AI Functionalities:

Keyword Extraction: Use NLP to extract keywords or skills mentioned in job descriptions.

Sentiment Analysis: Analyze the sentiment of job descriptions to gauge employer sentiment.

Candidate Matching: Develop a recommendation system to match candidates with suitable job postings based on their skills and experience.

Resume Parsing: Use NLP techniques to parse resumes and extract relevant information like education, experience, and skills.

Automated Messaging: Develop a feature to contact job owners via LinkedIn, email, or phone. Use AI techniques like natural language generation (NLG) to generate personalized messages based on job details.

Lead Scoring: Use AI to prioritize leads based on their likelihood to convert, helping sales teams focus their efforts on high-value prospects.

Customer Segmentation: Analyze customer data to identify distinct segments and tailor sales strategies accordingly.

Predictive Analytics: Use machine learning models to forecast sales trends, identify potential opportunities, or anticipate customer needs.

Dynamic Pricing: Implement dynamic pricing strategies based on market demand, competitor pricing, and other relevant factors.

AI Model Evaluation: Evaluate the performance of AI models using metrics like accuracy, precision, and recall. Refine models as needed based on feedback.

The Scraper:

Programming language: Suited for web scraping and AI, such as Python, which offers libraries like BeautifulSoup and Scrapy for scraping and TensorFlow or PyTorch for AI.

Scraping: Develop code to scrape job listings from targeted websites. Use web scraping libraries to extract relevant information like source, company name, contact person name, email, phone, country, city job titles, descriptions, salary and other details.

AI Integration: Implement AI functionalities into the scraper. For example, use NLP techniques to analyze job descriptions for keywords or sentiment. You can also use machine learning models to classify job postings into categories or predict candidate suitability.

Data Validation: Ensure scraped data is accurate and complete by validating against known sources or manual checks.

User Testing: Test the functionality and user interface of the scraper to ensure it meets user needs and expectations.

The scraper should be able to extract data from various online job boards, including major platforms like LinkedIn, Indeed, Glassdoor, Monster, Simply Hired, CareerBuilder, FlexJobs Corporation, ZipRecruiter, Job2Careers, as well as local ones such as Bayt, GulfTalent, Naukrigulf, Ladders, Naukri, AngelList, Wellfound, Getwork, Jobindex, and Jobnet. This list is not exhaustive, as each job board may have unique features.

When scraping a job listing, a key feature is the ability to find contact details outside of the job board if they are not provided within the listing. For example, if Coca-Cola is seeking a marketing manager and only provides an "Apply Now" button without contact information, the scraper and AI should search platforms like LinkedIn to find details for hiring managers, directors, CHROs, or other relevant contacts. This feature should be optional, but the top 10 contacts in higher-level positions should be displayed. These details must be validated, similar to services like RocketReach or ZoomInfo, to provide accurate email and phone information.

The next step involves AI functionality that can automatically send messages to potential clients via email, SMS, LinkedIn, and other platforms. These messages should contain key information from the job listing, such as:
"Hi [Recipient's Name], we noticed that your company, [Company Name], posted a job on [Job Board Name]. Good news for your company – we have a candidate who fits this description perfectly. He/she has [Qualifications]. Would you like to learn more? Please call me at [Phone Number] or schedule a meeting using a calendar invite system like Calendly. Alternatively, you can provide your phone number, and I'll contact you."

This message should be customizable as needed.

All leads should be visible in a dashboard, which includes information on reply rates and other relevant statistics.

For candidate scraping, the system should be able to extract data from candidates who are actively seeking employment or candidates in general (optional). The AI will then reach out to these candidates with potential job opportunities, matching their qualifications. If the candidate expresses interest, the AI will automatically send an email, SMS, LinkedIn message, or connection request with a personalized message.

Additionally, there should be a dashboard to monitor statistics, including reply rates and other relevant metrics.