



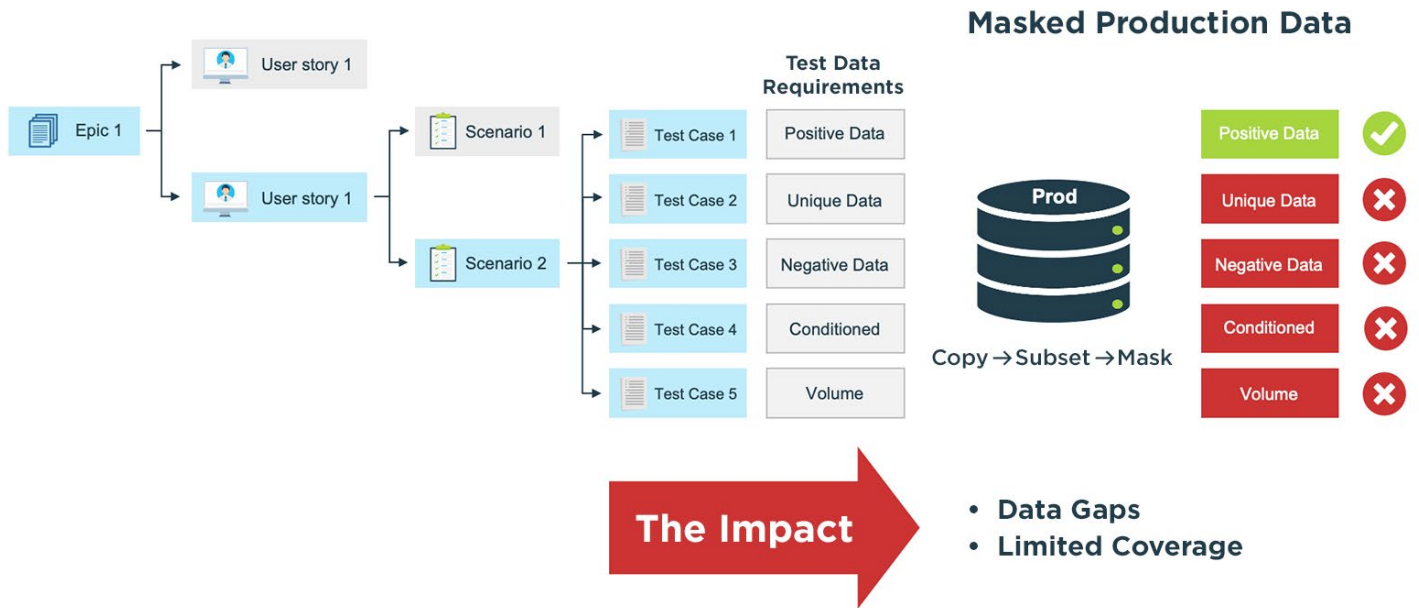
SYNTHETIC TEST DATA AUTOMATION SOLUTION OVERVIEW

Table of Contents

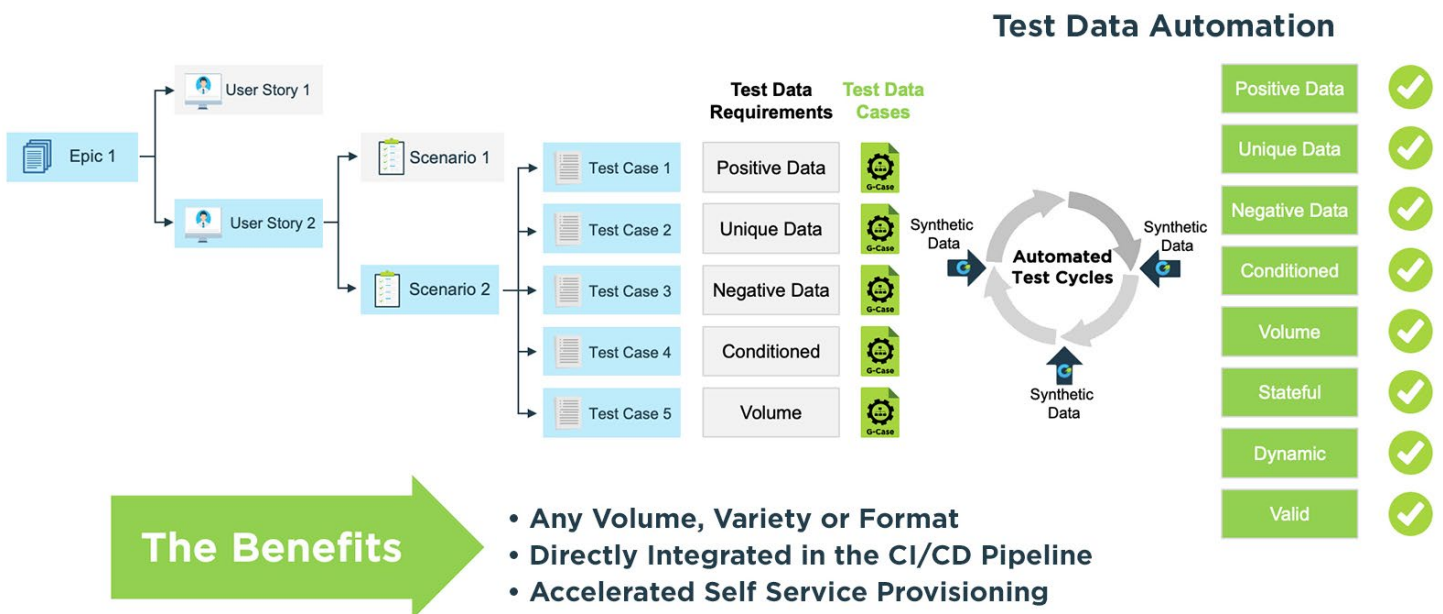
1. Synthetic Test Data Automation	3
2. Terminology	4
3. Methodology	5
4. Part 1 - Model	6
a. Modeling a Test Data Project - 14 Methods	6
b. Data Model Changes - G-Delta	7
5. Part 2 - Design	8
a. Data Generators	8
b. Test Data Case Management	9
• Test Data Cases (G-Cases)	
• Test Data Rules (G-Rules)	
• Test Data Queries (G-Queries)	
• Test Data Stories (G-Stories)	
• Test Data Epics (G-Epics)	
c. Test Data Families (G-Families)	11
d. Data Subsets with Synthetic Data Masking (G-Migration+)	12
6. Part 3 - Deploy	14
a. Receivers / Output Data Formats	14
b. Test Data Portal (G-Portal)	15
c. Auto Sync of Test Data Project Components (G-Repository)	15
d. Integration with Test Automation Frameworks	16
e. Dynamic Data with the GenRocket API	18
f. Self Service Test Data Changes (G-Questionnaire)	18
7. Part 4 - Manage	19
a. User Roles	19
b. User Authentication & Team Permissions	19
c. G-Analytics	20
d. Training - Flight School	21

