Firebase-Based Recipe Analytics Pipeline

Technical Report

Executive Summary

This report documents a complete data engineering pipeline built on Firebase Firestore for recipe data extraction, transformation, validation, and analytics. The system processes an authentic Maharashtrian cuisine dataset containing 20+ recipes, demonstrating modern ETL practices with NoSQL databases.

1. Introduction

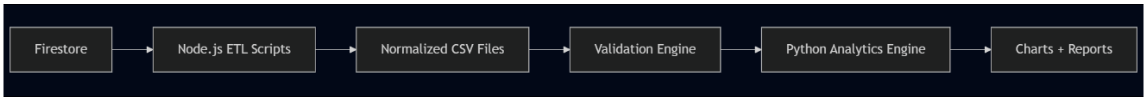
1.1 Project Overview

The Recipe Analytics ETL Pipeline is designed to extract recipe data from Firebase Firestore, transform it into normalized relational formats, validate data quality, and generate actionable insights through analytics and visualizations.

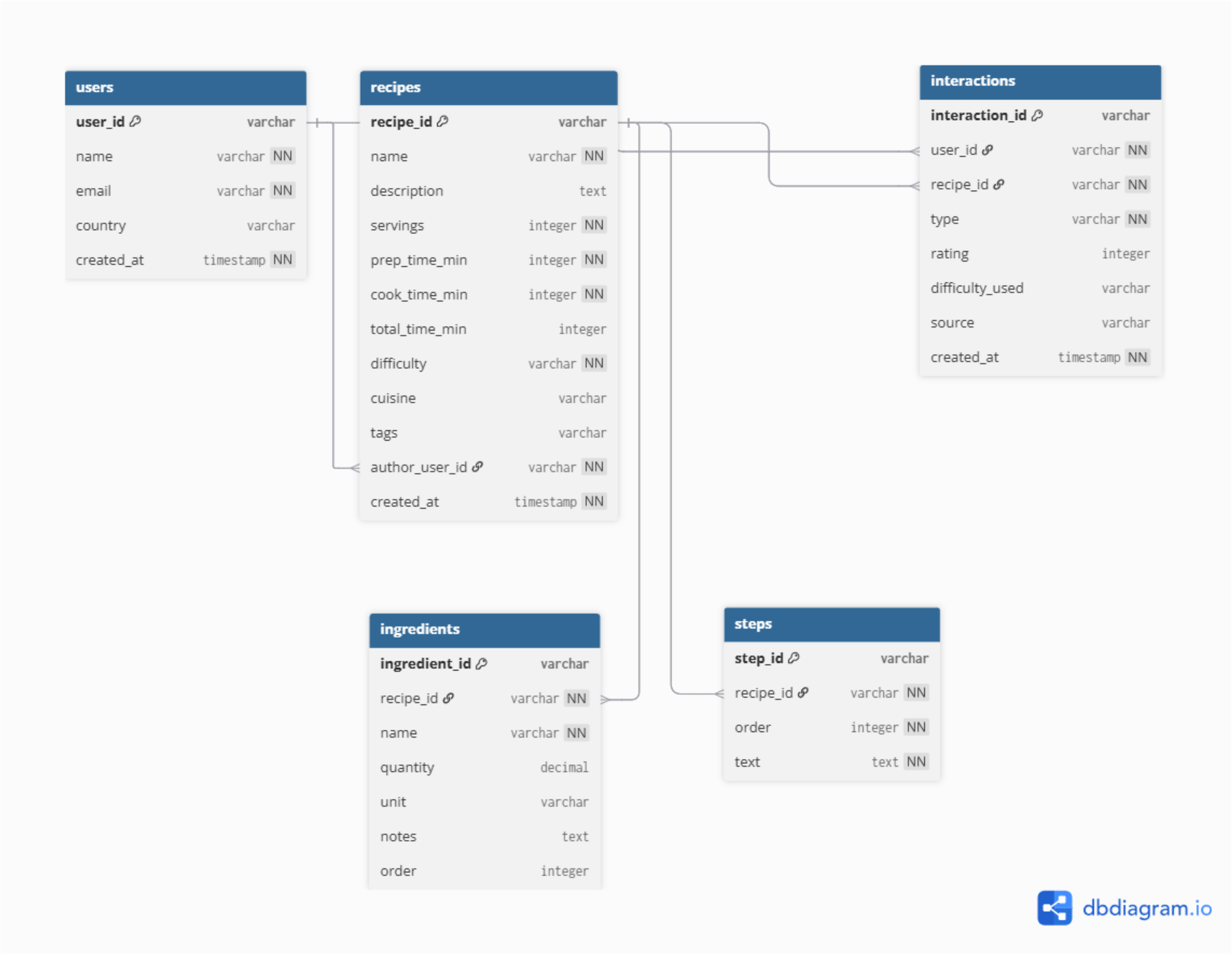
1.2 Technology Stack

Component	Technology
Database	Firebase Firestore
Programming Language	NODE JS & Python 3.8+
Data Processing	Pandas, NumPy
Visualization	Matplotlib
Firebase SDK	firebase-admin

