

GSFC ESDIS CMO
11/28/2018
Released

423-RQMT-014, Original, Revision -
Earth Science Data and Information Systems (ESDIS)
Project, Code 423

Earthdata Login Requirements Specification



Goddard Space Flight Center
Greenbelt, Maryland

National Aeronautics and
Space Administration

Earthdata Login Requirements Specification Signature/Approval Page

Reviewed by:

Signature obtained

Stephen Berrick
SSDO Manager
NASA GSFC Code 423

11/26/2018

Date

Signature obtained

Christopher Lynnes
ESDIS System Architect
NASA GSFC Code 423

11/27/2018

Date

Approved by:

Signature obtained

Andrew Mitchell
ESDIS Project Manager
NASA GSFC Code 423

11/28/2018

Date

**[Electronic] Signatures available in B32 Room E148
online at: / <https://ops1-cm.ems.eosdis.nasa.gov/cm2/>**

Preface

This document is under ESDIS Project configuration control. Once this document is approved, ESDIS approved changes are handled in accordance with Class I and Class II change control requirements described in the ESDIS Configuration Management Procedures. Changes to this document shall be verified by a document change notice (DCN) and implemented by change bars or by complete revision.

Any questions should be addressed to: esdis-esmo-cmo@lists.nasa.gov
ESDIS Configuration Management Office (CMO)
NASA/GSFC
Code 423
Greenbelt, Md. 20771

Abstract

This document provides the Earthdata Login (EDL) system requirements presented as a set of user stories following the Agile development approach.

Keywords: Earthdata Login, Authentication

Table of Contents

1	INTRODUCTION	1
1.1	Purpose	1
1.2	Scope	1
1.3	Related Documentation	1
1.3.1	Applicable Documents	1
1.3.2	Reference Documents	1
1.4	Agile Programming and Requirements Analysis	2
1.5	User Roles	2
2	REQUIREMENTS	3
2.1	URS FOUR	3
2.2	Apache Module	6
2.3	Glossary	7
	Appendix A Abbreviations and Acronyms	8

1 INTRODUCTION

The Earthdata Login (EDL) provides a centralized and simplified mechanism for user registration and profile management for all Earth Observing System Data and Information System (EOSDIS) system components. End users may register and edit their profile information in one location allowing them access to the wide array of EOSDIS data and services. The Earthdata Login also helps the EOSDIS program better understand the user demographics and access patterns in support of planning for new value-added features and customized services that can be directed to specific users or user groups resulting in a better user experience.

The Earthdata Login provides user registration and authentication services and a common set of user information to all EOSDIS data centers in a manner that permits the data center to integrate their additional requirements with the User Registration System (URS) services.

1.1 Purpose

The purpose of this document is to capture the system requirements for Earthdata Login.

1.2 Scope

This document covers requirements and user stories for EDL. This document does not contain release notes or any version reporting. This document does not cover test criteria for any requirements or user stories.

1.3 Related Documentation

The latest versions of all documents below should be used. The latest Earth Science Data and Information System (ESDIS) Project documents can be obtained from Uniform Resource Locator (URL): <https://ops1-cm.ems.eosdis.nasa.gov>. ESDIS documents have a document number starting with either 423 or 505. Other documents are available for reference in the ESDIS project library website at: http://esdisfmp01.gsfc.nasa.gov/esdis_lib/default.php unless indicated otherwise.

1.3.1 Applicable Documents

The following documents are referenced within or are directly applicable, or contains policies or other directive matters that are binding upon the content of this document.

Document Number	Document Title
423- CDRD- EED2	Contract Data Requirements Document for EOSDIS EED2
N/A	EED2 Statement of Work for Providing Development,

	Sustaining Engineering, and Continuous Evolution of the Earthdata
--	---

1.3.2 Reference Documents

The following documents are not binding on the content but referenced herein and, amplify or clarify the information presented in this document.

Document Number	Document Title
N/A	Earthdata Login documentation
N/A	Apache Module documentation

1.4 Agile Programming and Requirements Analysis

The EOSDIS Evolution and Development 2 (EED2) teams are using Agile Programming methodologies throughout the software development lifecycle, including the requirements analysis phase. The process of developing requirements has not changed much from other software programming management methods such as waterfall or iterative methods. The two major differences between Agile and other methods are:

- a. A greater emphasis on stakeholder participation; and
- b. Requirements are allowed to evolve throughout the Agile development process to account for changing priorities, new requirements that were left out initially, or changes due to issues encountered during development and testing.

