WHO Data Status Report Update

Lily Dourlain, Brandon Gordon, Wesley Miller, Declan Kelly

Our web application is taking data provided by the World Health Organization (WHO) and displaying it elegantly for the user. The inspiration for the project arose from the very difficult to understand presentation provided by the WHO website.

### Functional Requirements

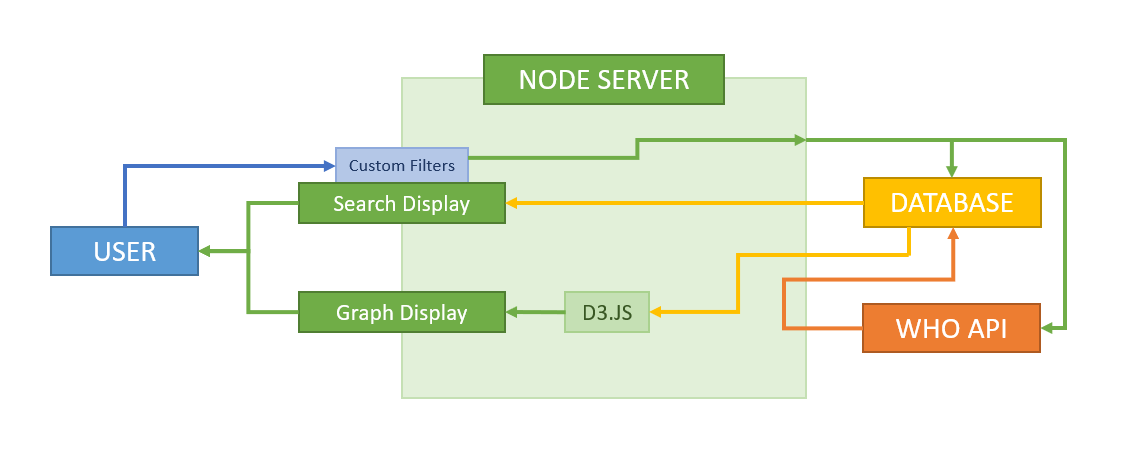
1. Interface Requirements:
   1. Users will be able to view data on a particular “indicator” from the WHO API
   2. Users will be able to specify a country or region to narrow down data.
   3. Data will be displayed on a map.
   4. Map will dynamically update when clicked on to show more information concerning a particular indicator and new country or region.
2. Data storage requirements:
   1. Data will be stored on a local MySQL database.
   2. If data is not found on the local database, a query will be made to the WHO API and new data will be inserted into the local database.
   3. Data such as countries, regions, and years will require scheduled (node-schedule) or server-side cron jobs querying the WHO API to verify updated data.