1.3 System Architecture



2. Data Model Design



users		
user_id 🔑	varchar	
name	varchar	NN
email	varchar	NN
country	varchar	
created_at	timestamp	NN

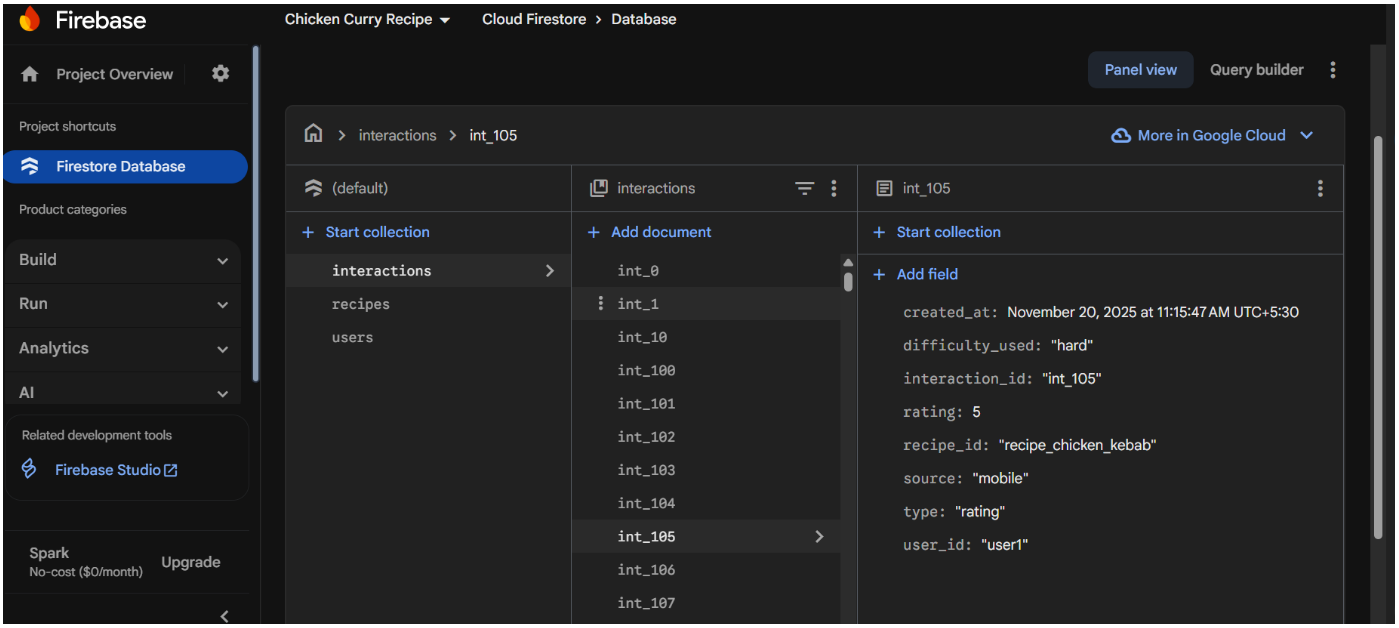
recipes		
recipe_id 🔑	varchar	
name	varchar	NN
description	text	
servings	integer	NN
prep_time_min	integer	NN
cook_time_min	integer	NN
total_time_min	integer	
difficulty	varchar	NN
cuisine	varchar	
tags	varchar	
author_user_id 🔗	varchar	NN
created_at	timestamp	NN

interactions		
interaction_id 🔑	varchar	
user_id 🔗	varchar	NN
recipe_id 🔗	varchar	NN
type	varchar	NN
rating	integer	
difficulty_used	varchar	
source	varchar	
created_at	timestamp	NN

ingredients		
ingredient_id 🔑	varchar	
recipe_id 🔗	varchar	NN
name	varchar	NN
quantity	decimal	
unit	varchar	
notes	text	
order	integer	

steps		
step_id 🔑	varchar	
recipe_id 🔗	varchar	NN
order	integer	NN
text	text	NN

2.1 Database Architecture



interactions > int_105

(default)

- interactions
- recipes
- users

interactions

- int_0
- int_1
- int_10
- int_100
- int_101
- int_102
- int_103
- int_104
- int_105
- int_106
- int_107

int_105

```
created_at: November 20, 2025 at 11:15:47 AM UTC+5:30
difficulty_used: "hard"
interaction_id: "int_105"
rating: 5
recipe_id: "recipe_chicken_kebab"
source: "mobile"
type: "rating"
user_id: "user1"
```

The system uses Firebase Firestore with a hierarchical document structure optimized for NoSQL patterns.

