

**TINF21C, SWE I Praxisprojekt 2022**

System Test Plan (STP)

**Project: AAS-Management**

**Customer:** Rentschler & Holder  
 Rotebühlplatz 41  
 70178 Stuttgart

**Supplier:** Team 2 (Selvana Dwi Ayunda, Paul Brenner, Jonas Alexander Graubner, Mohaddeseh Tibashi, Luka Dominik Pavic)  
Rotebühlplatz 41  
70178 Stuttgart

**Version 0.3**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Comment** |
| 0.1 | 24.04.2023 | Mohaddeseh Tibashi, Selvana Ayunda | Created and add executing plann |
| 0.2 | 25.04.2023 | Selvana Ayunda | Added Testschedule |
| 0.3 | 18.05.2023 | Selvana Ayunda | Change feature and executing plan |

1. **Introduction**
   1. **Purpose**

The AAS Management test plan outlines the strategies, approaches, and methodologies that the QA team will use to verify that the application meets the established requirements of the SRS. It includes testing activities such as unit testing, integration testing, system testing, user acceptance testing, and regression testing, as well as defining the testing environment, tools, resources, and timelines. The test criteria and test cases are based on the SRS requirements to ensure that the application meets the functional and non-functional requirements. The test plan is critical in guiding the testing efforts of the QA team to identify any defects or errors in the application and ensure that it is ready for release.

* 1. **Objectives**

Application

* meets the System Requirement Specification and System Architecture Specification,
* satisfies the business purpose
* satisfies the testing thresholds laid out in this document
  1. **Abbviation**

T: Test

TC: Test Case

QA: Quality Assurance

SRS:System Test Report

1. **Funnctional Scope**

The testing scope for AAS Management is determined by the functions specified in the System Requirement Specification. This document serves as a guideline for the QA team to ensure that all the required functions are tested thoroughly. The results of the testing activities are documented in the System Test Report, which provides a comprehensive summary of the testing efforts and results. The report serves as a valuable resource for stakeholders to evaluate the quality of the application and determine its readiness for release.

|  |  |  |  |
| --- | --- | --- | --- |
| Req-ID | Functionality | Priority | T ID |
| AASM-REQ1: Default landing page - GUI Components | browse the package explorer with AAS content, log in to an account, access to a customized package explorer with an extension of more functions and dynamically customized content depending on the user's user group, and logout | A | T001,T002,T003, T004, T005,T006 |
| AASM-REQ2: Identity & Access Management | centralized management of identities and access permissions, with role-based access rights | A | T008,T012,T013, T014 |
| AASM-REQ3: AAS content data management | admin has the possibility to upload and delete AAS content | B | T009,T010 |
| AASM-REQ4: Search functionalities | find desired AAS content using search functions | A | T011 |
| AASM-REQ6: Error Display | Handle and report errors like no entries found, unexpected errors, etc. | A | T003 |
|  | Allow user to change Server-URL | A | T012 |

1. **Features**
2. **Overall Strategy and Approach**

**4.1 Testing Strategy**

The testing strategy for the AAS Management application involves a comprehensive approach that includes manual and automated testing techniques. The testing will begin with unit testing, followed by integration testing, system testing, user acceptance testing, and regression testing. The testing team will work closely with the development team to ensure that any defects or errors are identified and fixed promptly. The testing will be conducted in a controlled environment that mimics the production environment, and the goal is to ensure that the application meets all the established requirements, is free from defects and errors, and is ready for release to end-users.

**4.2 System Testing Entrance Criteria**

The System Testing Entrance Criteria for AAS Management consists of a set of prerequisites that need to be fulfilled before the start of the system testing phase. These criteria include the completion of unit and integration testing activities, the availability of a stable build of the application that includes all the required features and functionality, and the approval of the test plan and test cases by relevant stakeholders. Additionally, any necessary test environment setup and configuration must be completed, all required testing resources must be available, and all defect fixes from previous testing phases must be verified. Meeting these criteria is essential to ensure that the system testing phase can proceed effectively and efficiently and helps to avoid delays and additional costs in the testing process.

**4.3 Test Preparation**

Test preparation for the AAS Management application involves several steps to ensure that the testing process is effective, efficient, and comprehensive. These steps include:

1. Reviewing the System Requirement Specification to identify the scope of testing and requirements that need to be tested.
2. Developing a test plan that includes testing objectives, approach, scope, schedule, and resource requirements.
3. Identifying and prioritizing test cases based on their relevance, complexity, and impact on the system.
4. Preparing test data and test environment to simulate real-world scenarios and ensure accurate and thorough testing.
5. Identifying and acquiring necessary testing tools and resources to support the testing process, including automation tools, defect tracking tools, and testing environments.
6. Ensuring that the testing team has the necessary skills, expertise, and training to carry out the testing activities effectively.
7. Defining and establishing testing metrics to measure the effectiveness and efficiency of the testing process.
8. Ensuring that all stakeholders are aware of the testing process, its objectives, and expected outcomes.

By following these steps, the testing team can prepare thoroughly for the testing activities and ensure that the application is thoroughly tested, meets all the requirements, and is ready for release.

**4.4 Testing Types**

**4.4.1 Usabily Testing**

Usability testing is an important aspect of testing for the AAS Management application that focuses on the ease of use and user-friendliness of the application. It involves defining the objectives and goals of the usability testing, identifying representative end-users, developing usability testing scenarios, determining the methodology to be used, conducting usability testing sessions, analyzing the results, implementing necessary changes, and repeating the process to validate the effectiveness of the changes made. By conducting usability testing, the testing team can ensure that the application provides an excellent user experience, meets the needs of its end-users, and is user-friendly.

**4.4.2 Functional Testing**

Functional testing is a crucial aspect of testing for the AAS Management application as it ensures that the application's functions are working as intended and meet the specified functional requirements. The testing process involves identifying the functional requirements that need to be tested, developing test cases that cover these requirements, executing the test cases, documenting any issues or defects found, validating defect fixes, and repeating the process until all functional requirements have been tested and validated. By conducting thorough functional testing, the testing team can ensure that the AAS Management application is functioning correctly, meets all the specified requirements, and is ready for release.

**4.5 Suspension Criteria and Resumption Requirements**

Defining Suspension Criteria and Resumption Requirements can improve the efficiency, effectiveness, and organization of the testing process, resulting in higher-quality testing and more reliable testing results.

**4.5.1 Suspension Criteria:**

* Unresolved critical defects that may cause further defects.
* Unavailability of necessary testing resources such as hardware, software, or personnel.
* Any environmental issues that may affect the accuracy of testing results, such as network or server downtime.

**4.5.2 Resumption Requirement**

* The resolution of all critical defects identified during the testing process.
* Availability of necessary testing resources.
* Confirmation that the environmental issues that caused the suspension have been resolved.
* Retesting previously failed test cases to ensure that they pass after the suspension.

