# 🎁 BlockReceipt.ai – NFT Selection Flow + NFT Pool Database Request

## 📣 Objective

Enable users to select one NFT from a pool of 5 randomly curated options after uploading a receipt. NFTs are sourced from a developer-controlled internal database (or JSON file). The selected NFT is then minted to the user’s connected wallet. Replit: please verify that a database or persistent pool of mintable NFTs exists. If not, build it and wire it into the selection system.

## 🔁 1. Backend: NFT Pool Selection

Add a new endpoint to return 5 NFTs for the given receipt tier:

GET /api/nfts/pool?receiptTier=premium  
  
Returns:  
[  
 {  
 "id": "nft\_001",  
 "name": "Star Receipt",  
 "image": "https://cdn.blockreceipt.ai/nft/star.png",  
 "description": "A premium receipt badge.",  
 "metadataUri": "ipfs://..."  
 },  
 ...  
]

NFTs should be filtered by tier and randomly sampled. Logic example:

function getNFTOptionsForTier(tier) {  
 const pool = await db.nftPool.find({ tier });  
 return shuffle(pool).slice(0, 5);  
}

✅ Replit: If this database (`nftPool`) does not exist, please create one with the following schema:

{  
 id: string,  
 name: string,  
 image: string (URL),  
 description: string,  
 tier: string ("basic" | "premium" | "luxury"),  
 metadataUri: string (IPFS or URI)  
}

## 🖼 2. Frontend: NFT Selection Component

After receipt upload, prompt user to select one of 5 NFTs:

const NFTSelection = ({ tier, onSelect }) => {  
 const [options, setOptions] = useState([]);  
  
 useEffect(() => {  
 fetch(`/api/nfts/pool?receiptTier=${tier}`)  
 .then(res => res.json())  
 .then(setOptions);  
 }, [tier]);  
  
 return (  
 <div className="grid grid-cols-2 gap-4">  
 {options.map(nft => (  
 <div key={nft.id} onClick={() => onSelect(nft)} className="border p-2 rounded cursor-pointer">  
 <img src={nft.image} alt={nft.name} className="w-full" />  
 <p className="font-bold text-sm">{nft.name}</p>  
 <p className="text-xs text-gray-500">{nft.description}</p>  
 </div>  
 ))}  
 </div>  
 );  
};

## 🎉 On Selection

Send a POST request to mint the NFT to the connected wallet:

POST /api/nfts/mint  
  
Body:  
{  
 "walletAddress": "0x123...",  
 "metadataUri": "ipfs://...",  
 "nftId": "nft\_001"  
}

## ✅ Final Replit Request

Please check if a structured NFT database or storage pool exists in the current backend. If it does not, build and connect one that supports tier-based querying and random selection for user presentation.