

# Context

# Triển khai hoá đơn điện tử tại Việt Nam: bối cảnh, thách thức và giải pháp trung gian

Đến năm 2025, gần như tất cả doanh nghiệp và hộ kinh doanh tại Việt Nam phải phát hành hoá đơn điện tử theo quy định của Bộ Tài chính và Tổng cục Thuế. Việc chuyển đổi này nhằm minh bạch hoá quá trình quản lý thuế và dựa trên dữ liệu giao dịch thực tế. Nhiều nền tảng thanh toán – chẳng hạn VNPay – đã tích hợp sẵn chức năng xuất hoá đơn điện tử vào phần mềm quản lý bán hàng của mình (xem hình minh hoạ), giúp tự động hoá quy trình lập hoá đơn và gửi dữ liệu đến cơ quan thuế.

## Khó khăn kỹ thuật và quy mô trong xuất hoá đơn điện tử

- **Khối lượng và hạ tầng lớn:** Nền tảng thanh toán phải xử lý hàng nghìn hoá đơn mỗi ngày. Theo quy định, các nhà cung cấp giải pháp hoá đơn điện tử phải duy trì kênh kết nối riêng với Tổng cục Thuế (VPN/MPLS) có băng thông tối thiểu 20 Mbps và giao thức truyền dữ liệu an toàn (SOAP/TCP). Đáp ứng yêu cầu này đòi hỏi đầu tư hạ tầng công nghệ thông tin lớn, trong khi mọi hoá đơn cần được ký số tập trung bằng HSM nhanh chóng.
- **Tốc độ và đồng bộ:** Hóa đơn điện tử có mã của cơ quan thuế phải được gửi ngay sau khi lập để xin cấp mã, còn hoá đơn không có mã cũng phải gửi chậm nhất trong cùng ngày lập. Điều này đòi hỏi hệ thống phải hoạt động ổn định và đồng bộ thời gian thực. Nếu xảy ra lỗi (ví dụ định dạng sai, chữ ký số hết hạn), hoá đơn có thể bị từ chối, gây chậm trễ kê khai thuế và rủi ro bị xử phạt. Các giải pháp “All-in-One” ra đời nhằm giảm thiểu sai sót này. Ví dụ, VNPay-Invoice tích hợp tất cả khâu từ xuất hoá đơn đến truyền dữ liệu; nhờ đó “giúp giảm rủi ro lỗi kết nối, sai chuẩn dữ liệu và thiếu thông tin khi kiểm tra, đối soát hoặc thanh tra”.
- **Đa dạng nghiệp vụ và tuân thủ:** Từ 1/7/2022, mọi hộ kinh doanh, cá nhân kinh doanh phải sử dụng hoá đơn điện tử, khiến số lượng người tham gia tăng vọt. Nhiều chủ đơn vị nhỏ quen sổ sách thủ công, nên quy trình chuyển đổi và đào tạo nhanh chóng là thách thức lớn. Ngoài ra, hoá đơn điện tử phải lưu trữ tối thiểu 10 năm theo quy định và tích hợp với các nghiệp vụ khác (bán hàng, ký hợp đồng, kê khai thuế). Các nhà cung cấp giải pháp như VNPay đáp ứng bằng bộ công cụ “all-in-one” trên di động: quản lý bán hàng, xuất hoá đơn, chữ ký số, và đồng bộ kê khai thuế. Giải pháp này cho phép doanh thu được cập nhật tức thì và giảm thiểu rủi ro sai lệch giữa doanh thu thực tế và số liệu kê khai.