**4.6 Test Data**

**4.6.1 Test Data Types:**

* User profiles
* Transactions
* System configuration
* Security permissions

**4.6.2 Test Data Creation:**

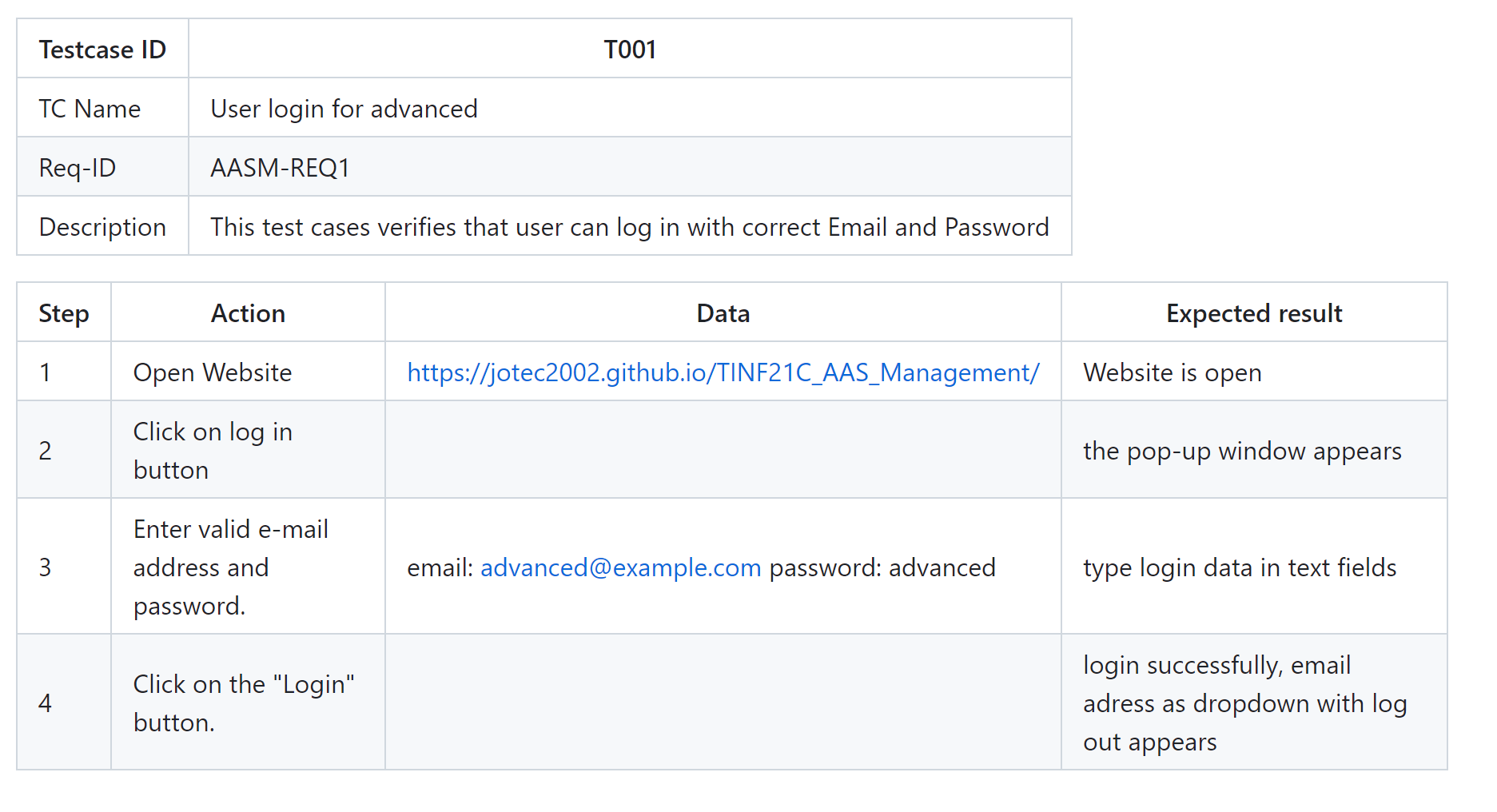
* User profiles will be created using realistic demographic information and academic history.
* Transactions will be created manually by the testing team to ensure all scenarios are covered.
* System configurations will be set up according to the specified requirements in the System Requirement Specification.
* Security permissions will be set up for different user roles such as guest, advance, and administrators.

**4.6.3 Test Data Management:**

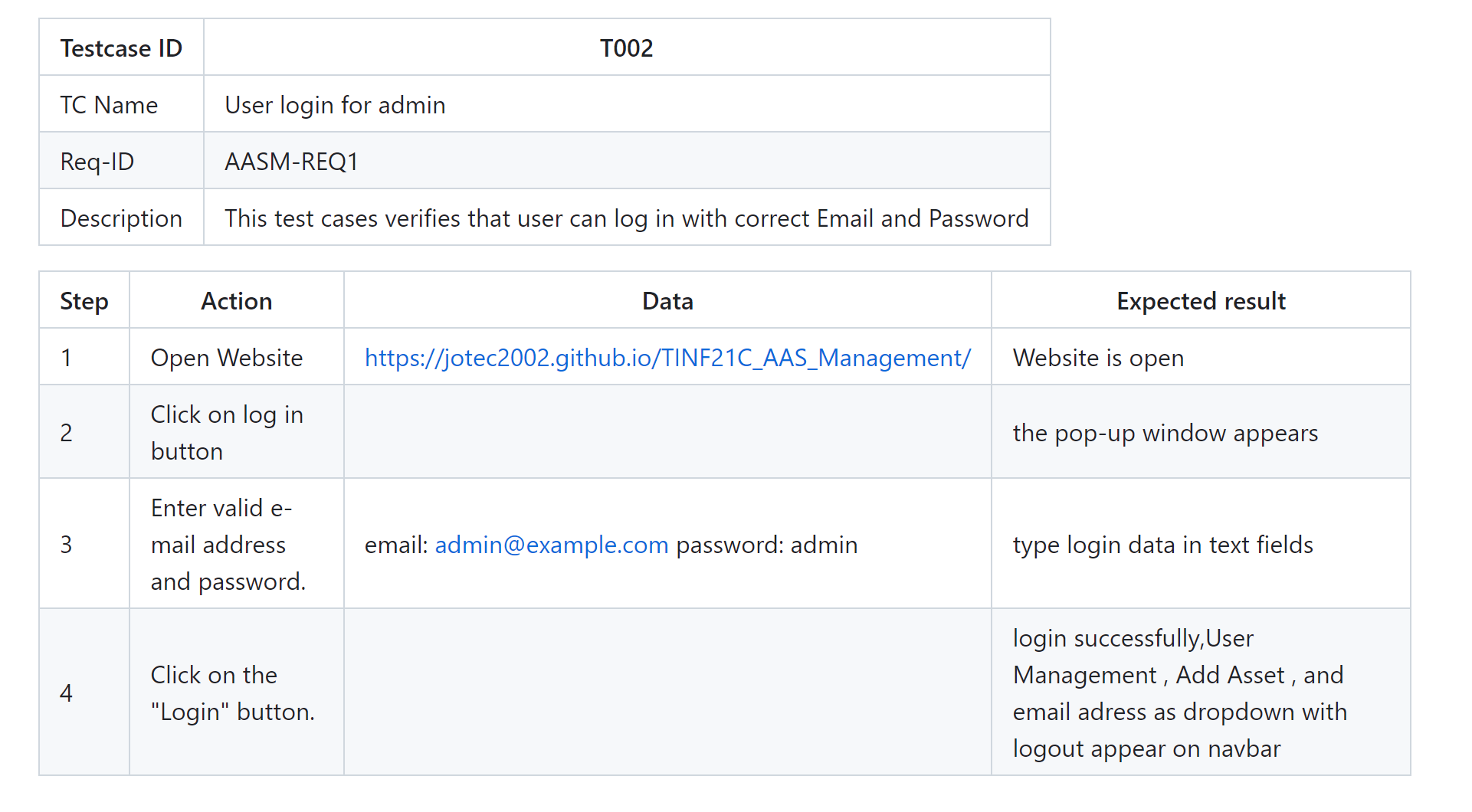
* All test data will be stored in a secure location with limited access to authorized personnel.
* The test data will be updated as needed to ensure that it remains relevant to the testing activities being performed.
* A tracking system will be used to keep track of all test data and ensure that it is being used correctly during testing.