These two differences may not seem all that different from other methods; however, requirements managed by Agile processes provide for a much more flexible framework from which to develop software.

In this document, requirements may/or may not be captured as a set of User Stories written in the form:

"As a <role> I want to <action> so that I can <explanation>".

The requirement may be told from the perspective of an EDL user, or developer role instead of the generic "The system shall...". Requirements written in this form help to refine the system requirement by identifying the user or consumer of the required functionality, the function that the user needs, and why this function is needed. The concept of providing value to the system user is reflected in the explanation of the user story form. This provides the rationale or the benefit of the requirement.

This also allows the developer to choose from "Best Practices" techniques on how to best implement a requirement. The first principle of the Agile Manifesto is:

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software. (<http://agilemanifesto.org/principles.html>)

The EDL requirements and user stories will be tracked and maintained using Atlassian's Jama Requirements Management tool. The user stories presented in this document are written to a Systems Requirements level and will be kept in sync with the user stories maintained in Jama.

1.5 User Roles

The following roles will be used to clarify Earthdata Login User Stories by defining system boundaries and those that are responsible for completing the tasks associated with these requirements:

- a. EDL Administrator - Administrator for Earthdata Login
- b. Client Application Owner - Owner of a Client Application that authenticates using EDL
- c. EDL User - User who logs into a Client Application via Earthdata Login
- d. Application Creator - EDL User who has authorization to create a Client Application that authenticates using EDL
- e. User Services - EDL user who has authorization to search for users within EDL

2 REQUIREMENTS

2.1 URS FOUR

This is the currently implemented version of Earthdata Login formerly known as URS Four

Story ID	User Story
URS-STORY-8	As a user, I can access URS 4.0 at urs.earthdata.nasa.gov
URS-STORY-9	As an app admin, I can add other app admins to my app
URS-STORY-12	As a registered user, I can see a list of apps which I have authorized
URS-STORY-13	As a registered user, I can revoke app authorizations.
URS-STORY-14	As a client app, I can fetch user profiles from EDL
URS-STORY-15	As a client app, I can search for users with certain properties.
URS-STORY-16	As a client app, I can set custom properties for a user in EDL
URS-STORY-17	As a client app, I can use the authorization code given by EDL to obtain an app token to use the EDL API
URS-STORY-18	As a registered user, I can update my EDL password.
URS-STORY-19	As a registered user I can login to the main EDL page directly and view my EDL profile.
URS-STORY-20	As an EDL admin user, I can edit any EDL user's profile.
URS-STORY-21	As an app admin, I can set my app's password.
URS-STORY-23	As an app admin, I can see my app's client ID.
URS-STORY-26	As a registered user, I can edit my basic EDL profile.
URS-STORY-27	As a user, when I submit my registration with incomplete information, I get informative feedback so that I can correct the errors and submit again.
URS-STORY-28	As a registered user, I can see a list of my authorized apps and a link to authorize a new app
URS-STORY-29	As an EDL admin user, I can access profile information for all EDL users.
URS-STORY-31	As a client app, I can send a user to EDL to sign in or create an account, authorize the app, and be redirected back to my app with an 'API authorization code'
URS-STORY-32	As a user, when I am not signed in and visit EDL in my web browser, I see a link to register for a new account.