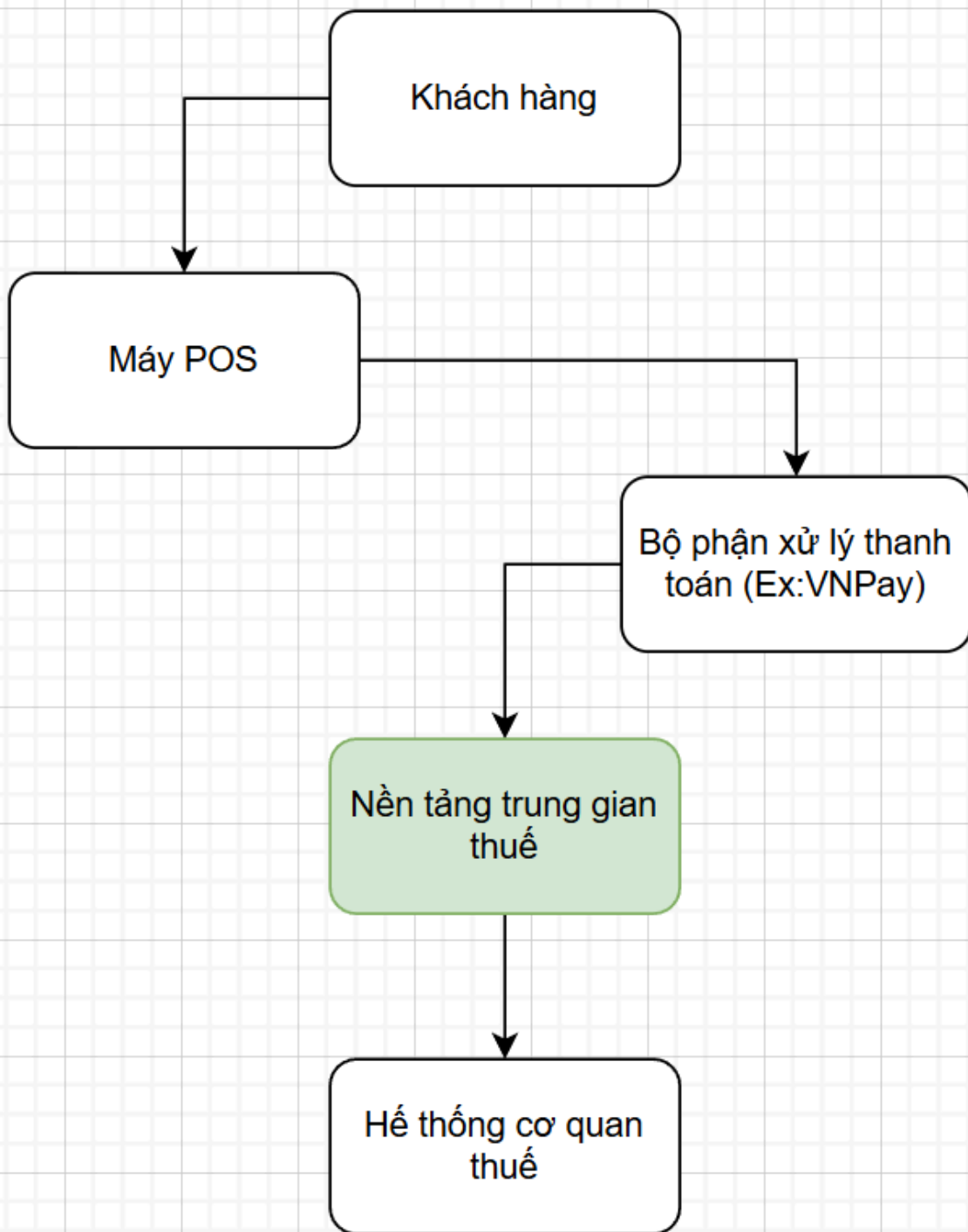
## Quy định pháp lý và nền tảng trung gian thuế

Theo quy định tại Nghị định 123/2020 và Thông tư 78/2021, tất cả hoá đơn điện tử phải được truyền lên Cổng thông tin của cơ quan thuế. Cụ thể, các tổ chức cung cấp giải pháp hoá đơn điện tử (như VNPay) phải đáp ứng hạ tầng CNTT và ký hợp đồng chính thức với Tổng cục Thuế để nhận, truyền dữ liệu hoá đơn. Từ 1/7/2022, hoá đơn điện tử áp dụng cho mọi hộ kinh doanh, cá nhân kinh doanh; Nghị định 70/2025 sẽ tiếp tục mở rộng phạm vi này. Việc kết nối với cơ quan thuế là bắt buộc, giúp Cục Thuế theo dõi giao dịch theo thời gian thực và tăng hiệu quả quản lý.

- **Kết nối bắt buộc với cơ quan thuế:** Hóa đơn có mã phải được gửi ngay để cơ quan thuế cấp mã, hoá đơn không mã phải gửi cùng ngày lập. Doanh nghiệp không kết nối trực tiếp sẽ ký hợp đồng với tổ chức cung cấp dịch vụ hoá đơn điện tử (T-VAN) để trung gian truyền dữ liệu. Ví dụ, Thông báo 422/TB-TCT (10/2021) của Tổng cục Thuế đã công nhận 9 tổ chức làm T-VAN (chẳng hạn MISA, Nacencomm, VNPT Hilo, FPT...) để kết nối kỹ thuật và truyền dữ liệu hoá đơn.
- **Lý do tồn tại của nền tảng trung gian thuế:** Trung gian hoá đơn điện tử tập trung giúp gom dữ liệu từ nhiều đơn vị, đảm bảo tuân thủ đúng chuẩn kỹ thuật của cơ quan thuế và xử lý đồng bộ. Thay vì mỗi doanh nghiệp tự xây hạ tầng phức tạp, họ sử dụng giải pháp được Tổng cục Thuế phê duyệt. Chẳng hạn, dịch vụ VNPAY-TVAN của VNPay tiếp nhận hoá đơn từ khách và chuyển lên cổng thuế với “kết nối ổn định và tuân thủ đúng chuẩn kỹ thuật”. Nhờ vậy, toàn bộ luồng hoá đơn được đẩy thẳng đến cơ quan thuế, giảm rủi ro lỗi hệ thống và đảm bảo doanh nghiệp tuân thủ đúng quy định.