1. **Executing Plan**

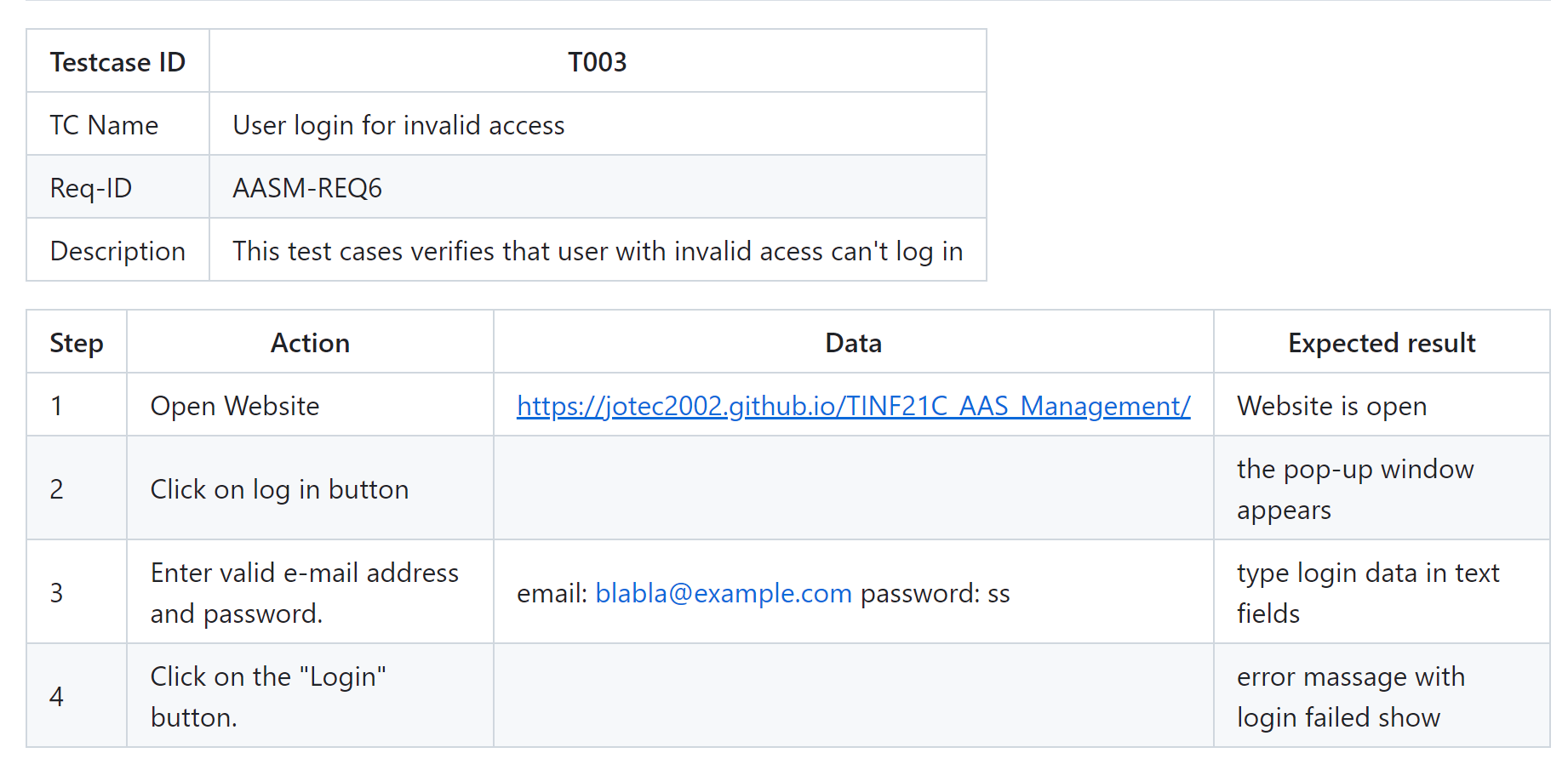
**5.1 Test Case User login for advanced**

