CREATE Operation SUCCESS CASE DIAGRAM UserManagementEntry UserManagementManager **IAuthorizationService** ILoggingService UserManagementService UserManagementDataAccess MariaDB (Application Layer) (Service Layer) (Service Layer) (Service Layer) (Data Access Layer) (Data Store) (Business Layer) SingleOperationRequest(CREATE : String) : String userManagementManager = new UserManagementManager() UserManagementManager() return UserManagementManager userManagementManager.CreateAccountInput(accountInfo : Map<String, String>) : Boolean authorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService AuthorizeRequest(username : String) : Boolean Refer to the True : Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementService = new UserManagementService() teturn UserManagementService user Management Service. Create Account Request (account Info: Map < String, String >): BooleanuserManagementDataAccess = new UserManagementDataAccess() UserManagementDataAccess() return UserManagementDataAccess userManagementDataAccess.IsAccountActive(username : String) : Boolean EstablishMariaDBConnection(): Boolean Query Processing: SELECT username FROM accounts WHERE username SelectAccount(searchAccountDML : String) : Boolean LIKE usernameInput False : Boolean False : Boolean Query Processing: INSERT accounts userManagementDataAccess.CreateAccount(accountInfo: Map<String, String>): Boolean VALUES (typeInput, InsertAccount(createAccountDML : String) : Boolean usernameInput, passwordInput, True : Boolean emailInput, True : Boolean ableInput) userManagementLoggingService = new UserManagementLoggingService() UserManagementLoggingService() return UserManagementLogging\$ervice LogRequestAsync(operation : String, isSuccess : Boolean) : Boolean Refer to the Logging Diagram True : Boolean True : Boolean "UM operation was successful" : String Success Scenario 1. Single Request is made for the CREATE operation 2. New instance of CreateAccountManager is created Input account info 4. Check if request is valid, the following is checked: does the input follow the appropriate regex
does the input contain all necessary info for the operation request 5. New instance of CreateAccountService is created 6. Call the CreateAccount Service to process operation with the validated input 7. New instance of CreateAccountDataAccess is created 8. Check if the account already exists using the username 9. Establish the MariaDB connection in DataAccess if not already opened 10. Perform a SELECT query on the Accounts table in the database: SELECT username FROM accounts WHERE username LIKE usernameInput Returns FALSE if not found. 11. Return FALSE to the Service if no account already exists, and continue 12. Create the account with the account info 13. Perform a INSERT query on the Accounts table in the database: INSERT accounts VALUES (typeInput, usernameInput, passwordInput, emailInput, Returns TRUE if account is successfully created. 14. Return TRUE to the Service if the account was successfully created 15. Return TRUE to the Manager if the request was successfully performed 16. Return TRUE to System and follow with the appropriate Logging for this operation FAILURE CASE DIAGRAMS Fail Case 13 New Account is not Inserted into MariaDB Reasons: incorrect DML syntax - MariaDB fails either by connection timeout or external factors JserManagementManager UserManagementService UserManagementEntry IAuthorizationService UserManagementDataAccess ILoggingService MariaDB (Application Layer) (Business Layer) (Data Access Layer) (Service Layer) (Data Store) (Service Layer) (Service Layer) SingleOperationRequest(CREATE : String) : Boolean userManagementManager = new UserManagementManager() UserManagementManager() UserManagementManager user Management Manager. Create Account Input (account Info: Map < String, String >): BooleanauthorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService authorizationService.AuthorizeRequest(username : String) : Boolean Refer to the True: Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementService = new UserManagementService() return CreateAccountService userManagementService.CreateAccountRequest(accountInfo : Map<String, String>) : Boolean userManagementDataAccess = new UserManagementDataAccess() UserManagementDataAccess() return UserManagementDataAccess userManagementDataAccess.lsAccountActive(username : String) : Boolean EstablishMariaDBConnection(): Boolean Query Processing: SELECT username FROM accounts WHERE username SelectAccount(searchAccountDML : String) : Boolean LIKE usernameInput False : Boolean Query Processing: INSERT accounts userManagementDataAccess.CreateAccount(accountInfo : Map<String, String>) : Boolean VALUES (typeInput, usernameInput, InsertAccount(createAccountDML : String) : Boolean passwordInput, emailInput, ableInput) False: Boolean userManagementLoggingService = new UserManagementLoggingService()

