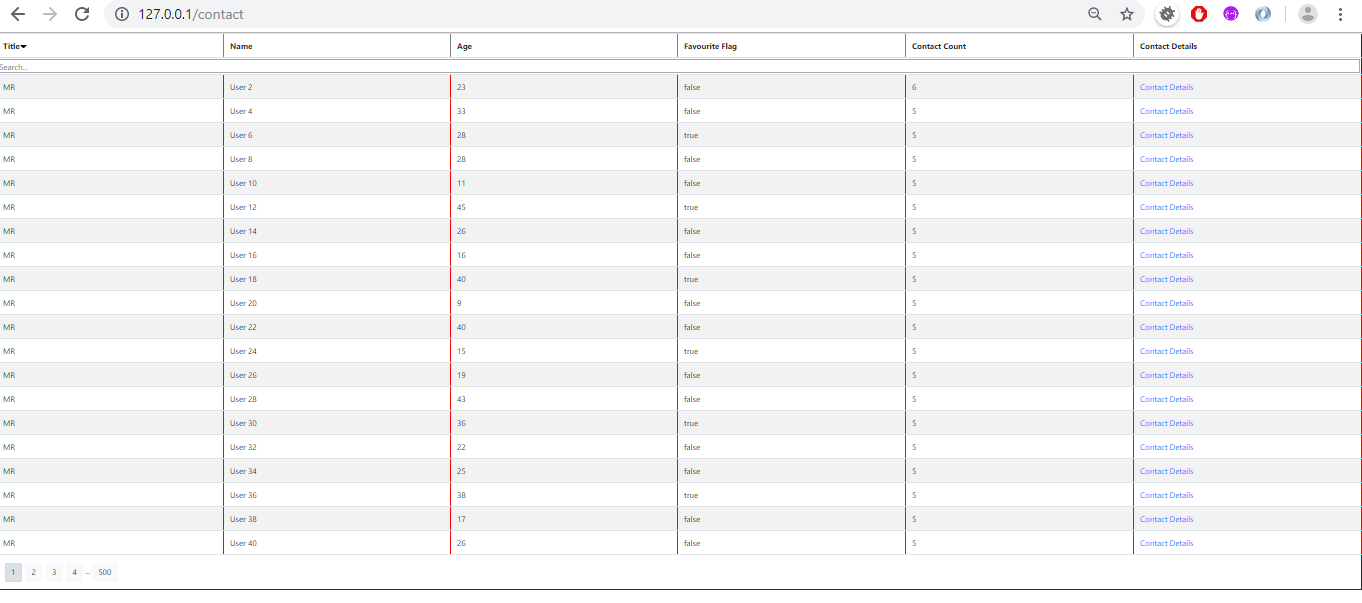
# Tech App Technical Doc

## API

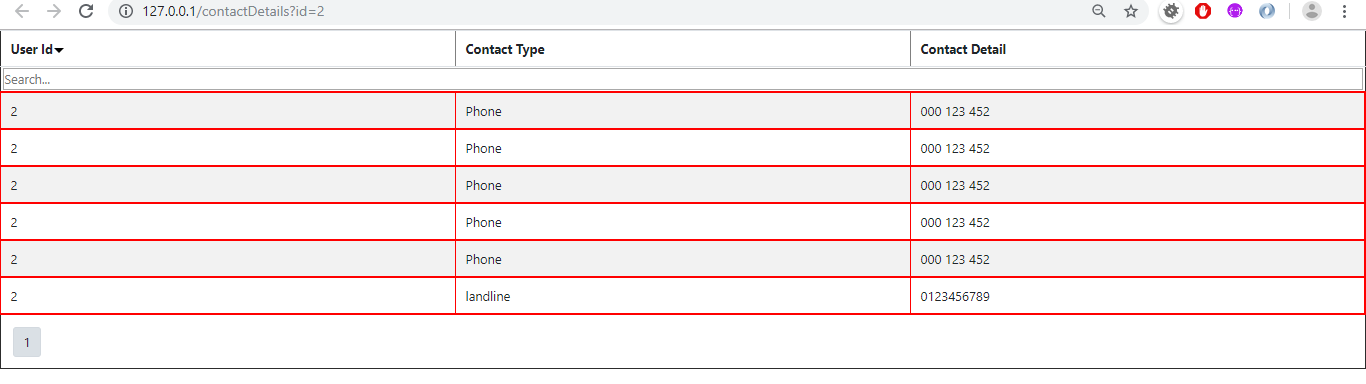
App consists of two restful APIs

1. <http://127.0.0.1/contact>
2. [http://127.0.0.1/contactDetails?id={query-param1}](http://127.0.0.1/contactDetails?id=%7bquery-param1%7d) where {query-param1} is any userid.

Sample request <http://127.0.0.1/contact> screen shot:-



Sample request for user id 2, <http://127.0.0.1/contactDetails?id=2> screen shot:



## Architecture:-

Restful architecture is used to develop these APIs, and expressjs is used at the front end.

* Contact API: It pulls the whole data from database and after that Jquery function run which adds pagination, sorting and search feature.
* Contact Details - It accepts one query parameter to fetch the records from the database.

### Use case 1:

**End user selects all contacts.**

The View (UI layer) interacts with the service layer APIs, which fetches the data from DB and formats the data, and finally returns the formatted data to UI layer (View).

### Use case 2:

**End user selects contacts details.**

The View (UI layer) interacts with the service layer APIs, which fetches the data from DB for a particular user and finally returns the formatted data to UI layer (View).

Deployment strategy (App + DB)

1. Install nodejs from <https://nodejs.org/en/download/>
2. Deployment can be done by moving the techapp folder to the system where node is installed.
3. Open command window and navigate to techapp folder.
4. To run the application fire the command **node server.js**
5. Open a browser and hit the url <http://127.0.0.1/contact>.

Technical choices (including the tools you have used)

1. Nodejs at the backend
2. EJS at the frontend
3. Restful architecture for APIs
4. Eclipse

## Possible future improvements

1. Pagination can be made customizable.
2. API level Security can be added.

## Strategy to track user actions and detect anomalies when in production

1. Proper Logging and monitoring is required for this.
2. CSRF implementation