



CONSULTING SERVICES PVT. LTD.

ETHOS TERMINAL SERIES · PRODUCT DATASHEET · DOC ETH-DS-CT01 · REV 1.0 · AUGUST 2026

# ethos.

## *Smart Canteen Terminal*

### MODEL ETH-CT01

Walk up, be recognised, collect a printed meal token — in seconds. A face-first canteen management terminal with an on-board rule engine, thermal token printing and serverless multi-kiosk sync over your LAN.

#### Three ways to identify

Face recognition · fingerprint (10 prints/user) · RFID card (optional)

#### 128-D face engine, on device

No cloud calls — detection, landmarks & matching run locally

#### Rule engine built in

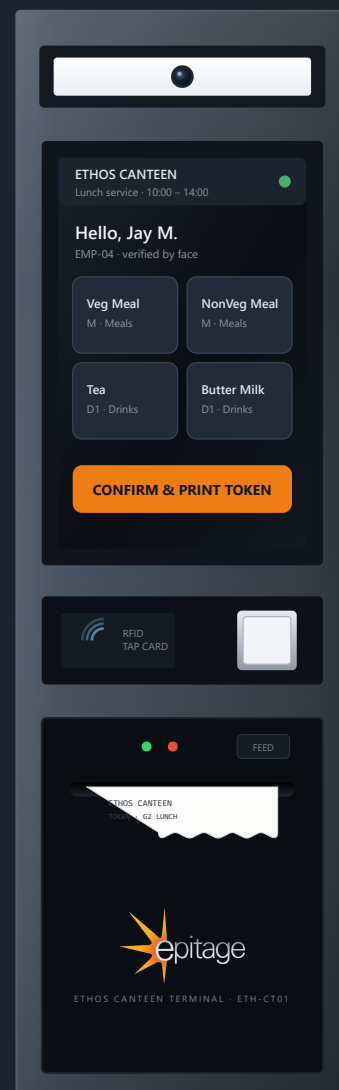
Shifts, time slots, menu quotas & duplicate-meal prevention

#### Instant thermal token

ESC/POS serial printer prints an ID slip & order receipt

#### Multi-kiosk mesh

Terminals replicate users & orders peer-to-peer over UDP — no central server



## 1 Product overview

The **Ethos Smart Canteen Terminal (ETH-CT01)** replaces coupon books, registers and crowded serving counters with a single wall-mounted device. An employee walks up, the illuminated camera bar identifies them by face in under a second of camera time, the built-in **rule engine** checks what the current service window and its shift entitle them to, they confirm their meal on the touch screen, and a **thermal token** prints on the spot — ready to hand to the serving counter.

Every terminal is self-contained: the kiosk UI, face matching, database and printing all run on the terminal itself. Multiple terminals on one LAN form a **peer-to-peer mesh** — registrations and orders made on any unit replicate to all others automatically, with no central server to install, license or maintain.

