

# Hotel Shift Log - Cloud SQL Migration Guide

---

Complete guide for migrating from Abacus.AI database to Google Cloud SQL.

## Table of Contents

---

1. [Overview](#)
  2. [Prerequisites](#)
  3. [Migration Steps](#)
  4. [Testing](#)
  5. [Deployment](#)
  6. [Troubleshooting](#)
- 

## Overview

---

This guide walks you through migrating your Hotel Shift Log application from the Abacus.AI hosted database to Google Cloud SQL (PostgreSQL).

### Current Setup

- **Database:** Abacus.AI PostgreSQL
- **Connection:** `db-170709f7da.db101.hosteddb.reai.io`
- **Contains:** All users, shift reports, comments, attachments

### Target Setup

- **Instance:** `hotel-shift-log-db`
  - **Region:** `us-central1`
  - **Database:** `hotelshiftlog`
  - **PostgreSQL:** Version 14
- 

## Prerequisites

---

### Before You Begin

1. **Google Cloud SQL Instance** (Already created ✓)
  - Instance name: `hotel-shift-log-db`
  - Connection: `houston-front-desk-shift-logs.us-central1.hotel-shift-log-db`
2. **Database Created** (Already created ✓)
  - Database: `hotelshiftlog`
  - Username: `hotel-shift-log-db`
  - Password: `ebf4vem5mzx8yuy*FJE`
3. **Required Tools**
  - `bash`

```
# Install Node.js dependencies for migration scripts
cd /home/ubuntu/hotel_shift_log/migration_data
npm install
```

#### 4. **Backup Current Data** (Optional but recommended)

- Your Abacus.AI database will remain intact
- This migration creates a copy, not a move

## Migration Steps

### Step 1: Export Data from Abacus.AI Database

Export all data from your current database:

```
cd /home/ubuntu/hotel_shift_log/migration_data
npm run export
```

#### Expected Output:

```
🔌 Connecting to Abacus.AI database...
✅ Connected successfully!

📦 Exporting users...
  ✅ Exported X rows from users
📦 Exporting shift_reports...
  ✅ Exported X rows from shift_reports
...

💾 Data exported successfully to: abacusai_data_export.json
```

#### What This Does:

- Connects to Abacus.AI database
- Exports all tables to JSON files
- Creates `abacusai_data_export.json` with all data
- Creates individual JSON files for each table

#### Exported Files Location:

```
migration_data/
├── abacusai_data_export.json (complete export)
├── users.json
├── shift_reports.json
├── comments.json
├── attachments.json
└── ... (other tables)
```

### Step 2: Create Database Schema in Cloud SQL

Create the schema using Prisma:

```
cd /home/ubuntu/hotel_shift_log
chmod +x migration_data/setup_schema.sh
./migration_data/setup_schema.sh
```

### Expected Output:

```
🔧 Setting up database schema in Cloud SQL...
1 Generating Prisma Client...
2 Pushing schema to Cloud SQL database...
✅ Schema setup complete!
```

### What This Does:

- Uses Prisma to create all tables in Cloud SQL
- Creates enums (UserRole, Priority, CommentType)
- Sets up relationships and constraints
- Creates indexes

### Verify Schema Created:

You can verify in Cloud SQL console or run:

```
cd migration_data
npm run test
```

This will show all tables created.

## Step 3: Import Data into Cloud SQL

Import your exported data:

```
cd /home/ubuntu/hotel_shift_log/migration_data
npm run import
```

### Expected Output:

```
📖 Reading export file...
✅ Export file loaded

🔌 Connecting to Cloud SQL database...
✅ Connected successfully!

📦 Importing X rows into users...
Progress: X/X rows
✅ Successfully imported X rows into users
📦 Importing X rows into shift_reports...
...

✅ Data import completed!
```

### What This Does:

- Reads the exported JSON data


- Imports all records into Cloud SQL
- Maintains data integrity and relationships
- Skips conflicts (if any data already exists)

## Step 4: Verify Migration


Test the Cloud SQL database:


```
cd /home/ubuntu/hotel_shift_log/migration_data
npm run test
```

### Expected Output:

 Google Cloud SQL Connection Test


 Testing: Direct IP Connection

 Connection successful!

 Database Info:

Time: 2025-10-30T...

Version: PostgreSQL 14.x

 Existing tables: 10

- users
- shift\_reports
- attachments
- comments
- comment\_likes
- ...

