MICHIGAN STATE UNIVERSITY

Project Plan Presentation

Synthetic Image Generation via Random Noise

The Capstone Experience

Team CSAA Insurance

William Long
Matthew Baxter
AJ Bensman
Zongyuan Li
John Park
Joe Romain

Department of Computer Science and Engineering
Michigan State University

Fall 2022

From Students…
…to Professionals

Project Sponsor Overview

- Short for California State Automobile Association
- Provide home, auto, and AAA insurance to millions of Americans
- Over 2.6 million policyholders
- Operate in 23 states

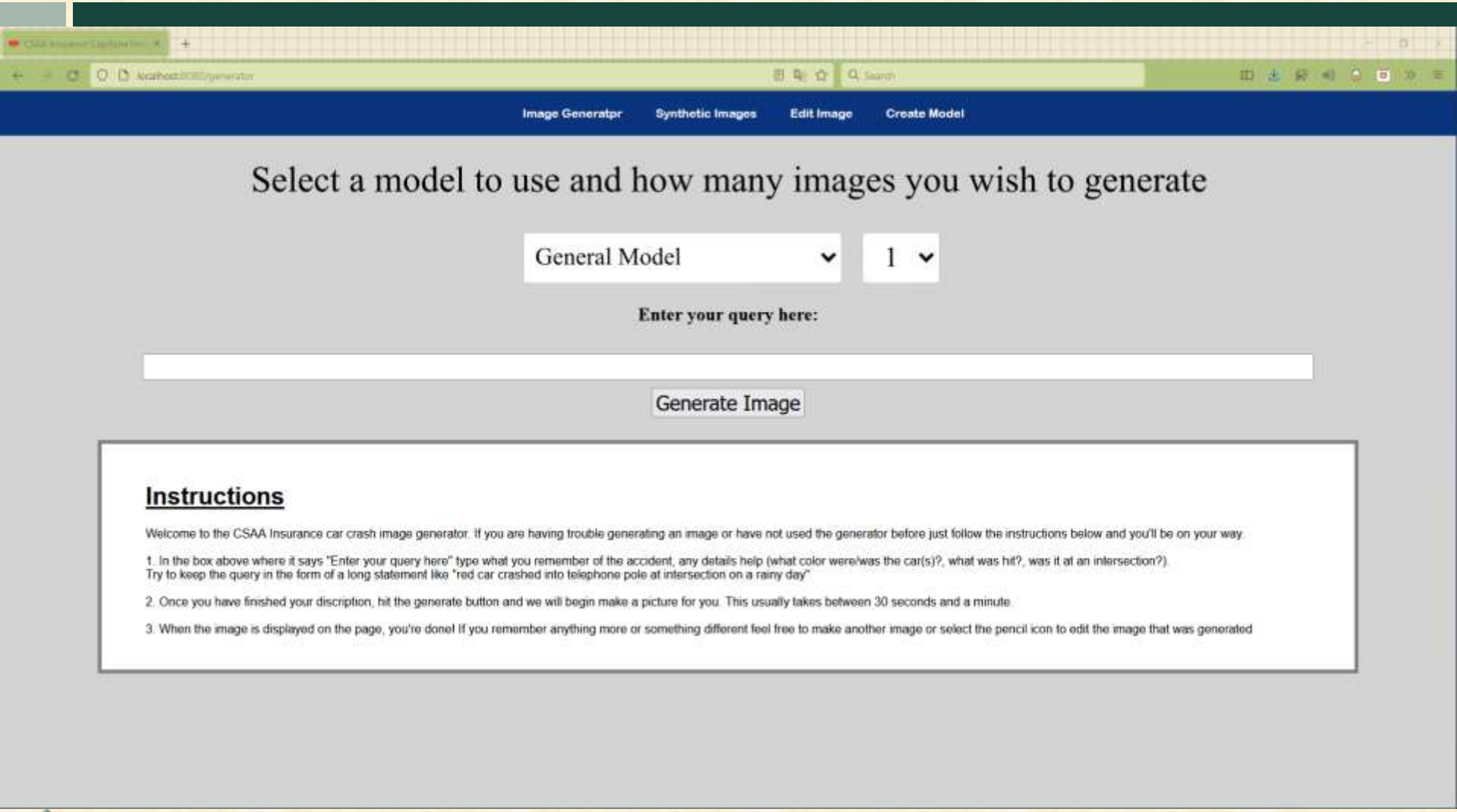
Project Functional Specifications

- CSAA has the need to generate large datasets of insurance incident images to give to various third-party partners, but they are often unable to share the images they have for privacy reasons and lack of organization
- The data is rarely ready for experimentation, manual searching is often required.
- Our project uses the power of stable diffusion models to create a tool to allow CSAA to generate synthetic image datasets of whatever subject is desired. With these models, we can generate datasets on demand with no privacy concerns.

