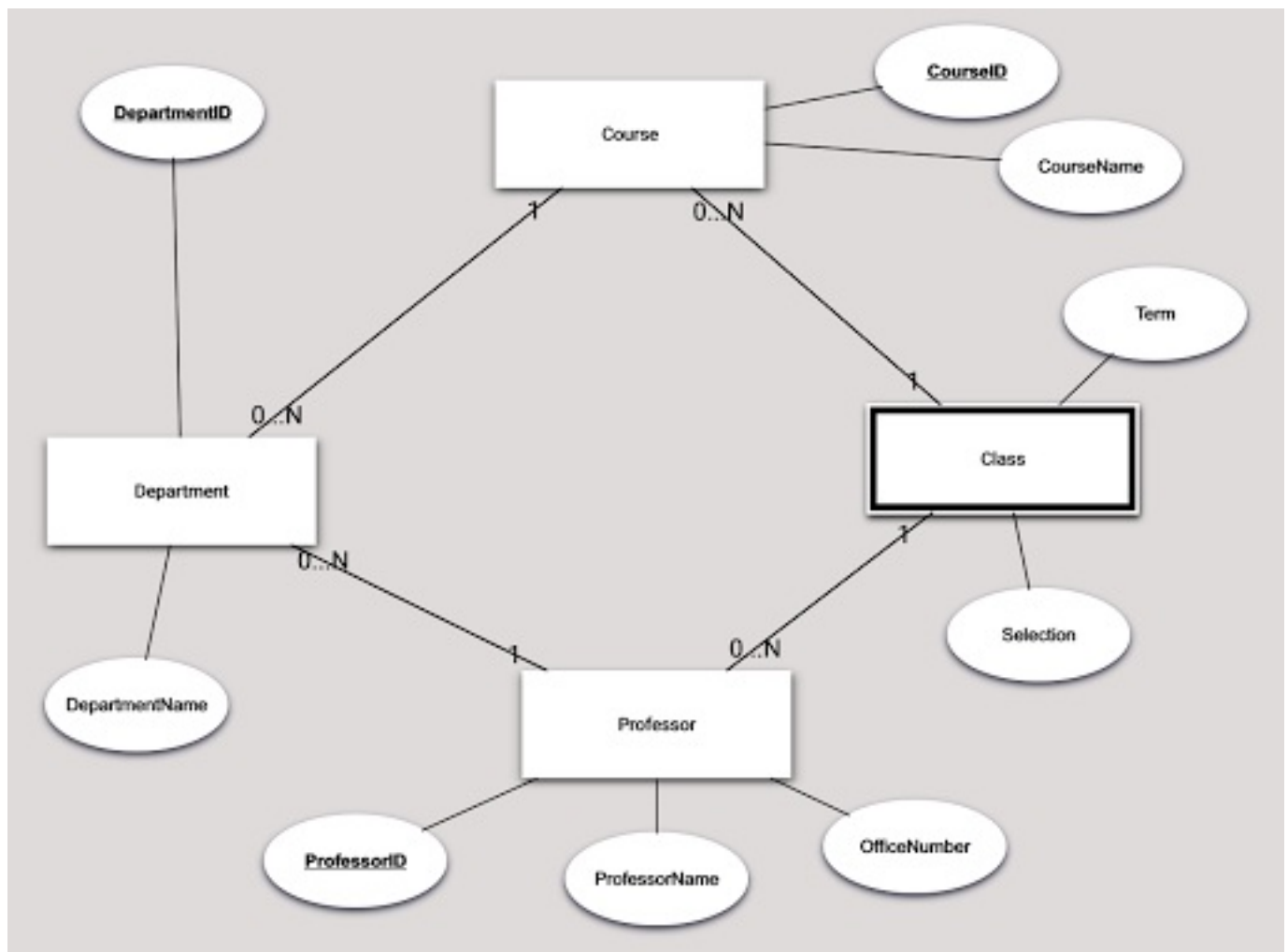
```json
                    "relationAttributes": []
                }
            ]
        },
        {
            "attributes": [
                {
                    "AttributeName": "CourseID",
                    "composedOf": [],
                    "isIdentifier": true,
                    "isMultiValued": false
                },
                {
                    "AttributeName": "CourseName",
                    "composedOf": [],
                    "isIdentifier": false,
                    "isMultiValued": false
                }
            ],
            "isStrong": true,
            "name": "Course",
            "relationships": [
                {
                    "Entity": "Department",
                    "RelationTypeForeign": "ExactlyOne",
                    "RelationTypeLocal": "ZeroOrMany",
                    "relationAttributes": []
                },
```

```json
            {
                "Entity": "Class",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ExactlyOne",
                "relationAttributes": []
            }
        ]
    },
    {
        "attributes": [
            {
                "AttributeName": "Selection",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            },
            {
                "AttributeName": "Term",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": false,
        "name": "Class",
        "relationships": [
            {
                "Entity": "Course",
```

```
                "RelationTypeForeign": "ExactlyOne",
                "RelationTypeLocal": "ZeroOrMany",
                "relationAttributes": []
            },
            {
                "Entity": "Professor",
                "RelationTypeForeign": "ExactlyOne",
                "RelationTypeLocal": "ZeroOrMany",
                "relationAttributes": []
            }
        ]
    }
    ]
}
```

