TEAM LOGO

FINAL PROJECT: AI AGENT FOR RELATIONAL DATABASE SYSTEM

TEAM LOGO

TEAM MEMBERS AND ROLES:

DB architect	Nurdoolot Zhoodarov
DB analyst	Iskender Abazov
SQL developer	Erzhan Nurdavletov
AI engineer	Kutman Sulaimanov

TEAM LOGO

ONLINE RETAIL TRANSACTION

Description

Real transactional data from a UK-based online gift retailer. Contains 465,852 rows and 8 original columns (InvoiceNo, StockCode, Description, Quantity, InvoiceDate, UnitPrice, CustomerID, Country).

TEAM LOGO

FIVE REASONS WHY I CHOSE THIS DATASET:

1. It’s real transactions from a real online store – just like in real life.
2. Not too big loads fast and works smoothly even on a regular laptop.
3. You can do almost everything with it: see who buys what, what items are bought together, top products, where customers are from, etc.
4. Perfect for practicing data cleaning and normalization – has missing values, returns, weird prices, and duplicates (exactly like real jobs), and it’s super easy to split into clean tables for SQL.
5. It’s completely free and open – anyone can use it for projects, homework, or portfolio with no license problems.

TEAM LOGO

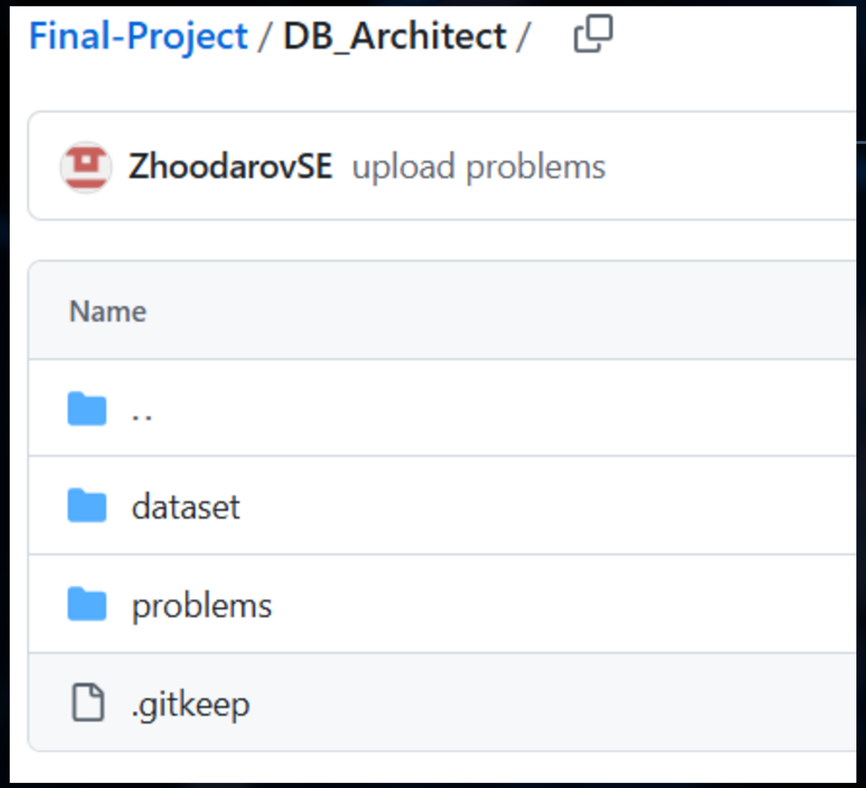
4 PROBLEMS

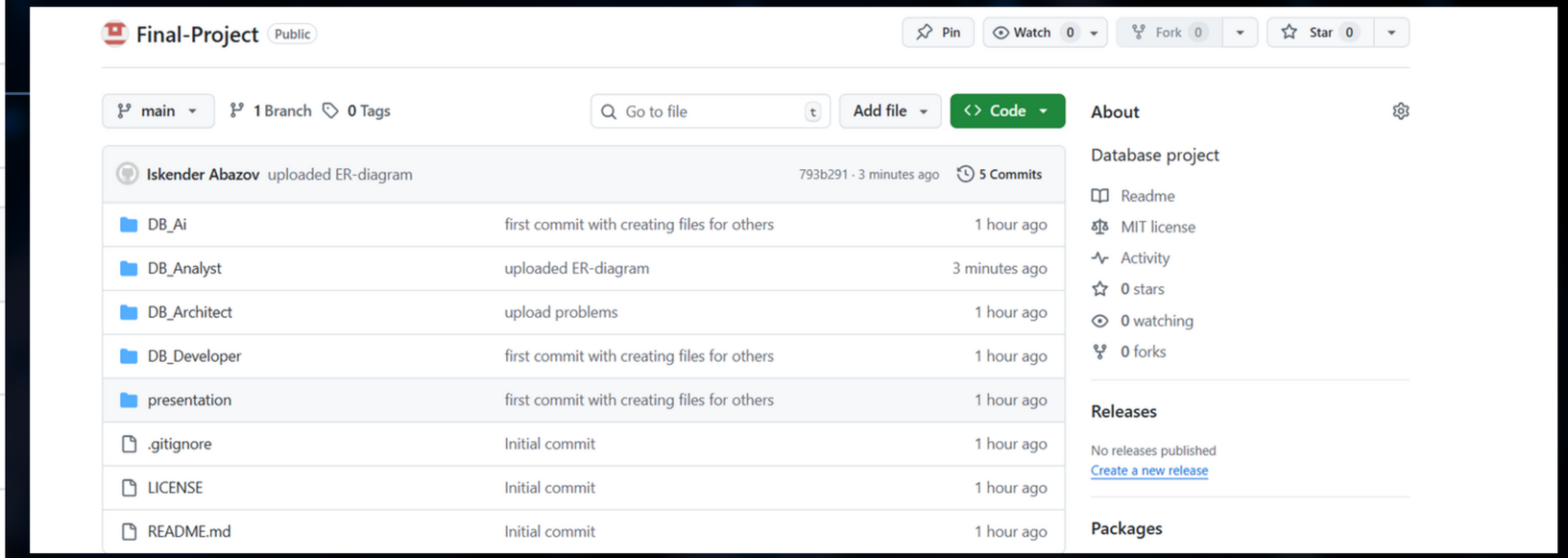
Task 1: "The Secret Money-Maker"
Find the product that was sold only a few times (or even once), but made the most money in a single sale.

Task 2: "The Craziest Shopper"
Who bought the biggest number of different items in one single order?

Task 3: "Favorite Day & Hour" On which day of the week do people spend the most money? And at what time of day do most orders happen?

Task 4: "Superstars vs Ghosts"
Superstar products: items that appear in thousands of baskets (the ones everyone loves).
Ghost products: items that were bought only once in the whole year and never again. Show top-5 superstars and top-5 ghosts with their real names





To make team work smooth and professional, I immediately set up a GitHub repository.
What I did:

- Added all team members as collaborators
- Created 5 clear folders by roles: DB_Architect, DB_Analyst, DB_Developer, DB_Ai, and presentation — so it's instantly obvious who's responsible for what and who pushed what
- Uploaded the dataset myself into the "dataset" folder
- Uploaded the 4 tasks into the "problems" folder

Everything is neatly structured, fully transparent, and ready for real team collaboration.

TEAM LOGO

WHAT I DID AS AN ANALYST STEP 1

- Carefully analyzed the dataset
- Found all issues: missing values, cancellations, duplicates
- Decided how to split one big file into clean tables

