

Payments — Online Payment Toggle (online_payment_enabled)

Online Payment Toggle (online_payment_enabled)

Overview

QODY supports per-venue online payment gating via the `venue.online_payment_enabled` flag. This allows operational control over whether guests can pay online (via card/Stripe) or must use pay-at-table, without requiring code changes or redeployment.

Database Schema

Column: `venue.online_payment_enabled`

Type: `BOOLEAN NOT NULL DEFAULT true`

Migration: `V20__online_payment_toggle.sql` (Flyway)

Location: `apps/api/src/main/resources/db/migration/V20__online_payment_toggle.sql`

Purpose & Context

This flag serves as a **Monri-readiness gate**. While QODY waits for Monri (the BiH payment gateway for production) and online card payments currently run on **Stripe TEST mode**, this toggle allows hiding online payment per-venue.

Intended production payment provider: Monri (Model B — per-venue merchant accounts). See payment architecture decision memo for details on the per-venue Monri integration model.

Behavior

When `online_payment_enabled = true` (default)

- Guest checkout displays online payment UI (Stripe card element via Stripe.js)
- Guest can pay online immediately via `pay_now` flow
- Pay-at-table option also available if `pay_at_table_enabled = true`

When `online_payment_enabled = false`

- Guest checkout **hides** the online payment section (Stripe PaySection component)
- Guest must pay via "Plati kod konobara" (pay-at-table) — this requires `venue.pay_at_table_enabled = true`
- Server-side guard: `POST /guest/order` returns **503 Service Unavailable** with error code `no_payment_path` if both `online_payment_enabled = false` AND `pay_at_table_enabled = false`

Guest API

The flag is surfaced to the guest frontend via:

`GET /guest/venue-settings`

Response field: `onlinePaymentEnabled: boolean`

Frontend Gate

File: `apps/guest/src/pages/CheckoutPage.tsx`

Logic: `venueSettings?.onlinePaymentEnabled !== false`

If the flag is false or undefined-as-false, the Stripe PaySection is not rendered.

Operational Procedure — Re-enabling Online Payment

Scenario

When Monri integration is complete and a venue is ready to accept online payments in production.

Steps

1. **No redeploy required** — this is a pure database change
2. Connect to the QODY database (Azure Postgres)
3. Run the following SQL:

```
UPDATE venue
SET online_payment_enabled = true
WHERE id = '<venue-id>';
```

4. Verify: refresh the guest checkout page — online payment UI should appear

Demo Venue Details

Venue ID: 000000000-0000-0000-0000-000000000002

Name: QODY Demo Bistro

Current state (as of 2026-06-26):

- online_payment_enabled = false
- pay_at_table_enabled = true

Current Deployment State

Environment: Azure rg-qody-demo (Sweden Central)

API revision: qody-api--0000044

Guest revision: qody-guest--0000033

Commit: a9668a5

Verification: Proveo PASS 3/3 (live Playwright UAT, 2026-06-26)

Payment Architecture Notes

Current (temporary): Stripe Connect (TEST mode) — online payment disabled on demo venue pending Monri.

Production target: Monri (Model B — per-venue merchant accounts). Each restaurant will have its own Monri + bank account; QODY stores encrypted per-venue Monri credentials in Azure Key Vault and routes payments through the venue's own merchant account. QODY invoices platform fees separately (0.5% of subtotal + 29-49 KM/month per PRD \$2.3). This model avoids QODY holding customer funds and requiring a BiH payment institution license.

Related Documentation

- QODY PRD: `/tmp/qody-prd/PRD.txt` (\$2.3 commercial model, \$4 payment requirements)
- Monri architecture decision: `/tmp/qody-prd/monri-architecture-decision.md`
- Gap analysis: `/tmp/qody-prd/GAP-ANALYSIS.md`
- Payment flow UAT evidence: `/tmp/qody-prd/uat-guest.md`

Safety Constraints

At least one payment method must be enabled:

- If `online_payment_enabled = false`, then `pay_at_table_enabled` MUST be `true`
- Server-side enforcement: `GuestRoutes.kt` in `POST /guest/order` returns 503 if both are disabled
- Error code: `no_payment_path`

Change Log

Date	Change	Commit
2026-06-26	Feature implemented and deployed (V20 migration, CheckoutPage gate, venue-settings API)	a9668a5

Document owner: John (ALAI Ops)
Last updated: 2026-06-26
Related MC tasks: #104391, #104389

Revision #2
Created 2026-06-26 19:42:59 UTC by John
Updated 2026-06-26 19:49:41 UTC by John