## Model

# Relationship attribute

## JSON

```json
{
"entities": [
    {
        "attributes": [
            {
                "AttributeName": "EmployeeID",
                "composedOf": [],
                "isIdentifier": true,
                "isMultiValued": false
```

```json
        },
        {
                "AttributeName": "EmployeeName",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
        },
        {
                "AttributeName": "EmployeeBirthDate",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
        }
    ],
    "isStrong": true,
    "name": "Employee",
    "relationships": [
        {
                "Entity": "Course",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ZeroOrMany",
                "relationAttributes": [
                    "DateCompleted"
                ]
        }
    ]
},
{
```

```json
        "attributes": [
            {
                "AttributeName": "CourseID",
                "composedOf": [],
                "isIdentifier": true,
                "isMultiValued": false
            },
            {
                "AttributeName": "CourseTitle",
                "composedOf": [],
                "isIdentifier": false,
                "isMultiValued": false
            }
        ],
        "isStrong": true,
        "name": "Course",
        "relationships": [
            {
                "Entity": "Employee",
                "RelationTypeForeign": "ZeroOrMany",
                "RelationTypeLocal": "ZeroOrMany",
                "relationAttributes": [
                    "DateCompleted"
                ]
            }
        ]
    }
]
```
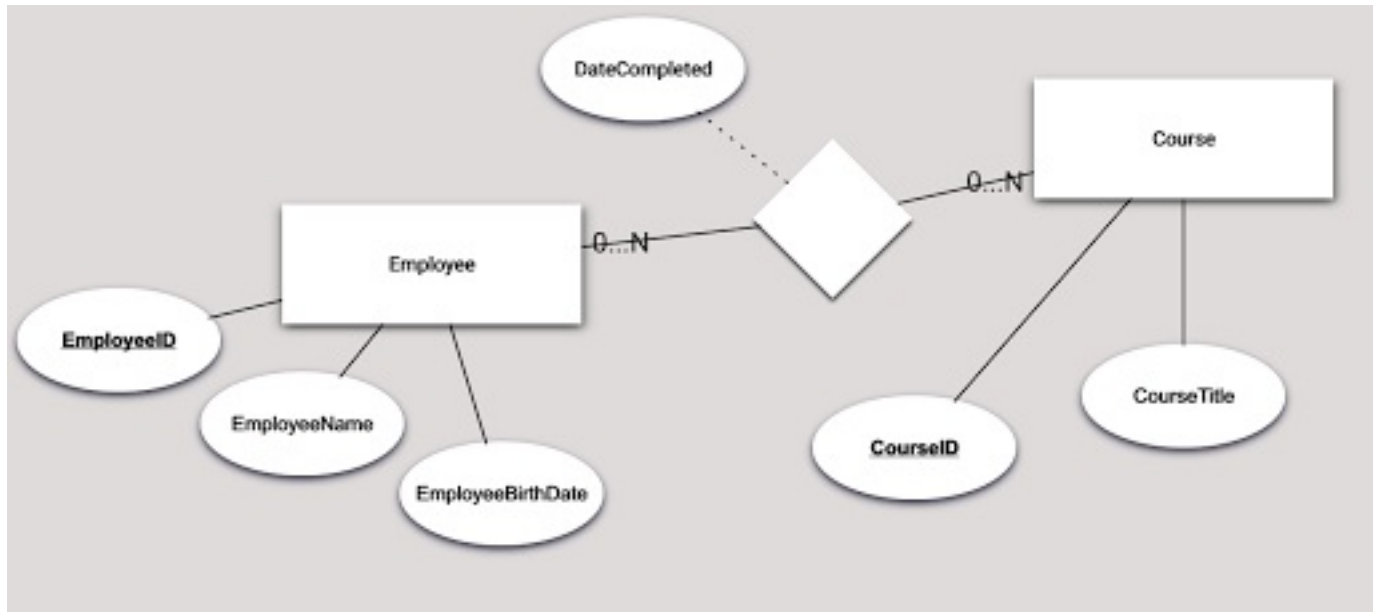
```
    }
```