****

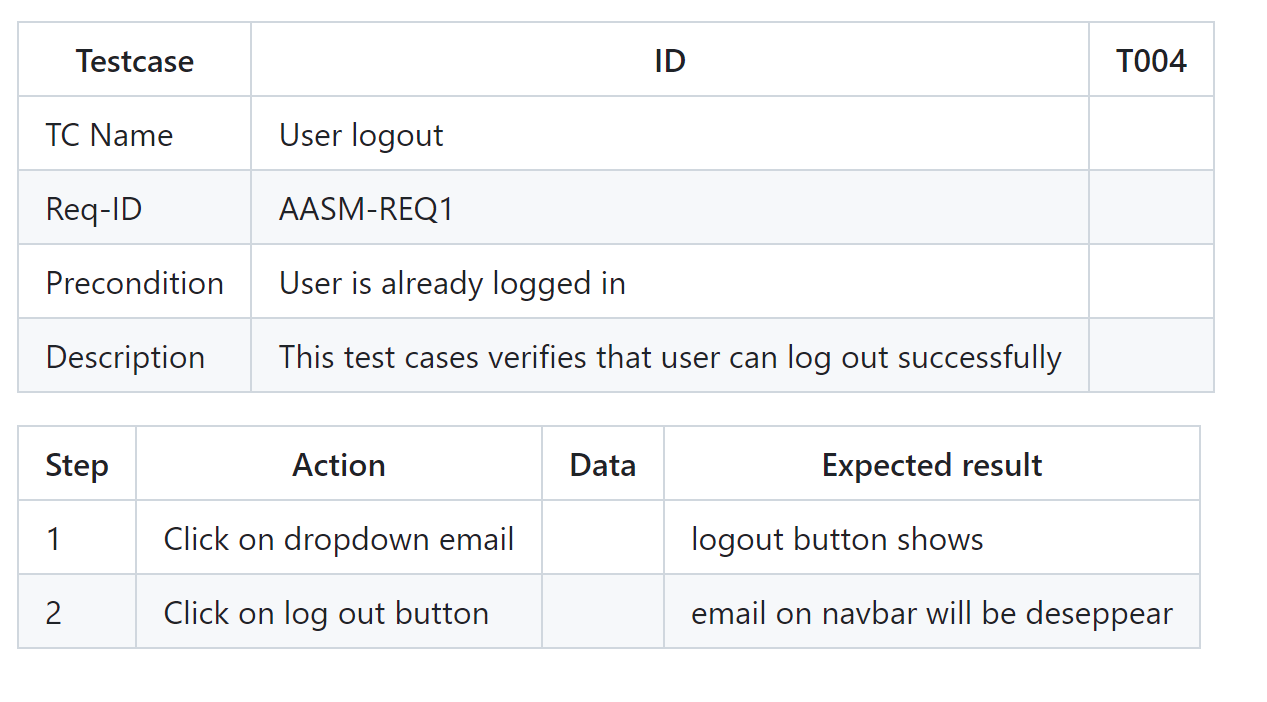
**5.2 Test Case User Login for Admin**

****

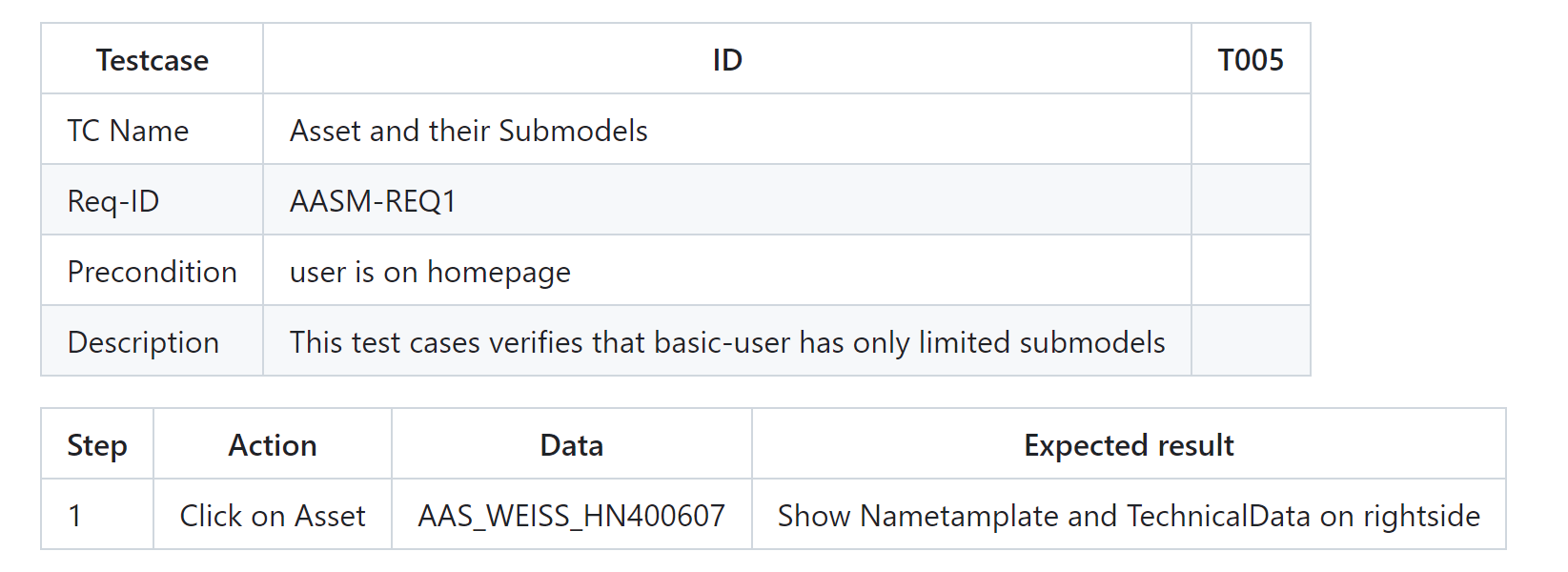
**5.3 Test Case User login for invalid access**

****

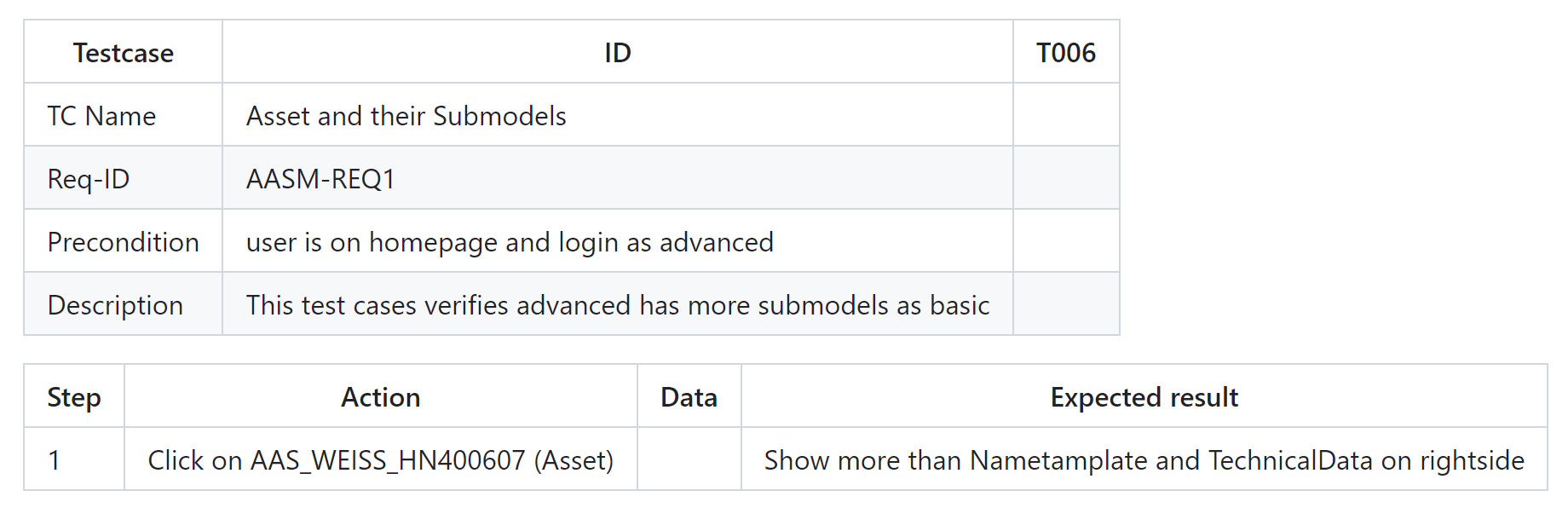
**5.4 Test Case Usr Logout**

****

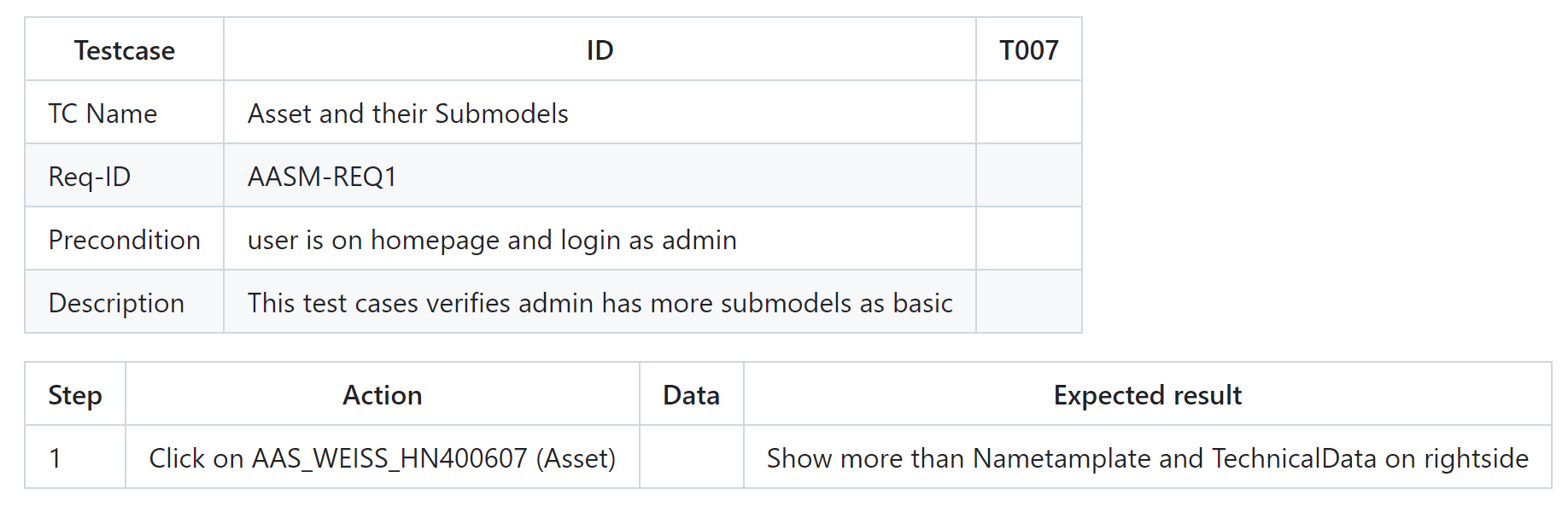
**5.5 Test Case Asset and their Submodels**

