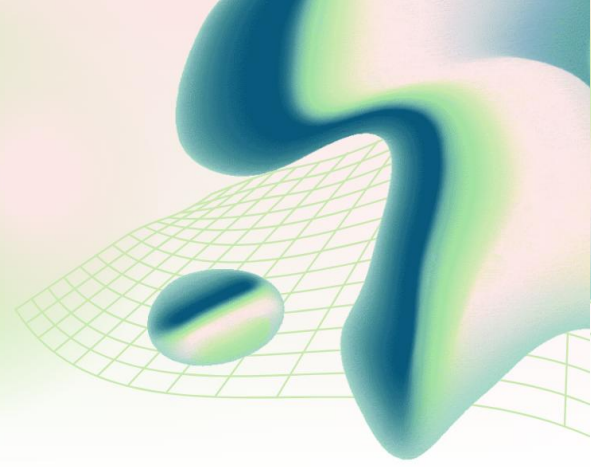


# GRS USER GUIDE



**GRS:** User Guide

**Version:** 3.1

**Author:** Rama Krishna K

**Date:** 09-November-2025

**Confidentiality:** Internal Use Only

## Table of Contents

1. [Summary](#)
2. [Introduction](#)
3. [System Overview](#)
4. [User Roles and Permissions](#)
5. [Interface Overview](#)
6. [Working with GRS](#)
  - 6.1 [Uploading Design Logs](#)
  - 6.2 [Validating Records](#)
  - 6.3 [Editing and Deleting Records](#)
  - 6.4 [Migrating Validated Records](#)
7. [Integration with SQL and Facets](#)
  - 7.1 [Front-End and Back-End Relationship](#)
  - 7.2 [GRS Process Flow](#)
8. [Troubleshooting](#)
9. [Best Practices](#)
10. [FAQs](#)
11. [Glossary](#)
12. [Document Control](#)

# 1. Summary

The GRS is a healthcare data management application that simplifies and automates configuration activities within the [Facets](#).

- It allows configuration teams to upload [design logs](#), validate records automatically, and migrate approved data between environments without manually updating Facets or [SQL database](#).
- By integrating with both the Facets front-end application and the SQL back-end database, the GRS ensures data consistency, auditability, and process efficiency
- across all stages of configuration.

## 2. Introduction

### Purpose

This guide provides step-by-step instructions for using the GRS. It is designed for Configuration, QA, and IT professionals who manage and validate design logs for healthcare products within the Facets.

### Background

Before the **GRS** was developed, the configuration team manually entered or updated each record from the design log in Facets or SQL.

- **Facets:** Supported single-record edits but required significant manual effort.
- **SQL:** Supported bulk operations but required technical expertise.

This manual process was time-consuming, prone to errors, and lacked automated validation. To resolve these issues, the **software team developed the GRS** — a centralized interface that allows users to **upload, validate, correct, and migrate design logs** quickly and securely.

#### How GRS Simplifies Configuration Work:

- Automatically validates uploaded data.
- Highlights errors for easy correction.
- Migrates validated data to Facets environments.
- Reduces manual SQL or Facets work.
- Maintains complete audit trails.

## 3. System Overview

### GRS Architecture

The GRS connects three key components:

1. **GRS Front-End:** Used by configuration users to upload and validate design logs.
2. **SQL Database (Back-End):** Stores validated records, supports bulk edits, and maintains audit history.
3. **Facets Application (Front-End):** Reflects validated and migrated data, allowing single-record edits.

### Workflow

1. Upload design log (.csv or .xlsx).
2. System performs multi-stage validation:
  - Format validation
  - Business rule validation
  - Relational validation
3. Validated data is stored in SQL database.
4. Configuration users correct failed records.
5. Admins migrate validated data to Facets via secure APIs.

### Supported Environments

- 1D (Development) – Validation and testing.
- 1R (QA) – Quality review and approval.
- 1P (Production) – Final deployment.

## 4. User Roles and Permissions

Role	Responsibilities	Access Level
Configuration User	Uploads and validates design logs; corrects failed records.	Upload, Validate, Edit
QA User	Reviews and approves records for migration.	Validate, Comment
Admin	Approves migrations and manages user permissions.	Full Access

## 5. Interface Overview

The GRS interface includes the following sections:

- **Dashboard:** Displays overall validation, migration, and error statistics.
- **Upload Design Log:** Interface to upload .csv or .xlsx files.
- **Validation Table:** Displays results of validation with color-coded status.
- **Migration Panel:** Used to initiate data migration to Facets.
- **Activity Log:** Tracks all user actions (upload, edit, delete, migrate).

## 6. Working with GRS

### 6.1 Uploading Design Logs

1. Log in to the **GRS**.
2. Navigate to **Upload Design Log**.
3. Select the .csv or .xlsx file to upload.
4. Click **Upload**.
5. Review the validation summary.

**Tip:** Include version and date in the filename (e.g., DesignLog\_2025-10-28\_v2.xlsx).

### 6.2 Validating Records

- GRS automatically validates each record during upload.
- Failed records are shown in red with detailed error messages (e.g., VAL-PLAN-001: Invalid Plan Code).
- Correct the records directly in GRS or re-upload the corrected file.
- Re-run validation until all records pass.