1. Synthetic Test Data Automation

TDM - Production Data Limits Test Coverage





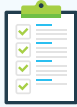


A New Paradigm - Test Data Automation (TDA)








2. Terminology

5 KEY COMPONENTS

GENROCKET TERM	THINK OF IT LIKE...	EXAMPLES
 Domain	A person, place or thing - Often equivalent to a database table	A User Domain
 Attribute	The characteristics of the Domain - the columns in the database	Name, Phone, SSN, DOB, Customer#
 737 Generator	Generates data or queries production data for the Attribute	NameGen, PhoneNumberGen, SSNGen, DateGen, QueryGen Generators
 104 Receiver	Receives the data from the Generator and morphs it into a usable format	XML, JSON, SQL, DB2, VSAM, CSV, REST API, SWIFT, etc.
 Scenario	Instruction set that tells the GenRocket engine how much data to generate	Generates data into a typical test case in about 100 milliseconds

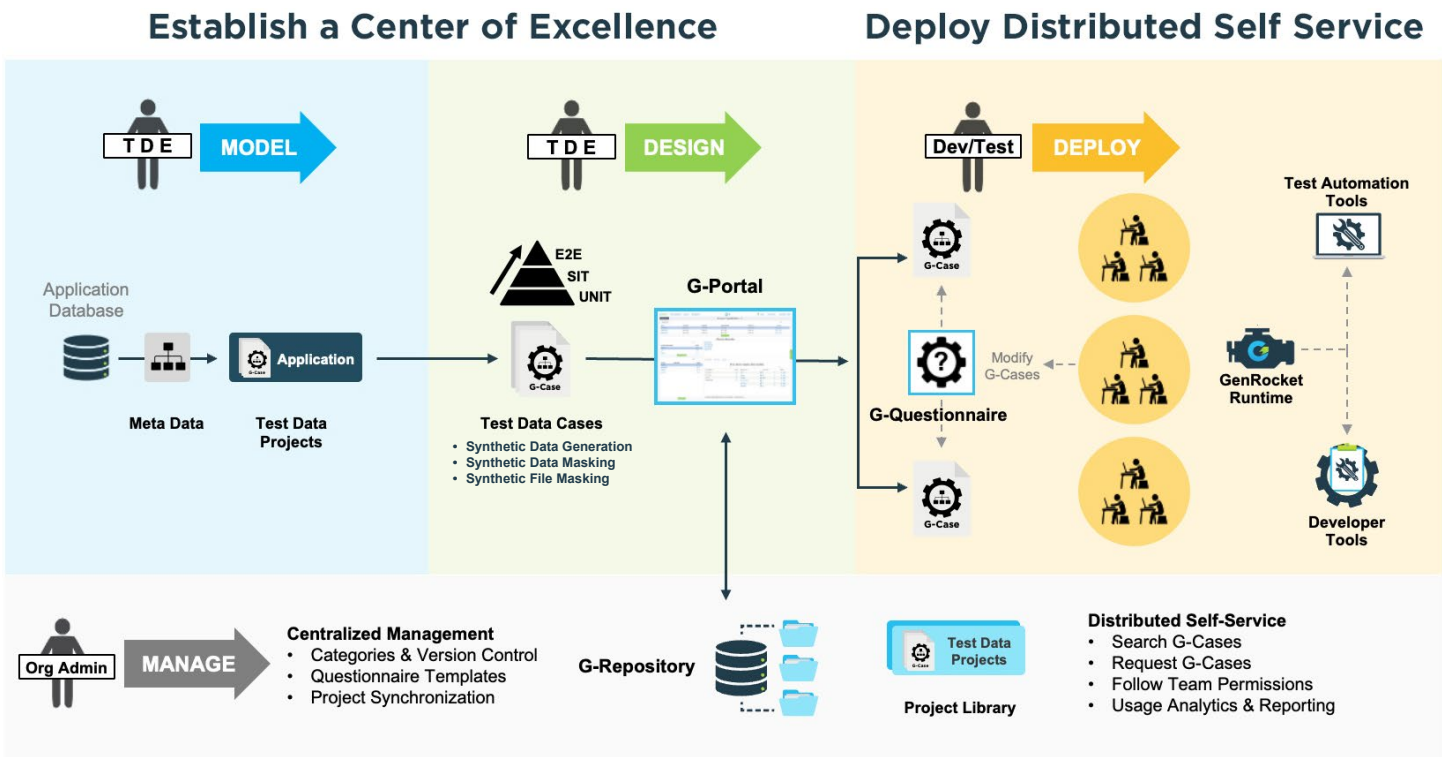
5 TEST DATA CASE MANAGEMENT COMPONENTS

GENROCKET TERM	THINK OF IT LIKE...	EXAMPLES
 Test Data Case	How you specify the exact test data needed for a test case	100 customers (unit), 5 Branches with 10 Accounts with 1,000 Customers (Integration)
 Test Data Rule	How to apply rules against your Test Data Case to only get the data you want	Platinum customer starting bank balance: \$10,000, Gold customer starting balance: \$5,000 - \$9,999
 Test Data Query	Ability to query specific production data values from a file or database	Query Platinum and Gold customer Account #'s to blend with synthetically generated data
 Test Data Stories	Stories combine a group of Test Data Cases	Combine Type Tables Test Data Cases with Account Table Test Data Cases
 Test Data Epics	Epics combine a group of Test Data Stories	Combine Integration Test #1 with Integration Test #2

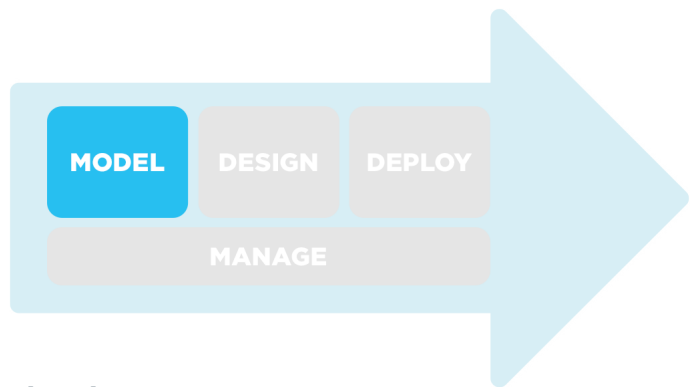
3. Methodology

- MODEL
- DESIGN
- DEPLOY
- MANAGE

A Distributed Self Service Platform



4. Part 1 - Model



4a. Modeling a Test Data Project - 14 Methods

Test Data Projects usually relate to all or part of a database / application. In general, modeling a Test Data Project entails the following steps:

1. Create a Test Data Project
2. Import or Create the Data Model
3. Establish Referential Integrity within the Test Data Project
4. For multiple Test Data Projects - establish referential integrity across Test Data Projects

There are currently 14 methods to model Domains in a Test Data Project in GenRocket:

Scratch Pad	Import from Presets
New Domain	Import from Salesforce
Quick Pattern	Import from Spark Schema
Import From CSV	Import from XSD
Import from DDL	Import from XTS
Import from GenRocket JSON Schema	Import from YAML
Import from JSON	Import from Avro JSON Schema

- XTS (Extract Table Schema) - The most commonly used method to create a test data project using the meta data (not the data). For more information about XTS setup go [here](#).
- Import from JSON Schema - ability to setup a test data project by connecting to Metadata management tools such as Ab.Initio and Alation
- Scratchpad - a useful tool to quickly build GenRocket Domains. To watch a 3 minute video go [here](#).
- X12 EDI Store + Accelerator - This is an optional Test Data Project setup method for healthcare customers using X12 EDI transaction sets such as 837 Claims, 835 Payments and 834 Benefit Enrollment. The Accelerator takes an EDI test data setup process that might take months and automates 90% of the setup process

4b. Data Model Changes – G-Delta

Data Models change over time with the addition or subtraction of Tables and Columns. G-Delta helps automate the maintenance of this “life cycle” task.

