Capstone Project Proposal

Nouv B. Al-Qahtani

Business Goals

Project Overview and Goal What is the industry problem you are trying to solve? Why use ML/AI in solving this task? Be as specific as you can when describing how ML/AI can provide value. For example, if you're labeling images, how will this help the business?	The industry problem the UML Diagram Generator aims to solve is the need for a quick and accessible tool to create and visualize UML diagrams, especially for software teams. ML/AI can provide value by: 1. Automating diagram generation from textual UML description. 2. Validating UML syntax in real-time. 3. Optimizing diagram layout and presentation. 4. Recognizing and extracting information from existing diagrams. This can increase productivity, improve diagram quality, broaden accessibility, and enable better integration with existing workflows.
Business Case Why is this an important problem to solve? Make a case for building this product in terms of its impact on recurring revenue, market share, customer happiness and/or other drivers of business success.	The UML Diagram Generator is an important business problem to solve due to its potential to: 1. Generate recurring revenue through premium features/services. 2. Capture significant market share in the growing UML diagramming tools market. 3. Improve customer satisfaction and adoption through AI-powered automation and optimization. 4. Establish a competitive advantage with unique ML/AI capabilities. 5. Enable ecosystem integration and strategic partnerships.
Application of ML/AI What precise task will you use ML/AI to accomplish? What business outcome or objective will you achieve?	1. **Automated Diagram Generation**: Use ML to translate textual UML into rendered diagrams, streamlining creation. 2. **Intelligent UML Validation**: Develop AI-powered syntax validation to ensure users create valid diagrams, reducing errors. 3. **Diagram Layout Optimization**: Employ ML to automatically optimize diagram layout, generating more readable visuals.

	4. **Diagram Recognition**: Train computer vision models to extract info from existing diagrams, enabling integration.

Success Metrics

Success Metrics What business metrics will you apply to determine the success of your product? Good metrics are clearly defined and easily measurable. Specify how you will establish a baseline value to provide a point of comparison.	1. User Adoption: Monthly Active Users, 20% quarterly growth 2. User Engagement: Avg. session, diagrams/user, 15%/25% increase 3. Paid Conversion: 10% free-to-paid in Year 1 4. Revenue Growth: 10% Monthly Recurring Revenue, 30% Annual Recurring Revenue in Year 1 5. Customer Satisfaction: Net Promoter Score of 50+

Data

Data Acquisition Where will you source your data from? What is the cost to acquire these data? Are there any personally identifying information (PII) or data sensitivity issues you will need to overcome? Will data become available on an ongoing basis, or will you acquire a large batch of data that will need to be refreshed?	• **Sources**: User-generated UML code, existing diagram repositories, technical documentation sites • **Cost**: Minimal, leveraging publicly available and user-contributed data • **PII/Sensitivity**: No PII, use anonymized/aggregated data only • **Availability**: Ongoing data streams, with periodic large dataset refreshes In summary, the data acquisition strategy focuses on cost-effective, privacy-preserving sources that provide a continuous flow of relevant UML diagram data to fuel the ML/AI capabilities of the product.
Data Source Consider the size and source of your data; what biases are built	Data source considerations for UML Diagram Generator: • **Size**: Large, diverse dataset from user-generated content and open-source repositories

into the data and how might the data be improved?	• **Source**: User-contributed UML code, technical documentation, open-source diagram libraries • **Biases**: Potential over-representation of certain programming domains/industries • **Improvements**: Targeted data collection to increase diversity, balance representation across domains
Choice of Data Labels What labels did you decide to add to your data? And why did you decide on these labels versus any other option?	Labels for UML Diagram Generator Data: • **UML Diagram Type**: (Class, Use Case, Sequence, etc.) - Enables targeted model training and generation • **Domain/Industry**: (Finance, Healthcare, Software, etc.) - Allows for domain-specific optimizations • **Complexity Level**: (Basic, Intermediate, Advanced) - Supports tailored diagram generation and layout • **Usability Metrics**: (Clarity, Conciseness, Aesthetics) - Enables training on high-quality diagram standards

