

**Software Engineering Department  
 Braude College**

Capstone Project Phase B

23-2-D-19

**VidWizard: Generate Smart Video Content Using AI**

A picture containing font, text, graphics, graphic design

Description automatically generated

**Supervisors:**

Alexander Keselman

**Authors:**

Shaked Arish  
Sean Friedman

**Table of Contents**

[1. Abstract 4](#_Toc166026189)

[2. Introduction 5](#_Toc166026190)

[**2.1 Motivation** 5](#_Toc166026191)

[**2.2 Overview** 5](#_Toc166026192)

[3. Background & Related Work 6](#_Toc166026193)

[**3.1 Artificial Intelligence** 6](#_Toc166026194)

[**3.2 Machine Learning** 7](#_Toc166026195)

[**3.3 API** 7](#_Toc166026196)

[**3.4 AI Tools…………………………………………………………………………………8   
3.4.1 Text-To-Speech (TTS)** 8](#_Toc166026197)

[**3.4.2 Text-To-Video (TTV)** 8](#_Toc166026198)

[**3.4.3 ChatGPT** 9](#_Toc166026199)

[**3.5.1 Bias In AI Systems** 10](#_Toc166026200)

[**3.5.2 Unfairness examples in AI systems** 10](#_Toc166026201)

[**3.6 Pexels** 10](#_Toc166026202)

[**3.7 Node.JS** 11](#_Toc166026203)

[**3.9 FFmpeg Library** 11](#_Toc166026204)

[**4.1 Process** 13](#_Toc166026205)

[**4.1.2 Create New Video Project** 13](#_Toc166026206)

[**4.1.3 Integrate User Input with Pre-Made Form.** 13](#_Toc166026207)

[**4.1.4 Present The Generated Script & Edit options** 15](#_Toc166026208)

[**4.1.5 Generate Voiceover** 15](#_Toc166026209)

[**4.1.6 Generate Subtitles** 15](#_Toc166026210)

[**4.1.7 Pexels API for background videos** 16](#_Toc166026211)

[**4.1.8 Edite and Merge** 16](#_Toc166026212)

[**4.1.9 Video Is Ready** 16](#_Toc166026213)

[**4.2 Methodology and Development process** 16](#_Toc166026214)

[**4.3 Constraints and challenges** 17](#_Toc166026215)

[**4.3.1 Integration with AI Tools and APIs** 17](#_Toc166026216)

[**4.3.2 User Input Variability** 18](#_Toc166026217)

[**4.3.3 Real Time performance** 18](#_Toc166026218)

[**4.3.4 Quality Control and Filtering** 18](#_Toc166026219)

[**4.3.5 Accuracy and Quality of Generated Content** 18](#_Toc166026220)

[5.Product 19](#_Toc166026221)

[**5.1 Requirements** 19](#_Toc166026222)

[**5.2 Diagrams** 20](#_Toc166026223)

[**5.2.1 Use Case** 20](#_Toc166026224)

[**5.2.2 Activity Diagram** 21](#_Toc166026225)

[**5.2.3 Package Diagram** 22](#_Toc166026226)

[**5.3 Development** 22](#_Toc166026227)

[6. Results and conclusion 23](#_Toc166026228)

[7. Verification and Evaluation 23](#_Toc166026229)

[**7.1 Testing Plan** 23](#_Toc166026230)

[**7.1.1 Frontend Testing** 23](#_Toc166026231)

[**7.1.2 Backend Testing** 25](#_Toc166026232)

[**7.2 Evaluation** 26](#_Toc166026233)

[9.References 27](#_Toc166026234)

# **1. Abstract**

In today's digital age, the ability to create captivating and interesting content is crucial, especially on popular social media platforms like Twitter, Facebook, and Instagram. Whether it's for marketing, education, or entertainment purposes, social media become a powerful tool for individuals, businesses, and organizations in order to succeed in the digital world.

The rapid advancement of artificial intelligence (AI) technology has opened up new possibilities for content creation. In this project, we present an innovative solution that leverages the power of AI tools to generate smart content, particularly short videos,  and integrate it into a user-friendly and accessible platform. Through the integration of multiple AI technologies and APIs, our platform streamlines the content creation process, eliminating the need for extensive manual production and technical skills. Users can simply input their preferences, topic, and desired visual elements, and our platform will take care of the rest. From generating customized scripts to synthesizing voiceovers and incorporating visual elements, our platform ensures a tailored and engaging video creation experience.

We address the limitations of existing approaches by offering a unique combination of automation and customization. Users have the freedom to personalize and edit the AI generated content throughout the process, making the final video truly unique and aligned with their vision.

Our goal is to make video content generation accessible to a wider audience, regardless of their video editing skills and technical expertise. empowering them to share their ideas and stories effectively through the help of AI powered technologies.

# **2. Introduction**

The effect of high-quality content in social media becomes crucial for businesses and individual people alike, in order to attract and engage audiences and ultimately boost their exposure. There are several approaches to generating smart content, each with varying levels of effectiveness and user-friendliness. In this introduction, we will explore two common methods: manual video production and template-based automated systems. These methods have their advantages and limitations, highlighting the need for a more innovative solution.

