NESD-QA Gherkin Scripting Guidelines

1. Purpose

This document defines the **approved Gherkin steps, parameters, and behaviors** for NESD-QA automation. It is intended for Test Analysts and Test Automation Engineers to ensure **consistent, reusable, and standardized** automation scripts.

Only the steps, syntax, and behaviours documented here are valid. Any automation or validation must strictly follow this document.

2. Glossary

Term	Definition
UNITS	Unit code for SMS bundles on Fusion
MINUTES	Unit code for Voice bundles on Fusion
MB, GB	Unit codes for Data bundles on Fusion

3. Subscriber Profile Management

3.1 Create New Subscriber

Gherkin Step

```
1  Given New subscriber <Subscriber_type> profile is
<Offerring_Code>
```

Parameters

- Subscriber_type: Any random test MSISDN (example: 27639568950)
- Offering_Code: Offering code (example: NOF5, WF5)

Behaviour

- Deletes any existing MSISDN profile
- Recreates the subscriber profile on CCS using the provided offering code

3.2 Create Multiple Subscribers

Gherkin Step

```
1  Given Multiple New subscribers <Hybrid_Subscriber> profile is
<Hybrid_Offerring_Code> and <Prepaid_Subscriber> profile is
<Prepaid_Offerring_Code>
```

Behaviour

- Creates two subscriber profiles:
    - Hybrid subscriber with Hybrid_Offerring_Code
    - Prepaid subscriber with Prepaid_Offerring_Code

3.3 Use Existing Subscriber

Gherkin Step

```
1  Given Existing subscriber is <Subscriber_type> with offering
code <Offerring_Code>
```

Behaviour

- Reuses an existing MSISDN
- Does not delete or recreate the profile

3.4 Create CUR Profile

Gherkin Step

```
1  And CUR profile is set with attributes
2  """
3  {
4    "billingplatformid": "300",
5    "paymenttype": "P"
6  }
7  """
```

Rules

- Attributes must be comma-separated
- Always use triple quotes

Payment Types

- P = Prepaid
- H = Hybrid
- C = Postpaid

4. Fusion Transactions

4.1 Set FI Details

Gherkin Step

```
1  And Set FI specific details "FI Code" is <FICode> and
schemeAgencyName is <Company>
```

4.2 Eligibility Checks

Swagger Eligibility

```
1  And send Swagger eligibility for "<Recharge_Amount>"
2  And send Swagger eligibility for <SOID>
```

Non-Swagger Eligibility

```
1  And send nonSwagger eligibility for "<Recharge_Amount>"
2  And send nonSwagger eligibility for <SOID>
```

4.3 Airtime Recharge

Swagger

```
1  When Sweggar recharge via FI Fusion recharge value of
<Recharge_Amount> ZAR
```

Non-Swagger

```
1  When nonSweggar recharge via FI Fusion recharge value of
<Recharge_Amount> ZAR
```

4.4 Bundle Purchase via Fusion

SOID-Based Purchase

```
1  When I purchase via fusion soid is <SOID> price is
<Purchase_Amount> bundle size is <Bundle_Size> unit code is
<unit_code> validity period is <validity_period>
```

Product ID-Based Purchase

```
1  When I purchase via fusion product id is <Product_ID> price is
<Purchase_Amount> bundle size is <Voice_Bundle> unit code is
<unit_code>
```

Delete Bundle

```
1  When I Delete Bundle via fusion soid is <SOID> price is
<Purchase_Amount> bundle size is <Data_Bundle> unit code is
<unit_code> validity period is <validity_period>
```

4.5 Add-To-Bill Purchase

Gherkin Steps

```
1  And Set Additional Fusion Properties <additionalstr>
2  When Subscriber using "Add-To-Bill" purchases bundle <SOID>
price is <Purchase_Amount> bundle size is <Data_Bundle> validity
period is <validity_period>
```

Additional Properties Format

```
1  FI_cost:12/FormatId:B005
```

5. Airtime Balance Adjustments

Gherkin Step

```
1  And Subscriber <MonetaryType_Code> balance is <Recharge_amount>
```

Examples

- MonetaryType_Code = C_VZA_PPS_MainAccount
- Recharge_amount = 4.00

6. Verification Steps

6.1 CCS Bundle Provisioning

Non-Prorated

```
1  Then Data bundle for OfferingID <OfferingID> allocated is
<Bundle_Size>
```

Prorated

```
1  Then Data bundle for OfferingID <OfferingID> allocated is
<Bundle_Size> free unit is <FreeUnit> <Prorated>
```

