- STABLE DIFFUSION -

CMPE 491

High-Level Design

30.12.2022

Arda Atakol 13624005832
Serdar Hoşver 10061964142
Furkan Özelge 14758028780
Yağız Hikmet Karakuş 44164883604

URL of the webpage: https://furkanozelge.github.io/seniorstedu/

1. Introduction

1.1 Purpose of the system

The Stable Diffusion project aims to develop an AI-powered application that allows users to input text descriptions and generate images based on those descriptions using deep learning and style transfer techniques. The application will use pre-trained open-source data to generate the images and will feature user accounts, suggestions based on previous searches, and restrictions on certain types of content. The ultimate goal of the project is to introduce stable diffusion and AI to a wide user base and enable users to use their words to bring their creative visions to life using artificial intelligence.

A team of 4 developers will work on the project for 4-5 months, with the intention of creating a sustainable, high-quality application that is updated regularly to fix bugs and improve functionality. The project will also include risk management to address potential challenges and ensure its success. Our hope is to create an application that is both user-friendly and powerful, allowing people from all walks of life to experience the transformative potential of AI and stable diffusion.

1.2 Design goals

User-friendly interface: It is important to design an interface that is easy for users to understand and navigate. This can involve using clear and concise language, organizing content in a logical and intuitive way, and using visual elements such as icons and images to help convey information.

Image generation system: The system will use deep learning and artificial intelligence to generate images based on text descriptions. This will involve training a machine learning model on a large dataset of images and text descriptions and using that model to generate new images based on user input. It is important to ensure that the system is able to generate high-quality images that accurately reflect the text descriptions provided.

Scalability: The system should be able to handle a large number of users and requests without experiencing performance issues. This may involve optimizing the system for performance, implementing caching and other performance-enhancing techniques, and using a robust infrastructure to support the system.

User accounts and additional features: Implementing features such as user accounts and suggestions based on previous searches can enhance the user experience and help to keep users engaged with the application. Additionally, implementing restrictions on certain types of content (such as violence or inappropriate language) can help to ensure that the application is appropriate for a wide audience.

Reliability and stability: It is important to ensure that the system is reliable and stable and that it performs well under various conditions. This may involve testing the system under different loads and scenarios and implementing fail-safes and other measures to ensure that the system continues to function properly even if certain components fail.

Sustainability: The project should be designed with sustainability in mind, with the goal of creating a system that can be updated and improved over time. This may involve designing the system with modular components that can be easily updated or replaced, and planning for future expansions or changes to the system.

Risk management: It is important to identify and address potential risks and challenges that may arise during the development and maintenance of the project. This may involve identifying potential issues and developing contingency plans to address them, as well as implementing measures to mitigate potential risks.

1.3 Definitions, acronyms, and abbreviations

Stable Diffusion: A text-to-image model using deep learning that can be used for various tasks including inpainting, outpainting, and creating image-to-image translations directed by text prompts.

AI (Artificial Intelligence): The ability of machines to perform tasks that typically require human-like intelligence, such as learning, problem-solving, and decision-making.

Deep learning: A subset of machine learning that involves training artificial neural networks on large datasets in order to enable the system to learn and make decisions on its own.

Text-to-image model: A machine learning model that is trained on a dataset of text descriptions and images and is able to generate images based on text input.

Inpainting: The process of filling in missing or damaged parts of an image.

Out painting: The process of generating new images based on a given image or set of images.

Image-to-image translation: The process of generating a new image based on a given image, with the goal of preserving certain features or characteristics of the original image while changing others.

User base: The group of users who are using or potentially interested in using a product or service.

Sustainability: The ability of a system or process to be maintained or continued over time without degrading or depleting resources.

Risk management: The process of identifying, assessing, and prioritizing risks and developing strategies to mitigate or eliminate those risks.

Style Transfer: Style transfer is a technique used to transfer the style of one image or piece of text to another. This can be done through a variety of methods, such as neural networks or other machine learning algorithms. The resulting output is a new image or text that has the content of the original input but is styled like the reference image or text.

1.4 Overview

The goal of our project is to develop an application that uses deep learning and artificial intelligence to generate images based on text descriptions provided by users. The application, which will be based on stable diffusion, a text-to-image model that can be used for various tasks including inpainting, outpainting also, we aim to add the styles of pre-selected images to the user's input image with style transfer, and creating image-to-image translations directed by text prompts, will be user-friendly and easy to use, and will accurately and reliably generate high-quality images based on text descriptions. In addition to generating images, the application will include various features such as user accounts, suggestions based on previous searches, and restrictions on certain types of content to enhance the user experience and ensure that the application is appropriate for a wide audience.