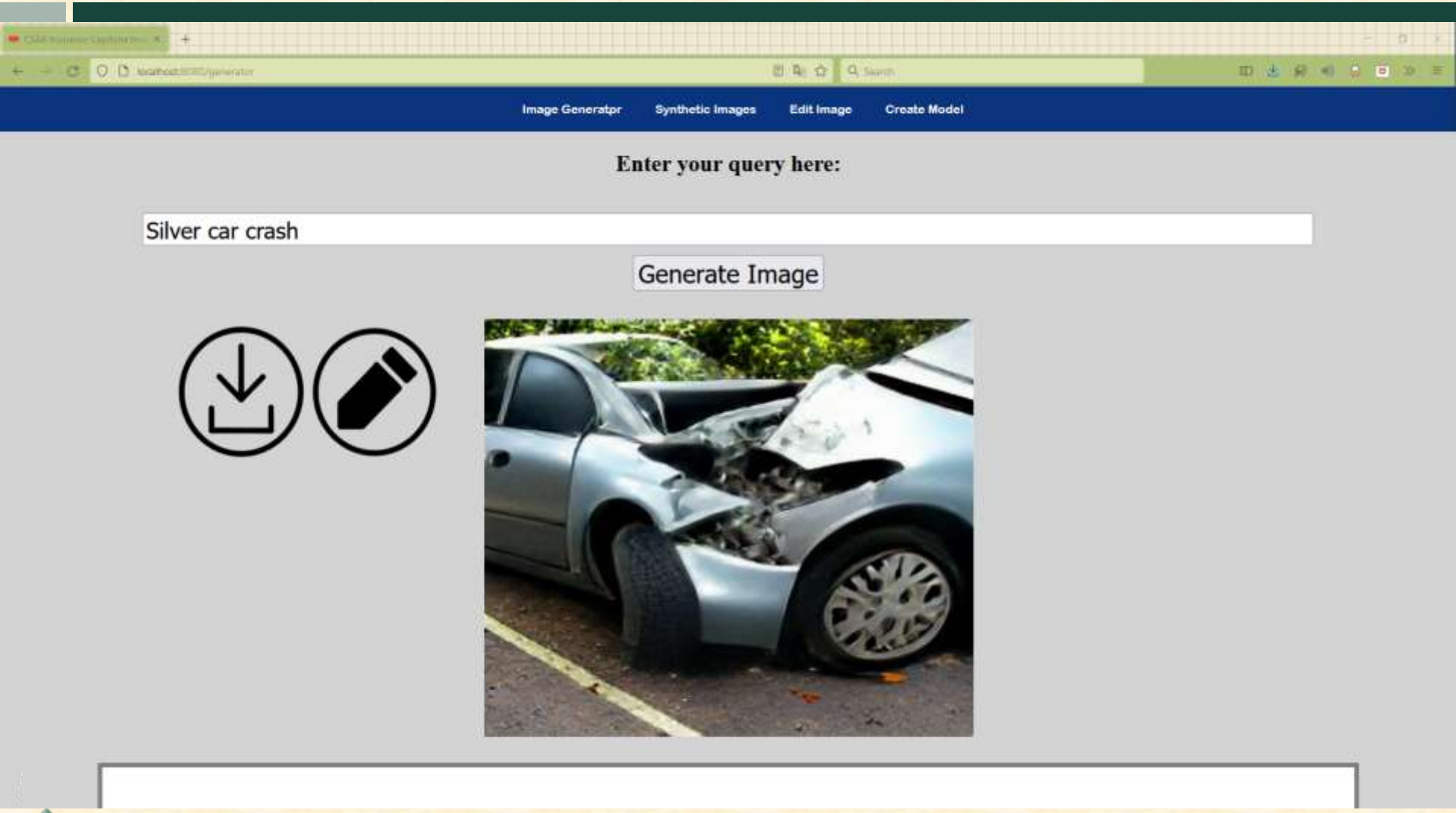
Project Design Specifications

- Synthetic Image Generation via Random Noise is a web-based application allowing users to generate images and edit those images.
- Image generator allows users to choose from a set of specialized models and, with a simple query, have that model create a number of images for them
- Models for specialized image generation can be created by users
- Defects in images can be selectively removed and regenerated with a simple image editor