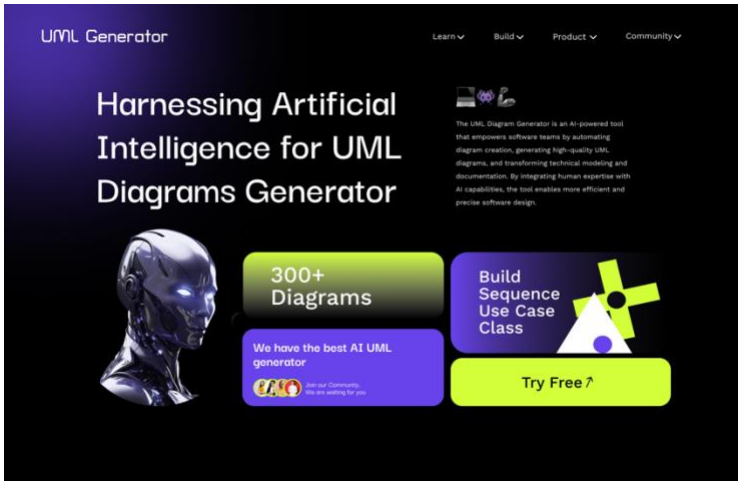
Model

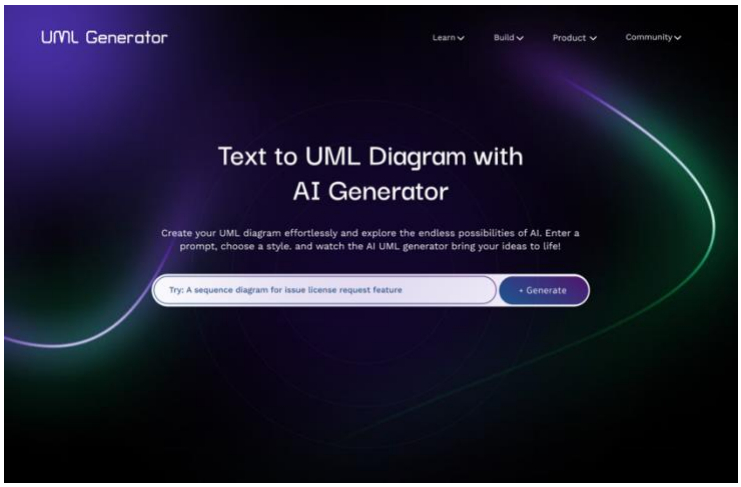
Model Building
 	
How will you resource building the model that you need? Will you outsource model training and/or hosting to an external platform, or will you build the model using an in-house team, and why?
	Model Building for UML Diagram Generator:
	• **Approach**: Leverage in-house ML/AI expertise to develop the models
	• **Rationale**:
	  ○ Maintain control over sensitive user data
	  ○ Customize models for domain-specific requirements
	  ○ Ensure seamless integration with the overall application
	• **Resources**:
	  ○ Dedicated data science team
	  ○ Access to high-performance computing infrastructure
	  ○ Ongoing model improvement and updates

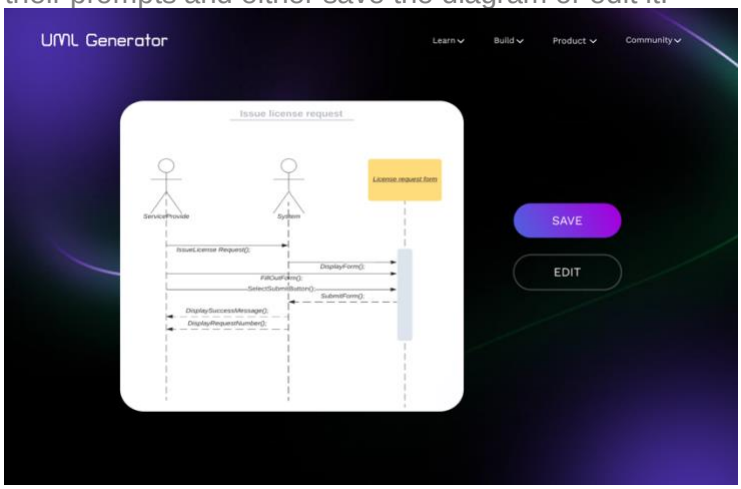
Evaluating Results Which model performance metrics are appropriate to measure the success of your model? What level of performance is required?	Evaluating UML Diagram Generator Model Performance: Key Metrics: • **Diagram Accuracy**: Measure how closely generated diagrams match expected outputs • **User Satisfaction**: Assess user feedback and ratings on diagram quality and usability • **Generalization**: Evaluate model performance on

	diverse diagram types and domains Target Performance: • **Accuracy**: 90% or higher on standard UML diagram types • **User Satisfaction**: NPS score of 70 or higher • **Generalization**: Consistent performance across a wide range of diagram complexity and domains

Minimum Viable Product (MVP)

Design What does your minimum viable product look like? Include sketches of your product.	Home Page: Here is the first screen that appears when the user opens the website.  Generating a diagram Page: Here is the screen where the user can enter their prompt to generate the desired diagram.

<table>
<tr>
<td></td>
<td> 

Result of the Diagram Page:
Here is the page where the user can see the result of their prompts and either save the diagram or edit it.
 
</td>
</tr>
<tr>
<td>Use Cases

What persona are you designing for? Can you describe the major epic-level use cases your product addresses? How will users access this product?</td>
<td>Target Persona: Software Developers, Project Managers, Technical Architects
Major Use Cases:
• Create UML diagrams from scratch
• Import and visualize existing UML code
• Collaborate on diagrams with team members
• Generate diagrams based on technical requirements
Access:
• Web-based application with responsive design
• Integrations with popular development tools and platforms</td>
</tr>
<tr>
<td>Roll-out

How will this be adopted? What does the go-to-market plan look</td>
<td>Roll-out Plan for UML Diagram Generator:

Adoption:
- Free trials, freemium model
- Onboarding support</td>
</tr>
</table>

like?	Go-to-Market: - Partner integrations, content marketing - Tiered pricing (basic, pro, enterprise) Goals: - 1,000 active users in 6 months - 10% free-to-paid conversion in Year 1 - Establish as go-to diagramming solution

Post-MVP-Deployment

Designing for Longevity How might you improve your product in the long-term? How might real-world data be different from the training data? How will your product learn from new data? How might you employ A/B testing to improve your product?	Longevity Plan for UML Diagram Generator: Improvements: - Expand diagram types, customization - Integrate with more tools/platforms - Enhance UI/UX based on user feedback Real-world vs. Training Data: - Diverse use cases, evolving standards Learning from New Data: - Continuous model retraining, active learning A/B Testing: - Experiment with layout, features - Measure impact on engagement
Monitor Bias How do you plan to monitor or mitigate unwanted bias in your model?	Bias Monitoring for UML Diagram Generator: Strategies: - Audit outputs, diversify training data - Implement bias testing frameworks Monitoring: - Track metrics, establish bias review Mitigation: - Retrain models, adjust data collection - Provide transparency on limitations