****

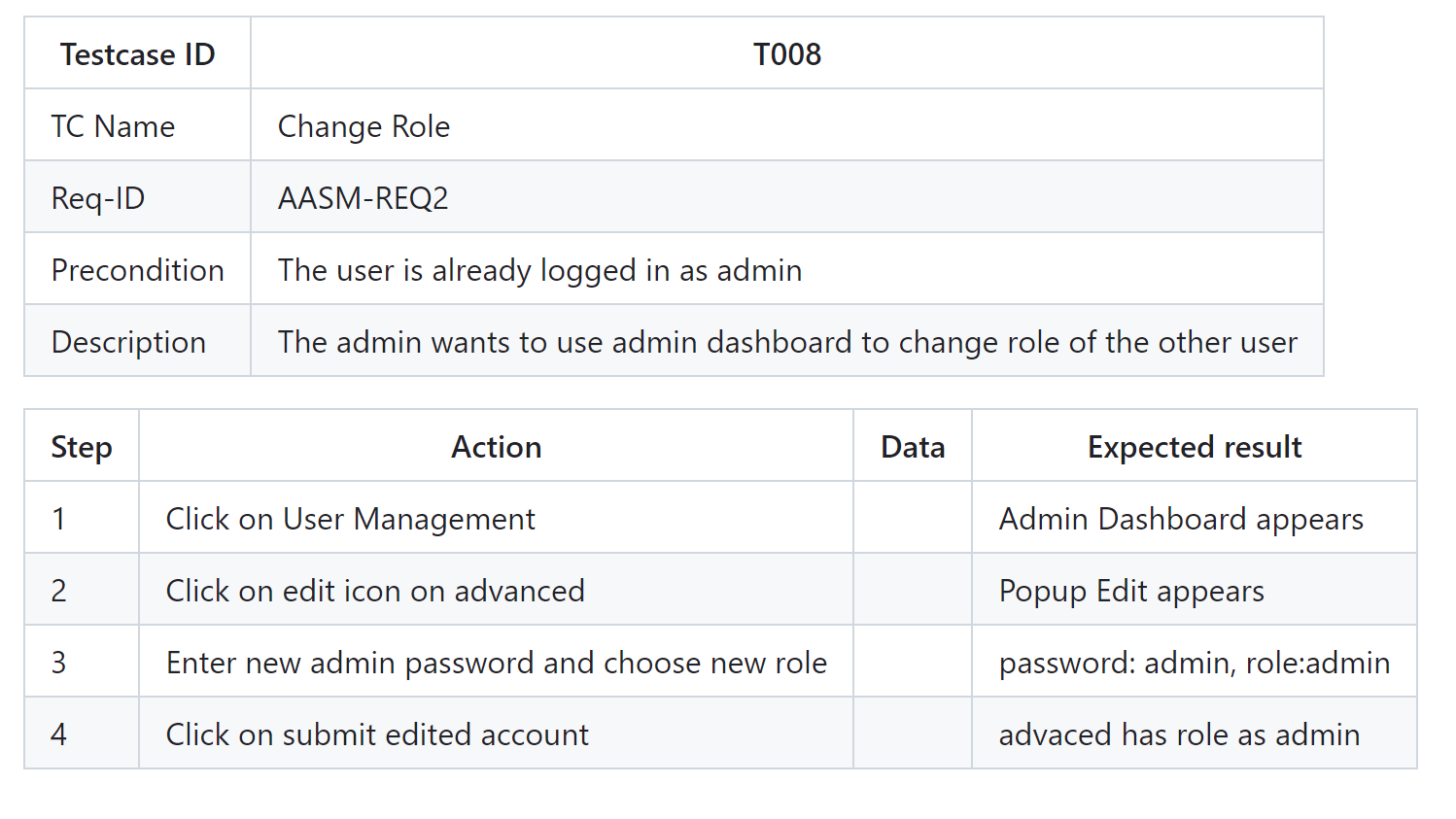
**5.6 Test Case Asset and their Submodels**

****

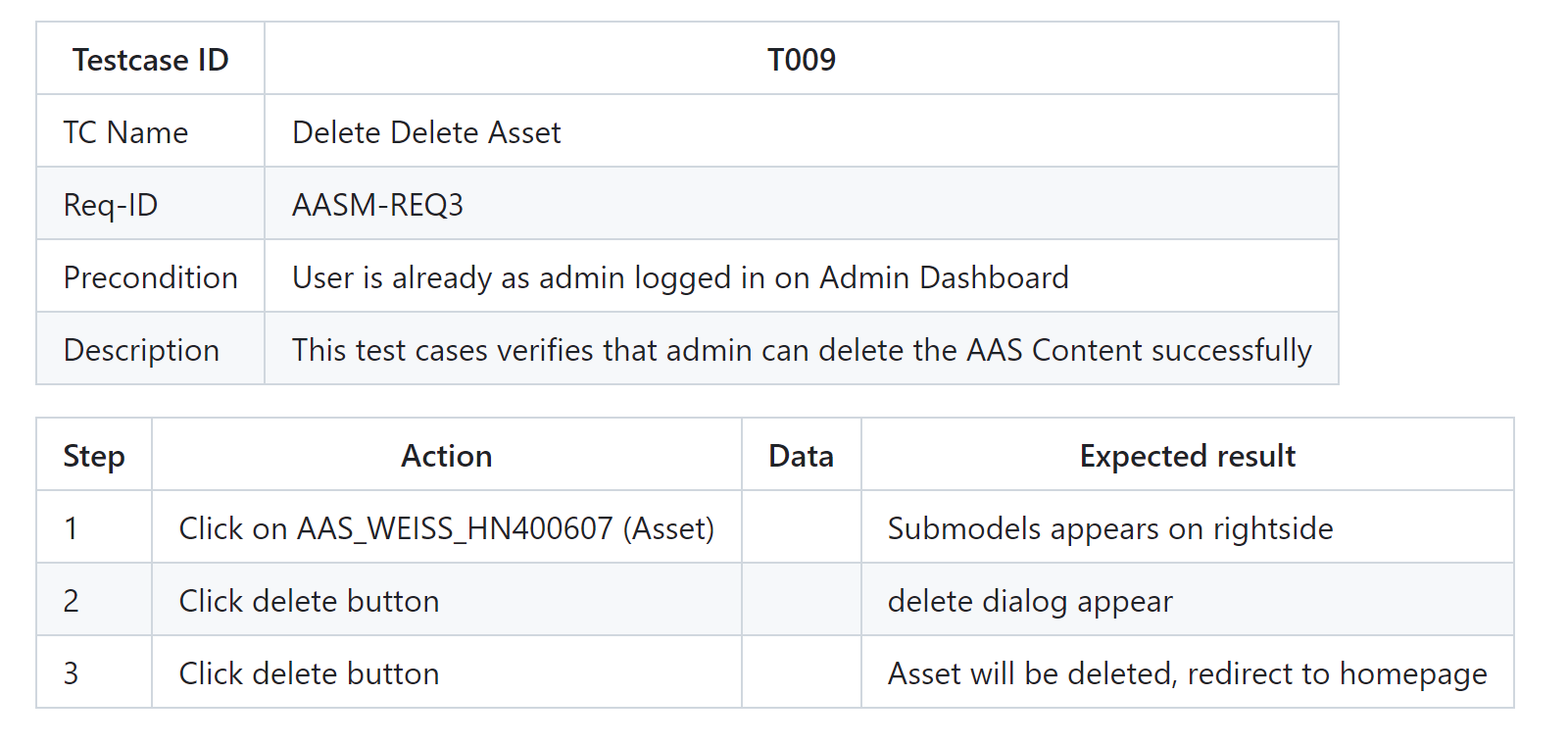
**5.7 Test Case Asset and their Submodels**