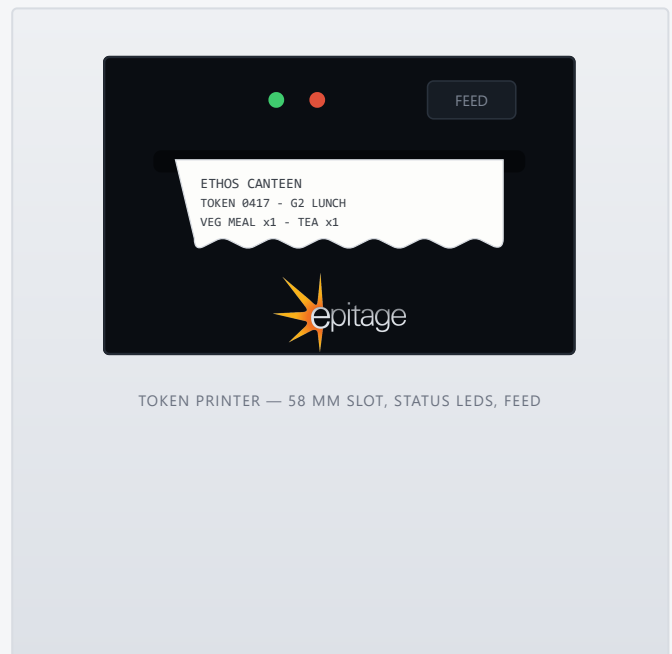
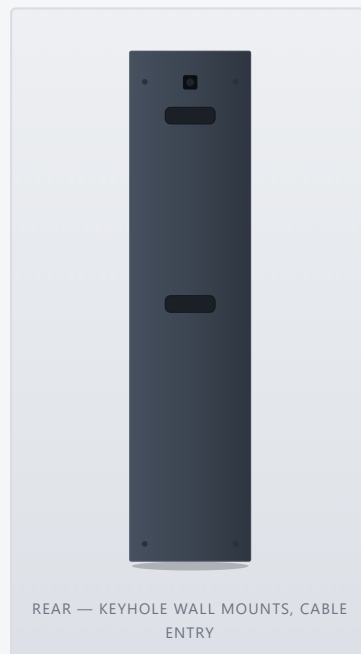
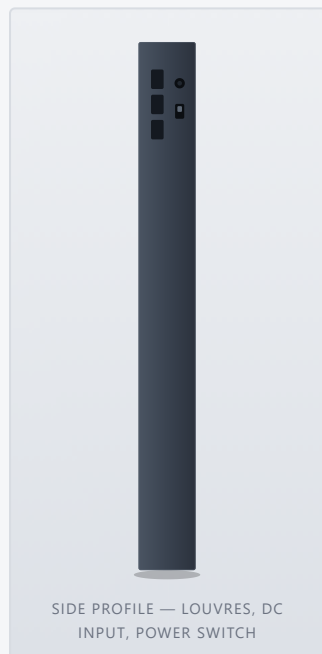
Ethos is designed and engineered in India by Epitage Consulting Services — and every layer of it (enclosure, menus, rules, token format, voice prompts) can be customised to your site.

### At a glance

|                |                                                                        |
|----------------|------------------------------------------------------------------------|
| Identification | Face recognition (primary) · Fingerprint, 10 prints/user · RFID card † |
| Face engine    | 128-D descriptors · match < 0.4 Euclidean · ~300 ms scan loop          |
| Display        | 7-inch capacitive touch panel †                                        |
| Token printer  | ESC/POS thermal · 58 mm roll† · serial UART @ 9600 bd                  |
| Compute        | Raspberry Pi-class Linux SBC †                                         |
| Connectivity   | Ethernet / Wi-Fi LAN · REST :5000 · UDP mesh :41234                    |
| Audio          | Voice prompts through built-in speaker                                 |
| Status LED     | 4-state indicator (idle / scanning / pass / fail)                      |
| Power          | 12 V DC barrel jack · rocker power switch †                            |
| Mounting       | Rear keyhole slots — wall mount, tool-free lift-off                    |

† Indicative — final values pending production drawings.

## 2 Product views



Rendered product illustrations are representative; final finish and dimensions per production drawings.

## 3 Where it fits

### Factory canteens

Shift-based workforces with fixed meal windows — the seeded configuration ships with 5 shifts and 8 meal windows spanning round-the-clock shift patterns, overnight included; windows are fully configurable.

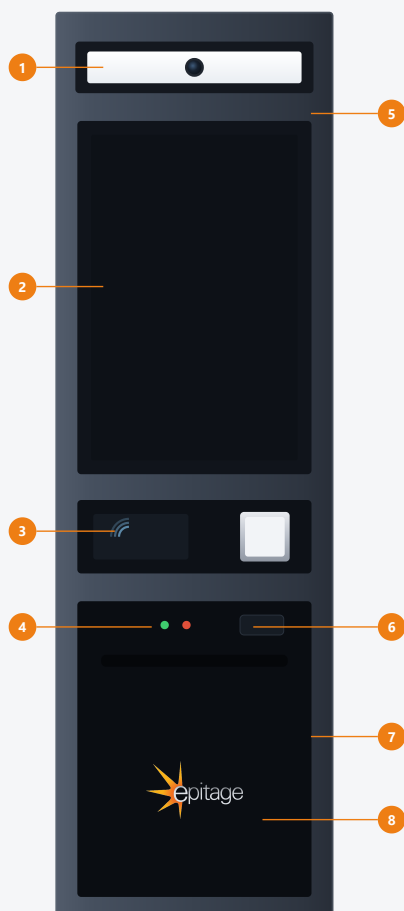
### Corporate cafeterias

Subsidised meal programmes where each menu group (meals, drinks, snacks) carries its own per-slot quota and per-item caps — enforced automatically at the terminal.