Story ID	User Story
URS-STORY-34	As a user, when I click the link to register an account, I see a form that allows me to provide the information needed to register.
URS-STORY-36	As an EDL admin, I can see a list of all registered apps.
URS-STORY-42	As an EDL user, I can provide free text information for Affiliation in user profile information.
URS-STORY-75	As a EDL admin, I can see a graph of authentications over time.
URS-STORY-76	As a EDL admin, I can see some graphs of authorizations over time
URS-STORY-77	As an app admin, I can see metrics for my application.
URS-STORY-79	As an app admin, I can delete one of my applications.
URS-STORY-80	As an app admin, I can configure my app to have more than one redirection URL, so that I can have multiple environments (production and test) for my app
URS-STORY-81	As a client app, I can invoke an API call for javascript to determine whether a user has authorized the application
URS-STORY-84	As a metrics user, in the subject of the EDL metrics email, I can see the EDL environment that the metrics are for.
URS-STORY-85	As a registered user I can search for applications to approve based on application name.
URS-STORY-87	As an app admin, I can receive an email when my app is deleted.
URS-STORY-89	As an application creator, any app I create is automatically approved.
URS-STORY-90	As EDL, I can only add registered users with Application Creator role as app admins.
URS-STORY-91	As an EDL user with Application Creator role, I can create and manage client applications.
URS-STORY-92	As EDL, I can scan uploaded images for viruses before processing them further.
URS-STORY-95	As an app admin, I can to view/edit my application specific metadata.
URS-STORY-96	As a EDL user who can see user information (app admin or EDL admin), I can see user's email address as part of a user's information
URS-STORY-97	As an EDL user who can see user information (app admin or EDL admin), I can sort the user list based on any listed user attribute on the 'Users' tab

Story ID	User Story
URS-STORY-99	As an EDL user who can see user information (app admin or EDL admin) I can page through user list on application admin page.
URS-STORY-101	As EDL, I provide NetInsight metrics
URS-STORY-102	As an app admin, I can search for users that have approved my application
URS-STORY-103	As an app admin, I can export a list of users that have authorized my application along with customizable list of their attributes
URS-STORY-104	As a client app, I can invoke an API call to verify whether a user ID exists in EDL
URS-STORY-105	As an app admin, I can see a list of 'primary users' of my application.
URS-STORY-107	As EDL, I provide Google Analytics metrics
URS-STORY-108	As a registered EDL user, I can pre-approve a client application.
URS-STORY-110	As a client app, I can invoke EDL javascript on my site using CORS headers
URS-STORY-114	AS EDL, I provide metrics on which applications are using URS4 for authentication, so I can gather information on what applications are driving the most traffic to URS4.
URS-STORY-116	As EDL, I provide documentation for app admins.
URS-STORY-117	As a client app, I can configure EDL to allow access to restricted API from my site.
URS-STORY-121	As a client app, I can update my restricted API configuration in EDL to allow/disallow access to restricted API from my site.
URS-STORY-192	As a registered user, I can opt-out of email notifications for a client application.
URS-STORY-123	As a registered EDL user with User Services role, I can search for any EDL users.
URS-STORY-124	As a client app, I can get username/password from my user and authenticate via EDL using its Resource Owner Password Credentials (ROPC) implementation of OAuth2
URS-STORY-125	As an app admin, I can reset the password for my application.
URS-STORY-126	As EDL, I provide metrics from DAP Google Analytics
URS-STORY-128	As EDL, my users can be migrated to another environment.

Story ID	User Story
URS-STORY-132	As EDL, I provide an EDL path that simulates the work flow for OAuth2/authorize with a default test application, so that client application administrators can understand/test the process.
URS-STORY-135	As a client app, I can require my users to provide value for configurable list of user profile attributes.
URS-STORY-136	As EDL, I can automatically unlock previously locked users (due to too many erroneous login attempts) after a configuration period of time.
URS-STORY-137	As a client app, I can see how long my user has been a registered EDL user.
URS-STORY-139	As a registered user, I see a list of client applications under "Approved Applications" and "Application Administration" in alphabetical order.
URS-STORY-143	As a registered user, I can search for EDL client applications that I have not yet approved.
URS-STORY-146	As a registered user, I can add/edit application required fields while approving an application.
URS-STORY-150	As a client app, I can redirect to EDL for additional EULA/Sentinel authorizations.
URS-STORY-152	As a registered user, I can login using my username as well as email address.
URS-STORY-174	As a client app, I can see the number of client applications a user has authorized as part of their user profile returned via API.
URS-STORY-176	As an app admin, I can create and edit a new user group via the EDL GUI.
URS-STORY-177	As a app admin, I can add or delete a user from a user group via the EDL GUI.
URS-STORY-178	As an app admin, I can view (via EDL GUI) and retrieve (via API), a list of all user groups for my client application.
URS-STORY-179	As an app admin, I can see a list of groups my user belongs to.
URS-STORY-180	As a client app, I can add a user to a group for my application via the EDL API.
URS-STORY-181	As a client app, I can retrieve a list of groups to which my user belongs via the EDL API.
URS-STORY-182	As a client app, I can retrieve a list of users in any of my groups via the EDL API.
URS-STORY-183	As MMT, I can access user profile information for all EDL users.

