Digital Diary with Encryption

Application Functionality

The Secure Digital Diary is a browser-based application designed for privacy-focused users to securely store and retrieve diary entries. Using encryption technology, the application ensures that entries can only be decrypted with the correct date and password.

Core Features:

1. **Entry Encryption**:
    - Title Field: Enter a short, descriptive title for the diary entry.
    - Content Field: Add the main diary content (text).
    - Password Field: Specify a password to encrypt the diary entry securely. The password is not stored and is essential for later decryption.
    - Timestamping: Automatically records the date (in YYYY-MM-DD format) when the entry is encrypted.
    - Result: The encrypted diary entry is displayed as a block of text. This encrypted text can be saved locally or stored securely for future use.

2. **Entry Decryption**:
    - Date Field: The user must input the exact date of the encrypted entry.
    - Password Field: The password used during encryption must also be provided.
    - Validation: If both the date and password match the encryption details, the entry is successfully decrypted and displayed.
    - Error Handling: Invalid credentials trigger an error message, maintaining security.

3. **Styling for Clarity**:
    - Input fields for **encryption** are styled with a **pale turquoise background** to distinguish them from decryption fields.
    - Input fields for **decryption** are styled with a **pale green background** for better usability.
4. **Responsive Design**:
    - Uses TailwindCSS for a clean, mobile-friendly interface.
    - Works seamlessly across different screen sizes.

Usage Instructions

---

1. **Encrypting a Diary Entry**:
    - Open the application in your browser.
    - In the **"Entry Title"** field:
        - Provide a meaningful title for your entry.
        - Example: "Vacation Memories" or "Meeting Notes."
    - In the **"Entry Content"** field:
        - Write the text of your diary entry.
        - Example: "Today I visited the beach and had an amazing time watching the sunset."
    - In the **"Encryption Password"** field:
        - Enter a secure password (e.g., "Beach2025!").
    - Click the **"Save Entry"** button:
        - The application encrypts your diary entry.
        - A block of encrypted text is displayed in the **Encrypted Entry** section.
    - Save this encrypted text manually or copy it to a secure location.

2. **Decrypting a Diary Entry**:
    - Paste the encrypted text into the **Encrypted Entry** section if not already displayed.
    - In the **"Decryption Date"** field:
        - Input the exact date the entry was encrypted.
        - Example: "2025-01-25" for an entry saved on January 25, 2025.
    - In the "Decryption Password" field:
        - Enter the exact password used during encryption (case-sensitive).
    - Click the **"Decrypt Entry"** button:
        - If both the date and password are correct, the decrypted title and content will be displayed.

- If the credentials are incorrect, an error message is displayed: "Invalid password, date, or corrupted data."

3. **Error Handling**:

- Missing Fields: If any required field is empty during encryption or decryption, an alert will notify the user.
- Invalid Credentials: If the date or password is incorrect during decryption, the application will display an error message.

Setup Process

1. **System Requirements**:
   - A modern web browser (Google Chrome, Mozilla Firefox, Microsoft Edge, etc.).
   - A text editor (optional) if you want to edit or modify the HTML file.

2. **Installation Steps**:
   - Download the HTML file containing the application code.
   - Save the file in a folder on your computer.
   - Double-click the HTML file to open it in your browser.

3. **Customization (Optional)**:
   - Modify the **color scheme** in the <style> section of the HTML code.
   - Adjust fonts, layout, or functionality as needed.

4. **Testing the Application**:
   - Perform test encryptions with sample titles and content.
   - Verify decryption by entering the correct password and date.
   - Experiment with incorrect credentials to ensure error messages function properly.

5. **Hosting the Application**:
   - For online use, upload the HTML file to a hosting platform such as GitHub Pages, Netlify, or Vercel.

Html Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Secure Digital Diary</title>
    <link href="https://cdn.jsdelivr.net/npm/tailwindcss@2.2.19/dist/tailwind.min.css"
rel="stylesheet">
    <link rel="stylesheet" href="styles.css">
    <script src="https://cdnjs.cloudflare.com/ajax/libs/crypto-js/4.1.1/crypto-js.min.js"></script>
</head>
<body>
    <div class="min-h-screen flex flex-col items-center justify-center">
        <div class="w-full max-w-md bg-white p-6 rounded-2xl shadow-lg">
            <h1 class="text-2xl font-bold text-center text-gray-800 mb-4">Secure Digital Diary</h1>