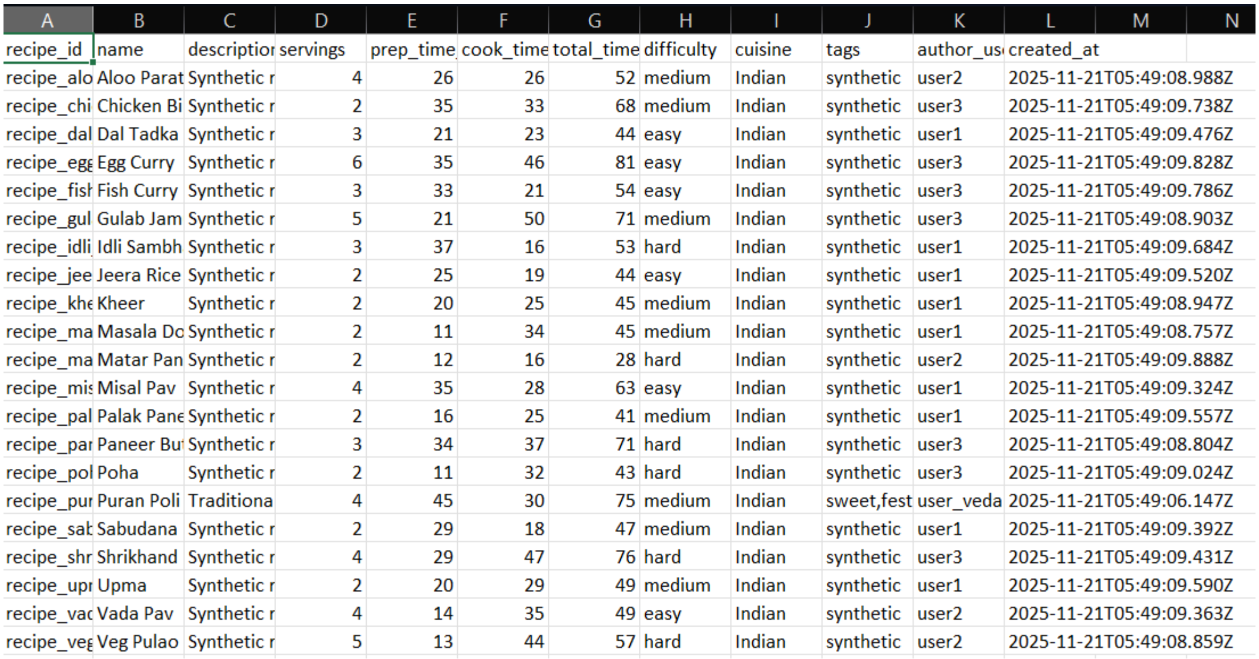
2.2 Design Decisions

Design Choice	**Reasoning**
Interactions under Recipes	Groups recipe activity together; enables fast queries for single recipe analytics
Activities under Users	Tracks user behavior across recipes; enables user-centric analytics
Denormalized author names	Avoids extra reads; Firestore doesn't support JOINs

2.3 Output Schema

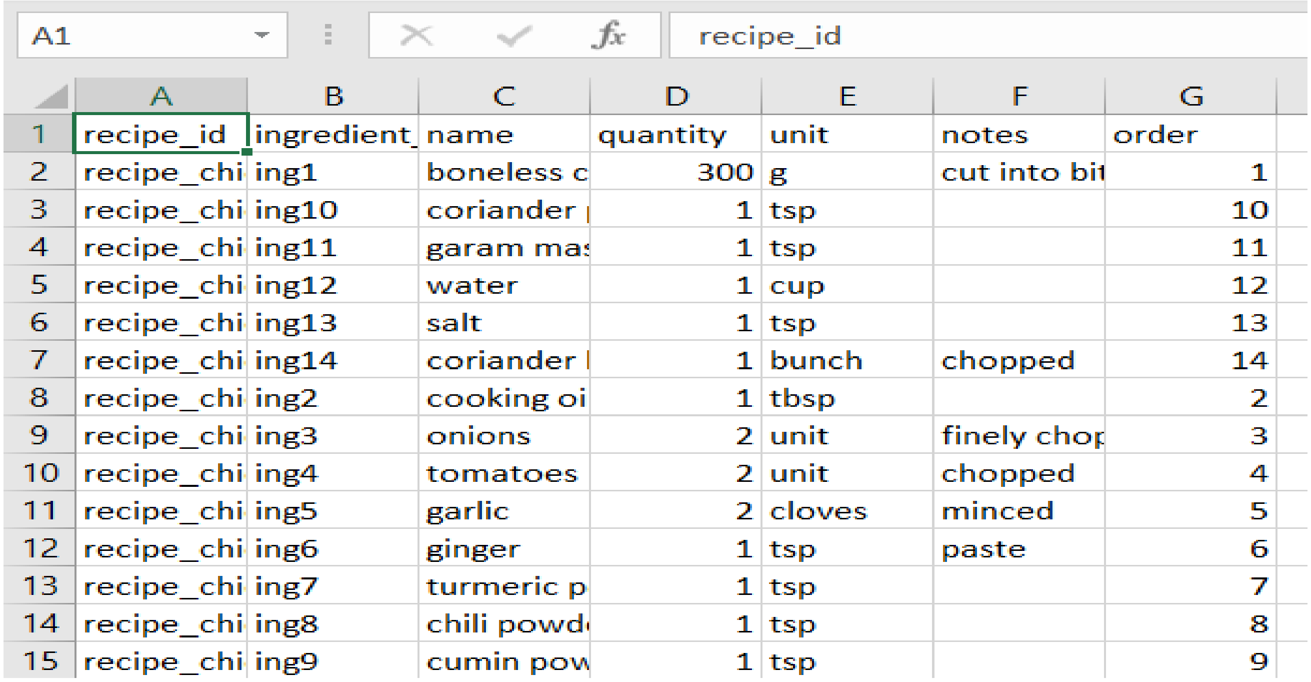
The pipeline produces four normalized CSV tables:

recipes.csv - Primary recipe information including title, description, timing, difficulty, category, dietary type, and author details.



