# Developing Cloud Solutions using AWS

## Project

**Case Study:**

Upgrade solution is an IT firm providing services to multiple EdTech companies, Since

they have to create and share huge number of documents, question set, eBooks, Video

libraries and all. Founder of organization decided that now it’s difficult for team to share

these details because system is getting slow and sometimes client/ students are not able

to get the required details because of the unavailability of database. Team now decided

to create and connect to a DB instance using A-RDS. By using AWS identity access

management to manage access to Amazon RDS resources.

**AWS Services Used:**

• EC2

• VPC

• RDS

**Approach to Solve:**

Please perform below mentioned tasks –

• You can use AWS IAM to create permissions that specify which Amazon RDS

actions a user, group, or role in your AWS account can perform, and on which RDS

resources those actions can be performed. You specify permissions using an IAM

policy.

• Create IAM under your AWS account.

• Create IAM policies which defines actions a user can take and specify the Amazon

- RDS resources required for each task using ARN’s by adding policies to

applicable users and groups.

• Allow user to create DB instance which uses a specified DB engine.

This project involves setting up **AWS IAM (Identity & Access Management)** to control who can access **Amazon RDS (Database)**. Follow these steps:

**Step 1: Sign in to AWS Console**

1. Go to [AWS Management Console](https://aws.amazon.com/).
2. Log in with your AWS account (or create one if you don’t have one).

**Step 2: Create an IAM Policy (Rules for Database Access)**

1. Open **IAM Dashboard** (Search "IAM" in AWS Console).
2. Click **"Policies"** > **"Create Policy"**.
3. Choose the **"JSON"** tab and paste this policy (allows RDS actions):

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": [

"rds:CreateDBInstance",

"rds:DescribeDBInstances"

],

"Resource": "\*"

}

]

}

1. Click **"Next"**, give it a name (e.g., RDS-Access-Policy), and **Create Policy**.

**Step 3: Create an IAM Group & Attach Policy**

1. Go to **"Groups"** > **"Create New Group"**.
2. Name it (e.g., RDS-Admins).
3. Attach the policy you just created (RDS-Access-Policy) , RDS-Full-Access and administrator access.
4. Click **"Create Group"**.

**Step 4: Create an IAM User & Add to Group**

1. Go to **"Users"** > **"Add User"**.
2. Enable **AWS Console Management.**
3. Enter a name (e.g., DB-User).
4. Add the user to the group (RDS-Admins).
5. Click **"Create User"** and **save the login details**(Username, old password)

**Step 5: Test the User’s Permissions**

1. Log out of AWS.
2. Log in as the new user (DB-User).
3. Go to **Amazon RDS** and try to **create a database**:
   * Click **"Create database"**.
   * Choose a database type (e.g., MySQL).
   * Fill in details (DB name, username, password).
   * Click **"Create"**.
4. If the database is created, your IAM setup works!

**Final Checks**

**IAM Policy** → Defines what the user can do (e.g., create RDS).  
**IAM Group** → Assigns the policy to multiple users.  
**IAM User** → Can now create/manage RDS databases.