HTML:  
<!DOCTYPE html>

<html lang="zh-TW">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>寄憶之海</title>

    <link

      href="https://fonts.googleapis.com/css2?family=Noto+Serif+TC:wght@400;500;600;700&display=swap"

      rel="stylesheet"

    />

    <style>

      /\* --- 您的原始 CSS 開始 --- \*/

      \* {

        margin: 0;

        padding: 0;

        box-sizing: border-box;

      }

      body,

      html {

        height: 100%;

        width: 100%;

        font-family: "Noto Serif TC", serif;

        color: white;

        overflow: hidden;

      }

      body {

        /\* 修改：使用您的直式照片 letter.jpg \*/

        background-image: url("letter.jpg");

        background-position: center;

        background-size: cover;

        background-repeat: no-repeat;

        background-attachment: fixed;

        position: relative;

      }

      .container {

        height: 100%;

        width: 100%;

        display: flex;

        flex-direction: column;

        justify-content: center;

        align-items: center;

        padding: 20px;

        position: relative;

        background-color: rgba(0, 0, 0, 0.3);

        transition: opacity 0.5s ease-out; /\* 新增：為畫面切換增加淡出效果 \*/

      }

      .content {

        text-align: center;

        max-width: 90%;

        margin-bottom: 25vh;

      }

      .main-title {

        font-size: 3.2rem;

        font-weight: 700;

        margin-bottom: 15px;

        letter-spacing: 5px;

        text-shadow: 0 0 5px rgba(255, 255, 255, 0.7),

          0 0 10px rgba(255, 255, 255, 0.5), 0 0 20px rgba(173, 216, 230, 0.5);

      }

      .sub-title {

        font-size: 1rem;

        font-weight: 500;

        margin-bottom: 40px;

        letter-spacing: 1px;

        text-shadow: 0 0 5px rgba(255, 255, 255, 0.6);

      }

      .text-content {

        font-size: 0.95rem;

        line-height: 1.6;

        margin-bottom: 15px;

        letter-spacing: 1px;

      }

      .buttons {

        display: flex;

        flex-direction: row;

        justify-content: center;

        gap: 15px;

        width: 100%;

        position: absolute;

        bottom: 30vh;

        left: 50%;

        transform: translateX(-50%);

      }

      button {

        background-color: rgba(255, 255, 255, 0.2);

        border: 1px solid rgba(255, 255, 255, 0.5);

        color: white;

        padding: 12px 20px;

        border-radius: 25px;

        cursor: pointer;

        font-family: "Noto Serif TC", serif;

        font-size: 0.9rem;

        transition: all 0.3s ease;

        backdrop-filter: blur(5px);

      }

      button:hover {

        background-color: rgba(255, 255, 255, 0.3);

        transform: translateY(-2px);

      }

      .copyright {

        position: absolute;

        bottom: 20px;

        font-size: 0.7rem;

        opacity: 0.7;

        width: 100%;

        text-align: center;

      }

      @media (min-width: 768px) {

        body {

          /\* 修改：使用您的橫式照片 banner.jpg \*/

          background-image: url("banner.jpg");

        }

        .mobile-break {

          display: none;

        }

      }

      @media (max-width: 480px) {

        .content {

          margin-bottom: 20vh;

        }

        .main-title {

          font-size: 2.2rem;

        }

        .buttons {

          flex-direction: column;

          gap: 10px;

          width: 80%;

          bottom: 15vh;

        }

        button {

          width: 100%;

        }

      }

      /\* --- 您的原始 CSS 結束 --- \*/

      /\* --- 新增的 CSS --- \*/

      .page {

        position: absolute;

        top: 0;

        left: 0;

        width: 100%;

        height: 100%;

        display: flex;

        justify-content: center;

        align-items: center;

        flex-direction: column;

        padding: 20px;

        background-color: rgba(0, 0, 0, 0.3);

        opacity: 0;

        visibility: hidden;

        transition: opacity 0.5s ease-in-out, visibility 0.5s;

        backdrop-filter: blur(5px);

      }

      .page.active {

        opacity: 1;

        visibility: visible;

      }

      .page-content {

        width: 100%;

        max-width: 600px;

        text-align: center;

