

```
from dataclasses import dataclass
from typing import Dict, List, Optional, Any

@dataclass
class DocumentClass:
    id: int
    name: str
    description: str
    fields: List[str]

# Define the document categories
DOCUMENT_CATEGORIES = [
    DocumentClass(
        id=1,
        name="Final Inspection Card",
        description="Certificate showing final building inspection approval",
        fields=["property_address", "fic_image", "non_fic_proof"]
    ),
    DocumentClass(
        id=2,
        name="Interconnection Agreement",
        description="Agreement between homeowner and utility company for grid connection",
        fields=["home_address", "homeowner_signature"]
    ),
    DocumentClass(
        id=3,
        name="PTO",
        description="Permission-To-Operate approval from utility",
        fields=["home_address", "pto_receive_date"]
    ),
    DocumentClass(
        id=4,
        name="Warranty Extension",
        description="Document extending warranty on solar equipment",
        fields=["warranty_proof", "serial_number"]
    ),
    DocumentClass(
        id=5,
        name="Interconnection / NEM Agreement",
        description="Agreement for Net Energy Metering and grid connection",
        fields=["document_name", "homeowner_signature", "utility_signature"]
    )
]

# Dictionary for quick lookup by ID
CATEGORIES_BY_ID = {cat.id: cat for cat in DOCUMENT_CATEGORIES}
CATEGORIES_BY_NAME = {cat.name: cat for cat in DOCUMENT_CATEGORIES}

@dataclass
class ProcessingResult:
    category: str
    extracted_fields: Dict[str, str]
    raw_text: str
    confidence: float
```