## **2.1 Motivation**

Manual video production has been the traditional approach for content creators, offering full creative control but requiring extensive time, resources, and technical expertise. On the other hand, template-based automated systems provide a simplified process but often result in generic and repetitive videos that fail to meet user preferences. These limitations call for a

transformative solution that combines the power of AI with a user-friendly interface.

## **2.2 Overview**

Our project aims to change the process of generating smart content by developing a platform that harnesses the power of AI while maintaining creative control and ease of use.

Through our platform, by just give a description of the video you want to create, users can easily express their content topic, preferences, and requirements. The information provided by the user through the form will serve as the foundation for generating the video, by integrating the user input with a precise, pre-built template that leverages advanced AI tools, our system generates a script of the intended video, offering the users selection of voices for voiceover and find relevant visuals for it.

This innovative approach bridges the gap between technical expertise and content creation, enabling individuals from all backgrounds to produce high-quality video content with ease with the help of AI tools.

# **3. Background & Related Work**

## **3.1 Artificial Intelligence** “Artificial intelligence (AI), the ability of a digital computer or computer-controlled robot to perform tasks commonly associated with intelligent beings. The term is frequently applied to the project of developing systems endowed with the intellectual processes characteristic of humans, such as the ability to reason, discover meaning, generalize, or learn from past experience.” [1]

Artificial intelligence (AI) describes the development of computer systems that are capable of handling tasks that traditionally require human intelligence. Computer science, mathematics, psychology, neuroscience, and other related fields are all combined in AI.

AI seeks to develop intelligent machines with human-like capacities for perception, learning, and problem-solving. The objective is to create systems that can comprehend and analyze natural language, spot trends, make judgments, learn, solve problems, and adapt themself.

Narrow AI and General AI are the two main subtypes of AI. Weak AI, also known as narrow AI, describes systems that are created to carry out functions inside specified domains, such as voice assistants and picture recognition systems. These AI systems are exceptional at what they are taught to do, but they are limited in their ability to generalize.

In comparison, general AI, usually referred to as strong AI, describes AI systems that can comprehend, learn, and apply knowledge across several domains and tasks, much like human intelligence. The goal of strong AI is to create machines that can reason and think like people.

תמונה שמכילה טקסט, עיגול, גופן, צילום מסך

התיאור נוצר באופן אוטומטיתמונה שמכילה טקסט, תרשים, צילום מסך, גופן

התיאור נוצר באופן אוטומטי

Figure 1: AI subfields

Figure 2: The combination of AI Generated Content in other fields

## **3.2 Machine Learning**

“ML is a field devoted to understanding and building methods that let machines "learn" – that is, methods that leverage data to improve computer performance on some set of tasks”.[2]  
  
Machine learning is a subcategory of artificial intelligence that focuses on enabling computers to learn and make predictions and decisions without being programmed to do it. It is a field that includes developing algorithms and models that learn patterns and relationships from the data. Machine learning can be separated into 4 main types:

* Supervised Learning
* Unsupervised Learning
* Semi Supervised Learning
* Reinforced learning

**Supervised learning:** The model is trained on labeled data.

**Unsupervised learning:** The model is trained on unlabeled data.

**Semi-supervised learning:** The model is trained on a mix of labeled and unlabeled data.

**Reinforced learning:**  involves training an agent to communicate with an environment and learn optimal actions based on rewards or feedback.

The difference between supervised and unsupervised learning is that in unsupervised learning the output data is not given. The learning process occurs by using the relation and connection in the data. Moreover, unsupervised learning doesn't have training data.  
Machine learning can be used in different applications, including image and speech recognition, natural language processing, recommendation systems, fraud detection, autonomous vehicles, and many more.

**3.3 API**  
"An Application Programming Interface (API) is a way for two or more computer programs to communicate with each other. It is a type of software interface, offering a service to other pieces of software." [3]

APIs are common in the world of software development. They are the communication feature that enables software applications to interact and exchange data with each other. We can find APIs in use in web-based systems, operating systems, database systems, and even the hardware in our computers and smartphones. For example, whenever we sign into a social media app on our phone, it's an API that retrieves the login information from the server. For developers, APIs simplify the process of integrating features into their software. Instead of building functionality from the start, developers can simply call the API and use the output results as they wish in their application.

## **3.4 AI Tools 3.4.1 Text-To-Speech (TTS)** Text-to-Speech (TTS) converts written text into spoken words, new technologies in the field make use of deep learning and artificial intelligence among other advanced techniques to capture linguistic patterns and generate high quality, fluency, and natural sounding speech. TTS systems usually offer the user a voice selection, allowing users to choose from a variety of voices that represent different genders, ages, and accents according to their needs and desire, resulting in a more personalized and engaging speech. The applications of TTS technology are vast, it plays a crucial role in various domains, ranging from accessibility for individuals with visual impairments to multimedia content creation and language learning tools.