Story ID	User Story
URS-STORY-184	As an app admin, I can see the date/time a user has logged into my app on the Users page for my application.
URS-STORY-185	As an App owner, I can request last logged-in date/time to be an exportable attribute.
URS-STORY-186	As an App admin, on the Users page for my application, I can view all users logged-in in the last month, etc.
URS-STORY-188	As URS Admin, I should be able to request a password reset link, as many times as needed to resolve a user's issue
URS-STORY-189	As an App Creator, I can specify what user attributes are required before a user can approve my application.
URS-STORY-190	As an unregistered user, I can optionally agree to Meris and Sentinel Eulas during registration.

2.2 Apache Module

The Apache Module is a drop-in module for Apache httpd server that provides basic client functionality to authenticate with Earthdata Login. It can be used to provide Earthdata Login authentication control for one or more independent resources being served by an Apache httpd server. Driven entirely by configuration, it can be used to protect files, directories, or entire applications, without requiring any change to the underlying resource.

This section contains the stories for the Apache Module component.

Story ID	User Story
URS-STORY-115	The Apache Module shall support server aliases
URS-STORY-162	The Apache Module shall provide a mechanism to require user login via Earthdata Login when a client performs a GET request for web pages served by the Apache http web server
URS-STORY-163	A user, once logged in, shall not be required to login in again until a configurable time period has elapsed.
URS-STORY-164	The Apache module shall terminate a user session if the session has no activity during a configurable time interval
URS-STORY-165	The Apache Module shall provide a mechanism for the user to end a session that can be invoked via a URL

Story ID	User Story
URS-STORY-166	The Apache Module shall provide a means to share user sessions between different instances of the httpd server.
URS-STORY-167	The Apache Module can support login via httpd configuration, so static web pages can also require login without changes to the page
URS-STORY-168	The Apache Module shall be configurable using apache configuration directives
URS-STORY-169	The Apache Module shall make EDL user profile information available to web pages via http sub-process environment
URS-STORY-170	The Apache Module shall be capable of acting in a reverse-proxy setup so that it can provide user login services to applications hosted on other servers
URS-STORY-171	The Apache Module shall support mod_autoindex providing directory listings
URS-STORY-172	The Apache Module shall be configurable to point to any instance of Earthdata Login.
URS-STORY-187	As Apache Module, I return any error_msg on redirect for oauth/authorize requests as a parameter.

2.3 Glossary

Glossary of terms with specific defined meanings in the context of the Earthdata Status application.

Term	Definition
app admin	Administrator of a Client Application within EDL
client app	Client Application within EDL
URS	Earthdata Login was formerly called URS. Any place URS is used can be replaced by EDL
authorization code	Code returned by EDL to a OAuth2 client application on authorization request
Client ID	Unique identifier for a Client Application within EDL
app token	Client credential token retrieved by a Client Application from EDL to use for subsequent EDL API calls
token	OAuth2 token retrieved from EDL by a Client Applications for a specific user
User Profile	User attributes like name, email address, etc., and optionally Client Application specific custom attributes
Registered	EDL user

Term	Definition
user	
EULA	End User License Agreement between the Client Application and EDL user
EDL User	End user of Earthdata Login

Appendix A Abbreviations and Acronyms

API	Application Programming Interface
CCB	Configuration Change Board
CCR	Configuration Change Request
CMO	Configuration Management Officer
CORS	Cross-Origin Resource Sharing
DAP	Delivered Algorithm Package
DCN	Document Change Notice
EDL	Earthdata Login
EED2	EOSDIS Evolution and Development 2
EISOC	Code 423 EOS Information Security and Compliance Office
EOSDIS	Earth Observing System Data and Information System
ESDIS	Earth Science Data and Information System
EULA	End User License Agreement
GSFC	Goddard Space Flight Center
GUI	Graphical User Interface
HTTPS	Hypertext Transfer Protocol
ID	Identifier
MMT	Metadata Management Tool
NASA	National Aeronautics and Space Administration
ROPC	Resource Owner Password Credentials
URL	Uniform Resource Locator
URS	User Registration System
URS4	URS Four