A	B	C	D	E	F	G	H	I	J	K	L	M	N
recipe_id	name	descriptior	servings	prep_time	cook_time	total_time	difficulty	cuisine	tags	author_us	created_at		
recipe_alo	Aloo Parat	Synthetic r	4	26	26	52	medium	Indian	synthetic	user2	2025-11-21T05:49:08.988Z		
recipe_chi	Chicken Bi	Synthetic r	2	35	33	68	medium	Indian	synthetic	user3	2025-11-21T05:49:09.738Z		
recipe_dal	Dal Tadka	Synthetic r	3	21	23	44	easy	Indian	synthetic	user1	2025-11-21T05:49:09.476Z		
recipe_egg	Egg Curry	Synthetic r	6	35	46	81	easy	Indian	synthetic	user3	2025-11-21T05:49:09.828Z		
recipe_fish	Fish Curry	Synthetic r	3	33	21	54	easy	Indian	synthetic	user3	2025-11-21T05:49:09.786Z		
recipe_gul	Gulab Jam	Synthetic r	5	21	50	71	medium	Indian	synthetic	user3	2025-11-21T05:49:08.903Z		
recipe_idli	Idli Sambh	Synthetic r	3	37	16	53	hard	Indian	synthetic	user1	2025-11-21T05:49:09.684Z		
recipe_jee	Jeera Rice	Synthetic r	2	25	19	44	easy	Indian	synthetic	user1	2025-11-21T05:49:09.520Z		
recipe_khe	Kheer	Synthetic r	2	20	25	45	medium	Indian	synthetic	user1	2025-11-21T05:49:08.947Z		
recipe_ma	Masala Do	Synthetic r	2	11	34	45	medium	Indian	synthetic	user1	2025-11-21T05:49:08.757Z		
recipe_ma	Matar Pan	Synthetic r	2	12	16	28	hard	Indian	synthetic	user2	2025-11-21T05:49:09.888Z		
recipe_mis	Misal Pav	Synthetic r	4	35	28	63	easy	Indian	synthetic	user1	2025-11-21T05:49:09.324Z		
recipe_pal	Palak Pane	Synthetic r	2	16	25	41	medium	Indian	synthetic	user1	2025-11-21T05:49:09.557Z		
recipe_par	Paneer Bu	Synthetic r	3	34	37	71	hard	Indian	synthetic	user3	2025-11-21T05:49:08.804Z		
recipe_pol	Poha	Synthetic r	2	11	32	43	hard	Indian	synthetic	user3	2025-11-21T05:49:09.024Z		
recipe_pur	Puran Poli	Traditiona	4	45	30	75	medium	Indian	sweet,fest	user_veda	2025-11-21T05:49:06.147Z		
recipe_sab	Sabudana	Synthetic r	2	29	18	47	medium	Indian	synthetic	user1	2025-11-21T05:49:09.392Z		
recipe_shr	Shrikhand	Synthetic r	4	29	47	76	hard	Indian	synthetic	user3	2025-11-21T05:49:09.431Z		
recipe_upr	Upma	Synthetic r	2	20	29	49	medium	Indian	synthetic	user1	2025-11-21T05:49:09.590Z		
recipe_vac	Vada Pav	Synthetic r	4	14	35	49	easy	Indian	synthetic	user2	2025-11-21T05:49:09.363Z		
recipe_veg	Veg Pulao	Synthetic r	5	13	44	57	hard	Indian	synthetic	user2	2025-11-21T05:49:08.859Z		

