Development Blueprint for Phase 1.

🏗️ Phase 1: Strategic Planning & Architecture

1. UX Strategy: The "Cultural Funnel"

Since we are selling culture (Magazines, Events), the standard "Grid of Products" will fail. We need a narrative-driven flow that sells the *experience* first, then the product.

A. Homepage Layout (Sales Funnel Structure)

Instead of a generic store, we treat the homepage as a Landing Page for the brand's current "Season" or "Issue."

1. **Hero Section (The Hook):**
   ○ *Content:* Full-screen cinematic video or high-res imagery of the latest Magazine Issue or Event atmosphere.
   ○ *Action:* "Read Latest Issue" or "Book Next Experience."
   ○ *Vibe:* Overlay text using a large, elegant Arabic serif font (e.g., *SST Arabic* or *IBM Plex Sans Arabic*) paired with a minimal English sans-serif.
2. **The "Featured Issue" Spotlight (Physical):**
   ○ *Layout:* Split screen (50/50). Left side: 3D-style book mockups of the magazine. Right side: The "Why" (Editor's Note snippet) + "Get Ship to Home" button.
3. **The "Cultural Calendar" (Events):**
   ○ *Layout:* A horizontal scrolling timeline (GSAP driven). Instead of product cards, they look like "Tickets" or "Passports."
   ○ *Data:* Dynamic countdown timers for upcoming events.
4. **The "Digital Archive" (PDFs):**
   ○ *Layout:* Masonry grid, minimalist. Focus on "Instant Access." Visuals should be abstract covers, not just generic PDF icons.
5. **Social Proof (The Tribe):**
   ○ *Concept:* "Who is reading?" – User-generated content (UGC) carousel showing people holding the magazine or attending events, framed in a specific Sadu-inspired border.

B. The "Saudi Touch" (Design System)

- **Motifs:** Use *Al-Sadu* patterns, but **deconstructed**. Instead of a full heavy strip, use the geometric triangle shapes as subtle SVG dividers between sections, or as a watermark with 5% opacity behind the product description.
- **Colors:**
  - *Primary:* Deep Najdi Clay (Terracotta) or Tuwaiq Mountain Blue (Dark Navy).
  - *Secondary:* Sand/Beige (for backgrounds to reduce eye strain).
  - *Accent:* Gold Foil (used sparingly on CTA buttons only).
- **Micro-Interactions:** When hovering over a "Book Seat" button, the icon changes from a standard arrow to a stylized "Dallah" or specific cultural icon.

---

2. Technical Logic (The Twilight Challenge)

Salla's Twilight engine is powerful but strict. To handle Physical, Service (Events), and Digital products seamlessly without cluttering one file, we will use a **Component-Based Architecture**.

A. Directory Structure Strategy

We will avoid writing 500 lines of `if/else` logic inside `single.twig`. Instead, we will break the views into smaller, manageable partials.

Proposed File Structure:

```
Plaintext
src/
├── views/
│   ├── pages/
│   │   └── product/
│   │       └── single.twig      <-- The Controller / Skeleton
│   ├── components/
│   │   └── product/
│   │       ├── layout-physical.twig  <-- For Magazines
│   │       ├── layout-service.twig   <-- For Events
│   │       └── layout-digital.twig   <-- For PDFs
```

B. The Logic Flow (Variable Detection)

In Salla, the `product` object exposes a `type` property. We will use this to dynamically route the user to the correct layout.

Logic Plan for `single.twig`:

1. **Check `product.type`:**
   - If `product` (standard) → Load `layout-physical.twig`.
   - If `service` → Load `layout-service.twig`.
   - If `digital` or `codes` → Load `layout-digital.twig`.
2. **Fallback:**
   - If the type is unknown, default to `layout-physical`.

C. Custom Fields vs. Native Features

- **Events (Maps & Dates):** Since Salla treats services generically, we will utilize **Custom Fields** (defined in `salla-theme.json` or the dashboard) to pass the "Event Location Link" and "Event Date."
- **Magazines (Previews):** We will use the "Product Gallery" but style the first image as a 3D cover and the subsequent images as a "Look inside" slider.