Our overall goal is to introduce stable diffusion and AI to a wide user base and allow users to use their words to draw their dreams to life using artificial intelligence. To achieve this goal, we will develop the application over a period of 4-5 months with a team of 4 people. We will work to ensure that the system is scalable and can handle a large number of users and requests without experiencing performance issues and will aim to create a system that is reliable, stable, and performs well under various conditions. We will also design the system with sustainability in mind, with the goal of creating a system that can be updated and improved over time.

The project will be funded through a combination of internal resources and external investment, and we will work to maximize the return on investment by expanding the user base and generating revenue through the application. To ensure the success of the project, we will also implement risk management strategies to address potential challenges and mitigate potential risks.

2. Current software architecture

The Stable Diffusion architecture is a deep learning model that is designed to generate high-quality images based on text descriptions. It consists of three main components: an autoencoder, a U-Net block, and a text encoder.

The autoencoder is responsible for reducing the input sample (a random noise of the size of the desired output) to a lower dimensional latent space. It uses the VAE (variational autoencoder) architecture, which consists of an encoder and a decoder, to achieve this reduction. The encoder converts the sample into a lower latent representation, which is then passed as input to the U-Net block. On inference, the denoised, generated samples undergo reverse diffusion and are transformed back to their original dimensional latent space. The VAE architecture is effective at reducing the dimensionality of the input sample while still preserving important information, which helps to improve the accuracy and quality of the generated images.

The U-Net block, which is comprised of ResNet (Residual Network), receives the noisy sample in a lower latent space and compresses it. It then decodes the sample back with less noise, using the estimated noise residual from the U-Net output to construct the expected denoised sample representation. The U-Net block is designed

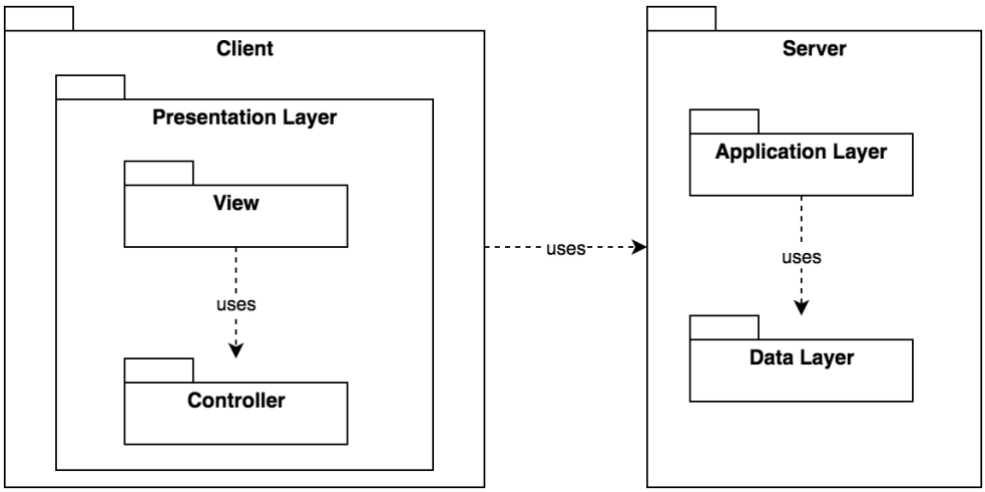
to preserve fine details and improve the accuracy of the generated images and is commonly used for image segmentation tasks such as inpainting and outpainting.

3. Proposed software architecture

3.1 Overview

The goal of the proposed project is to allow users to generate images based on text descriptions using artificial intelligence, with the aim of reaching a wide user base. To achieve this, the project will offer a limited number of image generation opportunities to each user, which will be managed through a system of registration and login. All users will be required to have an account in order to use the application. The project will also offer suggestions based on previous searches and provide suggested images to users. We will add the styles of pre-selected images to the user's input image with style transfer. In terms of technical implementation, the project will use PostgreSQL as a database, and the stack will include express, react, node, and python. The project will also create a custom pipeline for stable diffusion using a Crud API in node, which will be implemented as a separate API from python.