**Nguồn:** Các quy định về hoá đơn điện tử (NĐ 123/2020, TT 78/2021, NĐ 70/2025) và báo cáo thị trường thuế Việt Nam

# Problem



### 1. Khách hàng thanh toán tại cửa hàng

- Thanh toán bằng QR, thẻ, ví... qua **VNPay**
- Giao dịch diễn ra trên **máy POS**

### 2. POS tạo đơn hàng

- Mỗi lần thanh toán thành công = **1 đơn hàng**
- Đơn hàng này có:
  - Số tiền
  - Thời gian
  - Mã giao dịch
  - Mã cửa hàng
  - Hình thức thanh toán

### 3. VNPay nhận & xử lý giao dịch

- Xác nhận thanh toán
- Lưu transaction
- → **bắn (push) dữ liệu giao dịch ra ngoài**

### 4. Dữ liệu không đi thẳng tới Thuế

- Thực tế **không phải** cửa hàng nào cũng tự tích hợp với hệ thống Thuế
- ⇒ cần một **nền tảng trung gian**

### 5. Nền tảng trung gian thuế làm gì?

- Nhận dữ liệu từ VNPay / POS
- Chuẩn hóa dữ liệu theo chuẩn của Thuế
- Tạo:
  - Hóa đơn điện tử

- Báo cáo doanh thu
- Dữ liệu kê khai thuế
- Gửi dữ liệu sang **Hệ thống Thuế**

## Mối quan hệ cốt lõi

- Giao dịch thanh toán là gì?
- Hóa đơn là gì?
- Thuế gắn vào đâu?
- Vì sao **mỗi hóa đơn phải có thuế**?
- Vì sao **nhiều hóa đơn** → **hệ thống dễ nghẽn / sai sót**?

Thanh toán (transaction)

→ tạo hóa đơn (invoice)

→ hóa đơn có thuế

→ tổng hợp để kê khai

Thu thập số Thuế theo từng đầu vào.

- VNPay / POS tạo **rất nhiều hóa đơn**
- Không thể:
  - Mỗi cửa hàng tự gửi thuế
  - Mỗi hóa đơn xử lý thủ công
- Cần:
  - Gom dữ liệu

- Chuẩn hóa
- Gửi đi theo luồng

👉 Đây là **core problem** của đề tài.

Bạn NÊN hiểu: Phân loại hóa đơn

Chỉ cần biết khái niệm:

- Hóa đơn bán hàng
- Hóa đơn điện tử
- Có / không có mã (chỉ biết tên)

POS → Payment → Invoice → Tax → Report

nền tảng **phải tạo/chuẩn hóa dữ liệu** để DN **kê khai nhanh** (đặc biệt VAT) và **xử lý HĐĐT đúng quy định**: đăng ký HĐĐT, phát hành, sai sót, tổng hợp dữ liệu...

Dựa trên luật về thuế, mình giải thích data về luật thuế kê khai cho người ta dễ hiểu hơn



resource

Project Excel:

<https://docs.google.com/spreadsheets/d/1j93-kYkb5ZFIFHJAeB5sqPPN5dDdbQWqQ1SqY9JGojg/edit?gid=650502324#gid=650502324>

<https://vnpay.js.org/>

VN Pay: <https://github.com/lehuygiang28/vnpay>

MVP

Login + phân quyền cơ bản (Admin/Staff/Accountant)

Quản lý doanh nghiệp/khách hàng

Tạo + quản lý hoá đơn (draft → issued → cancelled)

Tính thuế theo rule đơn giản (VAT, tổng hợp)

Xuất PDF/Excel (hoặc template)

Tích hợp cổng thanh toán (vd VNPAY) + xử lý trạng thái giao dịch

Audit log, lịch sử thay đổi, chuẩn hoá trạng thái nghiệp vụ

Dashboard/report cơ bản

Deploy + demo flow end-to-end

Risk:

Nghiệp vụ hoá đơn/thuế (BR)

**Tích hợp bên thứ ba** (thanh toán/hoá đơn điện tử) + xử lý lỗi + retry

**Báo cáo & đối soát** (dữ liệu phải sạch và đúng)

# Kế hoạch sprint 1

## 1) Mục tiêu Sprint (Sprint Goal)

- Chốt **scope nhỏ nhưng hoàn chỉnh end-to-end** (ví dụ: đăng nhập → tạo 1 entity → lưu DB → hiển thị UI).
- Dự án **build/run được**, có demo được.