**תמונה שמכילה צילום מסך, טקסט

התיאור נוצר באופן אוטומטי**

Figure 3: Overview of TTS System

### **3.4.2 Text-To-Video (TTV)**

“A text-to-video (TTV) model is a machine learning model which takes as input a natural language description and produces a video matching that description.”[5]

Text-to-video tools streamline the time-consuming aspects of video production, making it effortless for users to generate engaging video content and opening up a world of possibilities. While text-to-video model innovative technology has gained significant traction in recent years, thanks to advancements in AI algorithms and machine learning techniques, the field is still in its early stages, with ongoing research and advancements driving its evolution. Access to text-to-video generation technology is currently limited, and the quality of the generated videos may not always meet the expectations of the users.

**תמונה שמכילה טקסט, צילום מסך, תרשים, מקביל

התיאור נוצר באופן אוטומטי**

Figure 4: Architecture of Text-To-Video Generation System

**3.4.3 ChatGPT**Is an advanced language model developed by OpenAI. As one of the most powerful language models available. It operates by utilizing deep learning algorithms to understand and process natural language inputs. It has been trained on an extensive array of text sources, including books, articles, websites, and other textual materials, allowing it to have a broad knowledge base on various subjects up until its knowledge cutoff date in September 2021. His primary goal is to engage in interactive conversations with users, offering informative and coherent responses to their queries. It is designed to comprehend and generate human-like text, providing assistance, answering questions, and offering suggestions based on the input it receives. ChatGPT has made a remarkable impact on the world, particularly in the field of artificial intelligence and natural language processing.

One of the notable breakthroughs of ChatGPT is its ability to generate human-like text responses. It has learned to understand and generate coherent and contextually relevant language. This breakthrough has opened up new possibilities for human-like conversations and interactions with AI systems.

Furthermore, Open Ai's introduction of the GPT-3.5 API has allowed developers and businesses to integrate ChatGPT into their own applications and services seamlessly. This accessibility has empowered a wide range of industries and individuals to leverage the power of ChatGPT for their specific needs. The impact of ChatGPT goes beyond its technical advancements. It has sparked widespread interest and engagement from researchers and developers especially in the field of AI ethics and fairness.  
  
**3.5 AI – Ethics**

“At the 2018 ACM Conference on Fairness, Accountability, and Transparency , an associate professor at Princeton University gave a talk outlining 21 different definitions of fairness. This goes to show how nuanced and context-sensitive the concept of “fairness” is. its definition differs across cultures, across societies, and even over time.”[6]

There is no single definition of fairness. When we're talking about AI systems, we need to think carefully about what fairness means in that context. If we don't carefully consider fairness throughout the entire process of building an AI system, it could unintentionally produce unequal outcomes for people from different gender, race, religion etc.

### **3.5.1 Bias In AI Systems**

Bias can enter your AI system at different stages of our development, starting from the diversity of the team working on it and how they gather and classify data, to the specific dataset chosen and the goals set for the system.

### **3.5.2 Unfairness examples in AI systems**

* In a paper by Joy Buolamwini and Timnit Gebru revealed that facial analysis algorithms misclassified darker-skinned females while achieving high accuracy for lighter-skinned males. The darker-skinned females saw error rates of up to **34.7%**, whereas the error rate among lighter-skinned males was **only 0.8%**. The training dataset didn’t contain enough examples of darker-skinned females for the model to perform well for that group.[7]
* in 2018, Amazon's hiring software was criticized for unfairly downgrading job applications that mentioned the word "women's”.[8]
* Apple's credit card algorithm showed a clear bias against women, this was evident when Steve Wozniak, the co-founder of Apple, discovered that his wife was given a credit limit that was 10 times lower than his, even though they shared all their financial resources and accounts. [9]

**3.6 Pexels**Was founded in 2014 by twin brothers Ingo and Bruno Joseph in Fuldabruck, Hesse. It's become a go-to platform for designers, content creators, and marketers and offers a vast collection of over 3.2 million free stock photos and videos. The website provides high-quality visual assets for personal and commercial use without the need for attribution. Pexels' library covers a wide range of categories, styles, and themes, ensuring suitability for various projects. All content on Pexels is licensed under the Creative Commons Zero (CC0) license, allowing free usage, modification, and distribution. It offers a user-friendly interface with easy browsing and searching capabilities. Pexels is a convenient and reliable resource for accessing visually appealing content without licensing fees or attribution requirements. Its commitment to free usage makes it valuable for individuals and businesses alike. In our project we can make a request to the Pexels free API and use its huge library of images and videos for our benefit.

**3.7 Node.JS**Is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting and running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts. Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications.  
For that reasons we chose to use Node.js EXPRESS as our server-side application.  
  
**3.8 Firebase Framework**Firestore, part of the Firebase platform, is a flexible, scalable database for mobile, web, and server development. It provides real-time data synchronization and robust querying capabilities. firestore is designed to easily store and sync app data at a global scale, making it an excellent choice for our project. In our Firestore database, we have 1 collection named "Users", each doc in this collection is unique user data that contains: mail, password, and username.