1. **G-Delta** is an automated platform, which allows scheduled monitoring of a database’s schema changes so that a given Project Version stays in sync with its database.
2. G-Delta prepares an Audit Report of any changes it finds in the database schema against the state of the Project Version. The user who created the G-Delta can review the Audit Report and execute the refactoring process, which will put the Project Version in sync with the database.
3. G-Delta and G-Repository work together to check and detect changes within the base Database Schema. By default, G-Delta checks for changes every 24 hours. This time interval can be changed to a minimum of 12 hours.

G-Delta Management Filter

G-Delta								Filter
Project Version	Scheduled Interval	Resource Path	Resource SubDir	Config File Name	Created By	Latest Delta (UTC)	G-Repository Server Name	Actions i
Version 1.0	Every 24 hours	#{resource.jdbc.directory}	config	config.properties	April Hatton	Nov 18, 2021 13:10:00	AprilTesting	
Version 1.1	Every 24 hours	#{resource.jdbc.directory}	config	config.properties	April Hatton	Nov 18, 2021 13:10:02	AprilTesting	

Create New G-Delta

G-Delta Audit Summary for Test Filter

Date Created (UTC)	Date Audited (UTC)	Audited By	Status i	Audits Found i	Audits Accepted i	Audits Modified i	Audits Rejected i	% Accuracy i	Actions i
Nov 18, 2021 13:10:00	Nov 18, 2021 13:10:00	April Hatton	Working	3	1	1	1	33 %	
Nov 18, 2021 00:09:00	Nov 18, 2021 00:09:00	April Hatton	Accepted	1	1	0	0	100 %	

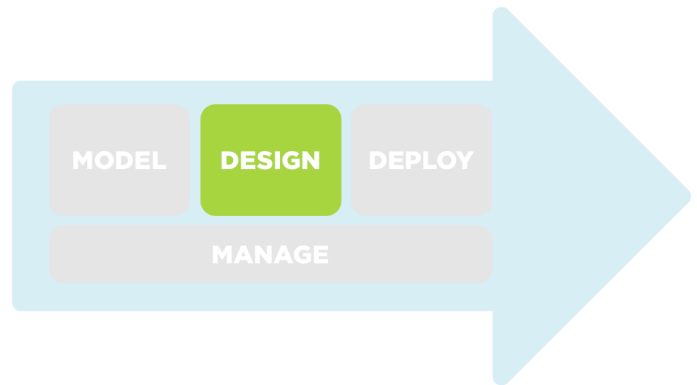
Request Audit

G-Delta Audit Summary Detail Filter

Domain	Attribute	Delta Recommendation	Action To Take	Audited By	Actions i
Address	N/A	Add Domain	Accepted	April Hatton	
User	age	Delete Attribute	Change Attribute Name	April Hatton	
User	N/A	Add Attribute	Rejected	April Hatton	

Hover on any row to see it's values to be changed

5. Part 2 - Design








5a. Data Generators

Generators are used to generate data for different Attributes such as names, addresses and credit cards. GenRocket currently offers 730+ different data Generators. Each Generator has parameters that allow user defined control over the data that is generated such as random, patterned, sequential, null, %, etc.

- Generators are intelligent and be able to maintain contextual relationships. For example, a city that matches a valid state and postal code.
- Generators can remember where they stopped generating data and start again at a later time while preserving state - this is called stateful data generation.
- [List Generators](#) are able to store or add re-usable assets into a Generator (e.g. a list of car types: sedan, convertible, hatchback)
- Permutation Generators are able to generate all valid permutations & combinations of data
- [Linked Generators](#) offers the ability to combine multiple generators to increase the versatility of the data that is generated
- Generators can be configured so that data is generated in exact length and types
- New Generators can usually be added to the platform in one Sprint cycle, at no cost.

5b. Test Data Case Management

GenRocket **Test Data Case Management** allows a small, central team of Test Data Engineers to design the data needed by developers, testers and data scientists. Test Data Case Management features make it easy for Test Data Engineers to quickly design test data that is rule-based, dynamic, and stateful.

GENROCKET TERM	THINK OF IT LIKE...	EXAMPLES
 Test Data Case	How you specify the exact test data needed for a test case	100 customers (unit), 5 Branches with 10 Accounts with 1,000 Customers (Integration)
 Test Data Rule	How to apply rules against your Test Data Case to only get the data you want	Platinum customer starting bank balance: \$10,000, Gold customer starting balance: \$5,000 - \$9,999
 Test Data Query	Ability to query specific production data values from a file or database	Query Platinum and Gold customer Account #'s to blend with synthetically generated data
 Test Data Stories	Stories combine a group of Test Data Cases	Combine Type Tables Test Data Cases with Account Table Test Data Cases
 Test Data Epics	Epics combine a group of Test Data Stories	Combine Integration Test #1 with Integration Test #2

Test Data Cases (G-Cases)

Test Data Cases are a key part of GenRocket's Self Service Test Data Module. We organize all the Test Data Case functionality into a part of the GenRocket platform we call "**G-Cases**". Test Data Cases are used any time you want to define test data for a test case.

- To watch a solutions video (5 minutes) for Test Data Cases go [here](#)
- For more information about Test Data Cases go [here](#)

Test Data Rules (G-Rules)

Test Data Rules are a key part of GenRocket's Self Service Test Data Module. We organize all the Test Data Rule functionality into a part of the GenRocket platform we call "**G-Rules**". Test Data Rules are used any time you want to apply business logic to test data that will be generated.

- To watch a solutions video (3 minutes) for Test Data Rules go [here](#)
- For more information about Test Data Rules go [here](#)

Test Data Queries (G-Queries)

With GenRocket, you can generate any type of synthetic test data needed to solve your test data challenges. However, synthetic test data does not always meet data requirements for some test cases where real data is needed such as:

- Real account #
- Real customer #

When you want to use real data values in your test case or a combination of real and synthetic data values that is where Test Data Queries comes in using one of our 18 different Query Generators.