### Manual Verification (Optional):

You can also connect using psql or pgAdmin:

```
# Using psql
psql "postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@34.133.148.252:5432/hotelshiftlog"

# Check record counts
SELECT 'users' as table_name, COUNT(*) FROM users
UNION ALL
SELECT 'shift_reports', COUNT(*) FROM shift_reports
UNION ALL
SELECT 'comments', COUNT(*) FROM comments;
```

## Testing

### Local Testing (Before Cloud Run Deployment)

1. Update local .env file:

```
cd /home/ubuntu/hotel_shift_log/nextjs_space
```

Create/update `.env.local` :

```
DATABASE_URL="postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@34.133.148.252:5432/hotelshiftlog"
NEXTAUTH_SECRET="fdnkd5d96U1"
NEXTAUTH_URL="http://localhost:3000"
```

### 1. Test the application locally:

```
npm install
npx prisma generate
npm run dev
```

### 1. Test login and data access:

- Visit `http://localhost:3000`
- Try logging in with existing credentials
- Verify reports, comments, and attachments load correctly

## Deployment

### Update Cloud Run Environment Variables

#### Option 1: Using Google Cloud Console (Recommended)

1. Go to [Cloud Run Console](https://console.cloud.google.com/run) (<https://console.cloud.google.com/run>)
2. Click on `hotel-shift-log` service
3. Click **"EDIT & DEPLOY NEW REVISION"**
4. Go to **"Variables & Secrets"** tab
5. Update these environment variables:

#### Critical Variables:

```
DATABASE_URL = postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@/hotelshiftlog?
host=/cloudsql/houston-front-desk-shift-logs:us-central1:hotel-shift-log-db

NEXTAUTH_SECRET = fdnkd5d96U1

NEXTAUTH_URL = https://hotel-shift-log-143559442445.us-central1.run.app
```

#### SMTP Variables (for email notifications):

```
SMTP_HOST = smtp.gmail.com
SMTP_PORT = 587
SMTP_USER = jpress@knighted.com
SMTP_PASSWORD = nbbv349vdshiuc9weq
```

1. **Important:** Under “Connections” tab:

- ☒ Ensure Cloud SQL connection is enabled
- Connection name should be: houston-front-desk-shift-logs:us-central1:hotel-shift-log-db

2. Click **“DEPLOY”**

## Option 2: Using gcloud CLI

```
cd /home/ubuntu/hotel_shift_log

gcloud run services update hotel-shift-log \
  --region=us-central1 \
  --update-env-vars="DATABASE_URL=postgresql://hotel-shift-log-
db:ebf4vem5mx8yuy*FJE@/hotelshiftlog?host=/cloudsql/houston-front-desk-shift-logs:us-
central1:hotel-shift-log-db" \
  --update-env-vars="NEXTAUTH_SECRET=fdnkd5d96U1" \
  --update-env-vars="NEXTAUTH_URL=https://hotel-shift-log-143559442445.us-cent-
ral1.run.app" \
  --add-cloudsql-instances="houston-front-desk-shift-logs:us-central1:hotel-shift-log-
db"
```

## Post-Deployment Verification

1. **Visit your application:**

```
https://hotel-shift-log-143559442445.us-central1.run.app
```

2. **Test login** with existing credentials

3. **Verify data:**

- Check that all shift reports are visible
- Check comments load correctly
- Check attachments are accessible
- Test creating a new report
- Test adding comments

4. **Check Cloud Run logs:**

```
bash
```

```
gcloud run services logs read hotel-shift-log --region=us-central1 --limit=50
```

Look for:

- ☒ No database connection errors
- ☒ Successful queries
- ☒ No authentication errors

## Troubleshooting

---

### Issue 1: “Cannot find module ‘/app/server.js’”

**Solution:** This happens during Cloud Run deployment. The build process uses Next.js standalone output.

**Fix:**

```
# In your Dockerfile, ensure:  
CMD ["node", "server.js"]  
# NOT  
CMD ["node", "/app/server.js"]
```

Your current Dockerfile is correct ✓

---

### Issue 2: “Type error: Module ‘@prisma/client’ has no exported member ‘UserRole’”

**Cause:** Prisma client not generated after schema changes

**Solution:**

```
cd /home/ubuntu/hotel_shift_log/nextjs_space  
npx prisma generate
```

Then rebuild and redeploy.

