# Overview of current Corical implementation

Backend (in AWS, Python3, gRPC)

Model file from GeNiE
(.xdsl)

Model structure and numbers are extracted
automatically into the backend Model Class

**Model Class**
* SetFact (set evidence/prior, e.g. n1_Pfizer_dose = None)
* Infer (e.g. get distribution of n20_Die_from_COVID19)

Frontend (in-browser, TypeScript, React)

**Model Inputs**
* Age, sex, etc
* Hardcoded to match GeNiE inputs

JSON request object

```
{
    "age": 34,
    "sex": "male",
    "dose": "One",
    "ct": "None_0",
    "tos": true
}
```

**Backend Handlers**
* Model specific logic (e.g. if doses = 1,
SetFact(n1_Pfizer_dose = One)
* Queries model multiple times (for each scenario)
* Generates all tables and exact output

**Model Outputs**
Display graphs and tables exactly as returned by backend

JSON response object

```
{
    "scenario_description": "",
    "bar_graphs": [
    {
    "title": "What is my chance of getting COVID-19...",
    "subtitle": "Results shown for a 30-39...",
    "risks": [
    {
    "label": "Chance of getting COVID-19 if received one dose",
    "risk": 0,
    "is_relatable": false,
    "is_other_shot": false
    },
    {
    "label": "Chance of getting COVID-19 if not vaccinated",
    "risk": 0,
```