**3.9 FFmpeg Library**Is a powerful open-source library designed for processing multimedia content such as audio, video, and even subtitles. It supports transcoding, streaming, and playback across multiple formats, making it indispensable for developers and content creators who need to manipulate video files efficiently.  
That is why we choose to use this library for video editing. **3.10 SweetAlert2 Library**SweetAlert2 is a customizable, promise-based JavaScript library that provides an easy way to display pop-up alerts, confirmations, and prompts in web applications. It offers a rich set of features for styling and animating dialogs, making it a versatile choice for enhancing user interaction and feedback mechanisms in nowadays web development. We use this library for success, error, and warning notifications to the user in style.  
  
**3.11 POSTMAN**is a powerful collaboration platform for API development that can simplifies testing and monitoring of API's. Postman simplifies each step of the API lifecycle and streamlines collaboration so you can create better API's.  
We chose to use that platform for testing and better understanding our server responses for different entries. It simplifies our server development process and improve our server performance.  
  
**3.12 Video Content on Social Media**Video content in particular has captured the attention of social media users due to its visual appeal and ability to convey information effectively. Short videos especially have gained popularity as they provide concise and easily consumable content that fits well within the fast-paced nature of social media like TikTok, YouTube and etc. [11]

תמונה שמכילה טקסט, צילום מסך, גופן, מספר

התיאור נוצר באופן אוטומטי**5. Engineering Process**In this section we will cover the main steps we identified in our project, as well as the potential challenges that we encounter during the engineering process of the web application.

Figure 5: System Architecture

**4.1 Process  
  
4.1.1 Website Page**  
Our website serves as the gateway to our application, we aim to develop a user interface that is intuitive and easy to navigate, even for users without any previous experience or technical expertise. To create a welcoming atmosphere on the website, we applied a clean and modern design approach with strategically placed elements. that approach helps to achieve a visually appealing interface that avoids overwhelming the user. The design consistent throughout all the video creation steps, ensuring that users can seamlessly progress from one step to the next. In order to optimize the process of developing the website we will use tools such as Tailwind CSS and React.

#### **4.1.1.1 React**

React is an open-source, frontend, JavaScript library for building user interfaces or UI components. It is maintained by Facebook and a community of individual developers and companies. React can be used as a base in the development of single-page or mobile applications. React  allows developers to create reusable components and efficiently update them based on data changes. It uses a virtual DOM for fast rendering and follows a declarative approach and enables the creation of dynamic and interactive UIs.  
  
**4.1.1.2 Tailwind CSS**

Tailwind is a free CSS framework for building custom user interfaces directly in HTML by their class name. It gives flexibility and control and allowing rapid customization without writing CSS files. Also automatically removing unused styles. Tailwind is a popular choice for web projects.

### **4.1.2 Create New Video Project**

The first phase in creating a video project involves collecting user input. Users are prompted to provide a description for their video, which serves as the foundation for the project. Key details gathered include:

* Topic or theme: Users specify the subject or theme around which they want their video to revolve.
* Keywords or text: Users can input keywords and text, aiding in the creation of a script tailored to their needs.
* Desired length: Users can indicate their preferred duration or approximate length for the video.

### **4.1.3 Integrate User Input with Pre-Made Form.**

Once the user provides their personalized details through the user information form, the next step is to integrate that input with a pre-prepared template form that allows us to harness the power of ChatGPT AI language model to generate an engaging script tailored to the user's preferences. The pre-prepared template must be effective and fine-tuned in order to get a high-quality script that leverages ChatGPT capabilities and do so for every user input.

**4.1.3.1 OpenAI API**To use the OpenAI API effectively its essential to follow a systematic approach that ensures seamless and efficient integration, the following steps outline the Process. Our first step is to create an API key from OpenAI. This key is a unique access code to enable secure communication and authentication with the API.

תמונה שמכילה טקסט, צילום מסך, תוכנה, דף אינטרנט

התיאור נוצר באופן אוטומטי

Figure 6: Generating an API Key

After obtaining your API key, the next step is to configure the chat model for the API call. In this step, we set up the necessary parameters to interact with the OpenAI API.

תמונה שמכילה טקסט, צילום מסך, תוכנה

התיאור נוצר באופן אוטומטי

Figure 7: OpenAI API request

For the final step we need to extract the relevant data from the API response. This data can be further processed, analyzed, and transferred to other AI tools. Also, we can use a JSON file that includes the necessary information for the ChatGPT API request, such as the model to use, the role and content of each message, and any other relevant parameters for your specific use case.

**תמונה שמכילה עיגול, טקסט, תרשים, גופן

התיאור נוצר באופן אוטומטי**

Figure 8: Interaction between ChatGPT and User

### **4.1.4 Present The Generated Script & Edit options**

After we generate the script by using ChatGPT, we present it to the user and provide them with the option to review, edit and make changes as desired.

The script will be displayed in a clear and readable format, with a user-friendly editing interface that allows the users to refine the script, make adjustments, add specific details, or rewrite sections entirely. By providing this editing capability, we empower users to customize the script to their liking and ensure it aligns perfectly with their requirements. When the user done editing the script and submit it, the script sends again to ChatGPT for second generation after the user change and also for exam the user changes if it's against the policy.