## Model



# ARM

## JSON

```
{
"relations": [
    {
        "attributes": [
            {
                "AttributeName": "self",
                "dataType": "OID",
                "isConcrete": false,
                "isFK": false,
                "isPathFunctionalDependancy": false
            },
            {
```

```
                "AttributeName": "DepartmentID",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": true
            },
            {
                "AttributeName": "DepartmentName",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            }
        ],
        "coveredBy": [],
        "disjointWith": ["Professor", "Course", "Class"],
        "inheritsFrom": "none",
        "name": "Department"
    },
    {
        "attributes": [
            {
                "AttributeName": "self",
                "dataType": "OID",
                "isConcrete": false,
                "isFK": false,
                "isPathFunctionalDependancy": false
            },
```

```
            {
                "AttributeName": "DepartmentID",
                "dataType": "OID",
                "isConcrete": false,
                "isFK": true,
                "isPathFunctionalDependancy": false
            },
            {
                "AttributeName": "CourseID",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": true
            },
            {
                "AttributeName": "CourseName",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            }
        ],
        "coveredBy": [],
        "disjointWith": ["Professor", "Department", "Class"],
        "inheritsFrom": "none",
        "name": "Course"
    },
```

```json
{
    "attributes": [
        {
            "AttributeName": "self",
            "dataType": "OID",
            "isConcrete": false,
            "isFK": false,
            "isPathFunctionalDependancy": false
        },
        {
            "AttributeName": "DepartmentID",
            "dataType": "OID",
            "isConcrete": false,
            "isFK": true,
            "isPathFunctionalDependancy": false
        },
        {