### Architectural Design

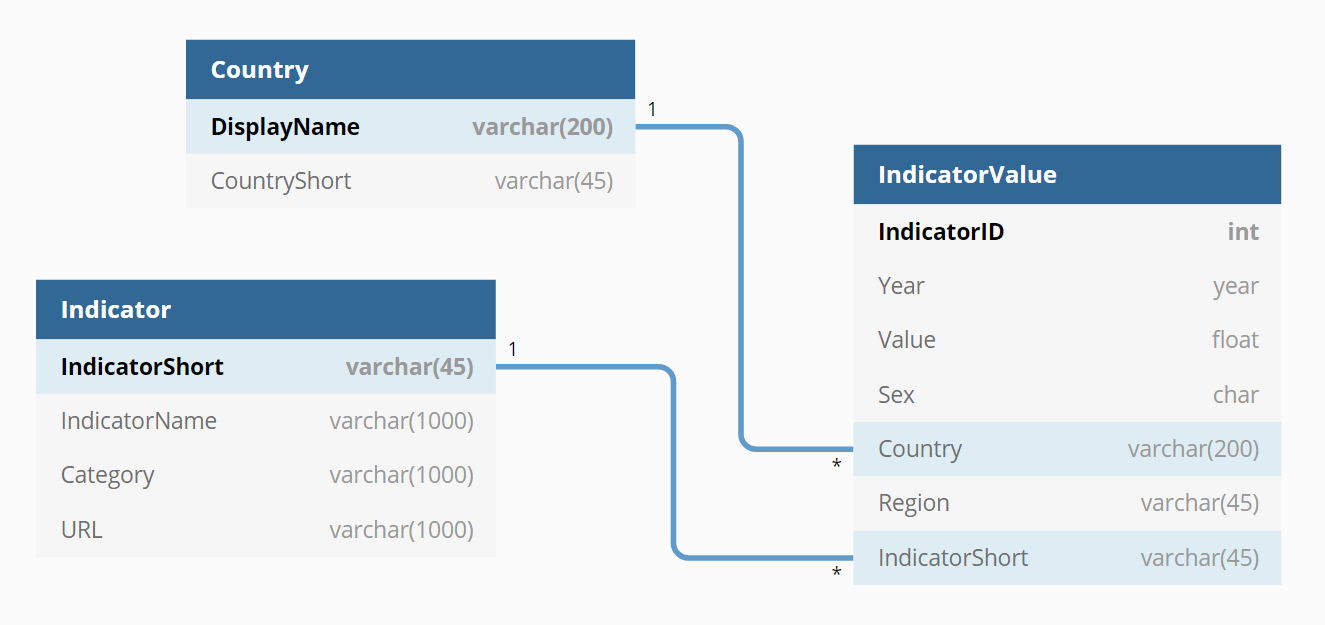
#### *Technologies To Utilize*

1. Front End
   1. Bootstrap
   2. D3.js
   3. topojson
   4. jQuery
   5. PinchZoomer jQuery Plugin
2. Back End
   1. Node.js
      1. Express, Request, MySQL
   2. MySQL

#### *Information Flow Diagram*



### Database Schema



CREATE TABLE `Indicator` (

`IndicatorShort` varchar(45) PRIMARY KEY NOT NULL,

`IndicatorName` varchar(1000) NOT NULL,

`Category` varchar(1000),

`URL` varchar(1000)

);

CREATE TABLE `Country` (

`DisplayName` varchar(200) PRIMARY KEY NOT NULL,

`CountryShort` varchar(45) NOT NULL

);

CREATE TABLE `IndicatorValue` (

`IndicatorID` int PRIMARY KEY NOT NULL AUTO\_INCREMENT,

`Year` year NOT NULL,

`Value` float NOT NULL,

`Sex` char,

`Country` varchar(200),

`Region` varchar(45) NOT NULL,

`IndicatorShort` varchar(45) NOT NULL

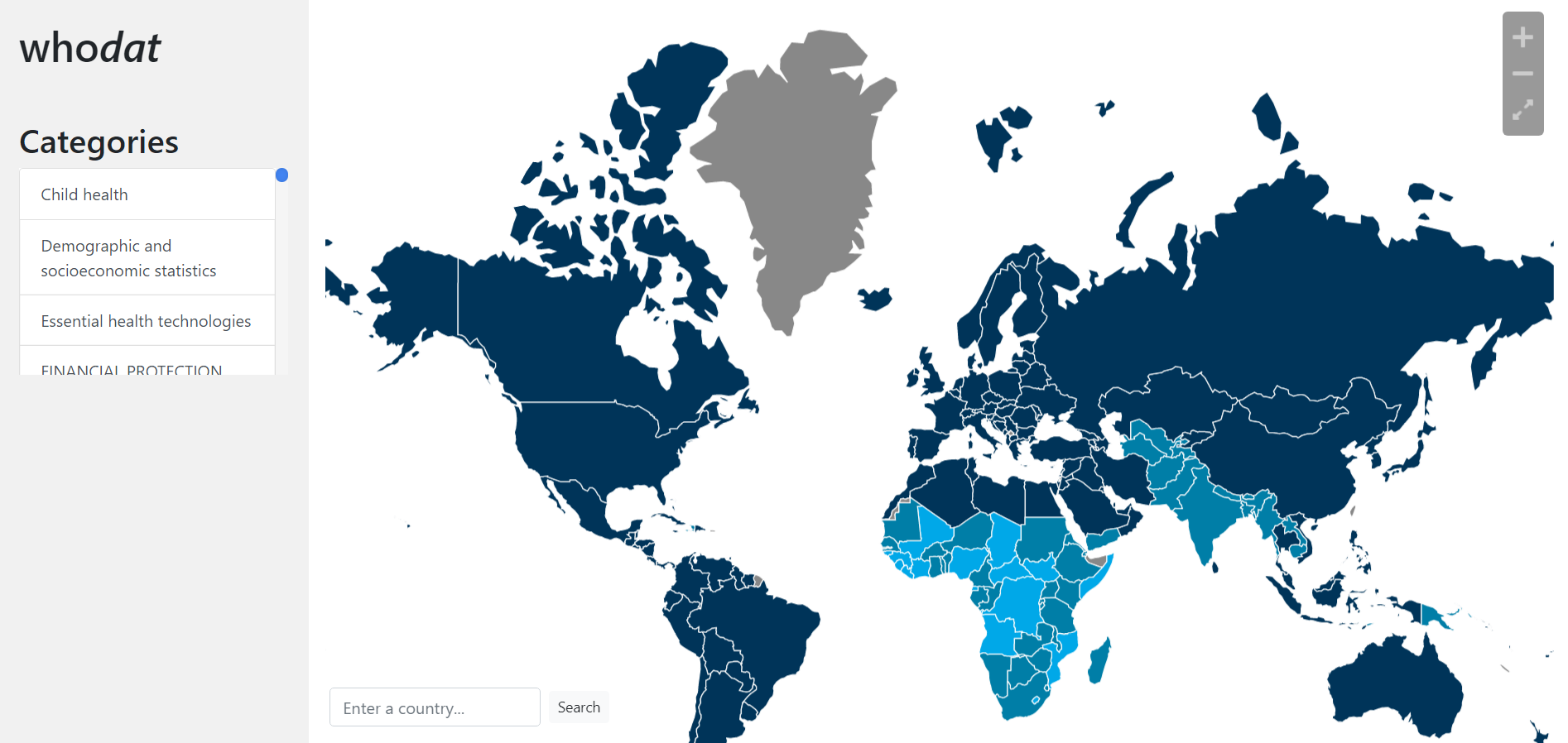
);