3.2 Subsystem decomposition

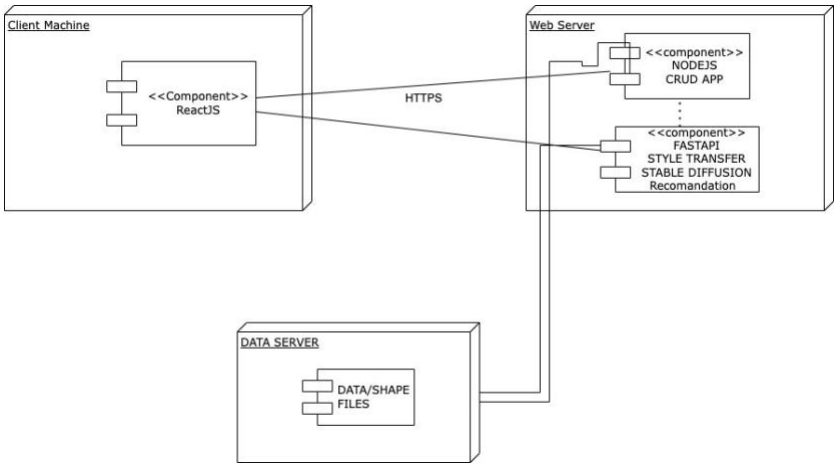


Subsystem decomposition is the process of dividing a system into smaller, more manageable components or subsystems. This can be done to make the system easier to understand and design, to simplify maintenance and repair, or to increase efficiency and performance.

In the context of the proposed project, subsystem decomposition could involve dividing the application into smaller components or modules that can be developed and tested independently but work together to achieve the overall goals of the project. For example, the application could be divided into subsystems for the login page, the text box window, and the image generation module. Each subsystem could then be developed and tested separately before being integrated into the final application.

Subsystem decomposition can help to break down a complex project into smaller, more manageable pieces, making it easier to track progress, identify and fix issues, and ensure that the final product is of high quality. It can also help to reduce the risk of problems arising from the interactions between different parts of the system.

3.3 Hardware/software mapping



3.3.1 Hardware

Computer or server: This is the physical hardware that will host the application and the Stable Diffusion model. It could be a desktop or laptop computer, or a dedicated server.

Operating system: This is the software that controls the hardware and provides a platform for the application and the Stable Diffusion model to run on. Examples include Windows, MacOS, or Linux.

3.3.2 Software

Programming language: This is the language that will be used to develop the application and the Stable Diffusion model. Examples could include Python.

Libraries and frameworks: These are pre-built software components that can be used to simplify the development process and provide additional functionality. Examples could include TensorFlow or PyTorch for the Stable Diffusion model, and React for the application.

Database: This is the software that will be used to store and manage the data generated by the application and the Stable Diffusion model. Examples could include PostgreSQL.

This is just one example of a hardware and software mapping diagram for the Stable Diffusion project. The specific hardware

3.4 Persistent data management

Persistent data management refers to the process of storing and managing data that needs to be preserved over a long period of time, even when the system or application that uses the data is not running.

In the context of the proposed Stable Diffusion project, persistent data management could involve storing and managing the data that is used to train the Stable Diffusion model, as well as the data that is generated by the model and the application. This could include things like the text descriptions that users enter, the generated images, and any other data that needs to be preserved and accessed over time.

To manage persistent data effectively, the project will likely need to use a database or other data storage system that can store and organize the data in a way that is easily accessible and searchable. The system will also need to have mechanisms in place to ensure the integrity and security of the data, such as backup and recovery procedures and data protection measures.

Overall, effective persistent data management is critical for the success of the Stable Diffusion project, as it ensures that the data that is used and generated by the application is properly stored and organized and can be accessed and used as needed.

3.5 Access control and security

Access control and security refer to the measures that are put in place to ensure the confidentiality, integrity, and availability of the data and systems used in the Stable Diffusion project.

In the context of the proposed project, access control and security measures could include things like authentication and authorization mechanisms to ensure that only authorized users can access the application and the data it generates, as well as data protection measures to ensure the confidentiality and integrity of the data. This could include things like encryption, backup and recovery procedures, and firewalls to protect against external threats.

Additionally, the project may need to consider security measures to protect against internal threats, such as employees who may have access to sensitive data or systems.