Other Bundle Types

- SMS bundle verification
- Voice bundle verification
- Total data bundle verification

6.2 Airtime Balance Verification

Deduction

```
1   Then amount <Purchase_Amount>ZAR is deducted from
<Recharge_amount> <MonetaryType_Code> money counter
```

Credit

```
1   Then amount <Purchase_Amount>ZAR is credited into
<MonetaryType_Code> money counter
```

Instant Recharge Limit

```
1   And Verify instant Recharge limit is
<InstantRechargeLimit>cents
```

6.3 Expiry Period Verification

Days-Based

```
1   And Expiry period is <Expiry_Date> with Offerring ID
<OfferingID>
```

Calendar Month-Based

```
1   And Expiry period is <Expiry_Date> with Offerring ID
<OfferingID> and free unit <FreeUnit> calendar month <Prorated>
```

6.4 Fusion Verification

Airtime Balance

```
1   Then verify "C_VZA_PPS_MainAccount" service balances remaining
airtime
```

Bundle Verification

```
1   Then fusion service bundle <FreeUnit> <OfferingID> for product
<Product_Name> is <Bundle_Size> <Prorated>
```

Validity Verification

```
1  Then fusion service validity <FreeUnit> <OfferingID>
   <Validity_Type> for product <Product_Name> is <Expiry_Date>
```

Description Verification

```
1  Then verify fusion <FreeUnit> <OfferingID> <Product_Name>
   <Product_desc>
```

7. Voice, Data, and Airtime Advance

Purchase

```
1  When I purchase via fusion product id is <Product_ID> price is
   <Purchase_Amount> and service fee <Servicefee>
```

Loan Verification

```
1  Then amount <Purchase_Amount>ZAR is added into <LoanAmount>ZAR
   loan counter
```

Deduction Verification

```
1  Then outstanding amount <Purchase_Amount>ZAR and service
   <Servicefee>ZAR is deducted from recharge amount
   <Recharge_Amount>ZAR
```

8. Buy for Another Subscriber

Setup

```
1  Given Multiple New subscribers <Subscriber_type1> profile is
   <Offerring_Code1> and <Subscriber_type2> profile is
   <Offerring_Code2>
```

Purchase

```
1  When I buy for another via fusion soid is <SOID> price is
   <Purchase_Amount> bundle size is <Data_Bundle> unit code is
   <unit_code> validity period is <validity_period>
```

Verification

```
1  And Data bundle for OfferingID <OfferingID> allocated is
   <Data_Bundle> is transfered to <Subscriber_type2> with free unit
   code <FreeUnit> <Prorated>
```

9. SMS Verification

SMS Template

```
1  And SMS template for SOID <SOID> is "Your {0} bundle has been
provisioned and will be valid until {1,date,yyyy-MM-dd 'at'
HH:mm:ss}."
```

Message Verification

```
1  And SMS message for SOID <SOID> bundle size <Data_Bundle> and
expiry date <Expiry_Date> and calendar month <Prorated> is sent
```

10. Free Change

Gherkin Steps

```
1  And Subscriber is eligible to change to <toOffer>
2  When subscriber performs free change from product offer
<Offerring_Code> to product offer <toOffer>
3  Then subscriber product offer is <toOffer>
```

11. Vodabucks

Add Vodabucks

```
1  When add {double} vodabucks for the subscriber
```

Deduct Vodabucks

```
1  Then verify that {double} vodabucks is deducted for the
subscriber
```

Verify Allocation

```
1  Then verify that {double} vodabucks is allocated for the
subscriber
```

Balance Check

```
1  And get vodabucks balance for the subscriber
```

Bank Vodabucks

```
1  And bank vodabucks for the subscriber
```

12. VOMS Voucher Redemption

Redeem via IVR

```
1   When Subscriber redeems valid voucher for voucher type
<Voucher_type> via IVR
```

Redeem via USSD

```
1   When Subscriber redeems valid voucher for voucher type
<Voucher_type> via USSD
```

Redeem via SMS

```
1   When Subscriber redeems valid voucher for voucher type
<Voucher_type> via SMS
```

Behaviour

- Performs VOMS voucher redeem based on Voucher_type
- Automatically generates a voucher PIN if none exists or is active

13. Voucher Reference Table

Product Description	Denomination	Voucher Type	Code
R 110 PIN vodago	110	vodago	31
R 275 PIN vodago	275	vodago	36
R 29 PIN vodago	29	vodago	16
R60 Yebo5 PIN	12	PIN	42
R49 SMS PIN	49	PIN	44
...	...	...	...

(Only voucher entries listed in the original source document are valid.)