## 2) Sprint Backlog (việc phải làm)

### A. Chuẩn bị dự án (setup)

- Tạo repo, **branch strategy**, code convention
- Setup CI cơ bản (lint/build/test), **cấu trúc thư mục**
- Setup môi trường dev: DB, env, Docker)

### B. Làm “vertical slice” đầu tiên

- Thiết kế Conceptual DB
- API (CRUD cơ bản + validation)
- UI (form + list)
- Kết nối end-to-end (FE ↔ BE ↔ DB) (SAD)

### C. Chất lượng & quản trị

- Unit test / smoke test tối thiểu
- Tài liệu: README, cách chạy, demo script, AGENTS.md

## 3) Phân công & ước lượng

- Mỗi task có: **owner + deadline**
- Ưu tiên: MVP

## 4) Definition of Done (tiêu chí hoàn thành)

Ví dụ:

- Code đã review/merge, build pass
- Chạy được ở máy người khác theo README
- Có dữ liệu test tối thiểu
- Demo được theo kịch bản

## **5) Sprint ceremonies (nhịp làm việc)**

- Sprint Planning (đầu sprint): chốt goal + backlog
- Daily (mỗi ngày 10–15p): hôm qua làm gì / hôm nay làm gì / vướng gì
- Sprint Review (cuối sprint): demo
- Retrospective: rút kinh nghiệm, cải tiến sprint 2

Tao thu thập được



## 1) “Nền tảng trung gian thuế” (Tax Middleware Platform) là gì?

Trong ngữ cảnh đề tài của bạn (POS/PSP/VNPAY → xuất HĐĐT → báo cáo/tuân thủ), “nền tảng trung gian thuế” có thể hiểu là **lớp phần mềm đứng giữa hệ thống bán hàng/thanh toán và hệ thống thuế – hoá đơn**, làm 3 việc chính:

- **Chuẩn hoá dữ liệu giao dịch\*** từ POS/PSP (mã đơn, merchant, line items, tổng tiền, trạng thái thanh toán...)
- **Áp luật thuế & quy tắc hoá đơn**: map hàng hoá/dịch vụ → nhóm thuế/thuế suất, tính thuế, quyết định loại hoá đơn (có mã/không mã...), quy trình xử lý sai sót, hủy/điều chỉnh.
- **Điều phối phát hành & truyền nhận**: gọi API nhà cung cấp hoá đơn điện tử / kết nối cơ quan thuế, lưu vết (audit log), đối soát và báo cáo.

Vì pháp lý ở VN đang bám chặt vào **hoá đơn điện tử** (quản lý, sử dụng, truyền dữ liệu), nên “trung gian thuế” trong thực tế thường **gắn với “nền tảng trung gian hoá đơn điện tử + tax rule engine + reconciliation”**\* hơn là làm thay toàn bộ khai thuế. (Nói đơn giản: bạn “làm cho việc tuân thủ thuế qua hoá đơn trở nên tự động và ít lỗi”.)

Các khung pháp lý bạn sẽ chạm mạnh nhất:

- Quy định chung về hoá đơn/chứng từ: **Nghị định 123/2020/NĐ-CP**.
- Hướng dẫn triển khai/giải pháp kết nối, truyền – nhận dữ liệu HĐĐT: **Thông tư 78/2021/TT-BTC**.
- Mảng “hoá đơn điện tử khởi tạo từ máy tính tiền” (liên quan POS): có cập nhật/nhấn mạnh hiệu lực từ **01/6/2025** và nội dung sửa đổi tại **Nghị định 70/2025/NĐ-CP** (sửa NĐ 123).

## 2) Ở Việt Nam có bao nhiêu loại thuế? (nhìn theo cách “bạn cần cho đề tài”)

Nếu nói “hệ thống thuế” theo nghĩa rộng thì có nhiều sắc thuế. Một bài tổng hợp pháp lý có liệt kê các sắc thuế hiện hành và giải thích theo từng luật thuế.