### Campus & hostel mess

High-throughput service where printed tokens keep the serving line moving and duplicate-meal prevention stops repeat trips within and across shifts.

## 4 Hardware anatomy



| # | ELEMENT                | DETAIL                                                                                                                                                   |
|---|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | Illuminated camera bar | USB camera behind a diffused white light strip — consistent face illumination indoors, day or night. Captures at 320 × 240 for the recognition pipeline. |
| 2 | Touch display          | Portrait-mounted capacitive panel running the kiosk UI in a full-screen browser. On-screen keyboard for operator entry — no physical keyboard needed.    |
| 3 | RFID reader zone       | Tap area for card-based identification as a fallback or companion to face ID. <sup>†</sup>                                                               |
| 4 | Fingerprint scanner    | Optical scanner with white surround illumination. Enrolment stores up to 10 fingerprints per user (all fingers, both hands).                             |
| 5 | Status LED & speaker   | Side-mounted indicator LED plus internal speaker for voice prompts; louvres double as ventilation.                                                       |
| 6 | Printer status & feed  | Green (ready) and red (fault/paper-out) LEDs with a paper feed button.                                                                                   |
| 7 | Thermal printer        | 58 mm <sup>†</sup> ESC/POS thermal mechanism on serial UART, 9600 baud. Prints identification slips and order tokens; tear-bar exit, front-loading roll. |
| 8 | Branding plate         | Customisable fascia — your company or canteen branding on request.                                                                                       |

### Status LED language

- ☐ **White, steady** — idle, ready for the next person
- ☐ **White, blinking** — scanning fingerprint / capturing
- ☒ **Green** — authentication successful
- ☐ **Red** — authentication failed

### Voice prompts

|                      |                                        |
|----------------------|----------------------------------------|
| "Thank You"          | Token issued successfully              |
| "User Already Taken" | Meal already availed for this window   |
| "No User Found"      | Person not registered on this terminal |

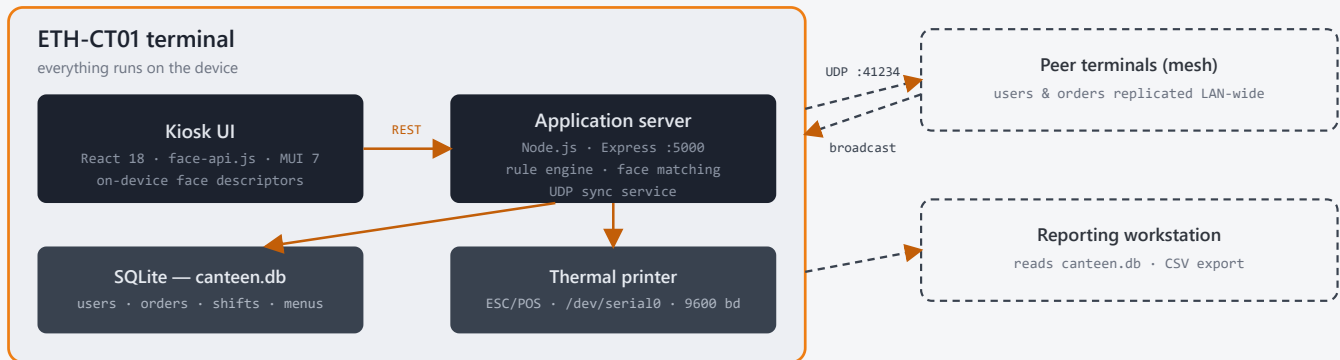
## 5 Physical & electrical

|                    |                                                                     |
|--------------------|---------------------------------------------------------------------|
| Enclosure          | Powder-coated sheet-steel tower, graphite grey; crisp folded fascia |
| Dimensions (H×W×D) | ≈ 500 × 150 × 90 mm <sup>†</sup>                                    |
| Weight             | ≈ 3.5 kg <sup>†</sup>                                               |
| Mounting           | Two rear keyhole slots on vertical centre-line; lift-off removal    |
| Recommended height | Camera bar at 1.4 – 1.6 m from floor                                |

