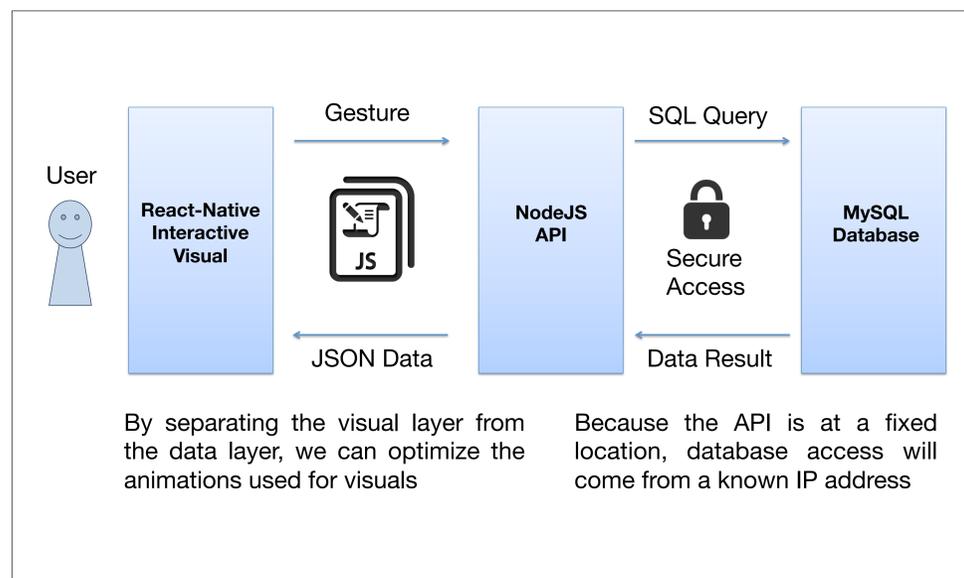


# GestureDB: An Accessible & Touch-Guided iPad App for MySQL Database Browsing

Danny Flax, Arnab Nandi, Vivek Subramanian • The Ohio State University • { flax.22, nandi.9, subramanian.121 } @ osu.edu

We present *GestureDB*, a responsive and efficient iPad App that maps SQL queries to familiar touch gestures such as swiping or dragging on individual data objects in the database. *GestureDB* makes database browsing more accessible by applying mobile-specific interface and navigation techniques to create a familiar interactive experience for exploring unfamiliar data sets.

## Architecture



## Why React-Native?

**Simple Layout Tools** – React-Native’s FlexBox CSS styling allowed us to quickly generate slick mobile layouts which automatically adjust to different screen sizes.

**Native UI Elements** – React-Native interprets our JavaScript interface code and renders it using native Cocoa UI elements, allowing us to retain the native feel while writing significantly less code than would be necessary with Objective-C.

**Better Compatibility with Web API** – With JavaScript on the front-end and back-end, the data exchange between the mobile client and server is as simple as passing a JavaScript object back and forth.

## Keeping Security in Mind

**Selective Remote Access** – Remote MySQL access is strongly discouraged due to the security threats it poses. To keep our application safe, we designed our architecture with an API layer that will sit at a fixed IP address, allowing database administrators to restrict remote access to a single point.

## Interactive Visual

EmployeeID	LastName	FirstName	Title
1	Adams	Andrew	General Manager
2	Edwards	Nancy	Sales Manager
3	Peacock	Jane	Sales Support Agent
4	Park	Margaret	Sales Support Agent
5	Johnson	Steve	Sales Support Agent
6	Mitchell	Michael	IT Manager
7	King	Robert	IT Staff
8	Callahan	Laura	IT Staff

## Gestures

- Sort** – By swiping right or left on a column header, the user may sort the data set (i.e. add an “order by” qualifier) by ascending or descending order of data in the chosen column.
- Filter** – By dragging and dropping a data block from the data set to a column header, the user may filter the data according to a specific value (e.g. “select \* from table where EmployeeID = 1”).
- Aggregate** – By dragging and dropping a column header to the main table header, the user can initiate an aggregation. This will group together rows that have the same value for this column.

Employee	MAX	MIN	COUNT	SUM
Title				

After selecting the grouping column, the user chooses how to combine the other columns in the row. The choices are MAX, MIN, COUNT, or SUM.

## Optimizations

### Optimized Animation Library

With React-Native’s ‘Animated’ library, we were able to create visual feedback for our custom gesture components that is just as responsive as the standard Cocoa components.

### Views Instead of Nested Queries

Views are temporary tables within database which store the state of the data after a query has been performed. We use Views to store the results of a query so that any successive queries can just be performed on the view. This way, we prevent the needless repetition of long and complicated queries at the cost of taking up more space in the database.

### State-Controlled Data Rendering

Table data is stored in React as a state variable, meaning that when it is updated by the API, only the parts of the data that have been changed get re-rendered.

## Implications

### Data Interaction More Accessible

Overall, the application provides for quick and responsive generation of complicated queries, all of which can be explored and applied to a live MySQL database over a cellular network. If we can make data exploration accessible to more users, large data sets can be explored with new perspective, potentially opening brand new market possibilities.

### Changing the Way We Think about Databases

The concept of abstracting away SQL queries from database interaction is not new. However, by allowing the user to physically interact with the data set, we bring back a sense of control and intuition that may otherwise be lost in the process of learning to manipulate information with a keyboard. We anticipate *GestureDB* to be a starting point for expansion of data analytics into devices that provide a more immersive user experience.