### 6.3 Editing and Deleting Records

- To edit: Click the **Edit** icon, make corrections, and click Save.
- To delete: Click **Delete** and confirm the deletion.
- Every change is logged with user ID, timestamp, and record ID.

## 6.4 Migrating Validated Records

- Only validated records can be migrated.
- Migration path: 1D → 1R → 1P.
- Admins approve migration requests.
- The triggers secure REST API calls to push validated data into Facets staging schemas.

## 7. Integration with SQL and Facets

### 7.1 Front-End and Back-End Relationship

The GRS bridges both systems:

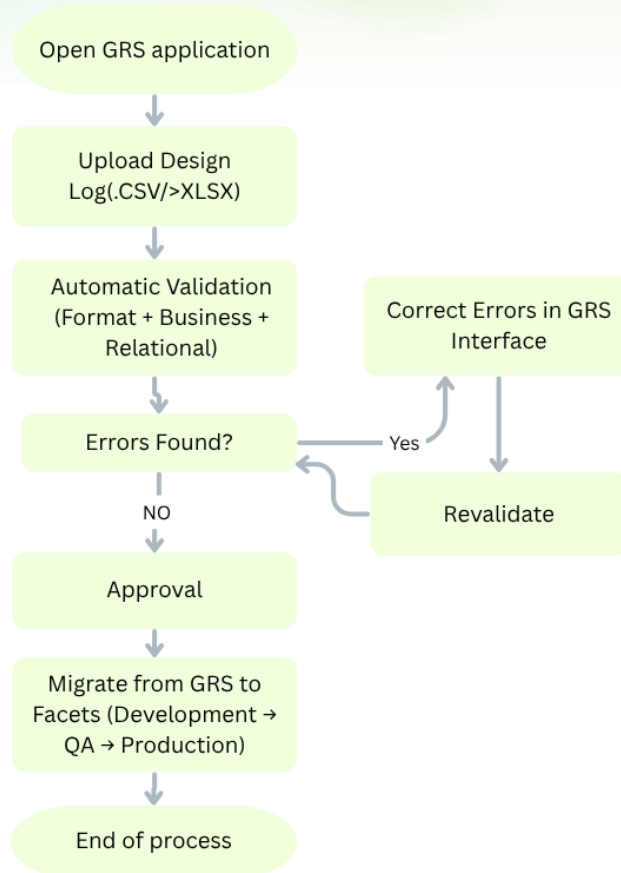
Function	System	Description
Single Record Editing	Facets Application	Used for individual record updates after migration.
Bulk Record Editing	SQL Database	Enables large-scale record updates at the database level.
Validation and Migration	GRS	Centralized platform for validation and migration.

### 7.2 GRS Process Flow

1. Open **GRS** Application
2. Upload **Design Log** (.CSV/.XLSX)
3. Automatic Validation (Format + Business + Relational)
4. Errors Found?
  - **If Yes:** Go to **Step 5**
  - **If No:** Go to **Step 7**
5. Correct Errors in GRS Interface
6. Revalidate: Return to **Step 4**
7. Approve
8. Migrate from GRS to Facets (Development → QA → Production)
9. End of process

## GRS Process Flow Chart

“The following diagram illustrates the end-to-end workflow from upload to migration.”



## 8. Troubleshooting

Issue	Cause	Resolution
Upload Error 400	Header mismatch	Ensure file matches' template.
Validation Timeout	File size too large	Split and re-upload smaller files.
Migration Blocked	Insufficient permissions	Contact Admin.
SQL Sync Delay	API or network lag	Wait for sync or retry later.

## 9. Best Practices

- Validate all records in **1D** before migrating to **QA**.
- Keep **backup copies** of all uploaded design logs.
- Avoid manual edits in Facets after migration.
- Monitor **Activity Logs weekly** for auditing.
- Ensure file templates are always up to date.

## 10. FAQs

### Q1. Why was the GRS created?

**A:** To automate configuration data uploads, reducing manual Facets and SQL work.

### Q2. Can I still update data in SQL or Facets?

**A:** Yes. SQL allows bulk updates, while Facets allows single-record edits. However, using GRS ensures validations and audit compliance.

### Q3. How does GRS ensure accuracy?

**A:** Through layered validation rules and secure data synchronization between SQL and Facets.

### Q4. How are deleted records tracked?

**A:** Deleted records are logged in SQL with user ID and timestamp for auditing.

## 11. Glossary

Term	Definition
<b>1D / 1R / 1P</b>	Development, QA, and Production environments.
<b>Design Log</b>	A configuration data file (.csv/.xlsx) uploaded to GRS.
<b>Facets</b>	A healthcare application used to manage member, provider, and plan configuration data across different environments
<b>SQL Database</b>	Back-end data storage used for validation and auditing.
<b>API</b>	Application Programming Interface enabling system communication.
<b>Validation</b>	Automatic checking of uploaded data against system rules.

## 12. Document Control

Version	Date	Author	Description
1.0	15-Sep-2025	Rama Krishna K	Initial Release
2.0	30-Sep-2025	Rama Krishna K	Reorganized Structure, Enhanced Clarity, Professional Formatting
3.0	15-Oct-2025	Rama Krishna K	SQL–Facets Integration
3.1	09-Nov-2025	Rama Krishna K	Flow chart added