<sup>†</sup> Indicative — final values pending production drawings.

|                       |                                                                |
|-----------------------|----------------------------------------------------------------|
| Input power           | 12 V DC barrel jack, external adapter <sup>†</sup>             |
| Power switch          | Side rocker switch                                             |
| Operating environment | Indoor, dry locations; keep direct sunlight off the camera bar |
| Cooling               | Passive, side louvres                                          |
| Consumables           | 58 mm <sup>†</sup> thermal paper roll                          |

## 6 System architecture



## 7 Software stack

| LAYER       | TECHNOLOGY                                          |
|-------------|-----------------------------------------------------|
| Kiosk UI    | React 18.2 · MUI 7 · react-simple-keyboard 3.8      |
| Face engine | face-api.js 0.22.2 (TensorFlow.js, in-browser)      |
| API server  | Node.js · Express 4.21 · CORS enabled               |
| Database    | SQLite 3 (sqlite3 5.1) — single file, zero admin    |
| Printing    | serialport 13.0 → ESC/POS commands                  |
| Sync        | Node dgram — JSON datagrams, UDP broadcast :41234   |
| Delivery    | Kiosk browser in full-screen on the terminal itself |

## 8 Face recognition engine

|              |                                                                     |
|--------------|---------------------------------------------------------------------|
| Detector     | TinyFaceDetector · input 160 px · score ≥ 0.5                       |
| Landmarks    | 68-point model                                                      |
| Descriptor   | 128-dimension float vector per face                                 |
| Matching     | Nearest neighbour, Euclidean distance < 0.4                         |
| Scan cadence | Live detection every 300 ms on 320×240 video                        |
| Enrolment    | Single capture per user, operator-assisted, on the terminal         |
| Privacy      | Only the numeric descriptor is stored — no face photos are saved    |
| Locality     | All models load from the device; recognition works with no internet |

## 9 Terminal REST interface

| ENDPOINT           | METHOD | FUNCTION                                                                                                                                                                                      |
|--------------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| /api/identify-user | POST   | Match a face descriptor; returns the employee, current shift & time slot, eligible menus with quotas, today's consumption and a previous-shift duplicate flag. Prints an identification slip. |
| /api/orders        | POST   | Record the confirmed selection (per item: menu, item, count, shift, slot), print the order token and broadcast the order to peer terminals.                                                   |
| /api/register-user | POST   | Enrol name + employee ID + face descriptor; rejects a face already enrolled; replicates the new user to all terminals.                                                                        |
| /api/employees     | GET    | Employee directory (ID + name) for the operator search on the registration screen.                                                                                                            |

## 10 A meal in seven steps



If rules deny — clear on-screen notice,  
voice prompt "User Already Taken",  
red LED — no order token prints.

## 11 The rule engine

Entitlement is decided **at identification time**, on the terminal, from four layers of configuration held in the local database:

|                    |                                                                                                                                                                                             |
|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 · Time slot      | The wall clock resolves the active meal window from <code>time_slot_master</code> (overnight windows supported — e.g. 22:00 → 04:00).                                                       |
| 2 · Menus for slot | <code>menu_timeslot_mapping</code> lists which menu groups serve in this window and the <b>maximum units per menu</b> (e.g. G2 Lunch: 1 Meal + 3 Drinks).                                   |
| 3 · Item caps      | <code>menu_item_mapping</code> caps single items inside a menu (e.g. max 2 × Coffee within Drinks).                                                                                         |
| 4 · Consumption    | Today's orders for this employee, shift and slot are summed live — exhausted menus disappear from the screen; a <b>previous-shift check</b> flags anyone returning under a different shift. |

