

S2S Spaza PSL POC

Functional & Technical Specification (v2.0)

Document Purpose

This document defines the **end-to-end WhatsApp-first flow**, business rules, and technical integration requirements for the **S2S Spaza PSL Proof of Concept (POC)**.

This version (v2.0) is rewritten to **accurately reflect the real-world operational flow**: - Voucher-first user journeys - Multi-entry support per user - WhatsApp as the primary UX layer - Backend as the single source of truth

The document is intended for **BSP developers, backend engineers, and product stakeholders**.

1. System Overview

The S2S Spaza PSL POC enables customers to: - Engage with participating Spaza shops - Receive spend-based vouchers (e.g. R100+) - Interact via WhatsApp to register and play - Submit one or more weekly prediction entries - Track participation via a leaderboard

WhatsApp is the **only required user interface**. No native app is required.

2. Actors & Responsibilities

2.1 WhatsApp BSP (S2S)

- Receives inbound user messages
- Delivers outbound messages and links
- Does **not** store business state
- Relays all logic decisions to backend

2.2 Backend (Kingsize / Game Backend)

- Owns all user, voucher, and entry state
- Validates SIDs and vouchers
- Generates registration and prediction links
- Generates and manages Leaderboard IDs
- Enforces weekly rules and immutability

2.3 Spaza Shop

- Issues spend vouchers to customers
- Has a unique Spaza Service ID (SID)

2.4 Customer

- Uses WhatsApp only
 - Registers once
 - Can submit multiple entries per week using vouchers
-

3. Core Concepts

3.1 Spaza Service ID (SID)

- Unique identifier per Spaza shop
- Used during initial registration

3.2 Voucher Code

- Issued after qualifying spend (e.g. R100+)
- Single-use
- Linked to issuing SID
- Valid only for the current PSL week

3.3 Leaderboard ID

- Public-facing user identifier
 - Generated by backend after registration
 - Alpha string with numeric suffix if needed
 - Filtered for profanity and uniqueness
-

4. User States

State	Description
UNKNOWN	WhatsApp number not yet known to system
PENDING_REGISTRATION	SID validated, registration not completed
ACTIVE	Fully registered and eligible to play

Note: ACTIVE users may submit **multiple entries per week** using **different vouchers**.

5. Supported User Inputs

Users are **not required** to know commands in advance.

Explicit Commands

- `new <SID>` – initiate registration manually
- `code <VOUCHER>` – redeem a voucher

Implicit Handling

- Any inbound message containing a voucher-like token is treated as a **voucher attempt**
 - Any other message triggers the **Help / Entry flow**
-

6. Primary User Flow (Voucher-First)

1. Customer spends R100+ at a Spaza
 2. Spaza issues a voucher code
 3. Customer sends voucher via WhatsApp
 4. System checks:
 5. Is the user registered?
 6. Is the voucher valid?
 7. If not registered → registration flow
 8. If registered → prediction link sent
 9. User submits predictions
 10. Entry is finalized
 11. Leaderboard link is sent
-

7. Registration Flow

Trigger Conditions

- User sends `new <SID>`
- OR user attempts to redeem a voucher while UNKNOWN

Backend Logic

1. Validate SID
2. Create or update user record
3. Set state to PENDING_REGISTRATION
4. Generate single-use registration link

WhatsApp Response

- Success: "Please complete your registration here: <registrationLink>"

- Failure: "Sorry, that spaza code is not valid. Please check and try again."
-

8. Registration Completion

Trigger

- Backend event after successful web registration

Actions

- Generate Leaderboard ID
- Generate weekly free prediction link

WhatsApp Message

Welcome! You are now registered.

Your Leaderboard ID is: ABC1234

Submit your predictions here:
<predictionLink>

9. Weekly Free Entry Broadcast

Trigger

- Start of PSL weekly cycle (scheduled job)

Audience

- All ACTIVE users

Message Template

PSL Weekly Predictions are live!

Submit before kickoff:
<predictionLink>

Want more entries? Spend R100+ and send your voucher code here.

10. Voucher Redemption Flow

Trigger

- User sends `code <VOUCHER>` or voucher-like message

Validation Rules

- User must be ACTIVE
- Voucher must:
 - Exist
 - Be unused
 - Be valid for the current week

Success Actions

- Mark voucher as redeemed
- Generate additional prediction link

WhatsApp Responses

- Success: "Code accepted. Here is your next prediction entry: <predictionLink>"
 - Invalid / used: "Sorry, that code is invalid or expired."
 - Not registered: "Please register first to use this voucher."
-

11. Prediction Submission Rules

- Each prediction link allows **one submission only**
- Once submitted, the entry is **final and immutable**
- Backend enforces no edits or resubmissions

Retry Message

"This entry has already been submitted and cannot be changed."

12. Post-Entry Handling

After every successful submission: - Entry is confirmed - Leaderboard link is sent

Message Template

Your entry has been accepted.

View your picks and history here:
<leaderboardLink>

13. Error & Edge Handling

- Expired / used link: "This link has expired. Watch out for the next round!"
 - Unknown input: "Use a voucher code or ask your Spaza for one to play."
-

14. Backend API Principles

- Transport: HTTPS
- Auth: Bearer Token (S2S → Backend)
- Idempotency supported
- UTC timestamps

Backend APIs follow previously agreed endpoint contracts.

15. POC Scope Guardrails

In Scope

- WhatsApp messaging
- Registration
- Voucher redemption
- Prediction submission
- Leaderboard viewing

Out of Scope

- Payments
 - Wallets
 - Prize payouts
 - Long-term user analytics
-

16. Success Criteria

The POC is successful if: - A user can register via WhatsApp - Redeem one or more vouchers - Submit predictions - See results on the leaderboard

End of Document (v2.0)