Matrix Rotatio GUI App



Submitted by:

**Jyothi Swaroop MB**

12306111

Lovely Professional University

Matrix Rotatio GUI App

Submitted by:  
Jyothi Swaroop MB  
Lovely Professional University

# Table of Contents

1. Introduction  
2. Project Overview & Objectives  
3. Core Concepts: BST & CRUD  
4. C++ Command-Line Implementation  
5. Web-Based Interactive Visualization  
6. Sample Input & Output  
7. Screenshots  
8. Conclusion & Future Work  
9. Appendix: Source Code

# 1. Introduction

The **Matrix Rotation GUI App** is a simple and interactive web-based tool designed to help users rotate a 3×3 matrix in different directions. It provides a clean, colorful interface where users can input matrix values, click rotation buttons (90° clockwise, 180°, or 90° counterclockwise), and instantly see the rotated result with smooth animations.

This app is useful for students, developers, and anyone working with matrix operations, offering a visual and intuitive way to understand how matrix rotation works. Built with HTML, CSS, and JavaScript, it runs entirely in your browser—no installations or logins required!

# 2. Project Overview & Objectives

The Matrix Rotation GUI App is a browser-based tool that lets users input a 3×3 matrix and rotate it 90° clockwise, 180°, or 90° counterclockwise. The objective is to provide a simple, interactive, and visual way to understand matrix rotations. It features a clean UI, smooth animations, and requires no installation—making it ideal for learners and developers alike.

# 3. Core Concepts: Matrix Rotation

Matrix rotation involves shifting the positions of elements in a 2D array (matrix) based on a specific angle. In a 3×3 matrix:

* **90° Clockwise Rotation**: Transpose the matrix, then reverse each row.
* **90° Counterclockwise Rotation**: Transpose the matrix, then reverse each column.
* **180° Rotation**: Reverse each row, then reverse the order of rows.

These transformations preserve the matrix shape but rearrange the data based on rotation direction.

# 4. C++ Command-Line Implementation

# #include <iostream>

# using namespace std;

# void printMatrix(int matrix[3][3]) {

# for (int i = 0; i < 3; i++) {

# for (int j = 0; j < 3; j++) {

# cout << matrix[i][j] << " ";

# }

# cout << endl;

# }

# }

# void rotate90(int mat[3][3], int res[3][3]) {

# for (int i = 0; i < 3; i++)

# for (int j = 0; j < 3; j++)

# res[j][2 - i] = mat[i][j];

# }

# void rotate180(int mat[3][3], int res[3][3]) {

# for (int i = 0; i < 3; i++)

# for (int j = 0; j < 3; j++)

# res[2 - i][2 - j] = mat[i][j];

# }

# void rotate270(int mat[3][3], int res[3][3]) {

# for (int i = 0; i < 3; i++)

# for (int j = 0; j < 3; j++)

# res[2 - j][i] = mat[i][j];

# }

# int main() {

# int matrix[3][3];

# int result[3][3];

# int choice;

# cout << "Enter 3x3 Matrix (row-wise):\n";

# for (int i = 0; i < 3; i++)

# for (int j = 0; j < 3; j++)

# cin >> matrix[i][j];

# cout << "\nChoose rotation:\n";

# cout << "1. 90° Clockwise\n";

# cout << "2. 180°\n";

# cout << "3. 90° Counterclockwise\n";

# cout << "Enter your choice: ";

# cin >> choice;

# cout << "\nRotated Matrix:\n";

# if (choice == 1)

# rotate90(matrix, result);

# else if (choice == 2)

# rotate180(matrix, result);

# else if (choice == 3)

# rotate270(matrix, result);

# else {

# cout << "Invalid choice.\n";

# return 0;

# }

# printMatrix(result);

# return 0;

# }

### ****5. Web-Based Interactive Visualization****

This web application is built using:

* **HTML** for structural layout
* **CSS** (Aqua/White gradient theme) for styling and responsiveness
* **JavaScript** for logic, rotation algorithms, and animations

**Key Features:**

* Clean interface with input fields for a 3×3 matrix
* Rotation options: **90° Clockwise**, **180°**, and **90° Counterclockwise**
* Live result display with **flip animation effect**
* Reset/clear matrix option
* Fully responsive across **mobile, tablet, and desktop devices**

### ****6. Sample Input & Output****

**Input Matrix:**

1 2 3

4 5 6

7 8 9

**Action:** Rotate 90° Clockwise  
**Output Matrix:**

7 4 1

8 5 2

9 6 3

# 7. Screenshots

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

### ****8. Conclusion & Future Work****

This project successfully demonstrates matrix manipulation through an intuitive, web-based GUI. The app allows users to rotate a 3×3 matrix in various directions with instant visual feedback and smooth animations, making learning matrix transformations engaging and accessible.

**Future Enhancements:**

* Support for dynamic NxN matrices
* Add step-by-step animation for rotation logic
* Enable export of matrix output as image or PDF
* Integrate dark/light theme toggle for better accessibility

### ****9. Appendix: Source Code****

* **C++ File:** matrix\_rotation.cpp – Console-based implementation of 90°, 180°, and 270° matrix rotation.
* **HTML File:** index.html – Contains the complete web-based GUI with embedded CSS and JavaScript.
* **Deployment:** Hosted via **GitHub Pages** for easy online access and interaction.

GitHub Repository:----

https://github.com/filthysnux/MatrixRotation.git