

# Basic Insurance Recommendation API

Basic insurance recommendation REST api written with Express.js

## Useful informations about project

```
* API URL: https://insurance-api-arinc.herokuapp.com
* Web Server: Heroku
* Database: MongoDB Cloud
* App Framework: Node.js/Express.js
* CI/CD: Github Actions, Docker
* Test framework: Jest, Supertest
```

This project assumes you had already installed these tools:

1. [nvm \(https://github.com/nvm-sh/nvm\)](https://github.com/nvm-sh/nvm)
2. [postman \(https://www.postman.com/\)](https://www.postman.com/)

## Production Level Information

- This repo is using Github Actions for CI/CD deployment to Heroku.
- This project is also using MongoDB Cloud for database.
- Below you can find the production level URLs:

- Recommendation endpoint, *Only POST method is allowed.*

```
https://insurance-api-arinc.herokuapp.com/recommendations
```

- Healthcheck endpoint, *Only GET method is allowed.*

```
https://insurance-api-arinc.herokuapp.com/healthcheck
```

## Installation

- Installation of node and npm using nvm:

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.0/install.sh | bash
```

- Running either of the above commands downloads a script and runs it. The script clones the nvm repository to ~/.nvm, and attempts to add the source lines from the snippet below to the correct profile file (~/.bash\_profile, ~/.zshrc, ~/.profile, or ~/.bashrc).

```
export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion" # This loads nvm
bash_completion
```

- After *nvm* successfully installed, use the commands below:

```
nvm install 14.16.0
nvm use 14.16.0
cd {path-to-basic-insurance-api}
npm install
```

## Postman Usage

- You can find the example POSTMAN collection as basic-recommendation-api.postman\_collection.json on the directory.  
You can import the file and try each request on POSTMAN.

## API Usage

- Sample configuration for .env file that should be available in the root of this project:

```
PORT=3000
DB_USERNAME=<mongodb-username>
DB_PASSWORD=<mongodb-password>
DB_CONNECTION_PATH=<mongodb-path>
```

- Run commands:

- For Development

```
npm run start:dev
```

- For Production

```
npm run start:prod
```

- For Development with Autorestart when code changes

```
npm run watch:dev
```

- For Testing

```
npm run test
```

- After starting the system, API will expose itself through port in the .env file or 3000.
- There are two different endpoints for this API:
  1. /healthcheck
    - This endpoint accepts *GET* request.
      - This endpoint is just useful for understanding of the application live status.
  2. /recommendations
    - This endpoint accepts *POST* request.
      - If email parameter of questionnaire is not in the database it puts the questionnaire data into database then returns the insurance recommendation.
      - If email parameter of questionnaire is in the database it updates the data related with that email inside database then returns the insurance recommendation.

- Body Params (express-validator has been used for validating the request parameters):

```
{
  "firstName": "Arinc Elhan", [must - string]
  "address": "Ankara", [must - string]
  "children": "yes", [must - enum['yes', 'no']]
  "childrenNumber": 4, [if `children` is `yes` then must else `optional` - number]
  "occupation": "Employed", [must - enum['Employed', 'Self-employed', 'Student']]
  "email": "elhanarinc@gmail.com" [must - string - email]
}
```

- Valid request response data:

```
{
  "status": "OK",
  "data": {
    "Personal Liability": 6,
    "Health Insurance": 420,
    "Household Content": 7.5,
    "Currency": "EU"
  }
}
```