Test Driven Development

1. Critical Modules of the Project

Module	Description
User Authentication	Provides secure access to the application
API: Get Filtered Findings	Fetches AWS Security Hub findings based on filters
Data Ingestion via Lambda	Automatically ingests findings from AWS Security Hub
Data Storage	Persists data in DynamoDB and RDS for querying and analysis
Data Visualization	Displays findings with chart/table UI
Export Functionality	Enables exporting findings as CSV or JSON
AWS Integration	Handles interactions with AWS services securely

2. Risk and Threat Analysis

Area	Identified Risk or Threat
Authentication	Unauthorized access, session hijacking, weak password policies
API Filtering	Injection attacks, excessive filtering breaking the API
AWS Credential Handling	Leaked or hardcoded credentials
Lambda Triggers	Failure to trigger or duplicate event processing
Database Storage (DynamoDB, RDS)	Data inconsistency, incomplete data writes
Data Export	Unauthorized export, information leakage
Frontend UI	Incorrect representation, inconsistent filter application

3. Test Cases

A. Authentication

Test Case 1: Valid Login

- Input: Correct username and password
- Expected Output: Status 200 with access granted

Test Case 2: Invalid Login

- Input: Invalid password
- Expected Output: Status 401 Unauthorized

Test Case 3: Injection Attempt

- Input: Username or password containing malicious SQL or script
- Expected Output: Request blocked or sanitized, status 401

Test Case 4: Login Rate Limiting

- Input: Multiple failed login attempts within a short time
- Expected Output: Later attempts blocked or throttled

B. API: Get Filtered Findings

Test Case 1: Single Valid Filter

- Input: ?Region=us-east-1
- Expected Output: List of findings filtered by region

Test Case 2: Multiple Valid Filters

- Input: ?SeverityLabel=CRITICAL&ResourceType=AwsEc2Instance
- Expected Output: Only findings matching all criteria returned

Test Case 3: Invalid Filter Field

- Input: ?InvalidField=test
- Expected Output: Field ignored or request rejected with 400 Bad Request

Test Case 4: Malicious Input Filter

- Input: Filter string containing code injection attempt
- Expected Output: No execution, input sanitized, findings unaffected

---

C. Data Export

Test Case 1: Export to CSV

- Input: User clicks export with valid session
- Expected Output: Downloadable CSV file with selected findings

Test Case 2: Export to JSON

- Input: User initiates JSON export
- Expected Output: JSON file correctly formatted and downloadable

Test Case 3: Export Without Authentication

- Input: Export request sent without valid session
- Expected Output: 403 Forbidden response

D. Data Visualization (Frontend)

Test Case 1: Load Dashboard

- Input: Login and access dashboard page
- Expected Output: Charts and tables are rendered with default data

Test Case 2: Empty Filter Result

- Input: Apply filters that match no findings
- Expected Output: UI shows "No Data Found" message

Test Case 3: Large Result Set

- Input: Query returns 1000+ results
- Expected Output: UI handles pagination or scrolling gracefully

---

E. AWS Services Interaction

Test Case 1: Credential Validation

- Input: Valid credentials in environment or AWS profile
- Expected Output: API connects to AWS and fetches data

Test Case 2: Credential Rotation

- Input: New access key rotated
- Expected Output: System continues to work without disruption

Test Case 3: Unauthorized IAM Role

- Input: AWS credentials without required permissions
- Expected Output: API request fails with AccessDeniedException