### Factory-seeded configuration

| SHIFTS (5)                                                              | MEAL SLOTS (8)                                                                                               |
|-------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| Morning 06:00–14:00                                                     | F1 B'fast 07–09 · F2 Lunch 10–13                                                                             |
| General 09:00–17:00                                                     | G1 B'fast 09–10 · G2 Lunch 10–14                                                                             |
| Split 10:00–14:00                                                       | S1 B'fast 14–17 · S2 Dinner 18–22                                                                            |
| Afternoon 14:00–22:00                                                   | T1 Dinner 22–04 · T2 Snacks 03–05                                                                            |
| Night 22:00–06:00                                                       |                                                                                                              |
| MENU GROUPS (5)                                                         | ITEMS (11)                                                                                                   |
| S1 B'fast Snacks · D1 Drinks · M Meals · D2 Drinks · S2 Midnight Snacks | Pohe · Misal · Vadapav · Tea · Coffee · Veg Meal · NonVeg Meal · Lime Juice · Butter Milk · Farsan · Khichdi |

Every table is data, not code — reshape shifts, windows, menus and quotas per site without touching the application. Note: the factory seed maps no service windows to the Split Shift — map or remove it per site.

## 12 What prints

```

ETHOS CANTEEN
-----
User ID : 4   Shift ID : 2
Name    : Jay Mehta
Date    : 3 August 2026
Time    : 12:41:07
-----
=====
  
```

```

ETHOS CANTEEN
-----
User ID : 4   Shift ID : 2
Date    : 3 August 2026
Time    : 12:41:32
Order Summary
Items :
- Veg Meal - 1
- Tea - 1
-----
=====
  
```

### Identification slip & order token

Each successful identification prints a slip with the employee, shift and timestamp; each confirmed order prints an itemised token. Both are timestamped in IST and cut manually at the tear bar. Fingerprint-mode tokens carry Name · Shift ID · Canteen, where the device name matches the canteen location so multi-canteen sites stay traceable.

**Customisable:** token layout, headers, language and QR/serial additions are engineering options — talk to Epitage. Samples above show the intended layout; the current build prints plain headerless text (see §20).

## 13 Commissioning guide

**Before you start:** terminal mounted with the camera bar at 1.4–1.6 m · 12 V supply connected · 58 mm paper roll loaded · terminal joined to the site LAN. Software prerequisites on the terminal: Node.js LTS with npm, and a Chromium-family kiosk browser.

### 01 Install the application server

Copy the server/ package to the terminal and install dependencies:

```
cd Canteen/server
npm install
npm start      # "Server running on port 5000"
```

Always launch from inside `server/` — the database path `./canteen.db` is resolved from the working directory.

### 02 Verify the database

`canteen.db` must sit beside `server.js`, pre-seeded with shifts, time slots, menus, items and quota mappings (section 11). The server does not create the schema on its own.

### 03 Connect the token printer

Wire the thermal printer to the SBC UART (/dev/serial0, 9600 baud). On Raspberry Pi, enable the UART and free it from the login console first:

```
sudo raspi-config # Interface Options → Serial Port:
                  # login shell OFF · serial hardware ON
```

Different printer port? Edit `SERIAL_PORT` in `services/printingService.js`.

### 04 Build the kiosk UI

```
cd Canteen/client
npm install
npm run build      # production bundle in client/build/
```

Serve `client/build/` from any static server on the terminal and open it in the kiosk browser, full-screen. The face models in `public/models/` ship with the build and load locally at boot.

### 05 Grant the camera

On first launch the browser asks for camera access — allow it, and set the kiosk browser to remember the grant. Camera capture requires the UI to be served from `localhost` or `HTTPS`.

### 06 Join terminals into a mesh (optional)

Put every terminal on the same LAN broadcast domain and allow UDP 41234 between them — that is all: discovery is automatic, peers replicate user and order events, and a terminal ignores its own broadcasts.

### 07 Repoint the UI at a remote server (optional)

The kiosk UI ships pinned to `http://localhost:5000`. To run the UI and server on different hosts, update the URL at its five call sites and rebuild:

```
# src/components/faceCapture.jsx      (identify)
# src/components/faceRegistration.jsx  (employees, register)
# src/components/foodSelect.jsx       (orders)
# src/App.js                          (legacy path)
npm run build
```

### 08 Smoke-test

Register a test user (section 15), identify them, place an order, and confirm: green LED, "Thank You" prompt, and a printed token. Then delete the test rows from `users / order_table`.

