ASCEND - Project Executive Summary

> **One-Line Pitch:** A college-specific mentorship platform connecting students with alumni at target companies (e.g., Google, Odoo) via structured, asynchronous Q&A.

---

1. The Core Problem

Junior students often lack structured access to alumni working at their dream companies. - **Cold Messaging Fails:** LinkedIn response rates are <10%. - **Generic Advice:** "Work hard" is not actionable; students need company-specific roadmaps. - **Mentor Burnout:** Alumni get spammed with repetitive requests like "How to get a job?". - **No Accountability:** No way to know if advice is good or if students actually follow it.

2. The Solution: ASCEND

ASCEND replaces chaotic DMs with a structured **Company-First** approach.

Key Differentiators:

1. **Company-Based Discovery:** Students search for "Odoo" or "TCS" to find alumni working *there*, rather than just searching by generic skills.
2. **Asynchronous Q&A:** No real-time chat. Students submit structured questions; Mentors answer on their own time.

   App-Specific Logic:
   - **FIFO Queue:** Mentors answer questions in the order received (Data Structure: Queue).
   - **Trust Scores:** Mentors earn scores based on student outcomes (e.g., "Got an interview").
   - **Knowledge Base:** Past answers become a searchable library, reducing duplicate questions.

---

3. How It Works (Core User Flows)

■■■ For Students (The Seekers)

1. **Discover:** Search for a company (e.g., "Google"). See a list of verified alumni working there.

   Ask: Submit a **Structured Question**.
   - *Required Fields:* Current Status, Specific Question, Urgency.
   - *Context:* "I have solved 100 LeetCode problems and built 1 Django project."
   - *AI Check:* System warns if the question is too vague or a duplicate.
3. **Learn:** Receive a detailed, written response.
4. **Verify:** After 4 weeks, provide feedback (Did it help? Did you get the job?).

■■■ For Alumni Mentors (The Guides)

1. **Queue Dashboard:** See incoming questions in a strict **First-In-First-Out (FIFO)** list.
2. **Control:** Set availability "On/Off" or limit questions to 5/month.
3. **Profile:** Build a reputation via the **Trust Score** (visible on their profile).
4. **Referrals:** Formal workflow to approve/reject job referral requests based on student portfolio.

■■ For Admins (The College)

1. **Verification:** Approve alumni profiles via LinkedIn links.
2. **Analytics:** Track placement success and alumni engagement.

---

4. Key Features & Modules

Module	Functionality
Company Directory	List of target companies with stats: "12 Alumni at Odoo", "Hiring Active".
Mentor Matching	Algorithm suggests mentors based on Student's Target Company + Mentor's Current Role.
Queue System	Ensures fairness. Mentors cannot "cherry pick" easy questions; they answer in order.
Knowledge Base	A "Stack Overflow" for career advice. Searchable by company/topic.
Referral System	Unlockable feature. Students can only request referrals after building trust/interaction history.

---

5. Technology Stack

- **Backend:** Python (Flask)
- **Database:** PostgreSQL (Production) / SQLite (Dev)
- **Frontend:** HTML5, Bootstrap 5, Jinja2 Templates (Responsive Web App)

Concepts Used:

- **Queue (FIFO):** Managing mentor workload.
- **Priority Queue:** Handling "Urgent" questions.
- **Hashing:** Efficient mentor-company lookups.

---

6. Database Overview (Key Tables)

1. **Users:** Login info (Email, Password, Role).
2. **Students:** Profile (Batch, Branch, Skills, CGPA).
3. **Alumni:** Work History (Current Company, Role), Trust Score, Availability.
4. **Companies:** Master list (Name, Industry, Logo).
5. **Questions:** The core content (Student ID, Company ID, Body, Status).
6. **Responses:** Mentor answers linked to Questions.

Summary for Developers: We are building a **Flask Web App** where Students query Company-Specific Alumni. The app enforces **Queuing Logic** for fairness and uses **Trust Scores** for quality control. It's designed to solve the "Networking Gap" in colleges.