            <form id="diaryForm" class="space-y-4">
                <div>
                    <label for="title" class="block text-sm font-medium text-gray-700">Entry Title</label>
                    <input type="text" id="title" class="mt-1 block w-full rounded-md border-gray-300
shadow-sm focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>
                </div>

                <div>
```

```
            <label for="content" class="block text-sm font-medium text-gray-700">Entry Content</label>
            <textarea id="content" rows="4" class="mt-1 block w-full rounded-md border-gray-300 shadow-sm focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required></textarea>
        </div>

        <div>
            <label for="password" class="block text-sm font-medium text-gray-700">Encryption Password</label>
            <input type="password" id="password" class="mt-1 block w-full rounded-md border-gray-300 shadow-sm focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>
        </div>

        <button type="submit" class="w-full bg-indigo-500 text-white py-2 px-4 rounded-md hover:bg-indigo-600 transition">Save Entry</button>
    </form>

    <div id="output" class="mt-6 hidden">
        <h2 class="text-lg font-bold text-gray-800">Encrypted Entry:</h2>
        <p id="encryptedData" class="bg-gray-100 p-3 rounded-md mt-2 overflow-auto"></p>

        <h2 class="text-lg font-bold text-gray-800 mt-4">Decrypt Entry</h2>
        <form id="decryptForm" class="space-y-4">
            <div>
                <label for="decryptDate" class="block text-sm font-medium text-gray-700">Decryption Date (YYYY-MM-DD)</label>
                <input type="date" id="decryptDate" class="mt-1 block w-full rounded-md border-gray-300 shadow-sm focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>
```

```
            </div>
            <div>
                <label for="decryptPassword" class="block text-sm font-medium text-gray-700">Decryption Password</label>
                <input type="password" id="decryptPassword" class="mt-1 block w-full rounded-md border-gray-300 shadow-sm focus:ring-indigo-500 focus:border-indigo-500 sm:text-sm" required>
            </div>

            <button type="submit" class="w-full bg-green-500 text-white py-2 px-4 rounded-md hover:bg-green-600 transition">Decrypt Entry</button>
        </form>

        <div id="decryptedOutput" class="mt-4 hidden">
            <h3 class="text-md font-semibold text-gray-800">Decrypted Entry:</h3>
            <pre id="decryptedData" class="bg-gray-100 p-3 rounded-md mt-2 overflow-auto"></pre>
        </div>
      </div>
    </div>
  </div>
  <script src="script.js"></script>
</body>
</html>
```

CSS Code :

```
body {
    background: linear-gradient(to bottom, #e2e8f0, #edf2f7);
    font-family: Arial, sans-serif;
}

#diaryForm input,
#diaryForm textarea {
    background-color: rgb(211, 234, 249);
}