      }

      textarea {

        width: 100%;

        height: 50vh;

        background-color: rgba(0, 0, 0, 0.4);

        border: 1px solid rgba(255, 255, 255, 0.3);

        border-radius: 10px;

        padding: 15px;

        font-family: "Noto Serif TC", serif;

        font-size: 1rem;

        color: white;

        resize: none;

        margin-bottom: 20px;

      }

      /\* --- 修改 1：調整提示文字顏色 --- \*/

      textarea::placeholder {

        color: rgba(255, 255, 255, 0.5); /\* 調整透明度讓顏色變深 \*/

      }

      textarea:focus {

        outline: none;

        border-color: rgba(255, 255, 255, 0.7);

      }

      .page-buttons {

        display: flex;

        gap: 15px;

        justify-content: center;

      }

      .letter-display {

        width: 100%;

        min-height: 50vh;

        background-color: rgba(0, 0, 0, 0.4);

        border: 1px solid rgba(255, 255, 255, 0.3);

        border-radius: 10px;

        padding: 25px;

        font-size: 1.1rem;

        line-height: 1.8;

        white-space: pre-wrap; /\* 保留換行 \*/

        margin-bottom: 20px;

        display: flex;

        justify-content: center;

        align-items: center;

        text-align: left;

        color: #e5e7eb; /\* 淺灰色字體 \*/

      }

      /\* 讀取信件前的動畫 \*/

      .bottle-animation {

        height: 50vh;

        display: flex;

        justify-content: center;

        align-items: center;

        position: relative;

        margin-bottom: 20px;

      }

      .wave {

        width: 100px;

        height: 100px;

        border-radius: 50%;

        border: 2px solid rgba(255, 255, 255, 0.8);

        position: absolute;

        animation: wave-animation 2.5s ease-out infinite;

      }

      .wave:nth-child(2) {

        animation-delay: 0.5s;

      }

      .wave:nth-child(3) {

        animation-delay: 1s;

      }

      @keyframes wave-animation {

        0% {

          transform: scale(0);

          opacity: 1;

        }

        100% {

          transform: scale(3);

          opacity: 0;

        }

      }

      #message-box {

        color: #6ee7b7; /\* 淺綠色 \*/

        height: 20px;

        margin-top: 15px;

        transition: opacity 0.5s;

      }

    </style>

  </head>

  <body>

    <!-- 主畫面 -->

    <div id="home-container" class="container">

      <div class="content">

        <h1 class="main-title">寄憶之海</h1>

        <h2 class="sub-title">一個只屬於你與偶然的療癒角落</h2>

        <p class="text-content">

          寫一封信，<br class="mobile-break" />給過去的遺憾、給未來的自己，<br

            class="mobile-break"

          />或給一個素未謀面的溫柔靈魂。

        </p>

        <p class="text-content">

          在這裡，每一封信都會<br

            class="mobile-break"

          />被溫柔地保存在時光的長河裡，<br

            class="mobile-break"

          />等待一位有緣人偶然拾起。

        </p>

      </div>

      <div class="buttons">

        <button id="send-button">投遞一封回憶信</button>

        <button id="receive-button">拾起一個漂流瓶</button>

      </div>

      <div class="copyright">© 2025 The Harbor of Memories by Vanessa</div>

    </div>

    <!-- 投遞信件頁面 -->

    <div id="write-page" class="page">

      <div class="page-content">

        <textarea

          id="letter-input"

          placeholder="寫給過去、未來，或某個再也無法對話的溫柔靈魂…"

        ></textarea>

        <div class="page-buttons">

          <button id="commit-send-button">投入大海</button>

          <button class="back-button">返回</button>

        </div>

        <div id="message-box"></div>

      </div>

    </div>

    <!-- 拾起漂流瓶頁面 -->

    <div id="read-page" class="page">

      <div class="page-content">

        <!-- 動畫容器 -->

        <div

          id="animation-container"

          class="bottle-animation"

          style="display: none"