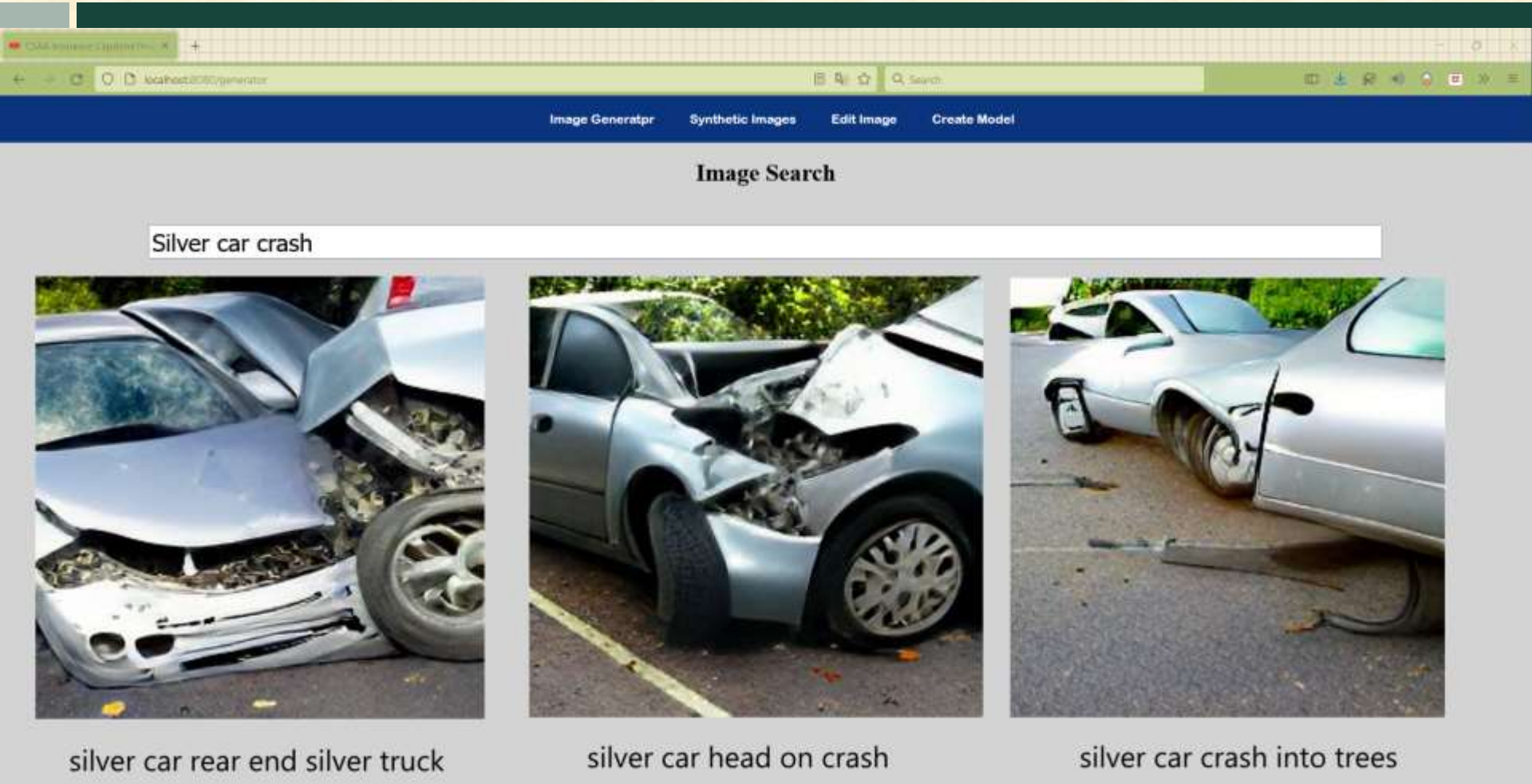
Screen Mockup: Generation Page



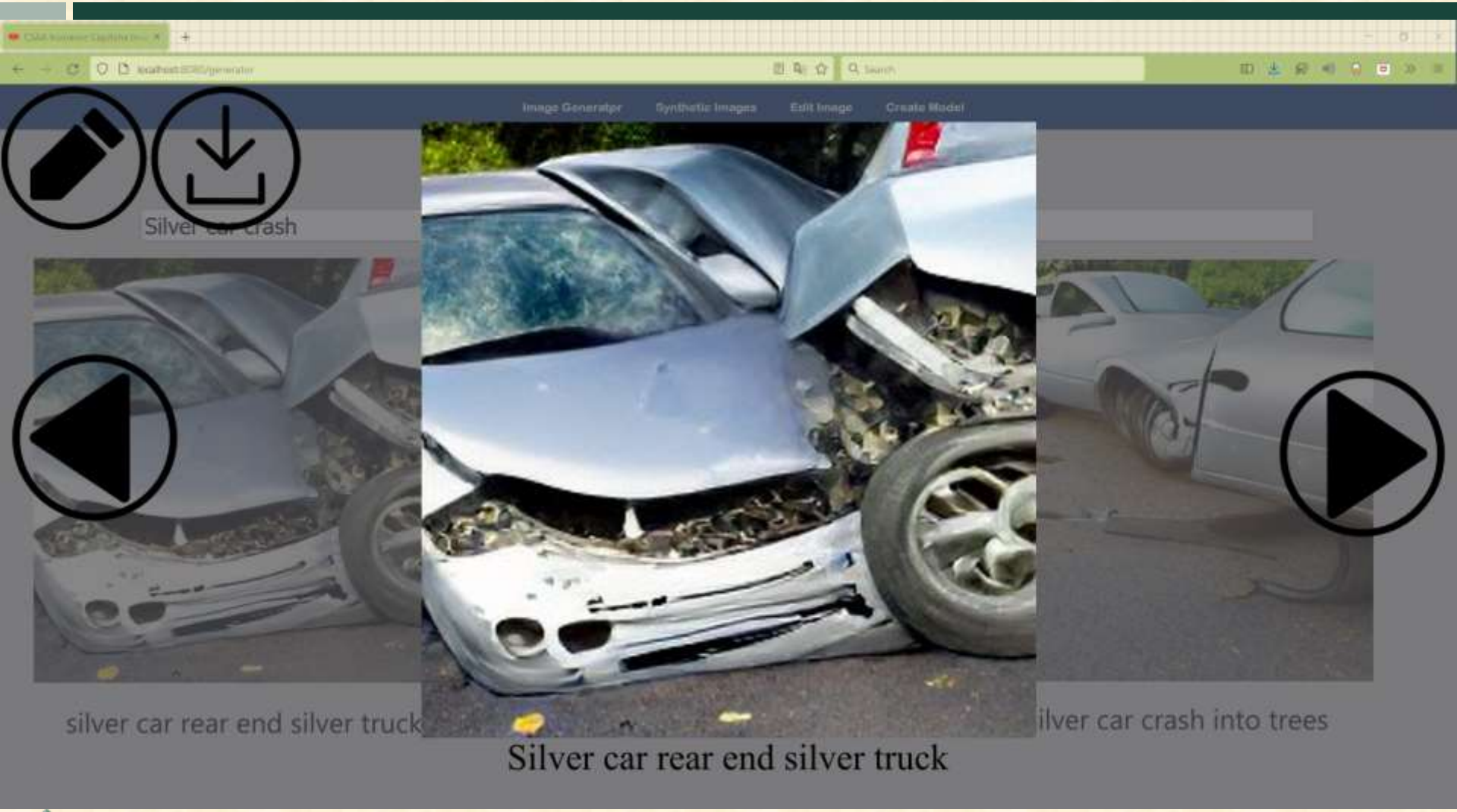
Screen Mockup: Generation page 2



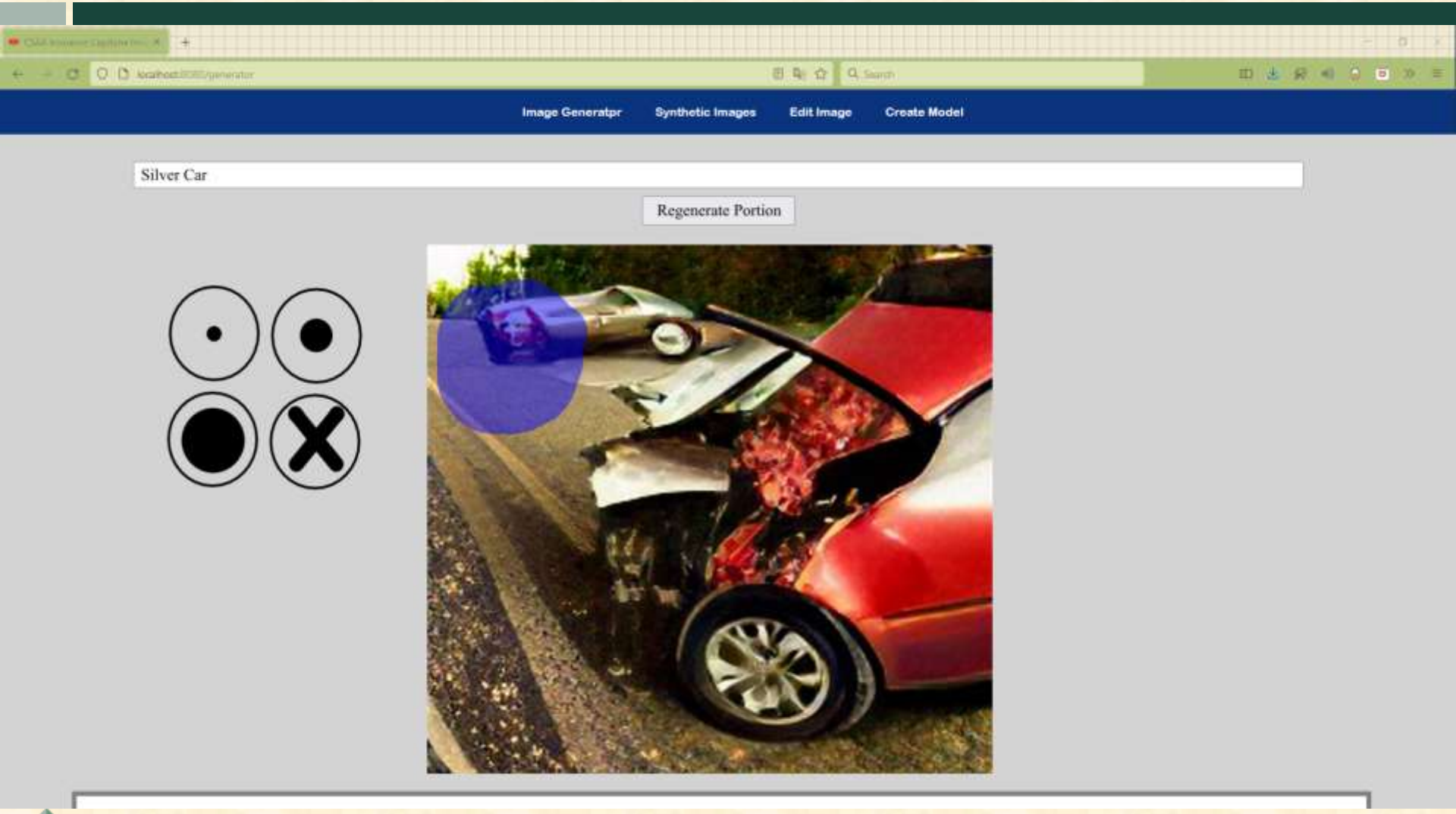
Screen Mockup: Image Gallery



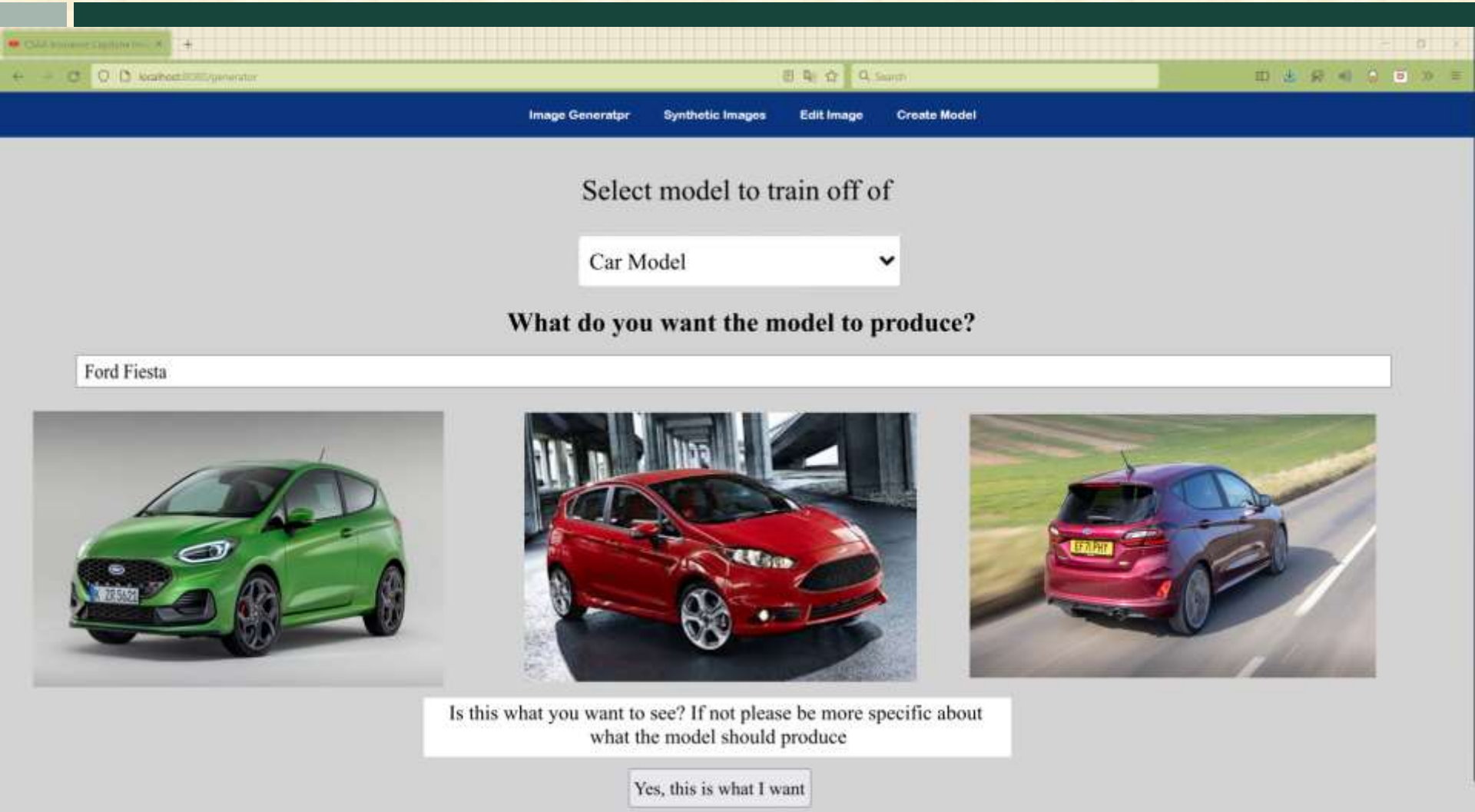
Screen Mockup: Image Gallery 2



Screen Mockup: Image Edit Page



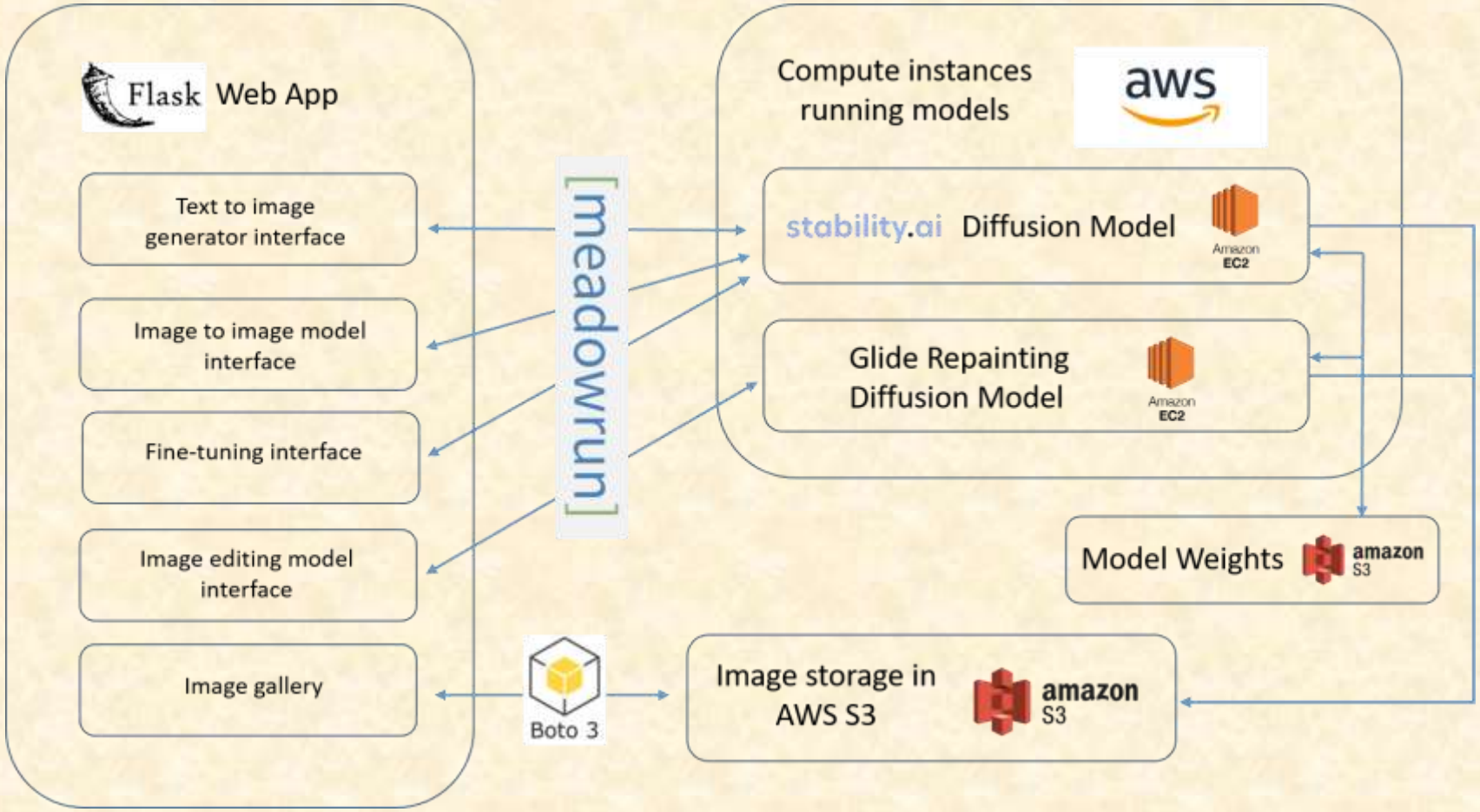
Screen Mockup: Model Creation Page



Project Technical Specifications

- Two diffusion-based models to generate and edit images from text descriptions and scratch images.