Phase 2: Implementation Roadmap. We are moving from strategy to the actual Twig architecture that supports your diverse product catalog.

1. Environment Setup

Before editing files, ensure your local development environment is active. Open your terminal in the theme directory:

Bash
```
# Install dependencies (if you haven't recently)
npm install

# Start the live preview server
salla theme:serve
```

---

2. File Structure Creation

To keep `single.twig` clean, we will create dedicated partials. Please create the following file structure in your theme:

Plaintext
```
src/
└── views/
    └── components/
        └── product/
            ├── layout-physical.twig   (For Magazines)
            ├── layout-service.twig    (For Events)
```

└── layout-digital.twig   (For PDFs)

---

3. Coding Task: The "Smart Router"

We will turn `single.twig` into a logic controller that detects the product type and serves the correct experience.

File: `src/views/pages/product/single.twig`

Twig

```
{% extends "layouts.master" %}

{% block content %}
<div class="product-page-container container mx-auto px-4 py-10 relative overflow-hidden">

  {# 🎨 The Saudi Touch: Deconstructed Sadu Pattern Background #}
  <div class="absolute top-0 right-0 opacity-5 pointer-events-none">
    <svg class="w-64 h-64 text-najdi-clay" viewBox="0 0 100 100"></svg>
  </div>

  {# 🧠 The Logic Router #}
  {% if product.type == 'service' or product.type == 'booking' %}
    {# Case B: Events #}
    {% include 'views.components.product.layout-service' %}

  {% elseif product.type == 'digital' or product.type == 'codes' %}
    {# Case C: Digital PDFs #}
    {% include 'views.components.product.layout-digital' %}

  {% else %}
    {# Case A: Default to Physical (Magazines) #}
    {% include 'views.components.product.layout-physical' %}
  {% endif %}

  {# Common Section: Upsell / Related Products #}
  <div class="mt-20 border-t border-sand-200 pt-10">
    <h3 class="font-serif text-2xl text-center mb-8">Discover More Culture</h3>
    <salla-products-list source="related" limit="4"></salla-products-list>
  </div>

</div>
{% endblock %}
```

4. Coding Task: The "Smart Views"

Type A: The Magazine Layout (Physical)

Goal: Focus on the tangible object, the "Issue Number," and shipping.

File: `src/views/components/product/layout-physical.twig`

Twig

```twig
<div class="grid grid-cols-1 md:grid-cols-2 gap-12 items-center min-h-[60vh]">
   {# Left: 3D Book Mockup View #}
   <div class="product-gallery relative group">
      <div class="aspect-[3/4] bg-sand-100 rounded-lg shadow-2xl transform rotate-1 transition-transform group-hover:rotate-0">
         <img src="{{ product.image.url }}" alt="{{ product.name }}" class="w-full h-full object-cover rounded-lg">
      </div>
      {# Badge #}
      <span class="absolute -top-4 -left-4 bg-najdi-clay text-white px-4 py-1 font-serif text-sm shadow-md">
         New Issue
      </span>
   </div>

   {# Right: Story & Shipping #}
   <div class="product-details space-y-6" data-aos="fade-up">
      <h1 class="text-4xl font-serif text-gray-900 leading-tight">{{ product.name }}</h1>
      <div class="prose text-gray-600 border-l-4 border-sand-300 pl-4">
         {{ product.description|raw }}
      </div>

      {# The CTA #}
      <div class="action-area bg-white p--6 rounded-xl shadow-sm border border-gray-100">
         <div class="flex items-center justify-between mb-4">
            <span class="text-2xl font-bold text-najdi-clay">{{ product.price|money }}</span>
            <span class="text-xs text-gray-500 flex items-center gap-1">
               <i class="sicon-shipping"></i> Ships within 24h
            </span>
         </div>
         <salla-add-product-button product-id="{{ product.id }}" class="w-full btn btn-primary btn-lg">
            Add to Library (Ship to Home)
```

```
        </salla-add-product-button>
      </div>
   </div>
</div>
```

Type B: The Event Layout (Service)

