# 3D Print Marketplace – Django File & App Structure

This document shows the recommended Django project and app-level folder structure for a multi-vendor 3D print marketplace. The structure emphasizes separation of concerns, scalability, and long-term maintainability.

## 1. Project Root

```
3dprint_marketplace/
■■■ manage.py
■■■ requirements.txt
■■■ .env
■■■ README.md
■■■ docker-compose.yml        (optional)
■■■ Dockerfile                (optional)
■
■■■ config/                   # Django project configuration
■   ■■■ __init__.py
■   ■■■ asgi.py
■   ■■■ celery.py             # background tasks
■   ■■■ settings/
■   ■   ■■■ __init__.py
■   ■   ■■■ base.py
■   ■   ■■■ dev.py
■   ■   ■■■ prod.py
■   ■■■ urls.py
■   ■■■ wsgi.py
```

## 2. Core Django Apps

```
■■■ accounts/                 # Authentication & roles
■   ■■■ models.py             # User, SellerProfile, ConsumerProfile
■   ■■■ views.py
■   ■■■ forms.py
■   ■■■ urls.py
■   ■■■ permissions.py
■   ■■■ templates/accounts/
■
■■■ storefront/               # Public-facing pages
■   ■■■ views.py
■   ■■■ urls.py
■   ■■■ services.py           # ranking, featured logic
■   ■■■ templates/storefront/
■
■■■ sellers/                  # Seller dashboard
■   ■■■ models.py             # Seller, StripeAccount
■   ■■■ views/
■   ■   ■■■ dashboard.py
■   ■   ■■■ products.py
■   ■   ■■■ orders.py
■   ■■■ urls.py
■   ■■■ templates/sellers/
```

## 3. Products, Orders & Media

```
■■■ products/
■   ■■■ models.py             # Product, Category, Media, License
■   ■■■ validators.py         # STL checks, size limits
■   ■■■ views.py
■   ■■■ urls.py
■   ■■■ templates/products/
■
■■■ orders/
■   ■■■ models.py             # Order, LineItem, Download
■   ■■■ fulfillment.py        # physical item handling
■   ■■■ views.py
■   ■■■ urls.py
■   ■■■ templates/orders/
```

## 4. Payments & Financial Flow

```
■■■ payments/
■   ■■■ stripe.py             # Stripe Checkout & Connect
■   ■■■ webhooks.py
■   ■■■ models.py             # Payouts, Fees
■   ■■■ views.py
■   ■■■ urls.py
```

## 5. Moderation, Reviews & Legal

```
■■■ moderation/
■   ■■■ models.py             # Reports, DMCA, Flags
■   ■■■ views.py
■   ■■■ admin.py
■
■■■ reviews/
■   ■■■ models.py             # Ratings, Feedback
■   ■■■ views.py
■
■■■ legal/
■   ■■■ templates/legal/
■   ■   ■■■ terms.html
■   ■   ■■■ privacy.html
■   ■   ■■■ refund_policy.html
```

## 6. Static Files & Media

```
■■■ static/
■   ■■■ css/
■   ■■■ js/
■   ■■■ images/
■
```

```
■■■ media/
■   ■■■ product_files/       # private STL storage
■   ■■■ product_images/
■   ■■■ seller_assets/
```

This structure is intentionally modular. You can build and ship the storefront first, then add seller tooling, payments, and moderation incrementally without refactoring the core.