

Auth Bill Pro

Billing System for Authorizations



May 27, 2024

Ocean waves tech

9800 45th Ave N Apt210 Plymouth, MN 55442

### Table of Contents

1. Database Overview
2. Table Descriptions
   * Participants Table
   * Activities Table
   * Auth Table
   * Links Table
   * Activity Types Table
   * Documents Table
3. Relationships
4. Key Queries
5. Modules and Macros
6. Forms
7. Maintenance and Backup Procedures
8. Migration Steps and Required Software

* Step 1: Configure ODBC Data Source
* Step 2: Link the New Table in Access
* Additional Steps for Data Migration and Linking

1. RDS Database Configuration
2. RDS Database Inbound and Outbound Configurations
3. IAM Roles and MFA

### 1. Database Overview

**Database Name:** auth\_bill\_pro  
**Purpose:** The database manages activities, authorizations, participants, and links for a small business application, specifically for billing system authorizations.

### 2. Table Descriptions

#### Table: participants

* **Description:** Stores participant information.
* **Columns:**
  + participant\_id (INT, Primary Key)
  + email (VARCHAR(255))
  + first\_name (VARCHAR(255))
  + last\_name (VARCHAR(255))
  + phone (VARCHAR(20))
  + registration\_date (DATETIME)

#### Table: activities

* **Description:** Logs activities related to participants.
* **Columns:**
  + activity\_id (INT, Primary Key)
  + activity\_date (DATE)
  + actType\_name (INT)
  + participant\_id (INT)
  + actCase\_notes (TEXT)
  + actBillable\_hours (INT)
  + auth\_number (VARCHAR(20))

#### Table: auth

* **Description:** Stores authorization details.
* **Columns:**
  + auth\_number (VARCHAR(20), Primary Key)
  + auth\_begin\_date (DATETIME)
  + auth\_end\_date (DATETIME)
  + auth\_details (TEXT)
  + auth\_rate (DECIMAL(10, 2))
  + auth\_billable\_hours (INT)
  + auth\_remaining\_billable\_hours (INT)
  + participant\_id (INT)

#### Table: links

* **Description:** Stores links associated with participants.
* **Columns:**
  + link\_id (INT, Primary Key)
  + link (VARCHAR(255))
  + participant\_id (INT)

#### Table: activity\_types

* **Description:** Defines the types of activities.
* **Columns:**
  + type\_id (INT, Primary Key)
  + type\_name (VARCHAR(255))
  + type\_desc (TEXT)

#### Table: documents

* **Description:** Stores document information.
* **Columns:**
  + DocID (INT, Auto Increment, Primary Key)
  + DocName (VARCHAR(255))
  + Attachment (LONGBLOB)

### 3. Relationships

* participants are linked to activities through participant\_id.
* auth records are linked to participants through participant\_id.
* links are associated with participants through participant\_id.
* activities are associated with activity\_types through actType\_name.

### 4. Key Queries

**Query 1: Total Billable Hours by Participant**

sql

SELECT p.first\_name, p.last\_name, SUM(a.actBillable\_hours) AS TotalBillableHours

FROM participants p

JOIN activities a ON p.participant\_id = a.participant\_id

GROUP BY p.participant\_id;

**Query 2: Authorization Details**

sql

SELECT auth\_number, auth\_begin\_date, auth\_end\_date, auth\_rate, auth\_billable\_hours, auth\_remaining\_billable\_hours

FROM auth;

**Query 3: Activities and Types**

sql

SELECT a.activity\_id, a.activity\_date, at.type\_name, a.actCase\_notes, a.actBillable\_hours, p.first\_name, p.last\_name

FROM activities a

JOIN participants p ON a.participant\_id = p.participant\_id

JOIN activity\_types at ON a.actType\_name = at.type\_id;

### 5. Modules and Macros

**Module:**

* **Name:** [ModuleName] (replace with actual name)
* **Description:** This module contains VBA code to handle specific tasks such as data validation, custom calculations, etc.
* **Key Functions:**
  + Function1(): Description of what Function1 does
  + Function2(): Description of what Function2 does

**Macro:**

* **Name:** [MacroName] (replace with actual name)
* **Description:** This macro automates specific tasks such as data import/export, report generation, etc.
* **Steps:**
  + Step 1: Description of the first step
  + Step 2: Description of the second step

### 6. Maintenance and Backup Procedures

**Backup Procedures:**

* Automated backups configured on AWS RDS with a retention period of 7 days.
* Manual snapshot taken before any major updates.