Goal: Hide shipping. Emphasize "Date," "Location," and "Booking." *Note: We assume you are using Product Custom Fields for Date/Location data.*

File: `src/views/components/product/layout-service.twig`

Twig
```
<div class="event-layout bg-gray-900 text-white rounded-2xl overflow-hidden shadow-2xl">

   {# Hero Banner #}
   <div class="relative h-80 md:h-96">
      <img src="{{ product.image.url }}" class="w-full h-full object-cover opacity-60">
      <div class="absolute inset-0 bg-gradient-to-t from-gray-900 via-transparent
to-transparent"></div>

      <div class="absolute bottom-0 left-0 p-8 w-full">
         <div class="flex flex-col md:flex-row justify-between items-end">
            <div>
               <span class="bg-gold text-gray-900 px-3 py-1 text-xs font-bold uppercase
tracking-widest mb-2 inline-block">Cultural Event</span>
               <h1 class="text-3xl md:text-5xl font-serif mb-2">{{ product.name }}</h1>
            </div>
            {# Timer Placeholder #}
            <div class="countdown-timer text-right">
               <p class="text-sm text-gray-400 mb-1">Starts In:</p>
               <div class="text-2xl font-mono text-gold">04 : 12 : 30</div>
            </div>
         </div>
      </div>
   </div>

   {# Event Details Grid #}
   <div class="grid grid-cols-1 md:grid-cols-3 divide-y md:divide-y-0 md:divide-x divide-gray-800
border-t border-gray-800">
      <div class="p-6 text-center">
         <i class="sicon-calendar text-gold text-2xl mb-2 block"></i>
         <span class="text-gray-300 text-sm">Date</span>
         <p class="font-bold text-lg">Oct 15, 2025</p> {# Dynamic data goes here #}
```

```
        </div>
        <div class="p-6 text-center">
            <i class="sicon-location-pin text-gold text-2xl mb-2 block"></i>
            <span class="text-gray-300 text-sm">Location</span>
            <p class="font-bold text-lg">Riyadh Front</p> {# Dynamic data goes here #}
        </div>
        <div class="p-6 flex items-center justify-center bg-gray-800">
            <salla-add-product-button product-id="{{ product.id }}" class="w-full btn btn-gold
text-black font-bold">
                Book Your Seat
            </salla-add-product-button>
        </div>
    </div>
</div>
```

Type C: The Digital Layout (PDF)

Goal: Minimal friction. "Instant Download" focus.

File: `src/views/components/product/layout-digital.twig`

Twig
```
<div class="max-w-4xl mx-auto pt-10">
    <div class="bg-sand-50 rounded-xl p-8 border border-sand-200 flex flex-col md:flex-row
gap-8 items-center">

        {# Abstract Thumbnail #}
        <div class="w-32 h-32 flex-shrink-0 bg-white rounded-full flex items-center justify-center
shadow-inner text-najdi-clay">
            <i class="sicon-file-pdf text-5xl"></i>
        </div>

        {# Content #}
        <div class="flex-grow text-center md:text-left">
            <h1 class="text-2xl font-bold text-gray-800 mb-2">{{ product.name }}</h1>
            <p class="text-gray-500 text-sm mb-4">{{ product.subtitle|default('Digital Edition • PDF
High Resolution') }}</p>

            <ul class="flex justify-center md:justify-start gap-4 text-xs text-gray-400 mb-4">
                <li class="flex items-center gap-1"><i class="sicon-cloud-download"></i> Instant
Delivery</li>
                <li class="flex items-center gap-1"><i class="sicon-screen"></i> Mobile/Tablet
Ready</li>
            </ul>
```

```
        </div>

        {# Action #}
        <div class="w-full md:w-auto">
            <salla-add-product-button product-id="{{ product.id }}" class="w-full md:w-48 btn
btn-primary btn-outline">
                Download Now ({{ product.price|money }})
            </salla-add-product-button>
        </div>
    </div>
</div>
```

Here is the **Phase 2: Implementation Roadmap**. We are moving from strategy to the actual
Twig architecture that supports your diverse product catalog.