UserManagementLoggingService() return UserManagementLoggingService LogRequestAsync(operation : String, isSuccess : Boolean) : Boolean Refer to the Logging Diagram True : Boolean False : Boolean False : Boolean False : Boolean "UM operation not successful" : String Fail Case 10a New Account already exists in MariaDB Reasons: account already exists UserManagementService UserManagementManager UserManagementEntry IAuthorizationService ILoggingService UserManagementDataAccess MariaDB (Service Layer) (Application Layer) (Business Layer) (Service Layer) (Service Layer) (Data Access Layer) (Data Store) SingleOperationRequest(CREATE : String) : Boolean userManagementManager = new CreateAccountManager() return userManagementManager user Management Manager. Create Account Input (account Info: Map < String, String >): BooleanauthorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService authorizationService.AuthorizeRequest(username : String) : Boolean Refer to the True : Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementService = new userManagementService() return UserManagementService $user Management Service. Create \cite{Account} Request (account Info: Map < String, String >): Boolean$ userManagementDataAccess = new userManagementDataAccess() UserManagementDataAccess() return UserManagementDataAccess user Management Data Access. Is Account Active (username: String): BooleanEstablishMariaDBConnection(): Boolean SelectAccount(searchAccountDML : String) : Boolean userManagementLoggingService = new UserManagementLoggingService() UserManagementLoggingService() return UserManagementLoggingService LogRequest A sync (operation: String, is Success: Boolean): BooleanRefer to the Logging Diagram True : Boolean True : Boolean False : Boolean False : Boolean "UM operation not successful" : String Fail Case 10b Could Not Process Query Reasons: incorrect DML syntax - MariaDB fails either by connection timeout or external factors UserManagementEntry (Application Layer) UserManagementManager ILoggingService (Service Layer) (Business Layer) UserManagementService UserManagementDataAccess MariaDB IAuthorizationService (Data Access Layer) (Data Store) (Service Layer) (Service Layer) SingleOperationRequest(CREATE : String) : Boolean userManagementManager = new UserManagementManager() return UserManagementManager userManagementManager.CreateAccountInput(accountInfo: Map<String, String>): Boolean authorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService authorizationService.AuthorizeRequest(username : String) : Boolean True : Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementService = new UserManagementService() UserManagementService() return CreateAccountService userManagementService.CreateAccountRequest(accountInfo: Map<String, String>): Boolean userManagementDataAccess = new UserManagementDataAccess() UserManagementDataAccess() return UserManagementDataAccess userManagementDataAccess.lsAccountActive(username : String) : Boolean Query Processing: SELECT username EstablishMariaDBConnection(): Boolean FROM accounts WHERE username SelectAccount(searchAccountDML : String) : Boolean LIKE usernameInput Error : String userManagementLoggingService = new UserManagementLoggingService() UserManagementLoggingService() return UserManagementLoggingService LogRequestAsync(operation : String, isSuccess : Boolean) : Boolean Refer to the Logging Diagram True: Boolean Error : String False : Boolean False : Boolean "UM operation not successful" : String Fail Case 9 MariaDB Connection could not be Established - EstablishMariaDBConnection() returns False -- NOTE: continue to call this method until True if still no connection after 5 seconds, log the error and end the process UserManagementManager UserManagementEntry IAuthorizationService UserManagementService ILoggingService UserManagementDataAccess MariaDB (Application Layer) (Business Layer) (Service Layer) (Data Access Layer) (Data Store) (Service Layer) (Service Layer) SingleOperationRequest(CREATE : String) : Boolean userManagementManager = new UserManagementManager() UserManagementManager() return UserManagementManager userManagementManager.CreateAccountInput(accountInfo : Map<String, String>) : Boolean authorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService authorizationService.AuthorizeRequest(username : String) : Boolean True: Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementService = new UserManagementService() UserManagementService() return UserManagementService userManagementService.CreateAccountRequest(accountInfo: Map<String, String>): Boolean userManagementDataAccess = new UserManagementDataAccess() UserManagementDataAccess() return CreateAccountDataAccess EstablishMariaDBConnection(): Boolean False : Boolean userManagementLoggingService = new UserManagementLoggingService() UserManagementLoggingService() return UserManagementLoggingService LogRequest A sync (operation: String, is Success: Boolean): BooleanRefer to the Logging Diagram True : Boolean False : Boolean False : Boolean "UM operation not successful" : String

Query Processing: SELECT username

FROM accounts
WHERE username

LIKE usernameInput

Fail Case 4 Invalid Request Reasons: - Inputs do not follow their respective regex - Missing required input(s) or does not match the inputs required for request ILoggingService (Service Layer) UserManagementEntry **IAuthorizationService** UserManagementDataAccess MariaDB UserManagementManager UserManagementService (Application Layer) (Business Layer) (Service Layer) (Service Layer) (Data Access Layer) (Data Store) SingleOperationRequest(CREATE : String) : Boolean userManagementManager = new UserManagementManager() **▶** UserManagementManager() return UserManagementManager user Management Manager. Create Account Input (account Info: Map < String, String >): BooleanauthorizationService = new UserManagementAuthorizationService(operation : String) UserManagementAuthorizationService(operation : String) return UserManagementAuthorizationService authorizationService.AuthorizeRequest(username : String) : Boolean Refer to the True : Boolean Authorization Diagram IsValidRequest(request : String[]) : Boolean userManagementLoggingService = new UserManagementLoggingService() UserManagementLoggingService() return UserManagementLoggingService LogRequestAsync(operation : String, isSuccess : Boolean) : Boolean Refer to the Logging Diagram True : Boolean False : Boolean "UM operation not successful" : String