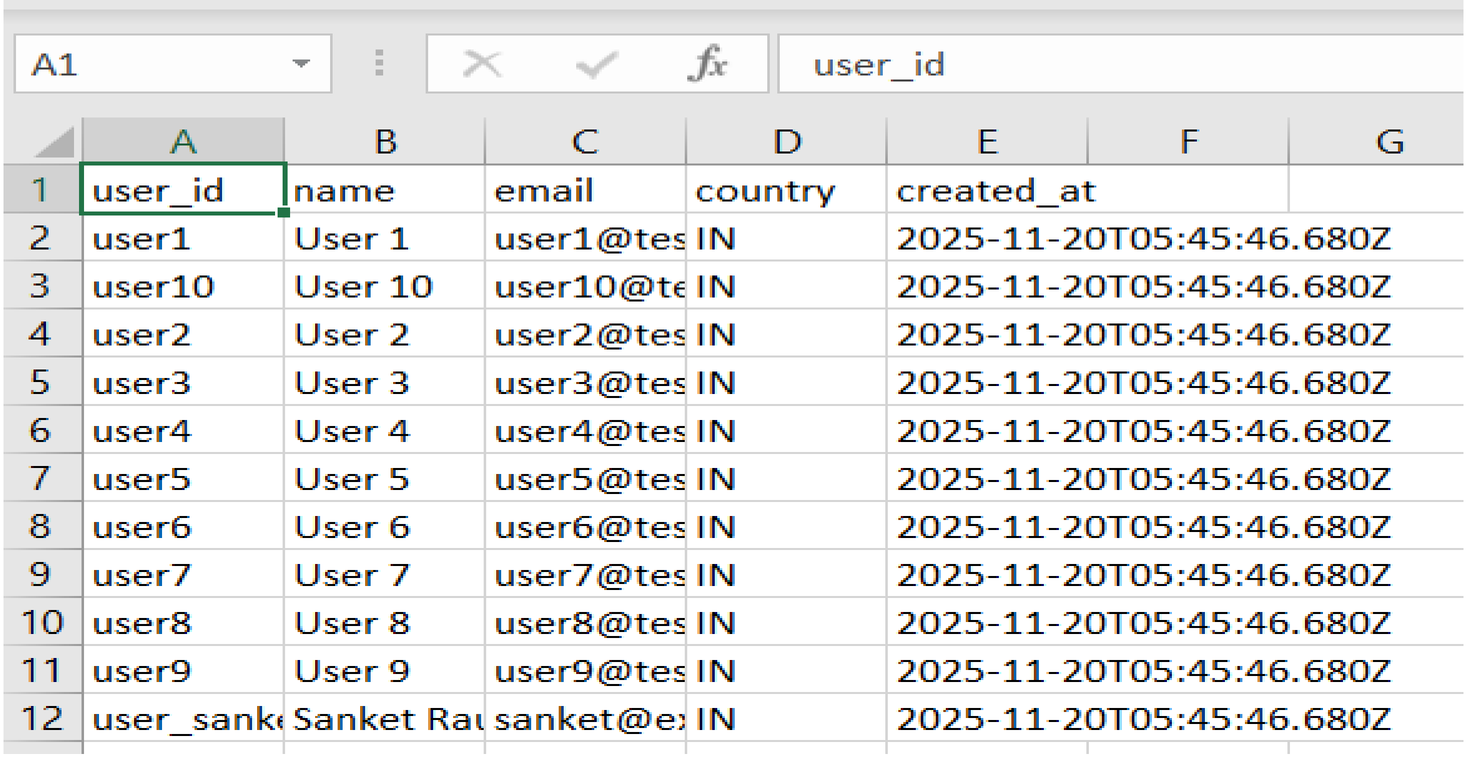
ingredients.csv - Normalized ingredient data linked to recipes via foreign key, containing name, quantity, unit, and optional flag.



A1 | recipe_id

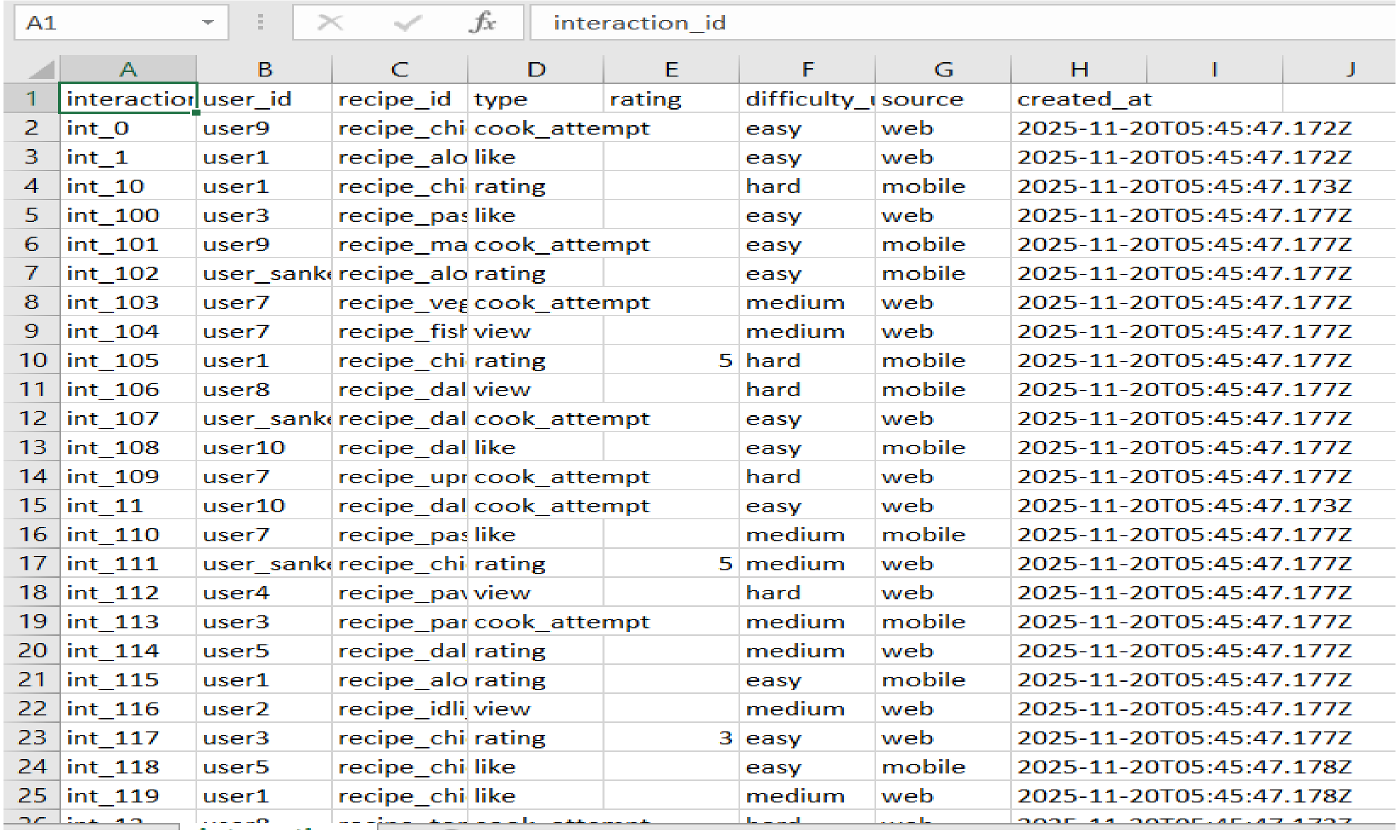
	A	B	C	D	E	F	G
1	recipe_id	ingredient_	name	quantity	unit	notes	order
2	recipe_chi	ing1	boneless c	300	g	cut into bit	1
3	recipe_chi	ing10	coriander	1	tsp		10
4	recipe_chi	ing11	garam mas	1	tsp		11
5	recipe_chi	ing12	water	1	cup		12
6	recipe_chi	ing13	salt	1	tsp		13
7	recipe_chi	ing14	coriander	1	bunch	chopped	14
8	recipe_chi	ing2	cooking oi	1	tbsp		2
9	recipe_chi	ing3	onions	2	unit	finely chop	3
10	recipe_chi	ing4	tomatoes	2	unit	chopped	4
11	recipe_chi	ing5	garlic	2	cloves	minced	5
12	recipe_chi	ing6	ginger	1	tsp	paste	6
13	recipe_chi	ing7	turmeric p	1	tsp		7
14	recipe_chi	ing8	chili powd	1	tsp		8
15	recipe_chi	ing9	cumin pow	1	tsp		9