Nhưng với **B2B trung gian thuế gắn POS/Payment → HĐĐT**, thực tế bạn sẽ tập trung vào **nhóm thuế phát sinh theo giao dịch bán hàng/dịch vụ** và nghĩa vụ liên quan:

**Nhóm “giao dịch bán hàng/dịch vụ” (đụng trực tiếp)**

- **Thuế GTGT (VAT):** thuế suất/đối tượng chịu thuế gắn với hàng hoá/dịch vụ (đây là phần “tax engine” hay nhất).
- **Thuế Tiêu thụ đặc biệt (TTĐB):** chỉ với một số ngành/hàng hoá (rượu bia, thuốc lá...); nếu mentor muốn “define category, tính thuế” thì đây là case điển hình.
- **Thuế/Phí theo ngành** (tài nguyên, BVMT, xuất nhập khẩu...): thường không phải core cho luồng POS phổ thông, nhưng là “edge cases” nếu bạn muốn mở rộng.

### Nhóm thuế doanh nghiệp hay nhắc (nhưng không tính theo từng bill bán lẻ)

- **Thuế TNDN (CIT):** tính theo lợi nhuận kỳ kế toán, không phải “mỗi hoá đơn ra một phát tính xong”.
- **Thuế TNCN (withholding):** liên quan bảng lương/khấu trừ, cũng không nằm trong “luồng thanh toán từng đơn”.  
(Doanh nghiệp hay liệt kê “4 loại cơ bản”: môn bài/GTGT/TNDN/TNCN).

👉 Vậy nên: **đề tài của bạn không phải “giải hết mọi loại thuế”**. Core hợp lý nhất là **VAT + quy định hoá đơn điện tử + đối soát thanh toán** (có thể thêm TTĐB làm case nâng cao).

## 3) Bạn đang đối mặt với “problem gì từ Doanh nghiệp”?

Nếu nhìn từ góc doanh nghiệp/merchant (đặc biệt bán lẻ/chuỗi cửa hàng + dùng POS + thanh toán QR/VNPAY), pain points thường nằm ở đây:

### 1. Tuân thủ HĐĐT ngày càng “real-time” và chặt hơn

- Nhất là hướng “HĐĐT khởi tạo từ máy tính tiền” và truyền dữ liệu cho cơ quan thuế.

### 2. Luồng thanh toán ≠ luồng xuất hoá đơn

- Thanh toán thành công/hủy/hoàn tiền/timeout... nếu không có lớp điều phối, rất dễ: **xuất nhầm, xuất trùng**, hoặc **không xuất** → rủi ro phạt/đối soát.

### 3. Dữ liệu POS bẩn và không chuẩn hoá

- Tên hàng linh tinh, thiếu SKU/nhóm thuế, discount/combo/phí ship/phụ thu... khiến việc map thuế suất và lập hoá đơn sai.

#### 4. **Quản lý sai sót hoá đơn rất tốn công**

- Điều chỉnh/hủy/thay thế... cần quy trình + log + đối soát (kế toán ghét nhất đoạn này).

#### 5. **Tích hợp nhiều bên** (POS, VNPAY/PSP, nhà cung cấp HĐĐT, hệ thống kế toán nội bộ)

- Mỗi bên 1 API/format/logic retry → doanh nghiệp muốn “một chỗ trung gian” để đỡ phải nối chằng chịt.

#### 6. **Audit & đối soát**

- Khi bị kiểm tra: cần chứng minh “đơn nào thanh toán lúc nào → hoá đơn số gì → gửi/nhận trạng thái ra sao”.

# Chuẩn hoá dữ liệu giao dịch

### A. Từ POS (bên bán hàng) bạn nhận kiểu:

- **Order:** mã đơn POS, thời điểm tạo, cửa hàng/chi nhánh, cashier
- **Line items:** danh sách món/hàng (tên, qty, unit price, discount, combo...)
- **Tổng tiền:** subtotal/discount/service charge/VAT (nếu POS có tính trước)
- **Thông tin người mua** (nếu có): tên, MST, email/SDT, địa chỉ
- **Trạng thái đơn:** created/paid/canceled/refunded... (tùy POS)

Vấn đề: mỗi POS đặt tên field khác nhau, cách tính discount/fee khác nhau.

### B. Từ PSP/VNPAY (bên thanh toán) bạn nhận kiểu:

- **Payment transaction:** paymentId/txnRef, amount, method (QR/card...), status (success/fail/pending)
- **Timestamp:** paidAt, callbackAt
- **Reference:** orderRef (tham chiếu về POS) hoặc invoiceRef (tham chiếu về bạn)
- **Refund:** refundId, refundAmount, refundStatus

Vấn đề: status/luồng callback có thể đến **trễ**, **trùng**, hoặc **out-of-order**.