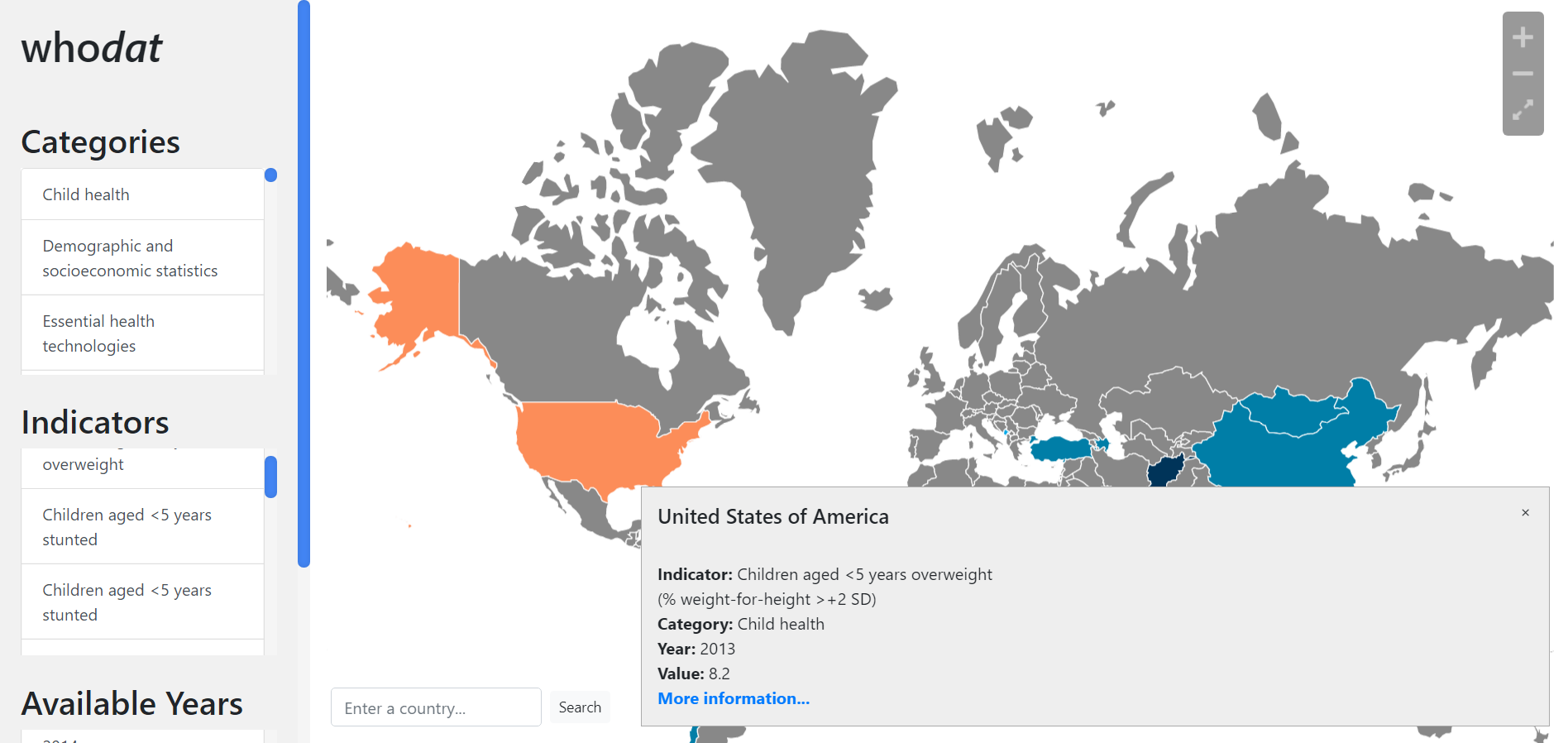
ALTER TABLE `IndicatorValue` ADD FOREIGN KEY (`IndicatorShort`) REFERENCES `Indicator` (`IndicatorShort`) ON DELETE CASCADE;