users.csv - Recipe instructions with step numbers, instructions, and duration, linked to parent .



	A	B	C	D	E	F	G
1	user_id	name	email	country	created_at		
2	user1	User 1	user1@tes	IN	2025-11-20T05:45:46.680Z		
3	user10	User 10	user10@te	IN	2025-11-20T05:45:46.680Z		
4	user2	User 2	user2@tes	IN	2025-11-20T05:45:46.680Z		
5	user3	User 3	user3@tes	IN	2025-11-20T05:45:46.680Z		
6	user4	User 4	user4@tes	IN	2025-11-20T05:45:46.680Z		
7	user5	User 5	user5@tes	IN	2025-11-20T05:45:46.680Z		
8	user6	User 6	user6@tes	IN	2025-11-20T05:45:46.680Z		
9	user7	User 7	user7@tes	IN	2025-11-20T05:45:46.680Z		
10	user8	User 8	user8@tes	IN	2025-11-20T05:45:46.680Z		
11	user9	User 9	user9@tes	IN	2025-11-20T05:45:46.680Z		
12	user_sank	Sanket Ra	sanket@e	IN	2025-11-20T05:45:46.680Z		

interactions.csv - User engagement data including ratings, cook notes, and timestamps.



	A	B	C	D	E	F	G	H	I	J
1	interactior	user_id	recipe_id	type	rating	difficulty_	source	created_at		
2	int_0	user9	recipe_chi	cook_attempt		easy	web	2025-11-20T05:45:47.172Z		
3	int_1	user1	recipe_alo	like		easy	web	2025-11-20T05:45:47.172Z		
4	int_10	user1	recipe_chi	rating		hard	mobile	2025-11-20T05:45:47.173Z		
5	int_100	user3	recipe_pas	like		easy	web	2025-11-20T05:45:47.177Z		
6	int_101	user9	recipe_ma	cook_attempt		easy	mobile	2025-11-20T05:45:47.177Z		
7	int_102	user_sank	recipe_alo	rating		easy	mobile	2025-11-20T05:45:47.177Z		
8	int_103	user7	recipe_veg	cook_attempt		medium	web	2025-11-20T05:45:47.177Z		
9	int_104	user7	recipe_fish	view		medium	web	2025-11-20T05:45:47.177Z		
10	int_105	user1	recipe_chi	rating	5	hard	mobile	2025-11-20T05:45:47.177Z		
11	int_106	user8	recipe_dal	view		hard	mobile	2025-11-20T05:45:47.177Z		
12	int_107	user_sank	recipe_dal	cook_attempt		easy	web	2025-11-20T05:45:47.177Z		
13	int_108	user10	recipe_dal	like		easy	mobile	2025-11-20T05:45:47.177Z		
14	int_109	user7	recipe_upr	cook_attempt		hard	web	2025-11-20T05:45:47.177Z		
15	int_11	user10	recipe_dal	cook_attempt		easy	web	2025-11-20T05:45:47.173Z		
16	int_110	user7	recipe_pas	like		medium	mobile	2025-11-20T05:45:47.177Z		
17	int_111	user_sank	recipe_chi	rating	5	medium	web	2025-11-20T05:45:47.177Z		
18	int_112	user4	recipe_pav	view		hard	web	2025-11-20T05:45:47.177Z		
19	int_113	user3	recipe_par	cook_attempt		medium	mobile	2025-11-20T05:45:47.177Z		
20	int_114	user5	recipe_dal	rating		medium	web	2025-11-20T05:45:47.177Z		
21	int_115	user1	recipe_alo	rating		easy	mobile	2025-11-20T05:45:47.177Z		
22	int_116	user2	recipe_idli	view		medium	web	2025-11-20T05:45:47.177Z		
23	int_117	user3	recipe_chi	rating	3	easy	web	2025-11-20T05:45:47.177Z		
24	int_118	user5	recipe_chi	like		easy	mobile	2025-11-20T05:45:47.178Z		
25	int_119	user1	recipe_chi	like		medium	web	2025-11-20T05:45:47.178Z		