- Query real data values from SQL databases
- Query real data values from a MongoDB database
- Query real data values from a CSV file

Queried data can be blended with synthetic data. Additionally, the real data value can be evaluated by one Generator and determine what data another Generator will generate. For example, if a bank account has \$10,000 in it, then data can be generated for a “gold” level customer.

- To watch a solutions video (3 minutes) for Test Data Queries go [here](#)
- For more information about Test Data Queries go [here](#)

Test Data Stories (G-Stories)

G-Stories can be used to create one or multiple suites of Test Data Stories with their own Chapters. Each Chapter can have a Test Data Case and a Scenario, ScenarioChain, or a ScenarioChainSet.

Test Data Stories can be used to package multiple Test Data Cases with defined Scenarios, Scenario Chains, and/or Scenario Chain Sets. Test data generation can then be *orchestrated* for a series of Test Data Story Suites with a single command.

For more information about Test Data Stories go [here](#)

Test Data Epics (G-Epics)

G-Epics can be used to organize and orchestrate the running of many Test Data Stories in a sequence. A Test Data Epic can look at a particular Test Data Story within a suite to generate test data.

Individual Test Data Stories can be combined to generate more complex test data. For example, Test Data Stories can be combined to populate an entire database based on defined test data cases using a single command.

Simple Workflow Example for Test Data Epics

Here is a sample tree view showing a simple workflow example for Test Data Epics:

- Test Data Epic
 - Test Data Story 1
 - Test Data Story 2
 - Test Data Story 3

For more information about Test Data Epics go [here](#)

5c. Test Data Families (G-Families)

G-Families discovers related Domains (“Families”) within an imported Database Schema based on relationships that have been set **among imported Domains**. Once families have been discovered, permitted users can view visual diagrams for families and create G-Cases from them.

G-Families provides a visual grouping of related Domains to help Test Data Engineers understand the data model and accelerate building of test data cases. With a few clicks, users can automatically create a Suite of G-Cases from a discovered G-Family.

The screenshot displays the GenRocket interface for 'G-Families for Project BankDemo - 1.0'. On the left, there is a 'Families' table with 'CardProduct' and 'Transaction' listed. Below it, a 'Find New Families' button and a table of 'Family Member' with 'Visibility' controls are shown. The right side features a visual diagram of the data model, showing entities like Branch, AccountLevel, AccountType, Account, Customer, CustomerAccount, CardType, and CardProduct connected by lines representing relationships.

With a few clicks, Test Data Engineers can **quickly create** a Suite of Test Data Cases (G-Cases) for a Test Data Family.

The screenshot displays the GenRocket interface for 'G-Cases for Project GFamiliesTest - 1.0'. It shows a table of G-Cases with columns for Name, Created By, Created On, Last Modified By, Modified On, Family, and Actions. Below the table, there is a 'G-Case Suite: Family1' section and a 'G-Case Editor' section. The 'G-Case Editor' shows a list of domain names and their loop counts for a specific G-Case.

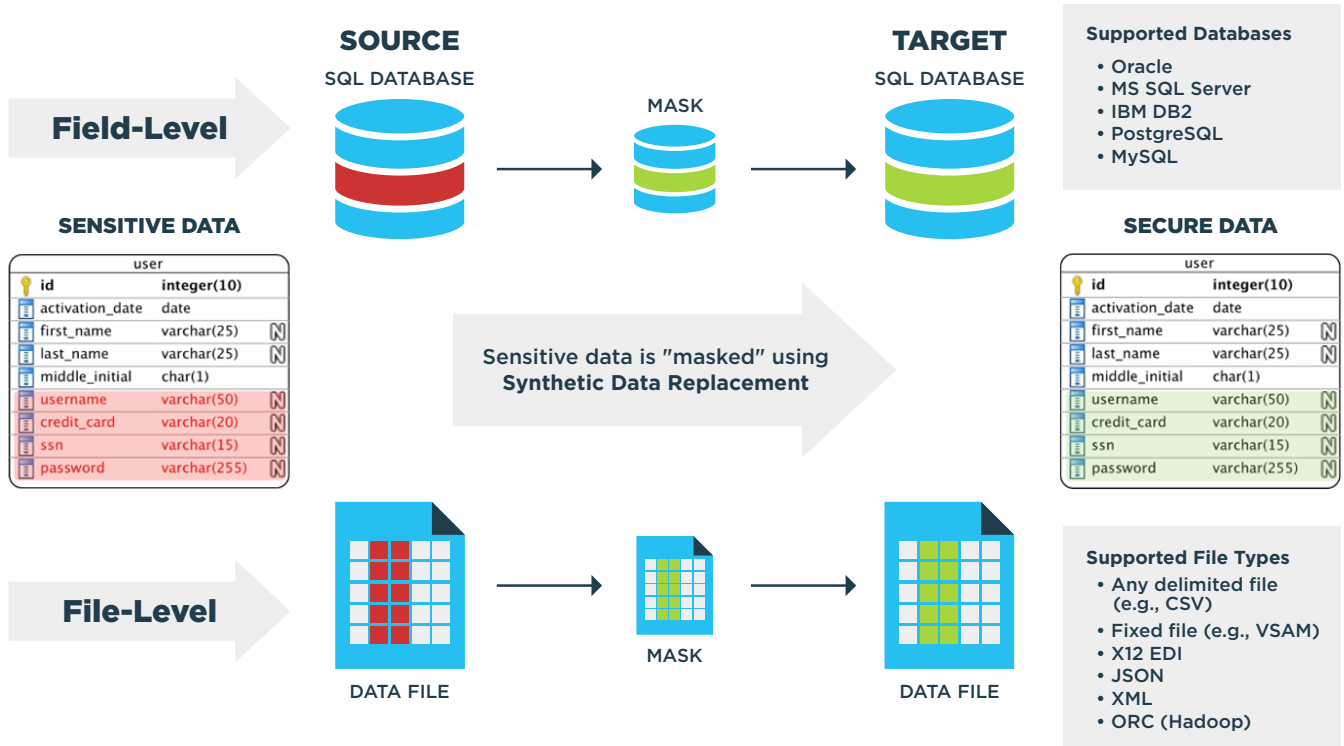
Domain Name	Actions
AccountLevel	∞ +
AccountType	∞ +
Branch	∞ +
CardProduct	∞ +
CardType	∞ +
Customer	∞ +
CustomerAccount	∞ +

Domain Name	Loop Count	Actions
Account	5	∞ +

5d. Data Subsets with Synthetic Data Masking (G-Migration+)

The G-Migration+ module intelligently queries a subset of data from a source database, replaces sensitive data values (“Synthetic Data Masking”) with synthetically generated data values, and inserts the data subset into an identical destination database using the same format and data model.

