## **🟡 Tier 2: Smart Intake + Metadata Automation**

### **🧩 Summary:**

Tier 2 streamlines file handling by automatically extracting metadata, storing files by client/project/date, and optionally applying branding watermarks. This is your productivity sweet spot.

### **📋 Workflow Logic:**

python

def tier2\_intake(file, user\_id, client\_tag):

save\_path = f"/intake/{client\_tag}/{date.today()}/"

os.makedirs(save\_path, exist\_ok=True)

checksum = hashlib.sha256(file.read()).hexdigest()

size\_mb = os.path.getsize(file.name) / (1024 \* 1024)

file.seek(0) # Reset after checksum read

metadata = {

"client": client\_tag,

"filename": file.name,

"size\_MB": round(size\_mb, 2),

"checksum": checksum,

"received": datetime.now().isoformat(),

"status": "received"

}

with open(f"{save\_path}/metadata.json", "w") as f:

json.dump(metadata, f)

shutil.copy(file.name, save\_path)

### **🖼️ Watermark Logic**

**Visible Watermark (.jpeg-based):**

python

def apply\_visible\_watermark(image\_path, watermark\_path, output\_path):

base = Image.open(image\_path).convert("RGBA")

mark = Image.open(watermark\_path).convert("RGBA")

mark = mark.resize((150, 75)) # scale for placement

base.paste(mark, (base.width - mark.width - 10, base.height - mark.height - 10), mark)

base.save(output\_path)

**Invisible Watermark: (metadata embedded)**

python

def embed\_invisible\_watermark(image\_path, metadata\_dict):

from PIL import PngImagePlugin

im = Image.open(image\_path)

meta = PngImagePlugin.PngInfo()

for key, val in metadata\_dict.items():

meta.add\_text(key, str(val))

im.save(image\_path.replace(".jpeg", "\_embed.jpeg"), "PNG", pnginfo=meta)

### **📝 Notes:**

* Watermark files stored in /watermarks/visible/ and /watermarks/invisible/
* Assets named using \_wm (visible) or \_embed (invisible) suffixes
* Review metadata.json before branding
* This tier supports light automation but allows human touchups

📁 Printing tip: Save this section as "JG-Tech Tier 2 Workflow – Smart Intake" in your physical backup binder.