- Multiple sets of model weights in S3 that can be used depending on what type of images are desired
- A Flask web application built with Python, HTML, CSS, and JavaScript used as an access interface to models.
- Models will be held separately on AWS (Amazon Web Services) EC2 instances and initiated from users calls using Meadowrun.
- All the generated images will be stored in an Amazon S3 bucket for later use and be extracted back to web page through boto3.

Project System Architecture



Project System Components

Development Environments

- **EC2 –** AWS Cloud computing platform provides powerful GPU to train and use the model.
- **S3 Bucket –** AWS storage provides storage for generated images.
- **Google Colab –** An easy access and share platform to build and run model code.

Models

- **CLIP –** Text encoder, convert text description to image like tensor for diffusion model.
- **DDPM –** Diffusion model, generate images from random noise.
- **Palette –** Image to image translation, used to colorize or un-crop images.
- **Glide –** Partially repaint the image based on text prompts.

Libraries

- **Meadowrun –** Python SDK to sychronize code and environment on EC2 Instances.
- **Boto3 –** AWS SDK for Python used to manage AWS services.
- **Pytorch –** The machine learning framework.

Project Risks

- Compute power used by Diffusion Models
  - Diffusion models run slowly in computers because they generate and consume large amounts of data while active
  - Fine tune models using Google Collab, and run models in backend of application using AWS EC2 instances with increased GPU quotas
- Communication between AWS and Web Applications
  - Web applications need to be able to control AWS EC2 instances
  - Use Meadowrun, boto3, and AWS Lambda to manage EC2 instances
- Determining if our models generate images with sufficient accuracy
  - Using computer vision to identify if all major elements of the prompt can be identified in the image

Questions?