---

### Issue 3: Login fails with “Internal Server Error”

**Check:**

1. **NEXTAUTH\_SECRET is set correctly:**

```
bash  
gcloud run services describe hotel-shift-log --region=us-central1 --  
format="value(spec.template.spec.containers[0].env)"
```

2. **DATABASE\_URL uses Unix socket format** (not direct IP):

```
✓ Correct: postgresql://user:pass@dbname?host=/cloudsql/...  
✗ Wrong: postgresql://user:pass@34.133.148.252:5432/dbname
```

3. **Cloud SQL connection is enabled:**

- Check in Cloud Run console under “Connections” tab

---

## Issue 4: “Cannot connect to database”

### Debugging steps:

#### 1. Check Cloud Run logs:

```
bash
gcloud run services logs read hotel-shift-log --region=us-central1 --limit=100
```

#### 2. Verify Cloud SQL connection:

- Instance must be in same project
- Public IP must be enabled (for setup)
- Cloud SQL Admin API must be enabled

#### 3. Test connection locally:

```
bash
cd migration_data
npm run test
```

#### 4. Verify database name:

- Use `hotelshiftlog` (not `postgres` )
- Database name is case-sensitive

## Issue 5: “Tables not found” or “Schema not created”

### Solution:

Run the schema setup again:

```
cd /home/ubuntu/hotel_shift_log
./migration_data/setup_schema.sh
```

Verify tables exist:

```
cd migration_data
npm run test
```

## Issue 6: Data imported but not showing in app

### Check:

#### 1. Correct database is being used:

- Verify `DATABASE_URL` points to `hotelshiftlog` database
- Not to `postgres` database

#### 2. User authentication:

- Passwords are hashed correctly
- Try logging in with known credentials

#### 3. Check data in Cloud SQL:

```
```bash
```



```
# Connect to database
psql "postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@34.133.148.252:5432/hotelshiftlog"

# Check record counts
SELECT COUNT() FROM users;
SELECT COUNT() FROM shift_reports;
...
```

---

## Database URLs Quick Reference

---

### For Local Development:

```
DATABASE_URL="postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@34.133.148.252:5432/hotelshiftlog"
```

### For Cloud Run (Production):

```
DATABASE_URL="postgresql://hotel-shift-log-db:ebf4vem5mzx8yuy*FJE@/hotelshiftlog?host=/cloudsql/houston-front-desk-shift-logs:us-central1:hotel-shift-log-db"
```

#### Key Difference:

- Local: Uses public IP and port (34.133.148.252:5432)
- Cloud Run: Uses Unix socket (/cloudsql/...)

---

## Important Notes

---

### 1. File Uploads

Your current setup stores uploaded files in the container's filesystem. This is **ephemeral** on Cloud Run.

**Recommendation:** Migrate to Google Cloud Storage for persistent file storage.

See: `DEPLOYMENT_GUIDE.md` section on "Setting Up Google Cloud Storage"

### 2. Database Backups

Enable automatic backups in Cloud SQL:

```
gcloud sql instances patch hotel-shift-log-db \
  --backup-start-time=03:00 \
  --enable-bin-log
```

### 3. Cost Optimization

Cloud SQL charges for:

- Instance uptime
- Storage
- Network egress

Consider:

- Using f1-micro or db-g1-small for low traffic
  - Enabling automatic storage increases
  - Setting up budget alerts
- 

## Success Checklist

---

- ✓ Data exported from Abacus.AI database
  - ✓ Schema created in Cloud SQL ( `hotelshiftlog` database)
  - ✓ Data imported successfully
  - ✓ Connection test passes
  - ✓ Local testing successful
  - ✓ Cloud Run environment variables updated
  - ✓ Cloud SQL connection enabled in Cloud Run
  - ✓ Application deployed and accessible
  - ✓ Login works with existing credentials
  - ✓ Data displays correctly
  - ✓ No errors in Cloud Run logs
- 

## Next Steps After Migration

---

1. **Monitor the application** for a few days
  2. **Keep Abacus.AI database** as backup (for 30 days)
  3. **Set up Cloud SQL backups**
  4. **Consider migrating file uploads** to Cloud Storage
  5. **Update documentation** with new database info
  6. **Notify team members** about the migration
- 

## Support

---

If you encounter issues:

1. Check the troubleshooting section above
  2. Review Cloud Run logs:

```
bash
gcloud run services logs read hotel-shift-log --region=us-central1 --limit=100
```
  3. Check Cloud SQL logs in Google Cloud Console
  4. Verify environment variables are set correctly
- 

## Migration Complete! 🎉

---

Your Hotel Shift Log application is now running on Google Cloud SQL with full data migration complete.

**Connection Details for Reference:**

- **Database:** hotelshiftlog
  - **Instance:** hotel-shift-log-db
  - **Region:** us-central1
  - **PostgreSQL:** Version 14
- 

Last Updated: October 30, 2025