Here’s a **structured flow** for your system, incorporating:

- Web scraping
- Problem regeneration with Gemini (including emotion tagging)
- Skill-based matching for individuals or groups
- Project idea generation from selected problems

IdeaGenie Flow: From Scraping to AI-Powered Project Idea

---

1. Problem Collection & Enrichment (Backend AI Flow)

a. Scraping Data:

- Platforms: Reddit, Stack Overflow, Twitter/X, GitHub Issues, Product Hunt Discussions, etc.
- Extract: Post content, tags, timestamp, author, community context

b. Problem Refinement (Gemini API):

- Clean and rephrase the raw problem into a concise **Problem Statement**
- Extract the **Emotion Behind the Problem** using sentiment/emotion analysis *(e.g., frustration, confusion, urgency, curiosity)*

```
{
  "original": "I’ve tried 10 AI tools but none of them summarize PDFs well.",
  "regenerated": "Lack of accurate PDF summarization tools powered by AI.",
  "emotion": "Frustration"
}
```

c. Store in Database:

- Store the following fields:
  - `id`, `source`, `tags`, `problem_statement`, `emotion`, `scrape_date`

---

2. User or Group Onboarding

a. Profile Creation:

- User or group signs up and creates a profile.
- Add tech stack manually or through GitHub/LinkedIn import.
- (For groups: aggregate all member skills + group interest tags)

b. Context Preferences (Optional):

- Users can optionally select themes they care about (e.g., AI, mental health, fintech, climate)

---

3. Problem Matching (Suggestion Stage)

a. AI-Based Filtering:

- Gemini or internal ranking logic matches user/group skillset with relevant problems.

- Filter by:
  - Skills (e.g., Python, Flutter, React, ML)
  - Interests
  - Emotion (optional: filter problems with specific emotion intensity like “urgent”, “confused”)

b. Display Suggested Problems:

- Show 10–15 **problem statements** with emotion tags and relevance score.

---

4. Problem Selection

- Users (or the group) **select 4–5 problem statements** they feel connected to.

---

5. Project Idea Generation (Gemini-powered)

- Gemini takes the selected problem statements + group skills + context
- Generates:
  - Project idea
  - Features list
  - Suggested tech stack
  - Estimated complexity/time
  - Real-world use case

```
Input: [
  "Lack of efficient PDF summarization tools",
  "College students find it hard to manage multiple deadlines",
  ...
], Skills: ["React", "Node.js", "LangChain", "Python"]

Output: "Build a smart academic dashboard that summarizes PDF study material, tracks
task deadlines, and suggests AI-generated notes using LangChain."
```

---

6. Save & Export

- Users can:
  - Save the idea to their dashboard
  - Export idea as PDF/Notion document
  - Share with team

High-Level Flow Diagram

```
[ Social Media Scraper ]
          ↓
 [ Gemini Regeneration + Emotion Tagging ]
          ↓
   [ Store to DB with tags + emotion ]
          ↓
    [ User/Group Signup + Tech Stack ]
```

```
          ↓
[ Gemini matches problems with users ]
          ↓
   [ User selects 4–5 problems ]
          ↓
  [ Gemini generates custom project idea ]
          ↓
        [ Output, Save, Export ]
```