### **4.1.5 Generate Voiceover**

After the user confirm the edited script, we will start to integrate with Azure AI text-to-speech API to convert the final script into high-quality voiceover for the video, add a human-like voiceover to the visual content, enhancing the overall impact and engagement of the final video. The users can select from a range of voice options, and genders. Being able to hear the different voiceover helps the user choose the exact voice they want for their video. Although there are many TTS technologies that offer APIs we can integrate with, we choose to use the Azure text-to-speech API because its suits our requirements well, both in terms of performance and cost.

### **4.1.6 Generate Subtitles**

Once the voiceover is generated, we use Azure Speech-to-text API to create accurate subtitles from the audio file. This process ensures that the subtitles are perfectly sync with the voiceover. The subtitles are automatically generated in real time, this feature makes the video more accessible to a wider audience and refine the final result.

### **4.1.7 Pexels API for background videos**

Pexels is a popular online platform that provides a vast collection of high-quality videos and images, at first, we use OpenAI API in order to analyze the user's final script and identify several key scenes in it. Then, by understanding the context of each scene, ChatGPT generate relevant keywords that encapsulate the desired visual elements. These keywords will be used to search for suitable videos that align with the script and overall narrative, from an extensive database of Pexels, also we have a backup more generic keyword for each scene if the primary keywords didn’t come up with a video. Moreover, we are filtering the video search result by the length, and resolution of each video for better performance.

### **4.1.8 Edite and Merge**

With the help of FFMPEG library we edit the videos. Starting with removing the sound stream and scaling the resolution to our desire resolution of each video (if needed), then we cut each video to the same duration time that has been set by our algorithm and merging those videos into a complete one while maintaining their order. Finally, we are adding on top of the complete video the voiceover, background music and the subtitles to create a cohesive and captivating high-quality video that truly brings the user's vision to life.

### **4.1.9 Video Is Ready**

Once the video editing is complete, we render the final video file, and presenting the user the final video. The user can ultimately download or sharing the final video in social media such as: Facebook, Twitter, Instagram, Email, WhatsApp, and Telegram. In case the user is unsatisfied with the generated video he can always go back to the beginning and create a new video and try different description.

**4.2 Methodology and Development process**  
The methodology we choose for our project is Agile. This methodology highlights iterative development, collaboration, and adaptability.  
By using the Agile methodology, our main goal is to encourage teamwork and adaptability. Agile allows us to work in small, manageable Sprints and easily adjust to changes in the project. The development process:

* API Research and selection: we needed to evaluate different AI APIs that suit with our project requirements and select the most suitable APIs based on performance, features and how smooth the integration process.

The API's we choose: OpenAI, TTS Azure, STT Azure, Pexels.

* Creating premade form: we have developed a sophisticated prompts for user interaction, designed to efficiently detect the user intent to ensure that the final product aligns closely with the user's needs.
* User interface Design and development: we needed to design and develop an intuitive and user-friendly interface, for example: content preview, customization options, big buttons etc. For that we used Tailwind CSS and React.

In the end of each Sprint period, we evaluated if there are any necessary changes needed to be made before we continue to the next Sprint with the help of managing versions on GitHub.

## **4.3 Constraints and challenges**

Creating this project comes with its fair share of challenges that we needed to overcome. These obstacles test our problem-solving skills and require careful consideration to ensure the success of our project. By addressing these challenges, we ensure a smoother, more successful, and more efficient development process.

**4.3.1 Integration with AI Tools and APIs**Integrating with AI tools and APIs presents challenges that requires careful implementation and management. We are utilizing several advanced APIs, each with specific requirements and dependencies:

* OpenAI API: Used for generating the user script, check the edited script by the user, and find keywords for each scene. this API requires us to handle large data sets and manage API call quotas efficiently to maintain performance without exceeding cost constraints.
* Azure Text-to-Speech API: This service converts text into voiceover audio. We need to ensure that the speech output is clear and accurately conveys the intended messages, which involves customizing voice settings and creating the right request.
* Pexels API: Provides access to high-quality videos to be the background of the generated video. Managing API requests and filtering results to retrieve relevant videos with the right resolutions and sizes is crucial for generating the optimal result.
* Azure Speech-to-Text API: Converts spoken audio to text for generating the subtitles. This requires us to handle different voices and dialects accurately, ensuring the text output is reliable and usable for further processing to the subtitles.

Each of these tools has its own set of integration challenges, from API limits and data to the technical demands and the cost of use. Addressing these challenges is crucial for the robustness and reliability of our project.

**4.3.2 User Input Variability**We faced an issue of handling the variability of user input. The user may provide input in different formats and styles, which can make it difficult to accurately capture and understand their intent. This diversity in user input adds complexity to our project. We overcome this challenge by providing clear and Precise prompt for the AI to deal with any input that can't be a description for a video generation. Moreover, In case of any errors while creating the script, we present the relevant error message via "SweetAlert2" in order to notify the user.