Overall, access control and security are critical considerations in the Stable Diffusion project, as they help to ensure the confidentiality, integrity, and availability of the data and systems used in the project and protect against external and internal threats.

3.6 Global software control

Global software control refers to the process of managing and coordinating the various software components and processes that are involved in a project. This can include things like version control, configuration management, and integration testing, as well as processes for managing changes and updates to the software.

In the context of the proposed Stable Diffusion project, global software control could involve establishing processes and tools to manage the various software components and processes involved in the project, including the Stable Diffusion model, the application, and any other software tools or libraries that are used. This could involve using version control systems to track changes to the software and manage different versions of the code, as well as configuration management systems to manage the various configurations and settings that are used in the project.

Global software control can help to ensure that the various software components and processes involved in the project are properly coordinated and work together effectively and can also help to ensure the quality and reliability of the final product. It is an important aspect of any software development project and is essential for the success of the Stable Diffusion project.

3.7 Boundary conditions

Boundary conditions refer to the constraints and limitations that are placed on a project or system. These can be external factors, such as regulatory requirements or technical limitations, or internal factors, such as project goals or resource constraints.

In the context of the proposed Stable Diffusion project, boundary conditions could include things like regulatory requirements related to the use of artificial intelligence and deep learning, technical limitations of the hardware and software that are used in the project, or resource constraints such as budget or time. Other potential boundary conditions could include limitations on the types of text descriptions that can be used to generate images or limitations on the number of images that can be generated by each user.

Understanding and managing boundary conditions is an important aspect of any project, as it helps to ensure that the project stays within the constraints and limitations that are imposed on it and helps to prevent the project from veering off course or encountering unexpected challenges.

4. Subsystem services

Login and authentication: This subsystem would handle the process of verifying user credentials and granting access to the application. It could include features such as password protection and two-factor authentication.

Text input and image generation: This subsystem would be responsible for accepting text descriptions from users and using the Stable Diffusion model to generate images based on the descriptions. It could also include features such as image tagging and categorization to help users search for and find specific images.

Data storage and management: This subsystem would handle the storage and management of the data generated by the application, including text descriptions and generated images. It could use a database or other data storage system to store and organize the data in a way that is easily accessible and searchable.

User account management: This subsystem would handle the creation and management of user accounts, including features such as password reset and account deletion. It could also include features such as user preferences and account history to track user activity and suggest images based on previous searches.

Security and access control: This subsystem would be responsible for implementing security measures to protect against external and internal threats and ensure the confidentiality, integrity, and availability of the data and systems used in the project. This could include things like encryption, backup and recovery procedures, and firewalls.

Global software control: This subsystem would handle the overall management and coordination of the software components and processes used in the project, including things like version control, configuration management, and integration testing.

These are just a few examples of potential subsystem services that could be included in the Stable Diffusion project. The specific subsystem services that are included will depend on the needs and goals of the project.

5. Glossary

User interface: The interface that users interact with when using the application, typically consisting of graphical elements such as buttons, menus, and text fields.

Deep learning: A subset of machine learning that involves training artificial neural networks on large datasets in order to enable the system to learn and make decisions on its own.

Text-to-image model: A machine learning model that is trained on a dataset of text descriptions and images and is able to generate images based on text input.

Pre-trained open-source data: Data that has been previously trained on a large dataset and made available for others to use for their own projects.

User accounts: Accounts that allow users to create a personalized profile and access features such as saved searches and history.

Suggestions based on previous searches: Recommendations for similar or related content based on a user's previous search history.

Restrictions on certain types of content: Limits on the types of content that can be accessed or displayed within the application, often implemented in order to ensure that the content is appropriate for a wide audience.

Modular components: Components of the system that can be easily updated or replaced without affecting the overall functionality of the system.

Effective data management and storage: Critical for the success of the Stable Diffusion project, as it ensures that the data that is used and generated by the application is properly stored and organized and can be accessed and used as needed.

Effective system monitoring and maintenance: Critical for the success of the Stable Diffusion project, as it ensures that the application and model are operating efficiently and effectively and that any issues or problems are identified and addressed in a timely manner.

6. References

6.1 https://dataconomy.com/2022/09/stable-diffusion-ai-art-generator/

6.2 https://www.howtogeek.com/830179/how-to-run-stable-diffusion-on-your-pc-to-generate-ai-images/#autotoc_anchor_1