****

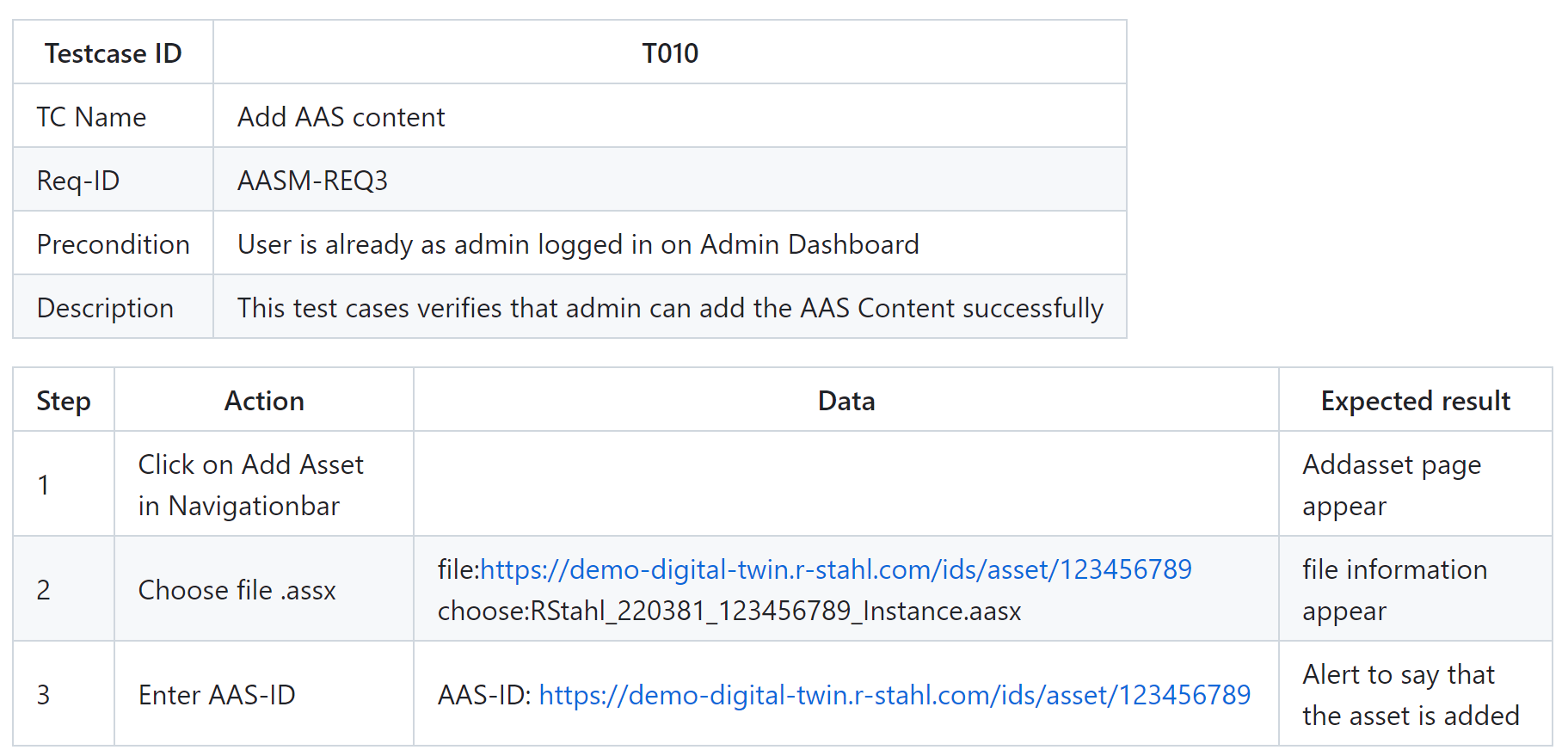
**5.8 Test Case Change Role**

****

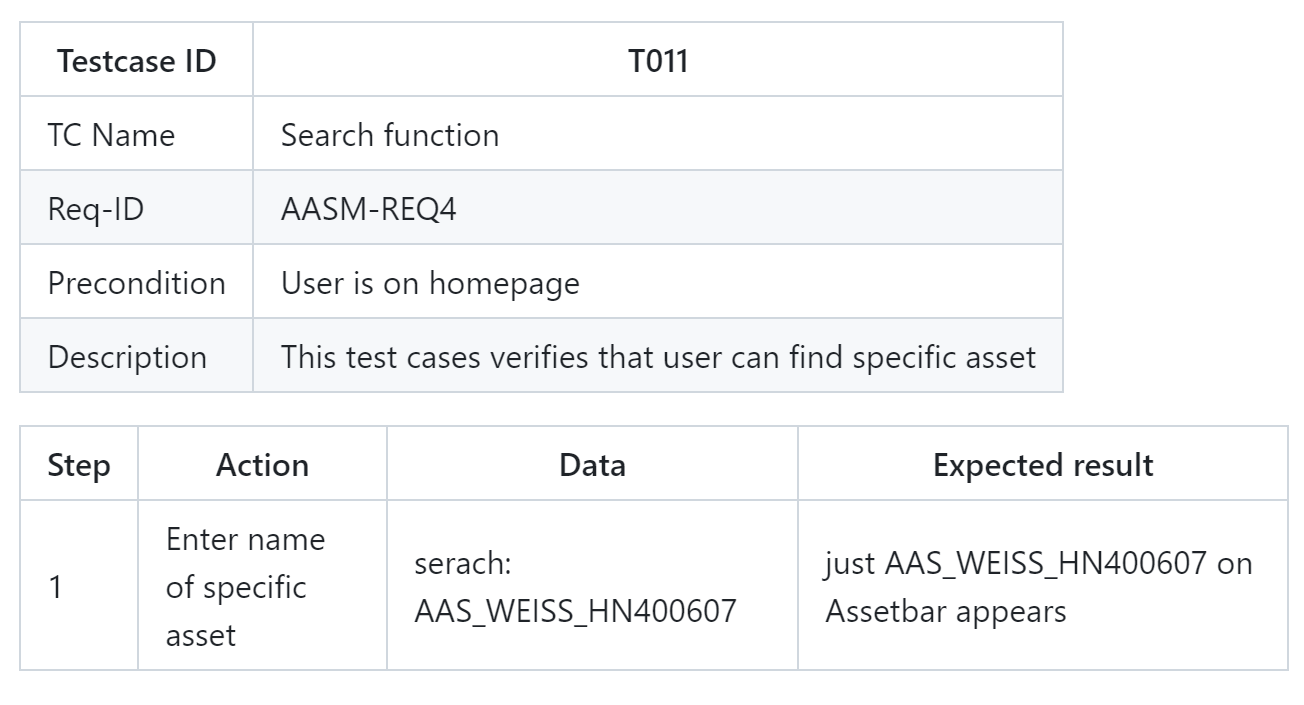
**5.9 Test Case Delete Asset**

****

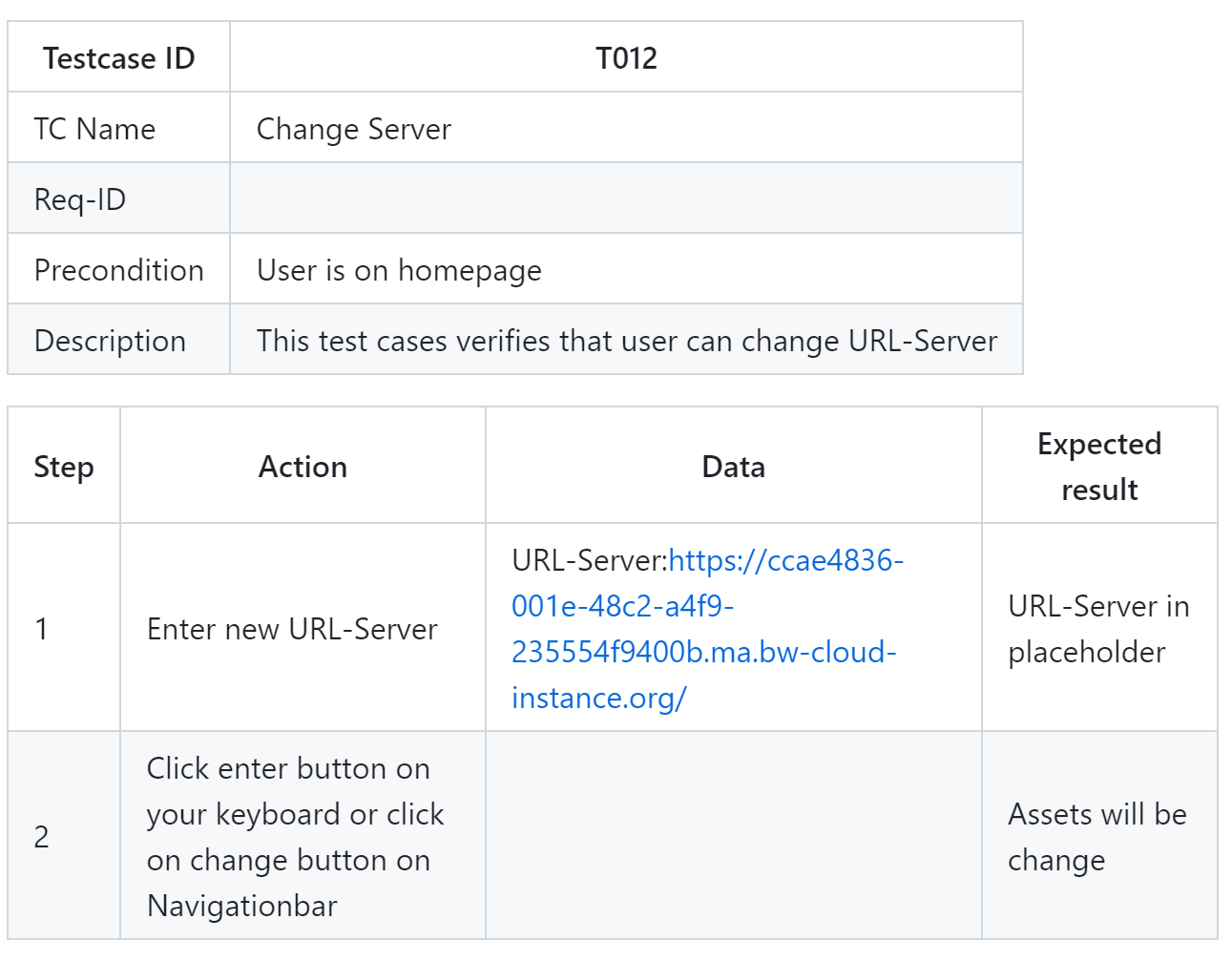
**5.10 Add Asset**

****

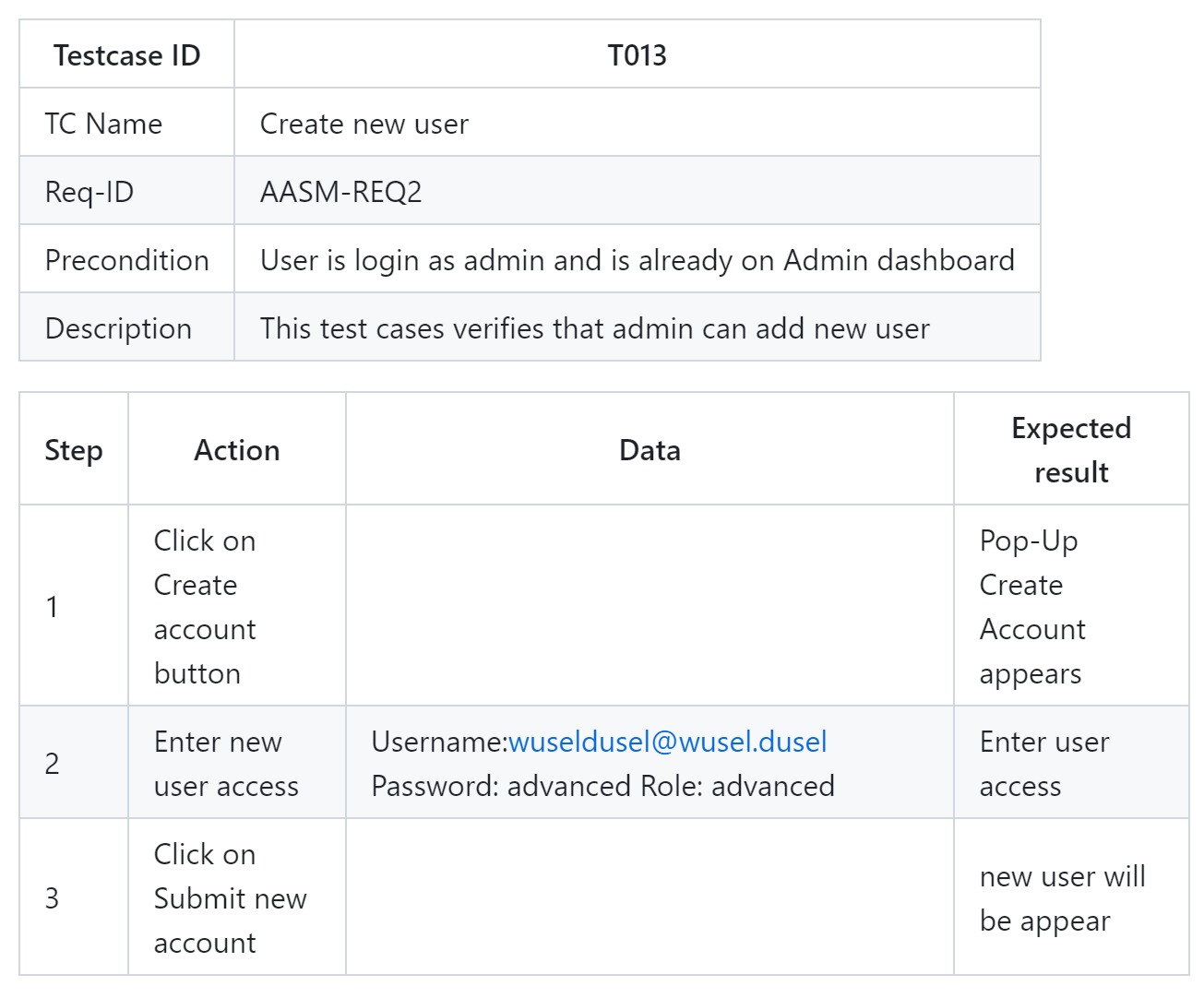
**5.11Test Case Search Asset**

****

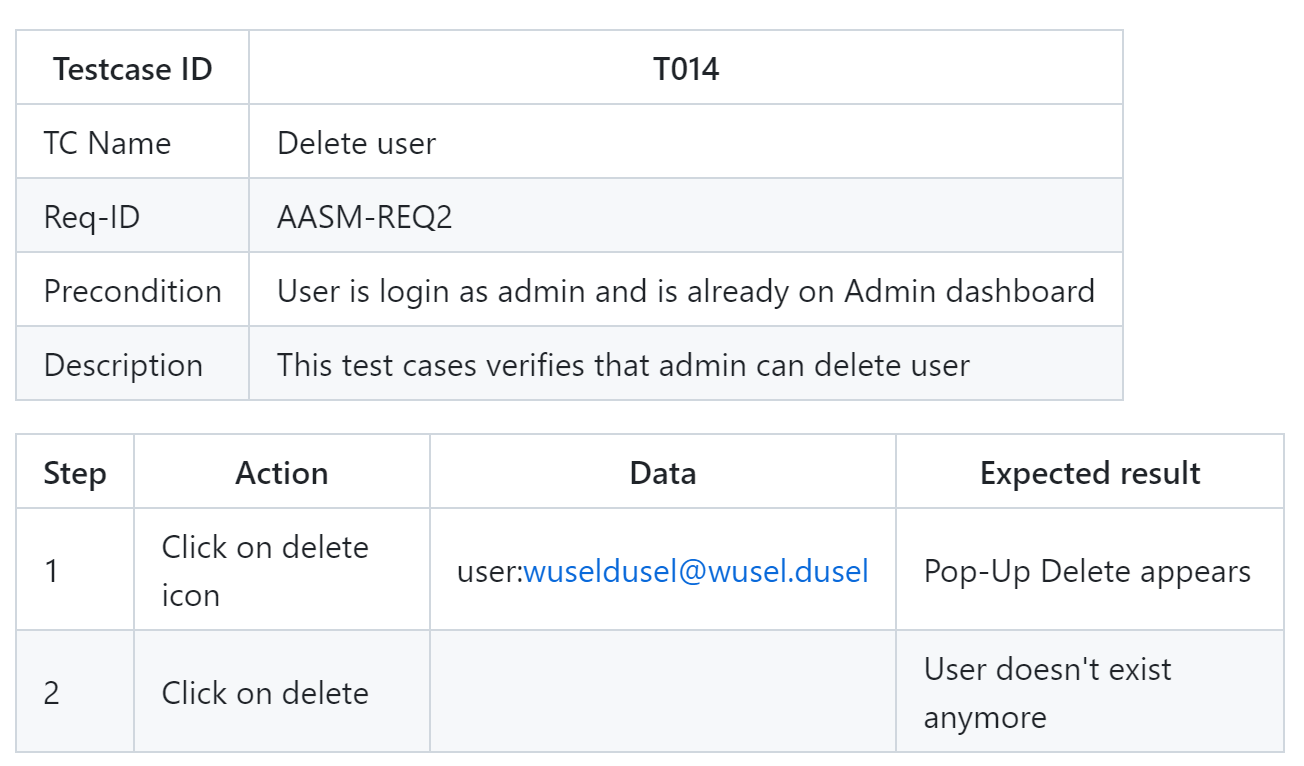
**5.12 Test CaseChange Server**

****

**5.13 Test Case Create New User**

****

**5.14 Test Case Delete**

****

1. **Defect Reporting**
   1. **Defect Tracking**

GitHub Issues will be to keep trackinng of any of failed tests.

* 1. **Defect Reporting and Reports**

 Defects will be reported by the System Test Manager to the Front End Development Team and Back End Development Team directly via arbitrary communication channel.

* 1. **Defect Management Prozess**

Defects discovered by testing will be fixed by the respective development team, status reports are delivered in the regularly scheduled meetings.

* 1. **Defect severity definitions**

used to categorize the severity of defects discovered during testing. In the context of the AAS Management application, defects are classified into three categories: critical, medium, and low.