        >

          <div class="wave"></div>

          <div class="wave"></div>

          <div class="wave"></div>

        </div>

        <!-- 信件內容顯示區 -->

        <div id="letter-display" class="letter-display"></div>

        <div class="page-buttons">

          <button id="pick-another-button">拾起下一個</button>

          <button class="back-button">返回</button>

        </div>

      </div>

    </div>

    <script>

      document.addEventListener("DOMContentLoaded", () => {

        // --- 獲取所有需要的 DOM 元素 ---

        const homeContainer = document.getElementById("home-container");

        const writePage = document.getElementById("write-page");

        const readPage = document.getElementById("read-page");

        const sendButton = document.getElementById("send-button");

        const receiveButton = document.getElementById("receive-button");

        const backButtons = document.querySelectorAll(".back-button");

        const letterInput = document.getElementById("letter-input");

        const commitSendButton = document.getElementById("commit-send-button");

        const messageBox = document.getElementById("message-box");

        const animationContainer = document.getElementById(

          "animation-container"

        );

        const letterDisplay = document.getElementById("letter-display");

        const pickAnotherButton = document.getElementById(

          "pick-another-button"

        );

        const LETTERS\_KEY = "harbor\_of\_memories\_letters";

        // --- 函數：切換頁面 ---

        function showPage(pageElement) {

          homeContainer.style.opacity = "0";

          writePage.classList.remove("active");

          readPage.classList.remove("active");

          if (pageElement) {

            pageElement.classList.add("active");

          } else {

            // 返回主頁

            homeContainer.style.opacity = "1";

          }

        }

        // --- 函數：儲存信件 ---

        function saveLetter() {

          const content = letterInput.value.trim();

          if (content === "") {

            messageBox.textContent = "信件不能是空的喔。";

            setTimeout(() => (messageBox.textContent = ""), 3000);

            return;

          }

          // 從 localStorage 獲取舊信件，若無則建立空陣列

          const letters = JSON.parse(localStorage.getItem(LETTERS\_KEY)) || [];

          // 新增信件

          letters.push(content);

          // 存回 localStorage

          localStorage.setItem(LETTERS\_KEY, JSON.stringify(letters));

          letterInput.value = ""; // 清空輸入框

          messageBox.textContent = "您的回憶已投入大海。";

          setTimeout(() => (messageBox.textContent = ""), 3000);

        }

        // --- 函數：顯示隨機信件 (包含動畫) ---

        function showRandomLetter() {

          const letters = JSON.parse(localStorage.getItem(LETTERS\_KEY)) || [];

          // 隱藏信件內容，顯示動畫

          letterDisplay.style.display = "none";

          animationContainer.style.display = "flex";

          // 動畫持續 2.5 秒

          setTimeout(() => {

            // 隱藏動畫，顯示信件內容

            animationContainer.style.display = "none";

            letterDisplay.style.display = "flex";

            if (letters.length > 0) {

              const randomIndex = Math.floor(Math.random() \* letters.length);

              const pickedLetter = letters[randomIndex];

              letterDisplay.textContent = pickedLetter;

              // --- 修改 2：移除已被讀取的信件 ---

              // 從陣列中移除該信件

              letters.splice(randomIndex, 1);

              // 將更新後的陣列存回 localStorage

              localStorage.setItem(LETTERS\_KEY, JSON.stringify(letters));

            } else {

              letterDisplay.textContent = "海面很平靜，還沒有漂流瓶...";

            }

          }, 2500);

        }

        // --- 綁定事件監聽 ---

        sendButton.addEventListener("click", () => showPage(writePage));

        receiveButton.addEventListener("click", () => {

          showPage(readPage);

          showRandomLetter(); // 第一次進入時就顯示一封信

        });

        backButtons.forEach((button) => {

          button.addEventListener("click", () => showPage(null));

        });

        commitSendButton.addEventListener("click", saveLetter);

        pickAnotherButton.addEventListener("click", showRandomLetter);

      });

    </script>

  </body>

</html>

CSS:  
/\* --- 您的原始 CSS 開始 --- \*/

\* {

  margin: 0;

  padding: 0;

  box-sizing: border-box;

}

body,

html {

  height: 100%;

  width: 100%;

  font-family: "Noto Serif TC", serif;

  color: white;

  overflow: hidden;

}

body {

  /\* 使用一個線上圖片作為背景 \*/

  background-image: url("letter.jpg");

  background-position: center;

  background-size: cover;

  background-repeat: no-repeat;

  background-attachment: fixed;

  position: relative;

}

.container {

  height: 100%;

  width: 100%;

