- The framework I will use is Flutter in the Dart language.  
- The software environment I will use is Cursor AI - for writing code;  
- I will use the interface layout from Figma.  
- The application will be available on Android and IOS operating systems.  
  
 **Introduction**

I want to create a mobile application that will become an intelligent processor for working with texts and images. Unlike a calculator, it will allow users to creatively interact with text and visual elements using metaphorical operations. The application will consist of two main sections: for text and images. In the future, I plan to expand the functionality for working with music and will continue to expand the application. This makes the application not just a technical tool, but an environment for generating new content, which can attract a wide audience.

**Concept applications**

**1. Main idea**

The app is designed for creative manipulation of texts and images, similar to how a calculator processes numerical data. It allows users to enter text or images, select an operation (such as “addition,” “multiplication,” or “exponentiation”), and receive results in a chat format (the AI ​​bot will respond in the chat).

**2. Target audience and uniqueness:**

The research shows that the target audience may include writers, artists, students, marketers and ordinary users interested in creativity. The uniqueness of my application lies in its simplicity, as it will be easy to use intuitively without unnecessary professional knowledge, and also in its focus on metaphorical operations that go beyond standard editors, offering users the ability to create something new, rather than simply editing existing content. This can be a key advantage in a market where competition among text and image processing applications is high.

**3. Functionality**

**3.1 Working with text**

**- Addition:**Combining two texts into a single paragraph so that it is harmonious or a more complex work in the future.

**- Multiplication:**Create a new text that combines both input parameters, such as a story or a poem.

**- Raising to a power:**Strengthening the semantic depth or emotional tone of the text, making it more expressive and impressive.

**Technical implementation:**To implement these operations, modern technologies of natural language processing (NLP) can be used. For example:

**- Addition:**This may not be a simple concatenation, but a more complex process involving context and style analysis. The study suggests that models such as T5 or BART, which can perform text summarization or text fusion tasks, are suitable for this. For example, the HyperWrite AI tool

already offers similar functions, combining texts into a single whole.

**- Multiplication:**This would require generating new text based on two inputs, which can be done using models such as GPT-3 or GPT-2, available through Hugging Face. These models are capable of creating texts based on given prompts, such as "Create a poem combining the themes of the first and second texts."

**- Raising to a power:**This may involve sentiment analysis followed by text generation with emphasis on certain attributes such as emotionality. Models that support controlled text generation such as CTRL may be useful.

I will train the neural network with more complex logic to work with operations. Perhaps I will develop my own neural network for this application.  
  
 **Problems and solutions:**

**- Generation quality:**AI can generate text with errors or not always as expected. The solution is to add an editing mechanism or the ability for the user to correct the results.

**- Language restrictions:**Make sure that the models support Russian and other languages. For example, Hugging Face offers multilingual models, such as mT5.

**- Model training:**If I plan to train my own neural network, I will need a dataset that reflects my tasks, for example, pairs of texts for "addition" or examples of emotion amplification for "raising to a power". This can be resource-intensive, so I will consider transfer learning, retraining pre-trained models.

**3.2 Working with images**

**- Overlay:**Create a collage by layering one image on top of another.

**- Mixing:**Combining or blending images to create a new visual result.

**- Improvement:**Enhance image quality, such as contrast or apply artistic effects.

**Technical implementation:**

**- Overlay:**This can be achieved using computer vision libraries such as OpenCV which offers similar functions for image overlay.

**- Mixing:**This may involve blending modes or more complex techniques such as style transfer, where the style of one image is transferred to another. Models such as CycleGAN or StyleGAN, available through the TensorFlow Hub, are suitable for this.

**- Improvement:**Standard filters (increasing contrast, brightness) can be implemented via OpenCV, and for AI processing, use super-resolution models such as ESRGAN, optimized for mobile devices via TensorFlow Lite.

**Problems and solutions:**

**- Performance**: Image processing on mobile devices can be resource-intensive. The solution is to use model quantization to reduce their size and optimize them for mobile devices, such as TensorFlow Lite or Core ML.

**- Quality:**AI models can produce unpredictable results. I will add the ability for users to adjust parameters, such as the intensity of the effect.

**- Format:**Support popular formats (JPEG, PNG) and make sure the application works with different resolutions.

**3. Interface**

**The application interface is created from several pages:**

- Loading page: Loading page with animated logo.  
  
- Login page: Login page, with input fields for email and password, and options to sign in via Apple/Google.

- Registration page: Registration page with verification code and password.

- Remember password: Password recovery page with verification code.

- Chat texts page: Chat for working with texts, with an operations panel and history.

- Chat images page: Chat for working with images, similar to text chat.  
  
And also additional widgets:  
  
- Menu widget: A menu that slides out from the left side of the screen.  
  
- History texts widget: History widget for Chat texts page. Slide out from the right side of the screen.  
  
- History images widget: History widget for the Chat images page. Slide out from the right side of the screen.  
  
The UI also includes:  
  
Button animations, svg icons, images, etc. Various interactive animated widgets that I haven't mentioned yet.

**4. Extensibility and Future Capabilities**

Plans to add processing of music and other media are a promising idea. To do this:

Develop a modular architecture by dividing the application into modules (text, images, music, etc.).

For music, consider libraries such as Librosa, which are adapted for mobile devices.

Think about video, 3D models or VR/AR content in the future by building flexibility into the architecture.

**5. Example of use**

**5.1 Working with text**

- The user enters: "Hello, world!"

- Selects the "Addition" operation.

- Enters: "Farewell, world!"

- Sends a request to the chat.

- AI replies: "Hello, we'll see each other again!"

For the operations "Multiplication" and "Exponentiation" other artificial intelligence algorithms will be used.

**5.2 Working with images**

- The user attaches an image of a sunset.

- Selects the "Addition" operation.

- Attaches an image of the ocean.

- Sends a request to the chat.

- AI creates a collage of a sunset over the ocean (AI logic will be improved in the future).

For the operations "Multiplication" and "Exponentiation" other artificial intelligence algorithms will be used.