### C. Chuẩn hoá = tạo “Canonical Schema” (schema chuẩn nội bộ)

Bạn thiết kế 1 model chuẩn, ví dụ tách thành 3 khối:

1. **Order** (sự kiện bán hàng từ POS)
2. **Payment** (sự kiện thanh toán từ PSP)
3. **Invoice Request/Invoice** (yêu cầu + kết quả phát hành HĐĐT)

Ví dụ schema chuẩn (minh hoạ):

```
{
  "merchantId": "M123",
  "storeId": "S01",
  "order": {
    "orderId": "POS-20260115-0008",
    "createdAt": "2026-01-15T10:05:12+07:00",
```

```
"currency": "VND",
"items": [
  {
    "sku": "CF-SUA-DA",
    "name": "Cà phê sữa đá",
    "qty": 2,
    "unitPrice": 25000,
    "discount": 0,
    "taxCategory": "VAT_8"
  }
],
"charges": {
  "serviceCharge": 0,
  "shippingFee": 0,
  "orderDiscount": 0
},
"amounts": {
  "subtotal": 50000,
  "tax": 4000,
  "total": 54000
},
"buyer": {
  "name": null,
  "taxCode": null,
  "email": null
}
},
"payment": {
  "psp": "VNPAY",
  "paymentId": "VNP-88990011",
  "orderRef": "POS-20260115-0008",
  "amount": 54000,
  "status": "SUCCEEDED",
  "paidAt": "2026-01-15T10:06:01+07:00"
}
}
```

nền tảng trung gian hoá đơn điện tử

Hãy hiểu nó như 3 module chính (có thể là 3 service hoặc 3 layer):

## **(1) E-invoice middleware (Trung gian hoá đơn điện tử)**

Vai trò: **điều phối phát hành HĐĐT** qua nhà cung cấp HĐĐT (hoặc tích hợp sâu hơn với cơ quan thuế tùy mô hình).

- Nhận “Invoice Request” (đủ dữ liệu chuẩn hoá).
- Gọi API provider xuất hoá đơn.
- Nhận kết quả: số hoá đơn, ký hiệu, mã cơ quan thuế (nếu có), PDF/XML.
- Lưu log, trạng thái, retry an toàn.

=> Nói nôm na: “đầu não xuất hoá đơn”.

## **(2) Tax rule engine (Bộ máy luật thuế)**

Vai trò: **từ line items → ra số thuế đúng**.

- Map SKU/category → thuế suất VAT / nhóm thuế.
- Áp quy tắc: VAT theo dòng, làm tròn, phân bổ giảm giá vào dòng, phí dịch vụ tính VAT hay không...
- Kết quả: breakdown thuế + tổng tiền chuẩn để đưa vào hoá đơn.

=> Nói nôm na: “máy tính thuế + luật”.

## **(3) Reconciliation (Đối soát)**

Vai trò: đảm bảo **3 thứ khớp nhau**:

- POS: đơn nào đã bán?
- PSP/VNPAY: giao dịch nào đã thu tiền/hoàn tiền?
- Invoice: hoá đơn nào đã phát hành/điều chỉnh/hủy?

Reconciliation thường trả lời các câu:

- “Có thanh toán thành công mà chưa xuất hoá đơn không?”



- “Có hoá đơn đã xuất nhưng thanh toán fail/hoàn tiền không?”
- “Có xuất trùng 2 hoá đơn cho 1 payment không?”

=> Nói nôm na: “kế toán/kiểm toán rất thích module này”.

---

## 4) Một ví dụ luồng “trung gian”

1. POS tạo order + line items
2. POS/PSP tạo QR thanh toán (chứa amount + orderRef)
3. PSP callback “payment succeeded” (có thể gửi trùng)
4. Nền tảng bạn:
  - ghép **orderRef** ↔ order detail,
  - chạy **tax rule engine** ra VAT breakdown,
  - tạo **invoice request**,
  - gọi **e-invoice middleware** để phát hành,
  - lưu kết quả,
  - reconciliation check “order-payment-invoice” khớp.

đứng ở hạ tầng nào trong chuỗi ngoài  
đời

## Lớp A — Merchant/POS layer (hạ tầng bán hàng)

- POS tạo **đơn hàng** + line items.
- POS/merchant cần “xuất hoá đơn điện tử” cho các đơn đã bán.