3. ETL Process

3.1 Pipeline Architecture

The pipeline follows a four-stage process: **Extract → Transform → Validate → Analyze**

3.2 Extract Phase

The extraction phase connects to Firebase Firestore using the Admin SDK with streaming for efficient data retrieval. Key operations include authentication via service account credentials, document streaming for large collections, subcollection extraction for interactions, and timestamp conversion to ISO 8601 format.

3.3 Transform Phase

Transformation	Description
Flatten Ingredients	Nested array converted to separate CSV with recipe_id FK
Flatten Steps	Nested array converted to separate CSV with recipe_id FK
Extract Subcollections	Firestore subcollection to interactions.csv
Normalize Time	Structured time object to individual minute columns
Handle Missing Data	Default values: "Uncategorized", "Unknown"

4. Data Validation

4.1 Validation Rules

Rule	Field	Criteria
Required Fields	title	Must not be empty
Valid Difficulty	difficulty	Must be: Easy, Medium, Hard, Expert

Rule	Field	Criteria
Prep Time	prep_time_min	Must be > 0
Cook Time	cook_time_min	Must be ≥ 0
Time Logic	total_time_min	Must be ≥ prep_time + cook_time
Ingredient Quantity	quantity	Must be > 0 if numeric
Rating Range	rating	Must be between 0 and 5
Has Steps	steps	At least one step required
Has Ingredients	ingredients	At least one ingredient required

4.2 Validation Output

The validator produces a JSON report containing total recipe count, valid/invalid counts, detailed error messages for invalid records, and list of valid record IDs.

---

5. Analytics & Insights

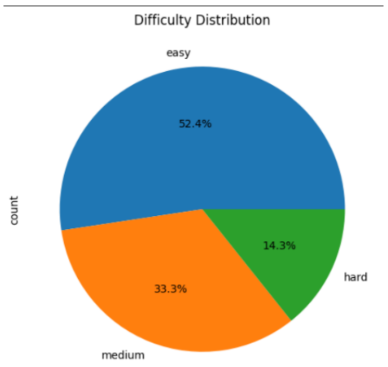
5.1 Generated Insights

The pipeline produces 11 analytical insights:

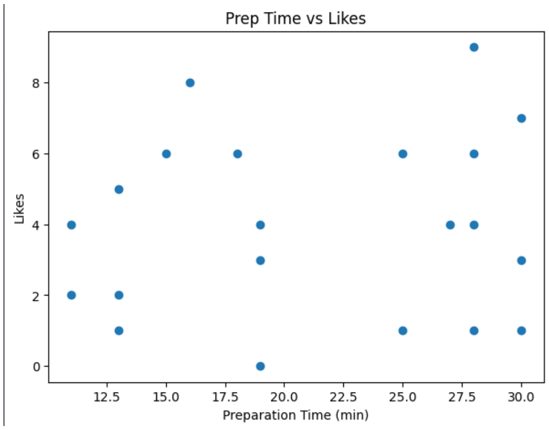
1. **Most Common Ingredients** - Top 20 ingredients by frequency
2. **Average Prep Time** - Mean preparation time in minutes
3. **Average Cook Time** - Mean cooking time in minutes
4. **Difficulty Distribution** - Recipe count per difficulty level
5. **Most Interacted Recipes** - Top 20 by interaction count
6. **Prep vs Rating Correlation** - Statistical correlation analysis

7. **High-Rating Ingredients** - Ingredients appearing in 4+ star recipes
8. **Top Rated Recipes** - Top 10 by average rating
9. **Steps Distribution** - Statistical summary of recipe complexity
10. **Most Commented Recipes** - Top 10 by cook note count
11. **Longest Recipes** - Top 10 by total preparation time