            "AttributeName": "ProfessorID",
            "dataType": "String",
            "isConcrete": true,
            "isFK": false,
            "isPathFunctionalDependancy": true
        },
        {
            "AttributeName": "ProfessorName",
            "dataType": "String",
            "isConcrete": true,
            "isFK": false,
```

```json
                "isPathFunctionalDependancy": false
            },
            {
                "AttributeName": "OfficeNumber",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            }
        ],
        "coveredBy": [],
        "disjointWith": ["Course", "Department", "Class"],
        "inheritsFrom": "none",
        "name": "Professor"
    },
    {
        "attributes": [
            {
                "AttributeName": "self",
                "dataType": "OID",
                "isConcrete": false,
                "isFK": false,
                "isPathFunctionalDependancy": false
            },
            {
                "AttributeName": "CourseID",
                "dataType": "OID",
                "isConcrete": false,
```

```json
                "isFK": true,
                "isPathFunctionalDependancy": true
            },
            {
                "AttributeName": "ProfessorID",
                "dataType": "OID",
                "isConcrete": false,
                "isFK": true,
                "isPathFunctionalDependancy": true
            },
            {
                "AttributeName": "Selection",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            },
            {
                "AttributeName": "Term",
                "dataType": "String",
                "isConcrete": true,
                "isFK": false,
                "isPathFunctionalDependancy": false
            }
        ],
        "coveredBy": [],
        "disjointWith": ["Course", "Department", "Professor"],
```
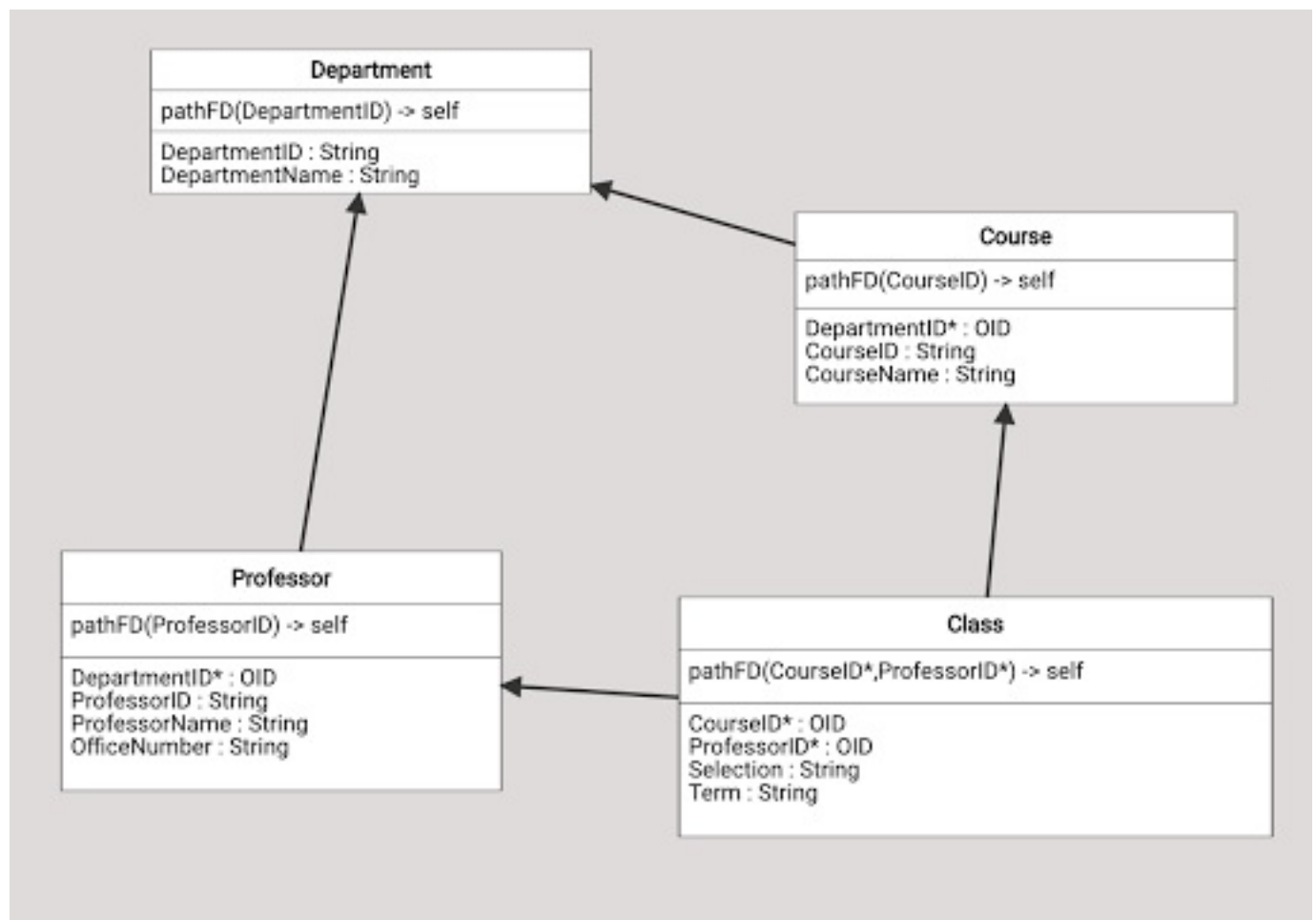
```
        "inheritsFrom": "none",

        "name": "Class"

    }

  ]

}
```

## Model



# Packages used

# Front-end

- Node js
- React js

- Jointjs

# Back-end

- Flask
- virtualenv
- numpy
- Flask-CORS