### **4.3.3 Real Time performance**

Generating content in real-time, including video scripts, voiceovers, and subtitles, and delivering it within acceptable response times, presents great challenges. These challenges include increased computational loads and potential delays, directly impacting performance and resource usage Such issues directly impact the user experience. To address these issues, we optimized our system structure, using efficient algorithms, and splitting the workload to multiple small tasks for making it simple and a smooth user experience. Moreover, we are limiting the video resolution with FFmpeg since our project uses the basic plan deployment that has restricted computational resources and bandwidth. By reducing the resolution, we ensure the video processing remains within the project's capacity, offering users a quicker and more consistent experience.

### **4.3.4 Quality Control and Filtering**

Ensuring that the generated content meets quality standards and filters out inappropriate or undesirable content it's a challenge. We overcome this challenge by leveraging the content moderation of OpenAI. This system is designed to not produce harmful or discriminatory content and filters responses accordingly. By integrating with OpenAI API, our system automatically rejects user inputs that violate these principles. If a user’s input is flagged by the ChatGPT’s moderation system for containing harmful, inappropriate, resist content that violates his policy, with an accurate prompt our system will not proceed with generating a response. Instead, it issues an informative alert to the user, explaining why the input was unacceptable and encouraging them to provide input that adheres to OpenAI's privacy and content standards.

### **4.3.5 Accuracy and Quality of Generated Content**

Ensuring the accuracy and quality of AI-generated content can be challenging, as AI models may sometimes produce irrelevant, nonsensical, or erroneous output. To mitigate this, our project utilizes a pre-built prompt that gathers detailed input and guidelines from users, optimizing the content generation process. Furthermore, we made research on how to write accurate prompts for ChatGPT, leveraging the expertise of a colleague who specializes in prompt generation to provided valuable tips that helped refine our approach. Also, we tested our prompt-generation strategy on multiple inputs to cover a wide range of scenarios and edge cases, ensuring that the prompts consistently generate high-quality results.

However, despite these measures, errors or unexpected behavior can still occur due to the nature of AI, rather than a specifically tailored model. Additionally, better results can be made by a higher-cost subscription plan that provides access to more refined options, features, and content accuracy.

**4.3.6 Deployment**During our deployment, we encountered challenges with using the FFmpeg library, in several deployment sites this library not supported. We choose to use Render site for our deployment as it supports the FFmpeg library. However the video generation process was particularly time-consuming on the free tier, that led us to upgrade to a paid plan to speed up the process and handle the demands of FFmpeg-based video production effectively.

# **5.Product**

**5.1 Requirements**  
Functional Requirements:

|  |  |
| --- | --- |
| **1.** | Allow users to input their content preferences and requirements such as: Topic, style, scenes, voice over and specific requirements |
| **2.** | Our system supports an input description from the user |
| **3.** | The system provides options for the user to customize and edit the generated content. |
| **4.** | Content filtering to prevent the generation of inappropriate, harmful content. |
| **5.** | The system integrates with AI APIs tools such as: OpenAI, Azure, Pexels |
| **6.** | Sharing the generated content with popular social media platforms for direct publishing such as: Facebook, Instagram, Telegram etc. |
| **7.** | The system generate voiceover |

Non-Functional Requirements:

|  |  |
| --- | --- |
| **1.** | processing user input and generating content within acceptable response times |
| **2.** | The system designed to handle a large number of concurrent users without compromising performance. |
| **3.** | The system measures to protect user data and ensure secure communication with AI APIs tools. |
| **4.** | The system needs to be reliable, with minimal downtime. |
| **5.** | The system needs to be compatible with different devices and browsers, |
| **6.** | The system generates accurate and high-quality content that aligns with user preferences and requirements. |

**5.2 Diagrams**Diagrams are an important tool for understanding how the system works. These diagrams help us visualize and clarify the system behavior, making it easier to design and explain how things should work. They are like maps that guide us in understanding and communicating the different parts and interactions of the system.

**תמונה שמכילה טקסט, תרשים, צילום מסך, ציור

התיאור נוצר באופן אוטומטי****5.2.1 Use Case**The following Use Case shows the user integration with our system.

Figure 9: Use Case Diagram

תמונה שמכילה טקסט, תרשים, מקביל, תוכנית

התיאור נוצר באופן אוטומטי**5.2.2 Activity Diagram**The following diagram describe the entire process starting with the user enter to the website until the video is ready to watch.

Figure 10: Activity Diagram

**5.2.3 Package Diagram**The package diagram presenting the general structural layout of our server, client application, showing the organization of key components, files within the project.