Bạn **không** phải ở lớp này nếu bạn làm “nền tảng trung gian”. Bạn chỉ *nhận dữ liệu* từ lớp này.

## Lớp B — Payment/PSP layer (hạ tầng thanh toán: VNPAY, MoMo, ngân hàng...)

- Tạo QR / xử lý thanh toán.
- Gửi **callback/webhook** “thành công / thất bại / hoàn tiền”.

Bạn cũng **không** phải là PSP (trừ khi đề bắt bạn làm hẳn payment provider). Bạn thường *nhận event thanh toán* từ lớp này.

## Lớp C — Tax & E-invoice middleware layer (hạ tầng “trung gian thuế/hoá đơn”) ← **đúng chỗ của đề bạn**

Đây là nơi bạn “đứng”:

- **Chuẩn hoá dữ liệu** (POS + Payment → 1 schema chuẩn)
- **Tính/áp quy tắc thuế** (VAT rules...)
- **Điều phối phát hành HĐĐT** (gọi API nhà cung cấp hoá đơn điện tử / tổ chức truyền nhận)
- **Đối soát** (order–payment–invoice có khớp không)

Ngoài đời, lớp này thường được xây bởi:

- Doanh nghiệp lớn tự build (chuỗi bán lẻ, platform lớn), hoặc
- Nhà cung cấp giải pháp hoá đơn / tích hợp, hoặc
- Một số PSP cũng cung cấp luôn e-invoice để thành “bộ đôi thanh toán + hoá đơn”.

Ví dụ VNPAY có sản phẩm **VNPAY-Invoice**.

Định nghĩa

POS: Point of Sale

PSP: Payment Service Provider

SKU: Stock-Keeping Unit (đơn vị lưu kho)

tôi tự hỏi

### 3) Nếu POS đã “in bill” rồi, sao còn cần “nền tảng trung gian thuế”?

Vì “nền tảng trung gian thuế/hoá đơn” không phải để *in bill*, mà để giải quyết các “đau” mà POS/PSP thường không xử lý tốt (hoặc xử lý không thống nhất) khi doanh nghiệp lớn vận hành thật.

Các lý do phổ biến:

#### A) POS ≠ Hệ thống thuế/hoá đơn

POS tạo đơn nhanh, nhưng **HĐĐT cần**:

- chuẩn hoá thông tin người bán/người mua,
- chuẩn hoá line items,
- áp thuế suất đúng,
- ký số, lưu trữ, tra cứu,
- xử lý điều chỉnh/hủy/thay thế.

POS thường **không muốn gánh hết** phần pháp lý phức tạp này.

#### B) Nhiều chi nhánh, nhiều POS, nhiều kiểu dữ liệu

Chuỗi cửa hàng có thể:

- chi nhánh A dùng POS X, chi nhánh B dùng POS Y,
- cách đặt tên món/discount khác nhau,
- cách tính phí khác nhau.

“Nền tảng trung gian” làm nhiệm vụ **chuẩn hoá** để hoá đơn ra **đồng nhất**.

#### C) Thanh toán và xuất hoá đơn hay lệch nhau

Thực tế hay gặp:

- webhook thanh toán gửi trùng,
- thanh toán pending rồi mới success,
- hoàn tiền,

- khách đổi món, huỷ đơn,
- mất mạng lúc trả kết quả.

Nếu chỉ dựa vào POS “bấm in”, rất dễ:

- **xuất trùng hoá đơn**
- **thu tiền rồi nhưng thiếu hoá đơn**
- **hoá đơn có rồi nhưng giao dịch bị hoàn**

Middleware đứng giữa để làm **idempotency + state machine + đối soát**.

#### **D) Doanh nghiệp muốn “đổi nhà cung cấp HĐĐT” mà không sửa POS**

Hôm nay dùng provider A, mai muốn đổi provider B vì giá/điều khoản.

Nếu POS tích hợp thẳng từng provider → đổi là **đau**.

Middleware giúp:

- POS chỉ nói chuyện với 1 API “chuẩn”
- còn adapter phía sau muốn đổi provider nào thì đổi.

#### **E) Đối soát & audit**

Kế toán/kiểm toán cần câu trả lời:

- đơn nào thu tiền lúc nào,
- hoá đơn số nào tương ứng,
- lỗi gì, retry mấy lần,
- ai thao tác.

POS thường không thiết kế sâu cho audit kiểu này.