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

  padding: 20px;

  position: relative;

  background-color: rgba(0, 0, 0, 0.3);

  /\* 修改：為畫面切換增加淡出效果 \*/

  transition: opacity 0.5s ease-out;

}

.content {

  text-align: center;

  max-width: 90%;

  margin-bottom: 25vh;

}

.main-title {

  font-size: 3.2rem;

  font-weight: 700;

  margin-bottom: 15px;

  letter-spacing: 5px;

  text-shadow: 0 0 5px rgba(255, 255, 255, 0.7),

    0 0 10px rgba(255, 255, 255, 0.5), 0 0 20px rgba(173, 216, 230, 0.5);

}

.sub-title {

  font-size: 1rem;

  font-weight: 500;

  margin-bottom: 40px;

  letter-spacing: 1px;

  text-shadow: 0 0 5px rgba(255, 255, 255, 0.6);

}

.text-content {

  font-size: 0.95rem;

  line-height: 1.6;

  margin-bottom: 15px;

  letter-spacing: 1px;

}

.buttons {

  display: flex;

  flex-direction: row;

  justify-content: center;

  gap: 15px;

  width: 100%;

  position: absolute;

  bottom: 30vh;

  left: 50%;

  transform: translateX(-50%);

}

button {

  background-color: rgba(255, 255, 255, 0.2);

  border: 1px solid rgba(255, 255, 255, 0.5);

  color: white;

  padding: 12px 20px;

  border-radius: 25px;

  cursor: pointer;

  font-family: "Noto Serif TC", serif;

  font-size: 0.9rem;

  transition: all 0.3s ease;

  backdrop-filter: blur(5px);

}

button:hover {

  background-color: rgba(255, 255, 255, 0.3);

  transform: translateY(-2px);

}

.copyright {

  position: absolute;

  bottom: 20px;

  font-size: 0.7rem;

  opacity: 0.7;

  width: 100%;

  text-align: center;

}

@media (min-width: 768px) {

  body {

    /\* 電腦版可以使用更高解析度的圖片 \*/

    background-image: url("banner.jpg");

  }

  .mobile-break {

    display: none;

  }

}

@media (max-width: 480px) {

  .content {

    margin-bottom: 20vh;

  }

  .main-title {

    font-size: 2.2rem;

  }

  .buttons {

    flex-direction: column;

    gap: 10px;

    width: 80%;

    bottom: 15vh;

  }

  button {

    width: 100%;

  }

}

/\* --- 您的原始 CSS 結束 --- \*/

/\* --- 新增的 CSS --- \*/

.page {

  position: absolute;

  top: 0;

  left: 0;

  width: 100%;

  height: 100%;

  display: flex;

  justify-content: center;

  align-items: center;

  flex-direction: column;

  padding: 20px;

  background-color: rgba(0, 0, 0, 0.3);

  opacity: 0;

  visibility: hidden;

  transition: opacity 0.5s ease-in-out, visibility 0.5s;

  backdrop-filter: blur(5px);

}

.page.active {

  opacity: 1;

  visibility: visible;

}

.page-content {

  width: 100%;

  max-width: 600px;

  text-align: center;

}

textarea {

  width: 100%;

  height: 50vh;

  background-color: rgba(0, 0, 0, 0.4);

  border: 1px solid rgba(255, 255, 255, 0.3);

  border-radius: 10px;

  padding: 15px;

  font-family: "Noto Serif TC", serif;

  font-size: 1rem;

  color: white;

  resize: none;

  margin-bottom: 20px;

}

textarea:focus {

  outline: none;

  border-color: rgba(255, 255, 255, 0.7);

}

.page-buttons {

  display: flex;

  gap: 15px;

  justify-content: center;

}

.letter-display {

  width: 100%;

  min-height: 50vh;

  background-color: rgba(0, 0, 0, 0.4);

  border: 1px solid rgba(255, 255, 255, 0.3);

  border-radius: 10px;

  padding: 25px;

  font-size: 1.1rem;

  line-height: 1.8;

  white-space: pre-wrap; /\* 保留換行 \*/

  margin-bottom: 20px;

  display: none; /\* 預設隱藏 \*/

  justify-content: center;

  align-items: center;

  text-align: left;

  color: #e5e7eb; /\* 淺灰色字體 \*/

}

/\* 讀取信件前的動畫 \*/

.bottle-animation {

  height: 50vh;

  display: none; /\* 預設隱藏 \*/

  justify-content: center;

  align-items: center;

  position: relative;

  margin-bottom: 20px;