#decryptForm input {
    background-color: rgb(199, 249, 217);
}
```

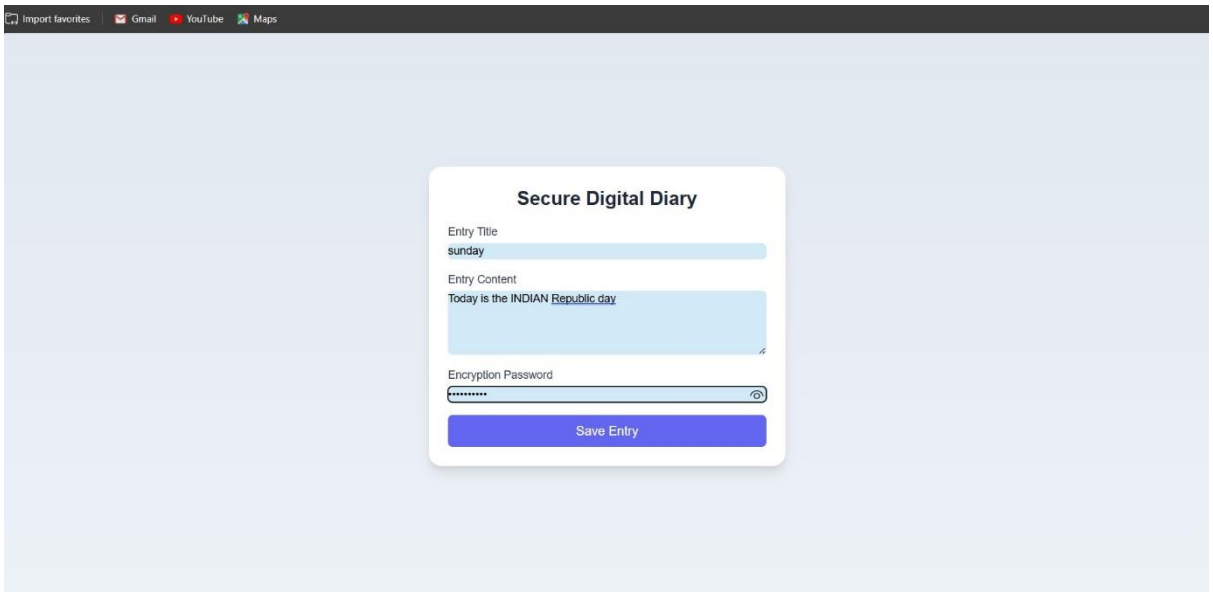
JavaScript Code :

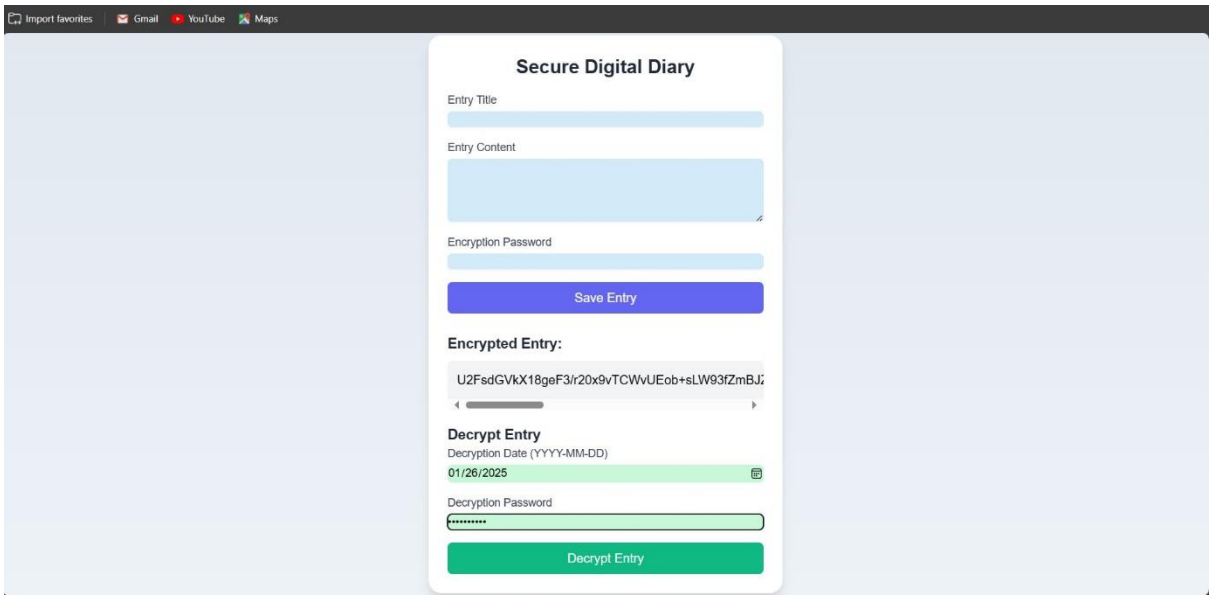
```
const form = document.getElementById('diaryForm');
const outputDiv = document.getElementById('output');
const encryptedDataElement = document.getElementById('encryptedData');
const decryptForm = document.getElementById('decryptForm');
const decryptedOutputDiv = document.getElementById('decryptedOutput');
const decryptedDataElement = document.getElementById('decryptedData');

form.addEventListener('submit', (event) => {
    event.preventDefault();

    const title = document.getElementById('title').value.trim();
    const content = document.getElementById('content').value.trim();
    const password = document.getElementById('password').value;
    const currentDate = new Date().toISOString().split('T')[0];

    if (!title || !content || !password) {
        alert('Please fill out all fields.');
        return;
    }

    const entry = { title, content, date: currentDate };
    const entryString = JSON.stringify(entry);