Critical defects are those that cause catastrophic or severe errors, resulting in major problems and rendering the functionality unavailable to the user. Examples of critical defects include system abends, data corruption, and the inability to post data to the database. Remedying a critical defect may require a high level of effort or the implementation of a complex manual procedure.

Medium defects, on the other hand, do not significantly impair system functionality and can be categorized as medium severity. Examples of medium defects include incorrect form navigation and inconsistent field labels with global terminology. Remedying a medium defect may require a medium level of effort or the implementation of a less complex manual procedure.

Low defects are those that have little to no impact on system functionality and are considered cosmetic. Examples of low defects include repositioning fields on screens and incorrect text font on reports. Remedying a low defect may require a low level of effort or the implementation of a straightforward manual procedure.

By using these severity definitions, the testing team can prioritize defect remediation and ensure that critical defects are addressed first to prevent major problems and ensure the application's functionality is available to the user.

1. **Environment**

The testing approach for the AAS Management application involves cloning the most recent version of the code from the git repository and running it in a suitable program environment. Front-end testing will be conducted by loading the application in various browsers and testing its functionality with various inputs. The results of the testing will be documented in the System Test Report.

1. **Test Schedule and Budget**

The AAS Management system testing is scheduled for one week, from May 8, 2023 to May 12, 2023. The System Test Manager will conduct all tests during the first week, while retesting of defects will occur on the last day of system testing. The dates for defect retesting may be modified as necessary to ensure that all defects are retested and closed. While it is ideal to close all defects, it is not necessary as long as the defects are not critical. The primary objective is to minimize the number of open defects.