## 14 Site configuration — shifts, windows & menus

Operational configuration lives in database tables (edited via SQL today — the Time, Shift, Menu Group and Menu Management admin modules per the platform specification are in development). Direct SQL:

```
-- a new meal window
INSERT INTO time_slot_master
(TimeSlotName, start_time, end_time)
VALUES ('Evening Snacks', '16:30', '17:30');

-- serve it to General Shift (id 2)
INSERT INTO shift_time_slot_mapping VALUES (2, 9);
```

```
-- put Drinks (menu 2) on the new window,
-- max 2 units per person
INSERT INTO menu_timeslot_mapping VALUES (9, 2, 2);

-- cap Coffee (item 5) at 1 within Drinks
UPDATE menu_item_mapping SET count = 1
WHERE menu_id = 2 AND item_id = 5;
```

**Cover the clock:** when a moment of the day falls outside every configured window, the terminal has no menu to offer — keep windows contiguous across your service hours (the factory seed leaves 05:00–07:00 and 17:00–18:00 open by design).

## 15 Enrolling people

- 01 Open registration**  
Tap **Register User** on the home screen.
- 02 Find the employee**  
Search by name or ID with the on-screen keyboard — the directory suggests the top matches; tap one to fill the form.
- 03 Start the camera**  
Position the person 40–60 cm away facing the camera; when the detection box appears the hint flips to *"Face detected! Click capture."*
- 04 Capture & register**  
One clear capture is enough — the terminal computes the 128-number descriptor and stores it. A face already on file is rejected with the existing employee named, so one face can never map to two IDs.
- 05 Fingerprints (optional)**  
Enrol up to 10 prints per user — all fingers of both hands — for glove-free fallback identification.

**Good captures:** even, front-on lighting (the camera bar provides it), glasses off if they glare, one face in frame, neutral expression. Re-enrol after major appearance changes.

## 16 Daily service

- 01 Look at the camera**  
Tap **Login User** (or simply step up on kiosks configured to auto-arm). Recognition completes on the first clean frame.
- 02 Review your allowance**  
The screen greets you by name and shows only the menus the current service window allows, with what you have already taken today.
- 03 Select & confirm**  
Use the + / – steppers; the terminal blocks over-quota picks as you go. Tap **Submit Order**.
- 04 Take your token**  
Green LED, "Thank You", and the itemised token prints. Tear and hand it to the counter.

## 17 Reporting

Every transaction lands in `order_table` with employee, menu, item, count, shift, slot and an IST timestamp — one query away from your MIS:

```
-- today's consumption by item
SELECT i.item_name, SUM(o.count) AS served
FROM order_table o JOIN item i USING (item_id)
WHERE DATE(o.timestamp) = DATE('now','localtime')
GROUP BY i.item_name ORDER BY served DESC;
```

Read the SQLite file directly, or export CSV from any SQLite tool on the reporting workstation.

## 18 Care & maintenance

### Daily

Wipe the screen and camera window with a dry microfibre cloth. Check the paper roll before each meal rush; the red printer LED signals paper-out.

### Monthly

Dust the side louvres. Verify terminal clock against IST — all rule decisions are wall-clock based. Back up `canteen.db` (a simple file copy while the canteen is closed).

### Per season

Review shift & window tables against HR calendars. Re-enrol faces flagged by staff as slow to match. Archive old `order_table` rows to keep queries brisk.

