

CBMRS

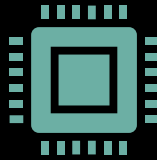
IN JAVA



Introduction



Multimedia Retrieval is one of the most important and fastest growing research areas in the field of Multimedia technology.



The large and growing amount of digital data, and the development of the Internet highlight the need to develop sophisticated access methods that provide more than just simple text-based queries.



Many programs have been developed with complex mathematics algorithms to allow the transformation of image or video data in a way that enhances searching accuracy. However it becomes difficult when dealing with large sets of multimedia data.

The image features a 3D rendering of several white puzzle pieces scattered on a teal gradient background. The pieces are arranged in a way that suggests a larger puzzle is being assembled. The lighting creates soft shadows, giving the pieces a three-dimensional appearance. The word 'development' is written in a white, serif font, positioned in the lower-left area of the image, partially overlapping the puzzle pieces.

development

Step 1: choosing the programming language

Our design was based making the best design in terms of memory efficiency, robustness and responsiveness so we choose java as programming language for the project.

Java vs Python

JAVA	PYTHON
statically typed and compiled language	dynamically typed and interpreted language.
Less memory usage	More memory usage
Faster	Slower
harder	easier

Step 2: database Design

We understand the importance of the database Design in this project so we made very efficient database design to increase the responsiveness of the application and give the user better impression

A wide-angle photograph of a coastal dune landscape. A light-colored wooden boardwalk curves from the foreground into the distance, flanked by tall, dry grasses and low-lying green shrubs. In the background, rolling sand dunes are visible under a heavy, overcast sky. The text "Step 3: Algorithms" is overlaid in a white serif font on a semi-transparent blue rectangular background in the lower half of the image.

Step 3: Algorithms

CBIR

Mean Color

Histogram

Color Layout



CBVR

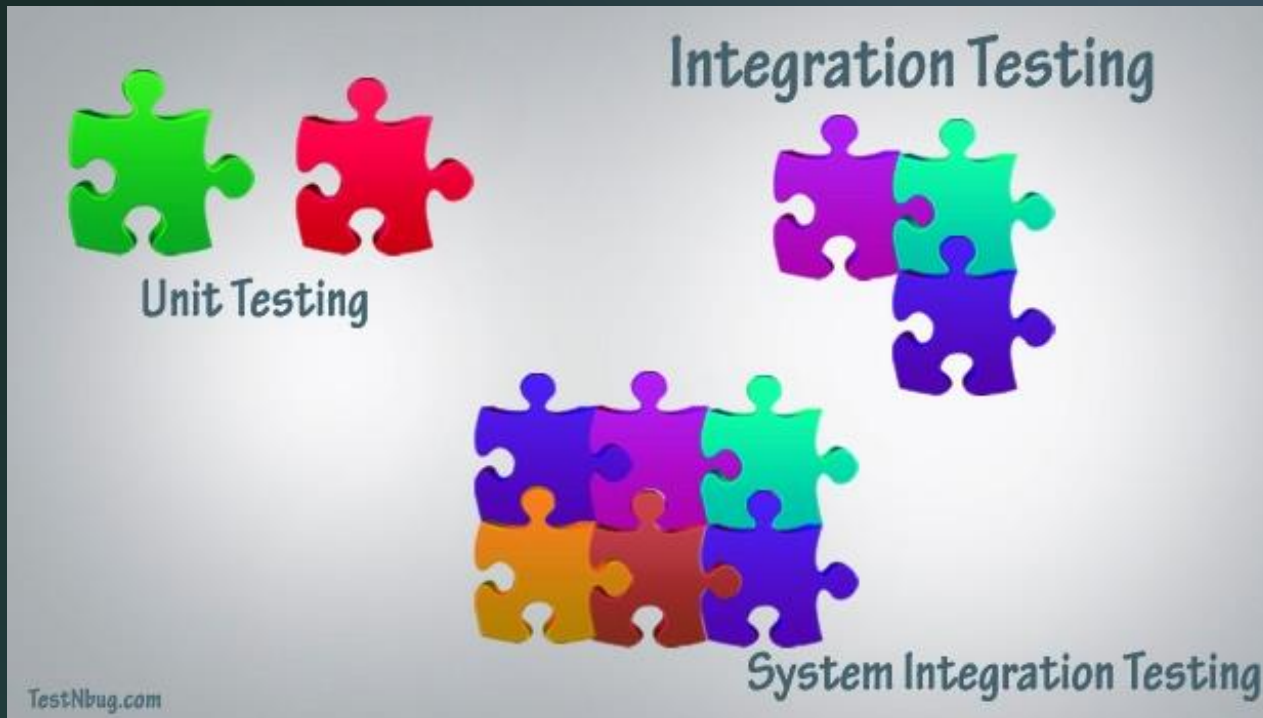
- NAIVE VIDEO SIMILARITY

Step 4: GUI Design

WE USED JAVAFX TO DESIGN OUR GUI
TO BE USER FRIENDLY APPLICATION



Step 5: Testing



We tested the project to meet the desired quality, the types of testing we made are unit testing, integration testing and system testing

Step 6: Error handling



We handled all user error that may pop up while using the application in order to reach the highest quality of the product .



Thank
you