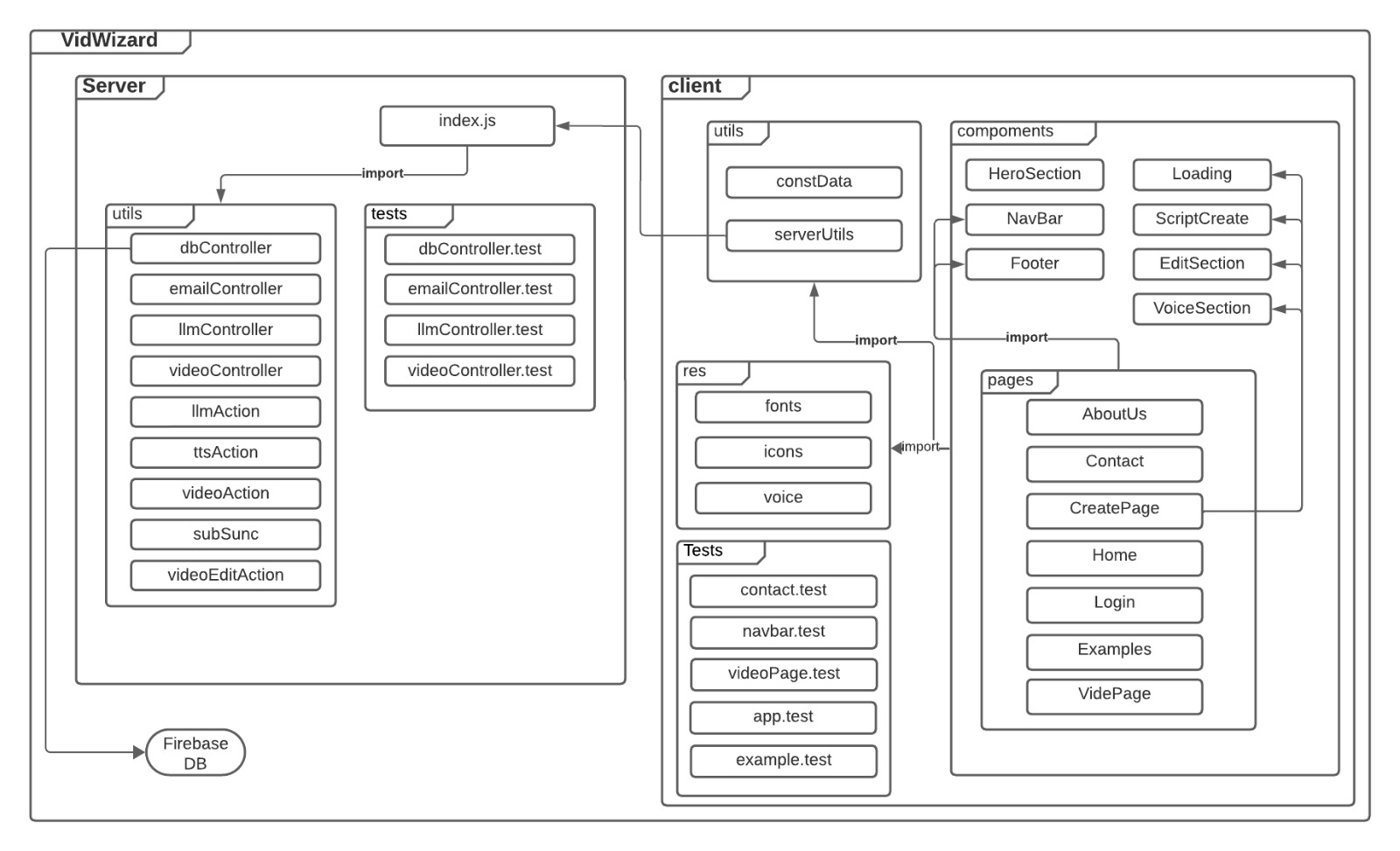


Figure 11: Package Diagram

## **5.3 Development**

* In our project for the Frontend development, we are using HTML, Tailwind CSS, SweetAlert2, and JavaScript to create and design the user interface. Additionally, we use JavaScript library such as React for simplify the process.
* On the Backend side, we are using JavaScript as well but with node.js EXPRESS for handling server side and interact with the client and database.
* Build Git repository for Version control. [[Link](https://github.com/shakedarish/Final-Project.git)]
* We use Firebase as our database for saving the user data.

# **6. Results and conclusion** in conclusion, we achieved our project goals, we successfully utilized AI and advanced tools to deliver video content creation with ease, and acceptable response times. However, given the nature of AI, some outputs were occasionally weird or incorrect. Currently, we are operating on a basic, non-paid plan, which while effective, shows potential for significant improvement with increased funding for an upgraded subscription. This investment would enhance our capabilities and refine the AI's accuracy, building on the strong foundation we've generated. The process of video editing is resource intensive and laborious, so tools and servers designed specifically for this purpose would significantly improve our project. Overall, we collaborated effectively and maintained a consistent workflow. Although, we would have conducted more research and selected editing video libraries to better suited our project, saving us valuable time. We learned that feedback loops, understanding the user intent, and consistent workflow structure are crucial for successfully generating high-quality videos. By applying these lessons, we can refine our process in future projects and achieve even greater results.

# **7. Verification and Evaluation**

**7.1 Testing Plan**We have divided our testing plan into two parts: the Frontend testing and the Backend testing.

### **7.1.1 Frontend Testing**

1. We are using a testing framework JestJS, with this framework we can perform testing by writing test cases to the important files to ensure the user interface is working correctly for example input forms, buttons, navigates and more.

2. Additionally, we used the Lighthouse plugin to assess the performance of our web application. Lighthouse, is a tool used by industry giants like Facebook, provides insights into various performance metrics.