Data Subset with Synthetic Data Masking



Import the Database Data Model

Table Schema

Table Schema

Filter

Table
account
account_level
account_type
branch
card_product
card_type
customer
customer_account
transaction
transaction_type

Columns of customer

Name	Type	Size	Foreign Key Table	Foreign Key Column	Nullable
id	BIGINT	19			Not Null
version	BIGINT	19			Not Null
email_address	VARCHAR	50			Not Null
first_name	VARCHAR	25			Not Null
last_name	VARCHAR	25			Not Null

Relationship

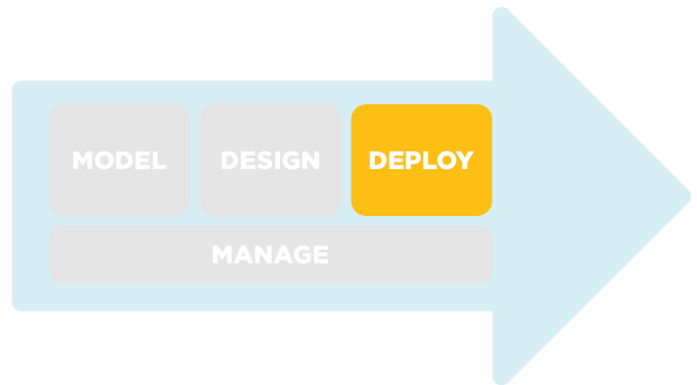
➔ Close

Select the Subset. Configure the Migration

G-Migration+ Configuration Dialog

*Name	<input type="text"/>	Description	<input type="text"/>
Primary Scenario Configuration		Migration Speed Optimization Configuration	
*Scenario Directory	<input type="text" value="resource.output.directory"/>	*Thread Count	<input type="text" value="1"/>
Default Scenario	<input type="text" value="Select Default Scenario"/>	*Batch Size	<input type="text" value="10000"/>
Data-Dump File Location Configuration		Receiver & Test Data Case Configuration	
*Data-Dump Directory	<input type="text" value="resource.output.directory"/>	Test Data Case	<input type="text" value="Select Test Data Case"/>
*Data-Dump SubDir	<input type="text"/>	*Partition Receiver	<input type="text" value="Select Receiver Type"/>
Source Database & JDBC Connection Configuration		Destination Database & JDBC Connection Configuration	
*DB / Schema Name	<input type="text"/>	*DB / Schema Name	<input type="text"/>
*JDBC Directory	<input type="text" value="resource.jdbc.directory"/>	*JDBC Directory	<input type="text" value="resource.jdbc.directory"/>
JDBC SubDirectory	<input type="text"/>	JDBC SubDirectory	<input type="text"/>
*JDBC Config Name	<input type="text"/>	*JDBC Config Name	<input type="text"/>

6. Part 3 - Deploy



Once test data DESIGN is finished, the Test Data Engineer has test data ready to be DEPLOYED. Test Data Scenarios and Test Data Cases defined during the DESIGN phase contain the instructions for volume and data variety that will be generated by the GenRocket Runtime Engine.

Receivers determine the test data output format for generated test data during deployment. Test data generated by GenRocket can be morphed into just about any format that is required. This data can be deployed in more than 100 different formats including:

- Test Automation Frameworks & Tools (e.g. GitLab, Azure DevOps, etc.)
- Databases (MS SQL, MySQL, MongoDB, Oracle, IBM DB2, etc.)
- Formats (XML, JSON, CSV, VSAM, PDF, JPEG)
- The Cloud (AWS, Azure, GCP, etc.)

6a. Receivers / Output Data Formats

GenRocket Receivers morph the raw test data generated by Generators into a desired output data format.

- 101 Receivers are currently available in GenRocket.
- To see all of the current supported output formats go [here](#)
- Non-proprietary Receivers can be added to the platform for our customers upon request, at no cost.

6b. Test Data Portal (G-Portal)

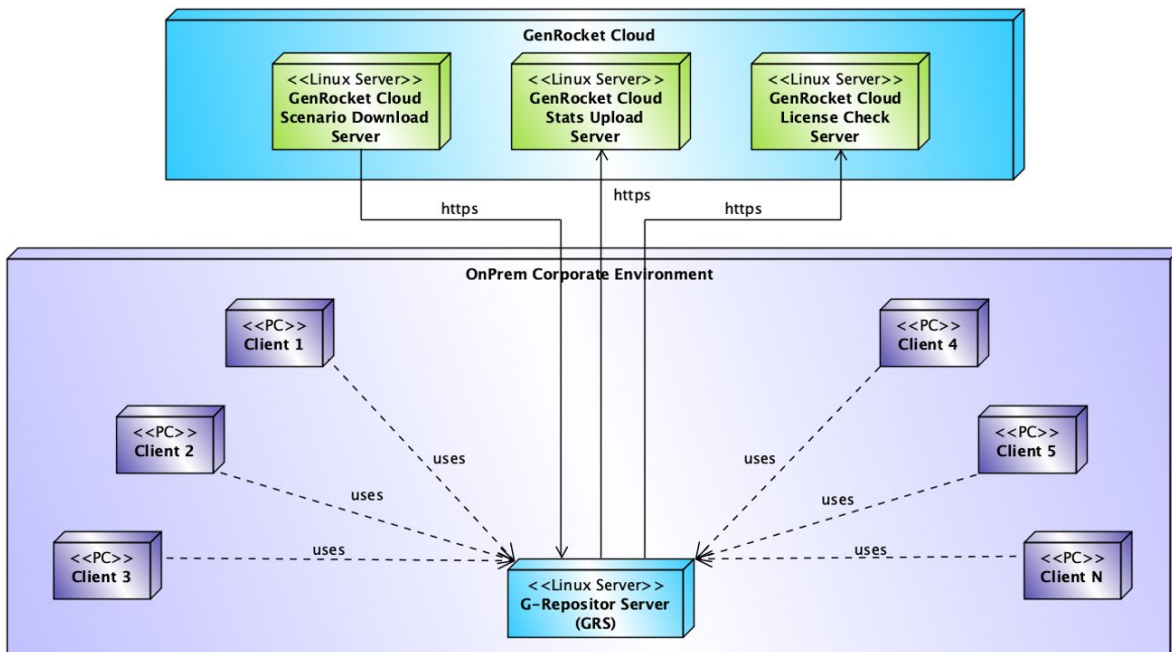
G-Portal is a central test data portal that allows developers and testers to find and request the test data they need for their test cases and allows Test Data Engineers to receive and respond to test data requests from large, distributed groups of test data users. The view below is for a Test Data Engineer.

The screenshot displays the G-Portal interface for Organization Administrators and Test Data Engineers. The interface is divided into several sections:

- Requesters:** A table with columns Name, New, and Active. It shows one requester: TDR Prabhath Kashyap with 1 New and 0 Active requests.
- TDEnginers:** A table with columns Name, Assigned, and Active. It shows one TDE: TDE Prabhath Kashyap with 1 Assigned and 0 Active requests.
- New Requests:** A table with columns ID, Time Created, Last Updated, Requester, Title, Assigned To, and Status. It shows one new request: ID 1003, Time Created Nov 15, 2022 12:59:10, Last Updated Nov 30, 2022 08:31:54, Requester TDR Prabhath Kashyap, Title Prabhath, Assigned To TDE Prabhath Kashyap, Status Assigned.
- Active Requests:** A table with columns ID, Time Created, Last Updated, Requester, Title, Assigned To, and Status. It shows no data available in the table.
- Completed Requests:** A table with columns ID, Time Created, Last Updated, Requester, Title, Assigned To, and Status. It shows one completed request: ID 1002, Time Created Nov 15, 2022 10:13:45, Last Updated Nov 15, 2022 10:57:12, Requester TDR Prabhath Kashyap, Title I need a requirement of BankDemo, Assigned To TDE Prabhath Kashyap, Status Completed.