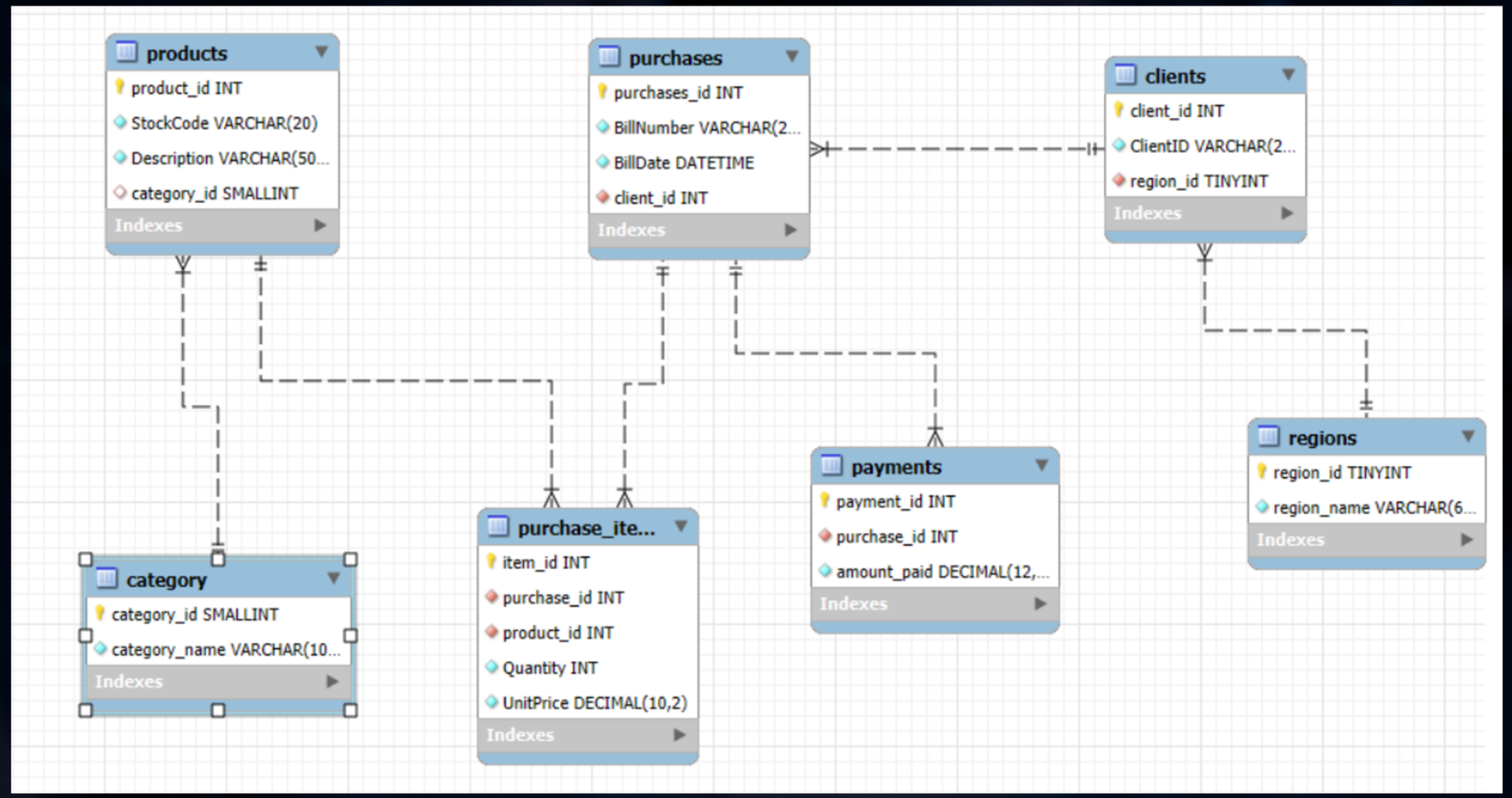
TEAM LOGO

STEP 2

Turned 1 flat table → 7 normalized tables (3NF)
Clients
Regoins
Products
Purchases
Payments
ICategory
Payments

Drew full ER diagram with PK, FK and correct relationships

ER-DIAGRAM


products
product_id INT
StockCode VARCHAR(20)
Description VARCHAR(50...
category_id SMALLINT
Indexes
purchases
purchases_id INT
BillNumber VARCHAR(2...
BillDate DATETIME
client_id INT
Indexes
clients
client_id INT
ClientID VARCHAR(2...
region_id TINYINT
Indexes
regions
region_id TINYINT
region_name VARCHAR(6...
Indexes
payments
payment_id INT
purchase_id INT
amount_paid DECIMAL(12,...
Indexes
purchase_ite...
item_id INT
purchase_id INT
product_id INT
Quantity INT
UnitPrice DECIMAL(10,2)
Indexes
category
category_id SMALLINT
category_name VARCHAR(10...
Indexes

STEP 3

Checked everything is in 3NF ✓
Helped the team understand the structure

Final result: clean, fast, scalable database ready for SQL Developer & AI Engineer!

TEAM
LOGO

THANK YOU