#### **4) Vậy trong đề của bạn, POS có thể “chỉ làm gì” là đủ?**

Trong scope hợp lý cho sinh viên, bạn có thể giả định POS làm tối thiểu:

- tạo order + line items + total



- phát sinh orderRef
- gọi tạo QR/thanh toán
- nhận trạng thái “invoice issued” để hiển thị/in link hoá đơn

Còn middleware của bạn làm:

- ghép order + payment,
- tính VAT,
- gọi provider HĐĐT (hoặc mock),
- lưu trạng thái, chống trùng, đối soát.

Provider nào tốt nhất, api có sẵn?

## Lựa chọn dễ làm nhất cho nhóm (có tài liệu API công khai rõ)

### 1) MISA melInvoice (Open API)

- Có **portal tài liệu tích hợp** và các bài “Tổng quan Open API” (các API phát hành, ký số, xem/tải, điều chỉnh/thay thế...).
- Có ghi rõ thường cần **đăng ký để được cấp app\_id / thông tin kết nối**.  
 ➡ Với đề của bạn (mentor nói “móc nối doanh nghiệp đã liên kết sẵn như MISA”) thì **MISA là hợp nhất** để làm “provider thật” hoặc ít nhất làm “spec chuẩn”.

## Các provider khác (có thông tin hỗ trợ nhưng mức “API public” không đồng đều)

### 2) Viettel S-Invoice

- Có trang hỗ trợ/tài liệu và có nhắc “Hướng dẫn tích hợp HDDT 2.0” (thường là tài liệu tải về/quá cổng SME).  
 ➡ Khả năng cao cần quyền truy cập/tài khoản để lấy đủ chi tiết API, nhưng vẫn dùng được để tham khảo luồng nghiệp vụ.

### 3) VNPT Invoice

- Có trang hỗ trợ/hướng dẫn sử dụng công khai.  
 ➡ Tài liệu API tích hợp chi tiết có thể không public hoàn toàn; nhiều trường hợp

phải “được cấp” thông tin.

#### **4) FPT.eInvoice**

- Có trang sản phẩm công khai.
  - ➡ API tích hợp chi tiết thường là tài liệu dành cho đối tác/khách hàng (tùy chính sách).

What to DEMO?

Bạn chuẩn bị 3 màn hình + 1 “POS mock” đơn giản:

### A) “POS Mock” (maybe payOS)

- Chọn món (2–3 món mẫu), số lượng, giảm giá.
- Bấm **Create Order** → hệ thống tạo `orderId`.
- Bấm **Pay** → giả lập tạo “payment request”.

Cái này chỉ để tạo dữ liệu đầu vào. Không phải làm e-commerce.

### B) “Payment Event Simulator” (1 trang hoặc 3 nút)

- Nút **Send webhook SUCCESS**
- Nút **Send webhook DUPLICATE SUCCESS** (gửi trùng)
- Nút **Send webhook REFUND**
- (tuỳ) **PENDING** → **SUCCESS**

Đây là điểm “wow” vì bạn chứng minh hệ thống xử lý event thực tế, không phải UI đẹp.

### C) Dashboard của “Tax Middleware”

Trang này mới là sản phẩm chính:

- **Orders**: danh sách đơn + trạng thái (CREATED/PAID/INVOICE\_ISSUED/FAILED...)
- **Payments**: danh sách giao dịch từ VNPAY mock
- **Invoices**: danh sách hoá đơn đã phát hành (mock) + link xem “PDF/XML” (file giả lập cũng được)
- **Reconciliation**: bảng mismatch (paid nhưng chưa invoice, invoice mà payment fail, duplicate events...)

### D) “Invoice Provider Adapter” (mock MISA)

Bạn làm 1 service mock trả về:

- `invoiceNumber`, `symbol`, `issuedAt`
- file PDF/XML giả lập
- đôi lúc trả lỗi (để demo retry)

Đây đúng với câu mentor: “không connect Tổng cục thuế, connect doanh nghiệp đã liên kết sẵn”.

## Cách chứng minh sản phẩm khi UI ít

Bạn chứng minh bằng **đúng thứ hội đồng thích ở hệ thống B2B**:

1. **Timeline event** của 1 order (audit trail)
  - Order created → Payment success webhook → Invoice issued → Stored PDF/XML
  - Nếu webhook trùng → hệ thống bỏ qua (idempotency)
2. **Reconciliation dashboard**
  - “Paid but not invoiced” xuất hiện khi provider lỗi
  - Bấm **Retry issue invoice** → chuyển sang issued
3. **Edge cases demo**
  - PENDING → SUCCESS
  - SUCCESS gửi trùng 3 lần vẫn chỉ có 1 invoice
  - REFUND → tạo trạng thái “needs adjustment/cancel” (mô phỏng)