ZORDIE AI — Data Scientist 1 & Data Scientist 2 Combined Task Breakdown

Tasks :

1. Resume Intelligence System

Owned Jointly by: DS1 + DS2

Tasks:

- Build a **Resume Analysis Engine** that performs:
    - Section Detection
    - Skill-to-JD Alignment
    - Project Relevance Analysis
    - Formatting and Trustworthiness Checks
    - Inflation/Deception Flags

Components:

A. Section & Structure Detector (DS1)

- Use NLP techniques (spaCy + Regex + Transformer models) to extract:
    - Education
    - Work Experience
    - Projects
    - Certifications
    - Skills
    - Summary
- Flag missing or underdeveloped sections.

B. Skill-to-JD Semantic Matcher (DS1)

- Use **SBERT/BERT embeddings** to:
    - Compare candidate skills with job JD
    - Calculate alignment % score
    - Highlight **missing critical skills**

C. Project Validation System (DS1)

- NLP analysis of project descriptions:

    - Compare with skills listed
    - Flag unrelated projects
    - Score depth and tech relevance

D. Formatting Scorer (DS2)

- Detect formatting issues:
    - Bullet usage
    - Font consistency
    - Section headers
    - Length limits
- Use rule-based scoring:
    - Penalize for walls of text, inconsistent fonts, etc.

E. Resume Trustworthiness & Inflation Detector (DS2)

- Text fingerprinting and skill-tag inflation detection
- Cross-check GitHub/Figma projects vs claimed experience
- Create "Resume Inflation Flags" if:
    - Skills > evidence
    - Copied text blocks
    - Unrealistic claims

F. Credibility & Verifiability Engine (DS2)

- Cross-check:
    - Certificates
    - LinkedIn activity
    - GitHub contributions
    - Use scrapers to verify names on certs

G. Final Resume Quality Score & Issue Report (DS1 + DS2)

- Output structured JSON with:
  - Resume Score (out of 100)
  - Alignment %
  - Formatting Score
  - Trust Score
  - Flags + Recommendations

2. Multi-Platform Link Crawling + Data Extraction System

Owned Jointly by: DS1 + DS2

Objective:

- Build link crawler that crawls **all links found in resume**, not just GitHub.

Platforms to crawl:

- GitHub (repos, commits, issues)
- LeetCode (solved problems, ranking)
- Figma, Dribbble, Framer (designs, stars, forks)
- Kaggle (notebooks, rankings)
- LinkedIn (activity posts, endorsements, skills)
- Medium, Behance, etc.

Tasks:

A. Universal Link Crawler Engine (DS2)

- Design a headless browser + crawler system using **Playwright/Selenium + BeautifulSoup**
- Accept any resume link and detect:
  - Type of platform
  - Whether it's public or not
  - Extract basic metadata (titles, project names, followers, engagement)

B. Platform-Specific Data Extractors (DS1)

- Write crawlers/scrapers:
  - GitHub: repo names, stars, forks, language distribution, commit graphs
  - Figma/Dribbble: project names, like counts, design tags
  - LeetCode: total solved, rating, top problem types
  - Kaggle: notebook count, competitions, ranks
  - LinkedIn: post frequency, endorsements, skill match

C. Metadata Normalizer & Schema Designer (DS2)

- Design unified schema to store:
  - Platform
  - Link Type
  - Engagement Stats
  - Content Types
  - Relevance Scores
  - Date of Last Activity

3. Scoring and Prediction Systems

Owned Jointly by: DS1 + DS2

Tasks:

A. GitHub Technical Skill Predictor (DS1 Lead)

- Analyze cloned GitHub repos using:
  - AST for structure
  - Cyclomatic complexity
  - Test coverage
  - Originality check (w/ plagiarism API)

- Generate:
  - Technical Skill Score
  - Code Quality Grade (A–F)
  - Originality %

B. Multi-Platform Weighted Score Engine (DS2)

- Create scoring formulas per platform:
  - LeetCode → Algorithmic Score
  - Kaggle → Data Science Depth Score
  - Figma/Dribbble → Creative Score
  - LinkedIn → Social Trust Score
- Final weighted score system:
- final_score = 0.3 * GitHub_score +
- 0.2 * LeetCode_score +
- 0.2 * Cert_score +
- 0.1 * Design_score +
- 0.1 * Resume_score +
- 0.1 * LinkedIn_activity_score

C. AI-Based Prediction Engine (DS1)

- Use training data from known hires to:
  - Train a classifier (e.g., XGBoost, Random Forest)
  - Predict:
    - "Likely to Pass Technical Round"
    - "Good Culture Fit"
    - "High Learning Adaptability"

D. Explainability Layer (DS2)

- Show HR:
  - Why someone got a score
  - Which strengths and weaknesses matter
  - What the prediction means

Tools & Frameworks:

Task	**Tools/Tech Stack**
Resume Parsing	PyMuPDF, spaCy, Regex
Skill-JD Matching	SBERT, BERT, cosine similarity
Crawling	Playwright, Selenium, BeautifulSoup
GitHub Analysis	GitPython, AST, Plagiarism API
ML Models	XGBoost, LightGBM, Sklearn Pipelines
Resume Score System	Custom Rule Engine + ML Features
Storage & Normalization	MongoDB, PostgreSQL, JSON Schemas
APIs	FastAPI / Flask (to serve scoring engine)
Monitoring	Logging w/ Prometheus, AlertManager, Cron Schedulers