    const encrypted = CryptoJS.AES.encrypt(entryString, password).toString();
```

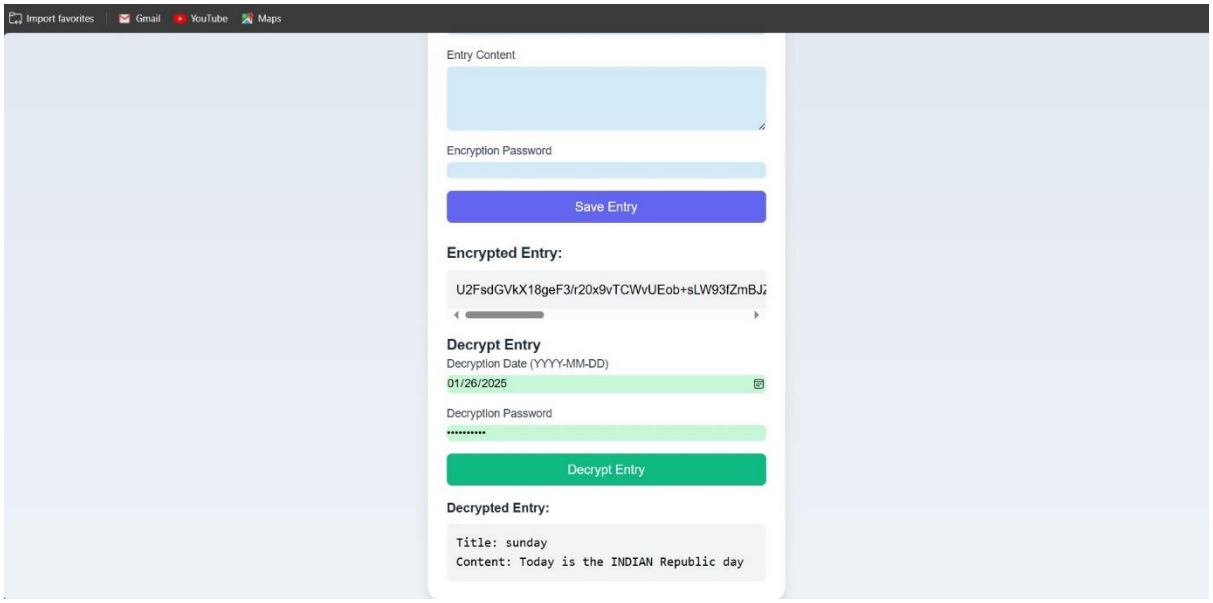
```
    encryptedDataElement.textContent = encrypted;
    outputDiv.classList.remove('hidden');
    form.reset();
});

decryptForm.addEventListener('submit', (event) => {
    event.preventDefault();

    const encryptedData = encryptedDataElement.textContent;
    const password = document.getElementById('decryptPassword').value;
    const inputDate = document.getElementById('decryptDate').value;

    if (!encryptedData || !password || !inputDate) {
        alert('Encrypted data, password, or date is missing.');
        return;
    }

    try {
        const bytes = CryptoJS.AES.decrypt(encryptedData, password);
        const decrypted = bytes.toString(CryptoJS.enc.Utf8);

        if (!decrypted) {
            throw new Error('Decryption failed.');
        }

        const entry = JSON.parse(decrypted);
```

```
        if (entry.date !== inputDate) {
            alert('Invalid date. Decryption failed.');
            return;
        }

        decryptedDataElement.textContent = `Title: ${entry.title}\nContent: ${entry.content}`;
        decryptedOutputDiv.classList.remove('hidden');
    } catch (error) {
        alert('Invalid password, date, or corrupted data.');
    }
});
```

Output :


Import favorites
Gmail
YouTube
Maps
Secure Digital Diary
Entry Title
sunday
Entry Content
Today is the INDIAN Republic day
Encryption Password
Save Entry


Import favorites
Gmail
YouTube
Maps
Secure Digital Diary
Entry Title
Entry Content
Encryption Password
Save Entry
Encrypted Entry:
U2FsdGVkX18geF3/r20x9vTCWvUEob+sLW93fZmBJ
Decrypt Entry
Decryption Date (YYYY-MM-DD)
01/26/2025
Decryption Password
Decrypt Entry


Import favorites
Gmail
YouTube
Maps
Entry Content
Encryption Password
Save Entry
Encrypted Entry:
U2FsdGVkX18geF3/r20x9vTCWvUEob+sLW93fZmBJ
Decrypt Entry
Decryption Date (YYYY-MM-DD)
01/26/2025
Decryption Password
Decrypt Entry
Decrypted Entry:
Title: sunday
Content: Today is the INDIAN Republic day