1. Environment Setup

Before editing files, ensure your local development environment is active. Open your terminal in
the theme directory:

Bash
```
# Install dependencies (if you haven't recently)
npm install

# Start the live preview server
salla theme:serve
```

---

2. File Structure Creation

To keep `single.twig` clean, we will create dedicated partials. Please create the following file
structure in your theme:

Plaintext
```
src/
└── views/
    └── components/
        └── product/
            ├── layout-physical.twig   (For Magazines)
            ├── layout-service.twig    (For Events)
            └── layout-digital.twig    (For PDFs)
```

---

3. Coding Task: The "Smart Router"

We will turn `single.twig` into a logic controller that detects the product type and serves the correct experience.

File: `src/views/pages/product/single.twig`

Twig

```twig
{% extends "layouts.master" %}

{% block content %}
<div class="product-page-container container mx-auto px-4 py-10 relative overflow-hidden">

    {# 🎨 The Saudi Touch: Deconstructed Sadu Pattern Background #}
    <div class="absolute top-0 right-0 opacity-5 pointer-events-none">
        <svg class="w-64 h-64 text-najdi-clay" viewBox="0 0 100 100"></svg>
    </div>

    {# 🧠 The Logic Router #}
    {% if product.type == 'service' or product.type == 'booking' %}
        {# Case B: Events #}
        {% include 'views.components.product.layout-service' %}

    {% elseif product.type == 'digital' or product.type == 'codes' %}
        {# Case C: Digital PDFs #}
        {% include 'views.components.product.layout-digital' %}

    {% else %}
        {# Case A: Default to Physical (Magazines) #}
        {% include 'views.components.product.layout-physical' %}
    {% endif %}

    {# Common Section: Upsell / Related Products #}
    <div class="mt-20 border-t border-sand-200 pt-10">
        <h3 class="font-serif text-2xl text-center mb-8">Discover More Culture</h3>
        <salla-products-list source="related" limit="4"></salla-products-list>
    </div>

</div>
{% endblock %}
```

---

4. Coding Task: The "Smart Views"

Type A: The Magazine Layout (Physical)

Goal: Focus on the tangible object, the "Issue Number," and shipping.

File: `src/views/components/product/layout-physical.twig`

Twig

```twig
<div class="grid grid-cols-1 md:grid-cols-2 gap-12 items-center min-h-[60vh]">
   {# Left: 3D Book Mockup View #}
   <div class="product-gallery relative group">
      <div class="aspect-[3/4] bg-sand-100 rounded-lg shadow-2xl transform rotate-1 transition-transform group-hover:rotate-0">
         <img src="{{ product.image.url }}" alt="{{ product.name }}" class="w-full h-full object-cover rounded-lg">
      </div>
      {# Badge #}
      <span class="absolute -top-4 -left-4 bg-najdi-clay text-white px-4 py-1 font-serif text-sm shadow-md">
         New Issue
      </span>
   </div>

   {# Right: Story & Shipping #}
   <div class="product-details space-y-6" data-aos="fade-up">
      <h1 class="text-4xl font-serif text-gray-900 leading-tight">{{ product.name }}</h1>
      <div class="prose text-gray-600 border-l-4 border-sand-300 pl-4">
         {{ product.description|raw }}
      </div>

      {# The CTA #}
      <div class="action-area bg-white p-6 rounded-xl shadow-sm border border-gray-100">
         <div class="flex items-center justify-between mb-4">
            <span class="text-2xl font-bold text-najdi-clay">{{ product.price|money }}</span>
            <span class="text-xs text-gray-500 flex items-center gap-1">
               <i class="sicon-shipping"></i> Ships within 24h
            </span>
         </div>
         <salla-add-product-button product-id="{{ product.id }}" class="w-full btn btn-primary btn-lg">
            Add to Library (Ship to Home)
         </salla-add-product-button>
      </div>
   </div>
</div>
```

Type B: The Event Layout (Service)

Goal: Hide shipping. Emphasize "Date," "Location," and "Booking." *Note: We assume you are using Product Custom Fields for Date/Location data.*