## 19 Limitations & deployment guidelines

| IDENTIFICATION     |                                                                                                                                                                | PLATFORM & NETWORK     |                                                                                                                                                                      |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Indoor use         | Camera pipeline is tuned for indoor light; avoid direct sun or strong backlight on the approach.                                                               | LAN product            | REST and UDP sync are open on the LAN with no authentication — deploy on an isolated/segmented canteen VLAN only.                                                    |
| One face at a time | The first detected face is processed — queue people single-file at the terminal.                                                                               | Biometric transport    | Face descriptors travel over plain HTTP on the local link (and appear in the browser debug console in this build) — keep the kiosk network isolated; TLS is planned. |
| No liveness check  | Descriptor matching does not detect photos; supervise enrolment and place terminals in staffed areas.                                                          | Mesh semantics         | UDP replication is fire-and-forget: no delivery guarantee or conflict resolution. Peers that were offline do not back-fill missed events.                            |
| Open enrolment     | Registration is reachable from the idle screen and employee details are operator-typed — restrict enrolment to supervised sessions until operator login ships. | Kiosk-side enforcement | Quota checks run in the kiosk UI; the orders API itself does not re-validate. Keep the API unreachable from user networks.                                           |
| Single enrolment   | One face capture per user; large appearance changes may need re-enrolment.                                                                                     | Printing               | One print job at a time; a printer fault does not block the order (token can be lost) — watch the red LED.                                                           |
| Population size    | Matching scans all users per identification — sized for workforces in the hundreds; contact us beyond ~1,000.                                                  | Time & locale          | IST (Asia/Kolkata) fixed; correct system time is essential.                                                                                                          |
|                    |                                                                                                                                                                | Server address         | UI is compiled against localhost:5000 — remote-server layouts need a rebuild (\$13.7).                                                                               |

## 20 Known issues — software build under review

| ISSUE                                     | IMPACT & PLANNED FIX                                                                                                                                                |
|-------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Order receipt prints with empty fields    | The order token renders placeholders instead of order data (argument defect in the print call); identification slips are unaffected. Fix scheduled.                 |
| Cross-shift duplicate check date handling | On the 1st–10th of a month the previous-shift window can be mis-computed (unpadded dates), weakening the cross-shift block on those days.                           |
| Peer “identified” status not applied      | The status-sync message references a column absent from the live schema, so the status-sync path is inoperative in this build; identification itself is unaffected. |
| On-screen keyboard stays open             | On the registration screen the keyboard cannot be dismissed until the camera step.                                                                                  |
| Camera stream lingers                     | The webcam can remain active after leaving the identify/registration screens until the flow resets.                                                                 |
| Per-item cap edge case                    | An internal field mismatch disables one of the two quantity guards; the menu-level quota still limits totals.                                                       |

Issues are listed for transparency toward pilot deployments; none block the core identify → order → print loop.

## 21 Troubleshooting

| SYMPTOM                                      | LIKELY CAUSE                                             | ACTION                                                                                            |
|----------------------------------------------|----------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| "Failed to load face recognition models"     | Model files missing from <code>public/models/</code>     | Restore the three model sets (detector, landmarks, recognition) and reload the kiosk page.        |
| Camera never starts                          | Permission denied, or page not on localhost/HTTPS        | Re-grant camera permission in the kiosk browser; serve the UI from localhost or HTTPS.            |
| "User not recognized" for enrolled staff     | Lighting change, appearance change, or match > 0.4       | Retry facing the camera bar squarely; re-enrol the user if it persists.                           |
| No menus shown after identify                | Clock outside every configured window, or shift unmapped | Check terminal time; verify the current time falls in a window mapped to a shift and menus (§14). |
| Token does not print                         | Paper out (red LED), UART disabled, or wrong port        | Load a 58 mm roll; confirm UART settings and <code>SERIAL_PORT</code> (§13.3).                    |
| Orders not appearing on other terminals      | Different subnet or UDP 41234 blocked                    | Put terminals in one broadcast domain; allow UDP 41234 in switch/firewall policy.                 |
| All users unrecognised / employee list empty | Started outside server/ — a fresh empty DB was created   | Stop, delete the stray <code>canteen.db</code> , start again from the server directory.           |

### Ordering & variants

|                 |                                                                                          |
|-----------------|------------------------------------------------------------------------------------------|
| <b>ETH-CT01</b> | Smart Canteen Terminal — this document                                                   |
| <b>ETH-AT01</b> | Smart Attendance Terminal — same platform, printer-less (separate datasheet ETH-DS-AT01) |
| <b>Options</b>  | Branding fascia · token layout · languages · enclosure colour · site-specific rules      |

© 2026 Epitage Consulting Services Pvt. Ltd. Specifications describe the software build reviewed in August 2026 and are subject to change; values marked + are indicative pending production drawings. This document does not constitute a warranty.

### Support

**Epitage Consulting Services Pvt. Ltd.**  
 enquiry@epitage.com  
 +91 98503 01345

Sales, commissioning, site customisation and AMC support for the entire Ethos Terminal Series.