# Snowflake CI/CD Project Onboarding & Marketing Project LOE

## 1. Level of Effort (LOE) - Marketing Project

The Marketing project involves a substantial number of stored procedures, tables, views, and tasks. The estimated effort to onboard and maintain this project includes:  
  
- Reviewing and understanding the existing SQL objects (SPs, tables, views, tasks)  
- Creating and updating JSON config files per project  
- Managing secrets and private keys in GitHub Secrets  
- Updating YAML workflow to detect changed files and trigger deployment  
- Writing and testing the deployment Python script  
- Handling schema and folder mappings  
- Debugging and troubleshooting Snowflake errors  
  
Estimated Development Effort:  
- Initial Setup: 2 days  
- Folder structuring ,deployment and testing: 3 days  
- Task Dependency Handling (Suspend/Resume Automation - not yet implemented): 5 days  
- Maintenance and Monitoring: -

## 2. Prerequisites for New Project Onboarding

Before onboarding a new project to the Snowflake CI/CD pipeline, ensure the following are prepared:  
  
- Create a JSON configuration file under configs/ named <PROJECT>.json with Snowflake credentials and schema info  
- Add the private key for Snowflake authentication as a GitHub secret named <PROJECT>\_PRIVATE\_KEY  
- Setup the project folder structure under dbscripts2/<PROJECT>/ with subfolders for schemas (e.g., XFRM, RPT)  
- Prepare SQL files (\*.sql) in appropriate folders following the naming conventions  
- Confirm Snowflake roles, warehouse, database access, and permissions are correct for the deployment user  
- Ensure YAML workflow references the new project configuration

## 3. Action Items for Onboarding a New Project

- Add new project JSON config file with proper Snowflake account, user, role, warehouse, database, key path, and schema mappings  
- Add the private key to GitHub secrets as <PROJECT>\_PRIVATE\_KEY  
- Place SQL scripts under the proper folder paths in the repo under dbscripts2/<PROJECT>/  
- Verify the GitHub Actions YAML detects and deploys only changed SQL files from the new project  
- Test deployments thoroughly for all object types: TABLE, VIEW, FUNCTION, STOREDPROC, TASK  
- Develop and integrate suspend/resume logic for task dependency handling (currently manual)  
- Document any project-specific nuances or deviations in the repo wiki or docs folder