**Maintenance Tasks:**

* Regular analysis and optimization of tables.
* Monitoring performance metrics using AWS CloudWatch.
* Regular review of security group settings to ensure secure access.

### 7. Migration Steps and Required Software

**Software Required:**

* Microsoft Access
* MySQL Workbench
* MySQL ODBC Driver
* AWS Account

**Migration Steps to MySQL RDS on AWS and Linking MS Access Front-End**

#### 1. Set Up MySQL RDS on AWS

1. **Create an RDS Instance:**
   * Log in to the AWS Management Console.
   * Navigate to RDS and click "Create database".
   * Choose MySQL as the database engine.
   * Select the appropriate version, instance class, storage type, and other configurations based on your needs.
   * Note the endpoint, username, and password for the RDS instance.
   * Launch the RDS instance.

#### 2. Install MySQL ODBC Driver

1. **Download and Install the MySQL ODBC Driver:**
   * Visit the [MySQL website](https://dev.mysql.com/downloads/connector/odbc/).
   * Download and install the appropriate version of the MySQL ODBC driver for your system.

### Steps to Link the New Documents Table to MS Access Front-End

#### Step 1: Configure ODBC Data Source

1. **Install MySQL ODBC Driver:**
   * If not already installed, download and install the MySQL ODBC driver from the MySQL website.
2. **Configure ODBC Data Source:**
   * Open ODBC Data Sources (32-bit or 64-bit) from your Windows Control Panel.
   * Go to the System DSN tab and click Add.
   * Select the MySQL ODBC 8.0 Driver and click Finish.
   * Configure the data source with the following details:
     + **Data Source Name:** A name for your data source (e.g., MySQLAuthBillPro).
     + **TCP/IP Server:** Your MySQL server's endpoint (e.g., auth-bill-pro.c3q6066gukuc.us-east-2.rds.amazonaws.com).
     + **Port:** 3306
     + **User:** Your MySQL username (e.g., admin).
     + **Password:** Your MySQL password.
     + **Database:** Your database name (e.g., auth\_bill\_pro).
   * Test the connection to ensure it is configured correctly.

#### Step 2: Link the New Table in Access

1. **Open MS Access Front-End:**
   * Open your Access front-end database.
2. **Link the New Table:**
   * Go to the External Data tab in the Ribbon.
   * Click New Data Source > From Other Sources > ODBC Database.
   * Select Link to the data source by creating a linked table and click OK.
   * In the Select Data Source dialog, go to the Machine Data Source tab.
   * Select the ODBC data source you configured (e.g., MySQLAuthBillPro) and click OK.
   * You will be prompted to select the tables to link. Select the documents table.
   * Click OK to finish linking the table.

### Additional Steps for Data Migration and Linking

#### Step 3: Export Data from Access to Text Files

1. **Export Tables to Text Files:**
   * Open your Access back-end database.
   * Go to External Data > Text File.
   * Follow the wizard to export each table to a text file (.csv or .txt).

#### Step 4: Import Data into MySQL Using MySQL Workbench

1. **Create Tables in MySQL:**
   * Open MySQL Workbench and connect to your MySQL RDS instance.
   * Create the necessary tables in your MySQL database.

sql

 CREATE TABLE documents (

DocID INT AUTO\_INCREMENT PRIMARY KEY,

DocName VARCHAR(255),

Attachment LONGBLOB

);

 **Use LOAD DATA INFILE to Import Data:**

* Import data from the text files into the corresponding tables in your MySQL database.

sql

1. LOAD DATA INFILE 'C:/path/to/your/file.csv'
2. INTO TABLE documents
3. FIELDS TERMINATED BY ',' ENCLOSED BY '"'
4. LINES TERMINATED BY '\n'
5. (DocID, DocName, Attachment);

#### Step 5: Link Access Front-End to MySQL RDS

1. **Link Tables in Access:**
   * Use the linked table manager in Access to link tables to the MySQL RDS instance.
   * Go to the External Data tab.
   * Click New Data Source > From Other Sources > ODBC Database.
   * Select Link to the data source by creating a linked table and click OK.
   * Choose your ODBC data source and link the required tables.