6c. Auto Sync of Test Data Project Components (G-Repository)

G-Repository automatically manages the downloading, updating and deleting of GenRocket Scenarios, G-Stories, G-Epics, and other Configuration files (G-Cases) between the GenRocket Cloud and the customer's secure On Premise and Private Cloud environments.

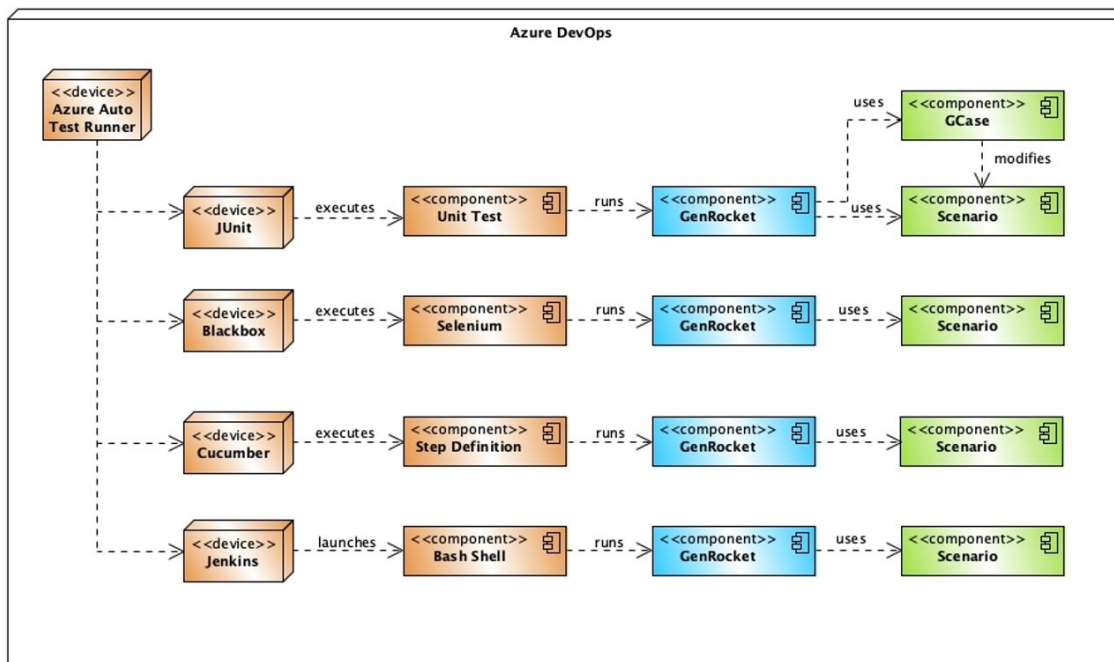


6d. Integration with Test Automation Frameworks

GenRocket was designed to directly integrate with just about any Test Automation Framework. A GenRocket Scenario + Test Data Case can be launched by:

- At the command line
- Batch file
- Shell scripts
- Jenkins (video [here](#))
- Scripting language
- Compiled language
- REST API

Azure DevOps examples:



Test Case Integration Example with Test Script:

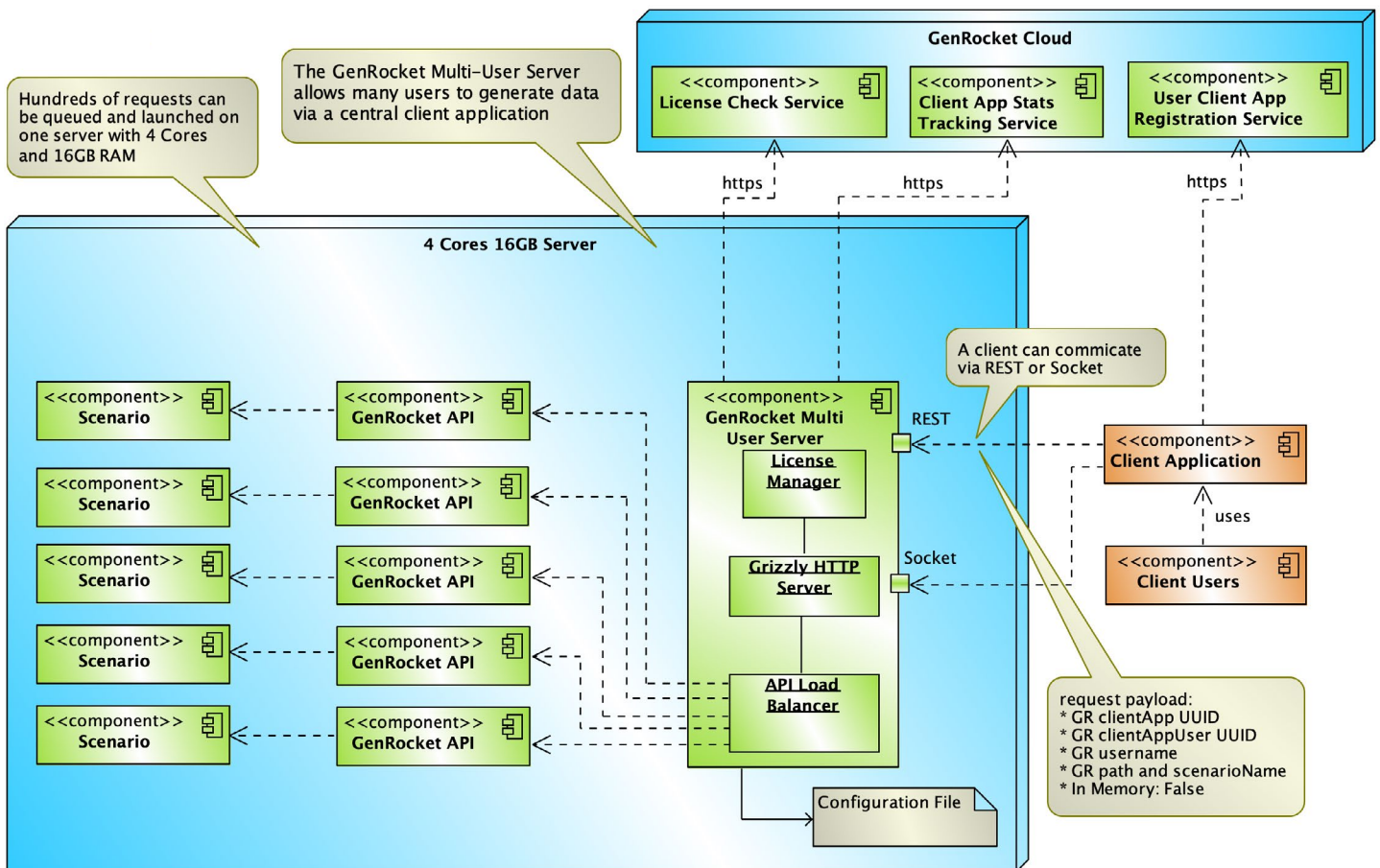
```
1 package com.genRocket
2
3 import com.genRocket.engine.EngineAPI
4 import com.genRocket.engine.EngineManual
5 import grails.test.spock.IntegrationSpec
6
7 class UserDomainServiceIntegrationSpec extends IntegrationSpec {
8
9     void "Test insert 10000 users"() {
10         final EngineAPI api = new EngineManual()
11
12         User.withTransaction {
13             given:
14
15                 api.clientRepoSet('BankDemo')
16                 api.clientRepoEpicRun('RootTableEpic')
17                 api.clientRepoRun()
18
19                 final Integer countBeforeInsert = User.count()
20
21             when:
22
23                 api.clientRepoRun('-grepo BankDemo -grs UserScenario -tdc UserCaseSuite.Load.load10000')
24
25             then:
26
27                 assert User.count() == countBeforeInsert + 10000
28         }
29     }
30 }
31
```