ALTER TABLE `IndicatorValue` ADD FOREIGN KEY (`Country`) REFERENCES `Country` (`DisplayName`);

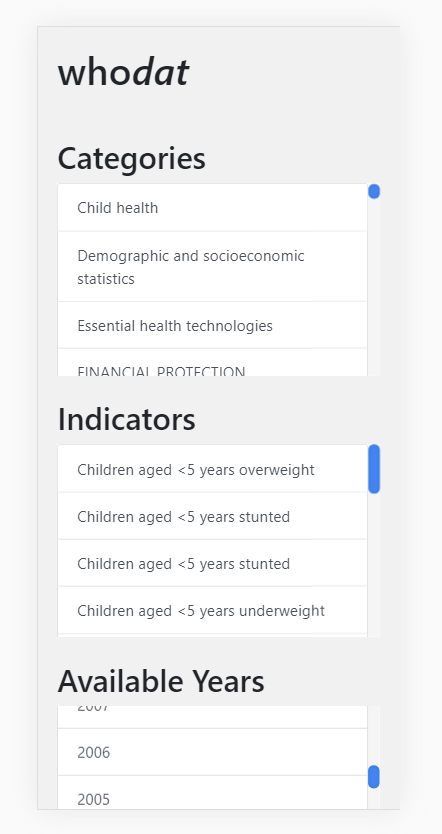
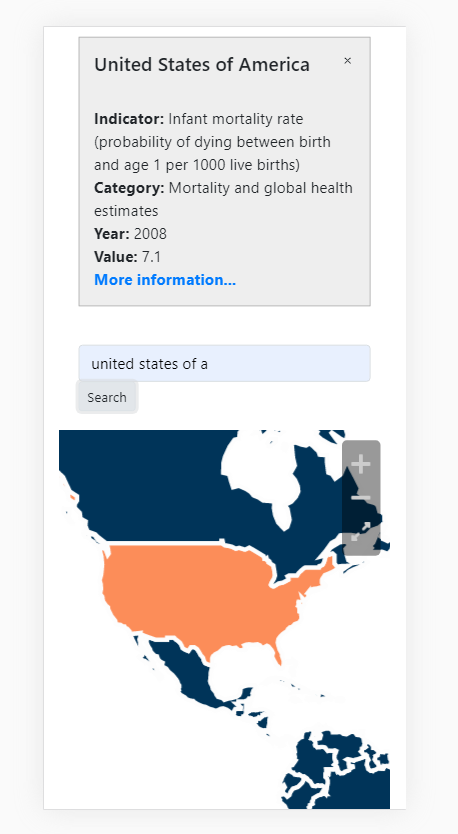
CREATE UNIQUE INDEX `IndicatorValue\_index\_0` ON `IndicatorValue` (`Year`, `Sex`, `Country`, `Region`, `IndicatorShort`);

### Final Design





Responsive Design

  text

### Gantt Chart