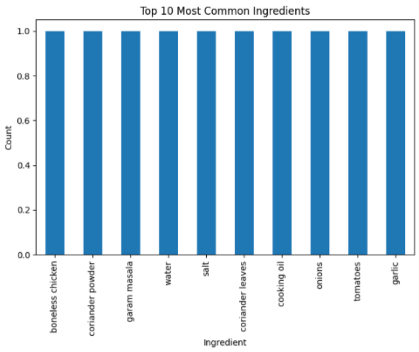
5.2 Visualizations



6.1 Diffculty Distribution

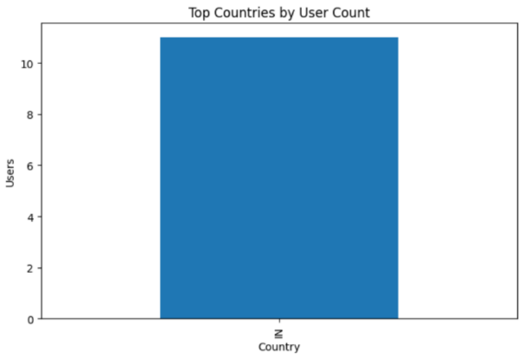


6.2 Prep Time vs Like Count

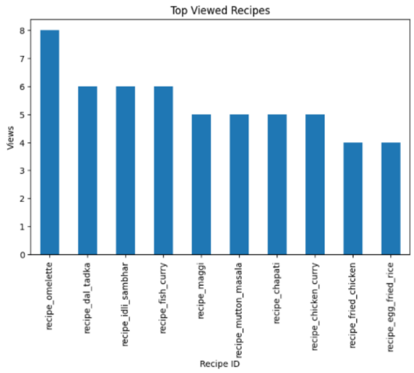


6.3 Top 10 Ingredients (by frequency)

7. User Analytics (Charts)



7.1 Users by Country



7.2 Top Users by Interaction Count

6. Constraints & Limitations

6.1 Firestore Limitations

Constraint	Impact	Mitigation
No native JOINs	Cannot query across collections	Denormalized data; subcollections
Read costs	Each document read is billed	Streaming instead of batch reads
No aggregations	No COUNT/SUM/AVG in queries	Aggregations in Python post-export
Subcollection queries	Cannot query all at once	Iterate per parent document

6.2 Pipeline Constraints

- Sequential execution required (seed → transform → validate → analyze)
- Full export only; no incremental/delta processing
- Memory-bound with pandas DataFrames
- Hardcoded relative paths in some scripts

6.3 Scalability Notes

- Current tested capacity: ~20 recipes, ~200 interactions
- For 1000+ recipes: implement pagination in export
- For larger datasets: consider chunked processing or Apache Spark

7. Project Structure

```
recipe-pipeline-node/
├── docs/        # Project documentation
│   ├── data_dictionary.md    # Detailed description of all tables/fields
│   └── recipe_erd_diagram.png # ERD diagram used in design
│
├── output/     # ETL outputs, analytics charts & reports
│   ├── ingredients.csv # Ingredients dimension data
│   ├── interactions.csv # User–recipe interaction fact data
│   ├── recipe.csv # Recipes master table
│   ├── steps.csv # Recipe steps data
│   ├── users.csv # Users dimension data
│   ├── validation_report.csv # Data quality / validation summary
│   ├── analytics_summary.txt # Text summary of key analytics findings
│   ├── charts.py # Script to generate visualizations
│   ├── difficulty_distribution.png
│   ├── prep_time_vs_likes.png
│   ├── top_10_most_common_ingredients.png
│   ├── top_countries_by_user_count.png
│   ├── top_liked_recipes.png
│   ├── top_viewed_recipes.png
│   ├── user_growth_by_month.png
│   └── users_with_the_most_recipes.png # All analytics charts (PNG)
│
├── .gitignore    # Git configuration to ignore temp files
├── analytics.js # Runs analytics on exported data
├── export_etl.js # ETL pipeline – exports Firebase data to CSV
├── insert_data.js # Seeds Firebase with sample recipe data
├── validate_data.js # Data validation & consistency checks
├── recipe_erd_diagram.png # ERD quick reference (copy in root)
└── README.md # Project overview & setup instructions
```

8. Installation & Execution

8.1 Prerequisites

- Python 3.8 or higher
- Firebase project with Firestore enabled
- Service account credentials (JSON)

8.2 Execution Steps

1. **Seed Initial Data** - Run seed_data.py to create base recipe and user
2. **Generate Synthetic Data** - Run genrate_sytetic.py for 20 Maharashtrian recipes
3. **Transform Data** - Run transform.py to export to CSV

4. **Validate Data** - Run validator.py for quality checks
5. **Generate Analytics** - Run analytics.py for insights and charts

---

9. Deliverables Summary

Deliverable	**Status**
Source files for ETL scripts	Complete
Validation script	Complete
Normalized CSV output	Complete
Analytics summary (JSON)	Complete
Documentation	Complete
Visualization charts	Complete

---

10. Conclusion

The Firebase-Based Recipe Analytics Pipeline successfully demonstrates a complete ETL workflow for NoSQL data, producing normalized relational outputs suitable for further analysis. The system handles the unique challenges of Firestore's document model while maintaining data quality through comprehensive validation.