תמונה שמכילה טקסט, צילום מסך, תוכנה, דף אינטרנט

התיאור נוצר באופן אוטומטי

Figure 13: LightHouse Performance

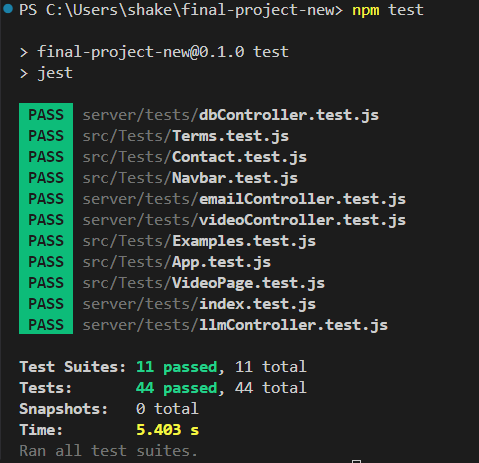


Figure 14: JestJS Tests

### 

### **7.1.2 Backend Testing**

In the Backend we decided to test our server running in EXPRESS NodeJs and our Firebase Database to verify that the data is correct for testing our backend, we used JestJS for testing the API response and our server response in several cases.   
  
In our project, the server structure was built with essential middleware like CORS and JSON processing enabled for handling API requests. Key functionalities are organized into separate controllers for videos, emails, login, and database operations.

Our error-handling structure uses try-catch blocks within asynchronous route handlers. If an error occurs, the server logs the error and responds with an error message if needed and a status code of 500 (Internal Server Error), indicating a server-side issue. In addition, the use of status code 200 is when responses are successfully sent without errors, indicating a successful response. This error-handling structure is crucial especially when dealing with dynamic content creation and user interactions.

Finally, we utilized Postman for server validation and to inspect the response of each server call.

תמונה שמכילה טקסט, צילום מסך, תוכנה, דף אינטרנט

התיאור נוצר באופן אוטומטי

Figure 15: Example of Postman

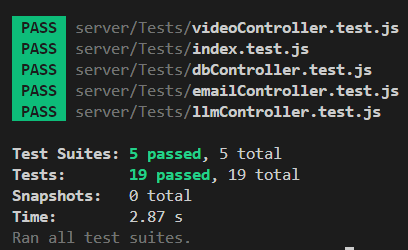
****

Figure 16: Example of JestJS Server Tests

**7.2 Evaluation**To effectively evaluate our platform, we are providing users with the option to contact us for various reasons, including site improvements, reporting errors, requesting explanations, and more through a user-friendly "Contact Us" page, designed to be straightforward to fill out.

# **9.References**

*[1] Copeland, B. (2023, June 21). Artificial intelligence (AI) | Definition, Examples, Types, Applications, Companies, & Facts. Encyclopedia Britannica.* [*https://www.britannica.com/technology/artificial-intelligence*](https://www.britannica.com/technology/artificial-intelligence) *(Page 1)*

*[2] Wikipedia contributors. (2023). Machine learning. Wikipedia.* [*Machine learning - Wikipedia*](https://en.wikipedia.org/wiki/Machine_learning) *(page 1)*

*[3] Wikipedia contributors. (2023a). API. Wikipedia.* [*API - Wikipedia*](https://en.wikipedia.org/wiki/API)

*[5] The state of AI in 2021. (2021, December 8). McKinsey & Company.* [*Global survey: The state of AI in 2021 | McKinsey*](https://www.mckinsey.com/capabilities/quantumblack/our-insights/global-survey-the-state-of-ai-in-2021)

*[6] Gregory, M. (2020, January 15). What Does Fairness in AI Mean? Forbes.* [What Does Fairness in AI Mean? (forbes.com)](https://www.forbes.com/sites/googlecloud/2020/01/15/what-does-fairness-in-ai-mean/?sh=4f9bd89d1574)

*[7] Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. In*Conference on Fairness, Accountability and Transparency*(pp. 77-91). PMLR.* [*buolamwini18a.pdf (mlr.press)*](http://proceedings.mlr.press/v81/buolamwini18a/buolamwini18a.pdf)

*[8] Jeffrey Dastin, (2018, October 11). Amazon scraps secret AI recruiting tool that showed bias against women.* [*Amazon scraps secret AI recruiting tool that showed bias against women | Reuters*](https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G)

*[9]* Telford, T. (2019, November 11). Apple Card algorithm sparks gender bias allegations against Goldman Sachs. *Washington Post*. [Apple Card algorithm sparks gender bias inquiry - The Washington Post](https://www.washingtonpost.com/business/2019/11/11/apple-card-algorithm-sparks-gender-bias-allegations-against-goldman-sachs/)

*[10] AI-Generated Content (AIGC): A Survey (Jiayang Wu, Wensheng Gan, Zefeng Chen, Shicheng Wan, and Hong Lin) Mar 2023.* [[2304.06632] AI-Generated Content (AIGC): A Survey (arxiv.org)](https://arxiv.org/abs/2304.06632)

*[11] AI-Generated Content, Creative Freelance Work and Hospitality and Tourism Marketing (Haaga-Helia University of Applied Sciences, Pajuniityntie 11, 00320 Helsinki, Finland)*