Integration Videos (Examples of Test Case Script Integration):

- Selenium: go [here](#)
- Tricentis Tosca: go [here](#)
- Ui Path: go [here](#)
- Cucumber Feature File: go [here](#)

GenRocket Multi User Server:

The GMUS is a versatile component that can communicate with customer's test automation frameworks via REST API or a Socket. The GMUS and G-Repository Server can be deployed as a package in a Container.

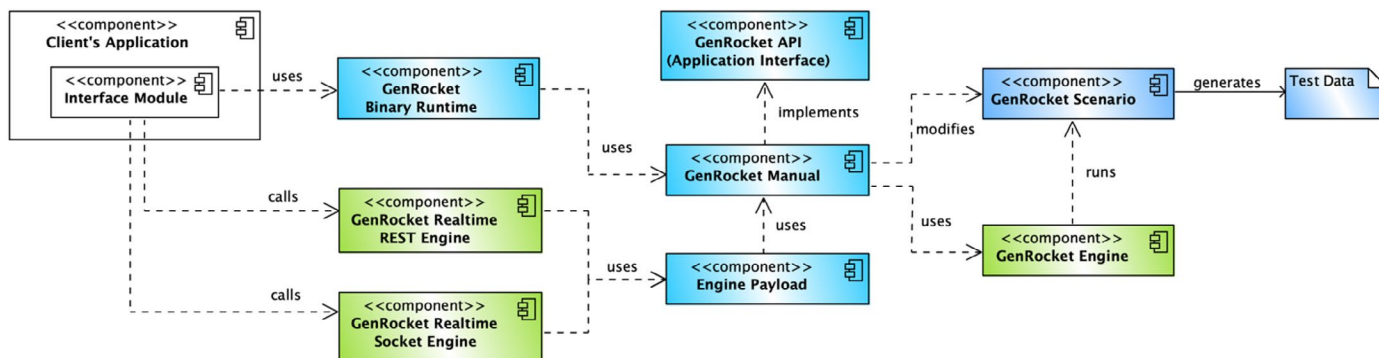


To learn more about GMUS integration go [here](#)

6e. Dynamic Data with the GenRocket API

The GenRocket runtime application interface (API) allows programmers to access the GenRocket runtime to directly modify and create scenarios from their own programs. Programmers will have the following control over a Scenario:

- Modify Domains, Attributes, Generators and Receivers
- Add Domains, Attributes, Generators and Receivers to an existing Scenario
- More information about the Runtime API [here](#)



6f. Self Service Test Data Changes (G-Questionnaire)

G-Questionnaire is designed for a developer or tester to modify test data cases (G-Cases) via “self-service” without having to be a GenRocket expert (Test Data Engineer).

1. Create volume and variety of test data quickly, with minimal effort.
2. Modify test data cases without changing the original test data case.
3. A simplified approach for users who are not as familiar with GenRocket.

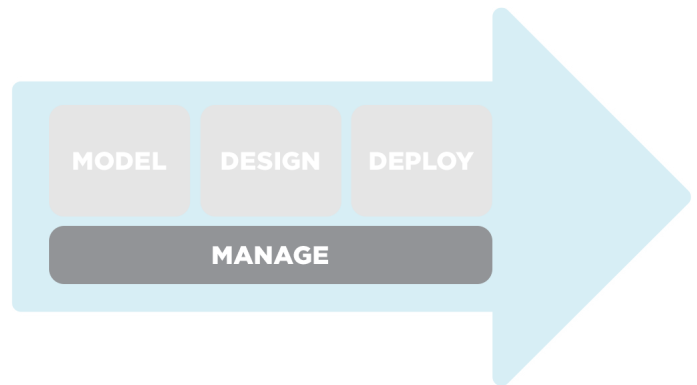
The screenshot shows the G-Questionnaire web application interface. The main area displays a table of G-Questionnaire Suites:

Name	Description	Category	Created By	Created On	Last Modified	Modified On	Actions
SampleGQuestionnaireSuite	A Sample G-Questionnaire Suite	Load	April Hatton	Oct 06, 2022	April Hatton	Oct 06, 2022	[Icons]

Below the table, there are several panels for configuring the questionnaire:

- Case**: A list of cases (Template, tester1case, tester2case) with a filter and action icons.
- Rules**: A section for defining rules, including a "G-Rule Set" and "UserRules".
- Domain and Attribute**: A table for defining domains and their attributes. The "Domain" column lists Organization, Department, User, and Address. The "Loop Count" column shows 1 for each. The "Attribute" column lists name for Organization and Department.
- Generators**: A section for defining generators, including a "Name" (gen1), "Generator" (F500Name2014Gen), "autoReference" (true), "caseType" (noCase), "sortOrder" (random), "seed", and "waitAmount" (1).
- Linked Generators**: A section for linking generators, including a "Name" (gen1), "Alias" (F500Name2014...), and "Generator" (F500Name2014...).

7. Part 4 - Manage



GenRocket makes it easy to securely manage Projects and User Permissions across your Organization. Each user can be assigned a User Role, which determines what features they are able to access within GenRocket.

7a. User Roles

Here are the different GenRocket user roles:

- [Org Admin](#) - Add / Remove users, deploy G-Repository, Analytics and Reporting
- **Test Data Engineer** - The “Power User” in the GenRocket ecosystem. Uses G-Families to build G-Cases, Apply G-Rules and share these instruction sets via G-Portal to Testers & Developers. Receive test data requests via G-Portal from testers and developers
- **Tester** - User of Scenarios and G-Cases to generate data for their testing requirements. Use G-Questionnaire to quickly modify G-Cases to meet their test case requirements. Searches for test data G-Cases in G-Portal and makes request for new test data requirements via G-Portal.
- **Developer** - User of Scenarios a G-Cases to generate data for their testing requirements. Use G-Questionnaire to quickly modify G-Cases to meet their test case requirements. Searches for test data G-Cases in G-Portal and makes request for new test data requirements via G-Portal.

