**Troubleshooting Log**

**Issue Description:** Unable to access the /api endpoint from a Node.js application hosted on DigitalOcean's App Platform. The web page loads, but the database data does not populate, indicating an inability to reach the /api endpoint.

**Initial Checks:**

1. **Code Review**
   * **Files Reviewed**: server.js, database.js, api.js, index.js
   * **Observations**: Code locally works fine; issues occur only in the production environment on DigitalOcean.
   * **Action Taken**: Reviewed code for errors; no issues found in local testing.
2. **Environment Setup**
   * **Local Setup**: Successful connection to the database and API endpoint functioning as expected.
   * **Production Setup**: Web page loads but API data does not populate.
   * **Action Taken**: Confirmed local environment variables and settings matched those in production.

**Network and Deployment Configuration:**

1. **DigitalOcean Configuration Checks**
   * **Port and IP Binding**: Confirmed correct usage of 0.0.0.0 and process.env.PORT for application binding.
   * **Firewall Settings**: Reviewed firewall settings to ensure that all necessary ports are open (particularly the application's port on DigitalOcean).
   * **DNS Configuration**: Validated that DNS settings are correct; hostname resolves properly.
2. **Connectivity Tests**
   * **Ping Tests**: Successfully pinged the database server and application URL, confirming network connectivity and correct DNS resolution.
   * **CURL Tests**: Attempted to access the application via CURL; faced issues when incorrectly using the database URL instead of the application URL.
   * **Action Taken**: Corrected the CURL command to use the appropriate application URL.

**API and Endpoint Accessibility:**

1. **API Route Configuration**
   * **Express Routing**: Added middleware to log when /api route is hit; confirmed routes are being accessed in the local environment.
   * **Endpoint Functionality**: Verified that the API retrieves data as expected locally.
   * **Action Taken**: Checked and reconfirmed all route handlers and middleware setups.
2. **HTTPS and Application Access**
   * **Protocol Handling**: Noted that DigitalOcean handles HTTPS termination at the load balancer; ensured application listens for HTTP traffic internally.
   * **HTTPS Access Tests**: Used correct HTTPS URL formats for accessing the application, avoiding the specification of ports in URLs.
   * **Action Taken**: Accessed the application using HTTPS through browser and CURL to test live functionality