### 8. RDS Database Configuration

**Instance Specifications:**

* **Instance Type:** db.t3.micro
* **vCPUs:** 2
* **RAM:** 1 GiB
* **Storage:** 20 GiB (General Purpose SSD)

**Database Engine:**

* **Engine Type:** MySQL
* **Engine Version:** 8.0.35

**Instance Settings:**

* **Instance Identifier:** auth-bill-pro
* **Master Username:** admin
* **Endpoint:** auth-bill-pro.c3q6066gukuc.us-east-2.rds.amazonaws.com
* **Port:** 3306

**Security:**

* **VPC:** Default VPC
* **Security Group:** Custom security group allowing inbound traffic on port 3306 from trusted IP addresses.
* **Encryption:** Enabled for data at rest using AWS-managed keys.

**Backup and Maintenance:**

* **Automated Backups:** Enabled with a retention period of 7 days.
* **Backup Window:** Configured to occur during low-traffic periods.
* **Maintenance Window:** Weekly maintenance window scheduled during low-traffic periods.

**Monitoring and Performance:**

* **CloudWatch Monitoring:** Enabled for performance metrics.
* **Enhanced Monitoring:** Enabled for real-time instance metrics.
* **Parameter Group:** Custom parameter group configured for optimal performance.

### 9. RDS Database Inbound and Outbound Configurations

#### Inbound Configuration

**Security Group Inbound Rules:**

* **Type:** MySQL/Aurora
* **Protocol:** TCP
* **Port Range:** 3306
* **Source:** Specific IP addresses or CIDR blocks that are allowed to connect to the database (e.g., your office IP, developer IP addresses)

Example:

* **Type:** MySQL/Aurora
* **Protocol:** TCP
* **Port Range:** 3306
* **Source:** 203.0.113.0/24 (replace with your specific IP range)

#### Outbound Configuration

**Security Group Outbound Rules:**

* **Type:** All traffic
* **Protocol:** All
* **Port Range:** All
* **Destination:** 0.0.0.0/0 (allowing all outbound traffic, which is the default setting for most security groups)

Example:

* **Type:** All traffic
* **Protocol:** All
* **Port Range:** All
* **Destination:** 0.0.0.0/0

**Summary of Inbound and Outbound Rules**

#### Inbound Rules

1. **MySQL/Aurora:**
   * **Protocol:** TCP
   * **Port Range:** 3306
   * **Source:** (Your specific IP range or address, e.g., 203.0.113.0/24)

#### Outbound Rules

1. **All Traffic:**
   * **Protocol:** All
   * **Port Range:** All
   * **Destination:** 0.0.0.0/0

### 10. IAM Roles and MFA

#### IAM Roles

**Role for Database Access:**

* **Description:** This role provides the necessary permissions for the application to access the RDS database.
* **Policies:**
  + AmazonRDSFullAccess
  + CloudWatchFullAccess

**Steps to Create IAM Role:**

1. Go to the IAM console.
2. Click on "Roles" and then "Create Role".
3. Choose "AWS Service" and select "EC2" (if your application is running on EC2).
4. Attach the necessary policies.
5. Name the role and create it.

#### MFA (Multi-Factor Authentication)

**Steps to Enable MFA:**

1. Go to the IAM console.
2. Click on "Users" and select the user for whom you want to enable MFA.
3. Under the "Security credentials" tab, click "Manage" next to "MFA device".
4. Follow the prompts to set up MFA using a virtual MFA device (e.g., Google Authenticator).

**Recommended MFA Policies:**

* Require MFA for console access.
* Attach policies that require MFA for sensitive operations.