7b. User Authentication & Team Permissions

All GenRocket users are authenticated with username and passwords. GenRocket has an approach where the usernames and company email addresses can be obfuscated, if required.

- SSO (Single Sign On) is available as an additional layer of security
- MFA (Multi Factor Authentication) is available as an additional layer of security

GenRocket [Team Permissions](#) provides an additional layer of security for managing Projects across your Organization. Access and permissions can be managed with a Team, which has its own Projects and Team Members. This feature prevents unwanted changes such as deleted Projects, Domains, and Scenarios in GenRocket.

7c. G-Analytics

Reporting Dashboard for project and user activity. To learn more, go here.

G-Analytics Dashboard | OrganizationSetupActivity | GenRocket | Currently Public

1 Organization Overview | 12 Months

Name	Info
Org Name	GenRocket
License Type	Enterprise
License End Date	Dec 31, 2025
Total Teams	0

2 Organization Created Activity | Duration: 12 Months

Name	Number Created
Project	407
Project Version	453
Domain	4653
Scenario	4426

3 Organization Created Projects | Total: 407 | Duration: 12 Months

Name	Date Created	Last Modified
NextJS/JSON	2020-12-29	2022-02-07 05:43:10
BankingDemo	2021-01-04	2021-09-14 14:29:42
GreyboxDemo	2021-01-15	2022-01-07 11:58:44
GMigrator-Demo	2021-01-25	2021-12-03 12:56:41
TestDataCasesDemo	2021-03-06	2022-05-28 11:31:39
MT940Swift	2021-03-06	2022-07-11 16:03:41

4 Organization Updated Projects | Total: 407 | Duration: 12 Months

Name	Last Modified
NextJS/JSON	2022-02-07 05:14:01
AB_PoC	2022-03-24 09:10:30
BankingDemo	2021-12-08 22:30:41
EDI-437-P-0010-X322	2022-02-23 16:10:14

G-Analytics Dashboard | ProjectSetupActivity | GenRocket | Currently Public

1 Organization Created Projects

Name	Date Created	Last Modified
NextJS/JSON	2020-12-29	2022-02-07 05:43:10
BankingDemo	2021-01-04	2021-09-14 14:29:42
GreyboxDemo	2021-01-15	2022-01-07 11:58:44
GMigrator-Demo	2021-01-25	2021-12-03 12:56:41
TestDataCasesDemo	2021-03-06	2022-05-28 11:31:39
MT940Swift	2021-03-06	2022-07-11 16:03:41

2 Created Project Versions | Total: 1 | Duration: 12 Months

Name	Date Created	Last Modified
1.0	2021-01-25	2021-12-03 12:56:41

3 Project Version Created Domains | Total: 1 | Duration: 12 Months

Name	Date Created	Last Modified
Address	2021-12-03	2021-12-03 12:56:40

4 Project Version Created Scenarios | Total: 1 | Duration: 12 Months

Name	Date Created	Last Modified
AddressScenario	2021-12-03	2021-12-03 12:56:41

G-Analytics Dashboard | UserSetupActivity | GenRocket | Currently Public

1 Organization Overview | 3 Months

Name	Info
Org Name	GenRocket
License Type	Enterprise
License End Date	Dec 31, 2025
Total Teams	0

2 Organization Users | Total: 11 | Duration: 3 Months

Name	Last Active
Prashant Kashyap	2022-06-29 12:15:22
Utkay Seta	2022-07-13 13:11:49
Prasen Jeet Gupta	2022-06-18 12:08:01
Jatin Dominic	2022-06-23 13:05:38

3 User Created Projects | Total: 37 | Duration: 3 Months

Name	Date Created	Last Modified
TestDataCasesDemo	2021-03-06	2022-05-28 11:31:39
MT940Swift	2021-03-06	2022-05-17 16:03:41
ExternalAPIDemo	2021-03-11	2022-05-17 07:31:11
PartWorkEngineTestData	2021-03-20	2022-06-16 16:09:50

4 User Created Project Versions | Total: 2 | Duration: 3 Months

Name	Date Created	Last Modified
1.0	2022-05-28	2022-07-06 17:25:52
1.1	2022-05-09	2022-06-16 12:31:30

G-Analytics Dashboard | DataRunActivity | GenRocket | Currently Public

1 Organization Overview | 3 Months

Name	Info
Org Name	GenRocket
License Type	Enterprise
License End Date	Dec 31, 2025
Total Teams	0

2 Organization Runtime Activity | Duration: 3 Months

Name	Activity
Scenario	157
Scenario Chain	8
Scenario Chain Set	0
Test Data Case	95

3 Organization Runtime Projects | Total: 7 | Duration: 3 Months

Name	Last Run
TestDataCasesDemo	2022-06-16 12:13:44
StoresDemo	2022-06-11 17:22:31
BankDemo	2022-05-28 18:13:06
CheckImageDemo	2022-07-11 22:21:16

4 Project Runtime Activity | Duration: 3 Months

Name	Activity
Scenario	8
Scenario Chain	2
Scenario Chain Set	0
Test Data Case	0

5 Project Runtime Project Versions | Total: 1 | Duration: 3 Months

Name	Last Run
1.0	2022-06-16 12:13:44

6 Project Version Runtime Activity | Duration: 3 Months

Name	Activity
Scenario	8
Scenario Chain	2
Scenario Chain Set	0
Test Data Case	0

7d. Training - Flight School



Available Flight School Classes

Class Title	Description	Class Title	Certification
GCE 1/Test Data Engineer Curriculum	Detailed understanding of GenRocket with Examples	<ul style="list-style-type: none"> Test Data Engineers (TDEs) Value Stream Managers Quality Engineers GenRocket Power Users 	GenRocket Certified Engineer - Level 1 (after passing the optional exam)
GenRocket Organization Admin	Learn how to administer GenRocket platform	<ul style="list-style-type: none"> Platform Administrators Managers 	N/A
GenRocket POC Training (Non-Healthcare EDI)	High level overview of the GenRocket functionality (except EDI)	<ul style="list-style-type: none"> Organizations undertaking a POC 	N/A
GenRocket POC Training (Healthcare EDI)	High level overview of the GenRocket healthcare EDI Functionality	<ul style="list-style-type: none"> Organizations undertaking a POC to evaluate EDI functionality 	N/A
GenRocket Sales Enablement	Understand the value of using Synthetic test data and synthetic test data marketplace	<ul style="list-style-type: none"> GenRocket channel partners Executives and managers 	N/A
GCE Level 1 (Healthcare EDI)	Detailed training with examples on how to use GenRocket for Healthcare EDI	<ul style="list-style-type: none"> Test Data Engineers (TDEs) Value Stream Managers Quality Engineers GenRocket Power Users 	N/A