File: `src/views/components/product/layout-service.twig`

Twig
```twig
<div class="event-layout bg-gray-900 text-white rounded-2xl overflow-hidden shadow-2xl">

  {# Hero Banner #}
  <div class="relative h-80 md:h-96">
    <img src="{{ product.image.url }}" class="w-full h-full object-cover opacity-60">
    <div class="absolute inset-0 bg-gradient-to-t from-gray-900 via-transparent to-transparent"></div>

    <div class="absolute bottom-0 left-0 p-8 w-full">
      <div class="flex flex-col md:flex-row justify-between items-end">
        <div>
          <span class="bg-gold text-gray-900 px-3 py-1 text-xs font-bold uppercase tracking-widest mb-2 inline-block">Cultural Event</span>
          <h1 class="text-3xl md:text-5xl font-serif mb-2">{{ product.name }}</h1>
        </div>
        {# Timer Placeholder #}
        <div class="countdown-timer text-right">
          <p class="text-sm text-gray-400 mb-1">Starts In:</p>
          <div class="text-2xl font-mono text-gold">04 : 12 : 30</div>
        </div>
      </div>
    </div>
  </div>

  {# Event Details Grid #}
  <div class="grid grid-cols-1 md:grid-cols-3 divide-y md:divide-y-0 md:divide-x divide-gray-800 border-t border-gray-800">
    <div class="p-6 text-center">
      <i class="sicon-calendar text-gold text-2xl mb-2 block"></i>
      <span class="text-gray-300 text-sm">Date</span>
      <p class="font-bold text-lg">Oct 15, 2025</p> {# Dynamic data goes here #}
    </div>
    <div class="p-6 text-center">
      <i class="sicon-location-pin text-gold text-2xl mb-2 block"></i>
      <span class="text-gray-300 text-sm">Location</span>
      <p class="font-bold text-lg">Riyadh Front</p> {# Dynamic data goes here #}
    </div>
```

```twig
        <div class="p-6 flex items-center justify-center bg-gray-800">
            <salla-add-product-button product-id="{{ product.id }}" class="w-full btn btn-gold text-black font-bold">
                Book Your Seat
            </salla-add-product-button>
        </div>
    </div>
</div>
```

Type C: The Digital Layout (PDF)

Goal: Minimal friction. "Instant Download" focus.

File: `src/views/components/product/layout-digital.twig`

Twig
```twig
<div class="max-w-4xl mx-auto pt-10">
    <div class="bg-sand-50 rounded-xl p-8 border border-sand-200 flex flex-col md:flex-row gap-8 items-center">

        {# Abstract Thumbnail #}
        <div class="w-32 h-32 flex-shrink-0 bg-white rounded-full flex items-center justify-center shadow-inner text-najdi-clay">
            <i class="sicon-file-pdf text-5xl"></i>
        </div>

        {# Content #}
        <div class="flex-grow text-center md:text-left">
            <h1 class="text-2xl font-bold text-gray-800 mb-2">{{ product.name }}</h1>
            <p class="text-gray-500 text-sm mb-4">{{ product.subtitle|default('Digital Edition • PDF High Resolution') }}</p>

            <ul class="flex justify-center md:justify-start gap-4 text-xs text-gray-400 mb-4">
                <li class="flex items-center gap-1"><i class="sicon-cloud-download"></i> Instant Delivery</li>
                <li class="flex items-center gap-1"><i class="sicon-screen"></i> Mobile/Tablet Ready</li>
            </ul>
        </div>

        {# Action #}
        <div class="w-full md:w-auto">
            <salla-add-product-button product-id="{{ product.id }}" class="w-full md:w-48 btn btn-primary btn-outline">
```

```
            Download Now ({{ product.price|money }})
        </salla-add-product-button>
      </div>
   </div>
</div>
```

Immediate Next Step

Once you have pasted these files, the views will work logically, but they will lack the specific styling (CSS) for the classes like `text-najdi-clay`, `bg-sand`, and the 3D transforms.

generate the `tailwind.config.js` or SCSS file to define these "Saudi Palette" colors and fonts