}

.wave {

  width: 100px;

  height: 100px;

  border-radius: 50%;

  border: 2px solid rgba(255, 255, 255, 0.8);

  position: absolute;

  animation: wave-animation 2.5s ease-out infinite;

}

.wave:nth-child(2) {

  animation-delay: 0.5s;

}

.wave:nth-child(3) {

  animation-delay: 1s;

}

@keyframes wave-animation {

  0% {

    transform: scale(0);

    opacity: 1;

  }

  100% {

    transform: scale(3);

    opacity: 0;

  }

}

#message-box {

  color: #6ee7b7; /\* 淺綠色 \*/

  height: 20px;

  margin-top: 15px;

  transition: opacity 0.5s;

}

JS:  
document.addEventListener("DOMContentLoaded", () => {

  // --- 獲取所有需要的 DOM 元素 ---

  const homeContainer = document.getElementById("home-container");

  const writePage = document.getElementById("write-page");

  const readPage = document.getElementById("read-page");

  const sendButton = document.getElementById("send-button");

  const receiveButton = document.getElementById("receive-button");

  const backButtons = document.querySelectorAll(".back-button");

  const letterInput = document.getElementById("letter-input");

  const commitSendButton = document.getElementById("commit-send-button");

  const messageBox = document.getElementById("message-box");

  const animationContainer = document.getElementById("animation-container");

  const letterDisplay = document.getElementById("letter-display");

  const pickAnotherButton = document.getElementById("pick-another-button");

  const LETTERS\_KEY = "harbor\_of\_memories\_letters";

  // --- 函數：切換頁面 ---

  function showPage(pageElement) {

    // 先隱藏主頁，並移除其他頁面的 active 狀態

    homeContainer.style.opacity = "0";

    homeContainer.style.pointerEvents = "none"; // 讓主頁無法被點擊

    writePage.classList.remove("active");

    readPage.classList.remove("active");

    if (pageElement) {

      // 顯示指定的頁面

      pageElement.classList.add("active");

    } else {

      // 如果沒有指定頁面，就是返回主頁

      homeContainer.style.opacity = "1";

      homeContainer.style.pointerEvents = "auto"; // 恢復主頁點擊

    }

  }

  // --- 函數：儲存信件 ---

  function saveLetter() {

    const content = letterInput.value.trim();

    if (content === "") {

      messageBox.textContent = "信件不能是空的喔。";

      setTimeout(() => (messageBox.textContent = ""), 3000);

      return;

    }

    // 從 localStorage 獲取舊信件，若無則建立空陣列

    const letters = JSON.parse(localStorage.getItem(LETTERS\_KEY)) || [];

    // 新增信件

    letters.push(content);

    // 存回 localStorage

    localStorage.setItem(LETTERS\_KEY, JSON.stringify(letters));

    letterInput.value = ""; // 清空輸入框

    messageBox.textContent = "您的回憶已投入大海。";

    setTimeout(() => (messageBox.textContent = ""), 3000);

  }

  // --- 函數：顯示隨機信件 (包含動畫) ---

  function showRandomLetter() {

    const letters = JSON.parse(localStorage.getItem(LETTERS\_KEY)) || [];

    // 隱藏信件內容，顯示動畫

    letterDisplay.style.display = "none";

    animationContainer.style.display = "flex";

    // 動畫持續 2.5 秒

    setTimeout(() => {

      // 隱藏動畫，顯示信件內容

      animationContainer.style.display = "none";

      letterDisplay.style.display = "flex";

      if (letters.length > 0) {

        const randomIndex = Math.floor(Math.random() \* letters.length);

        letterDisplay.textContent = letters[randomIndex];

      } else {

        letterDisplay.textContent = "海面很平靜，還沒有漂流瓶...";

      }

    }, 2500);

  }

  // --- 綁定事件監聽 ---

  sendButton.addEventListener("click", () => showPage(writePage));

  receiveButton.addEventListener("click", () => {

    showPage(readPage);

    showRandomLetter(); // 第一次進入時就顯示一封信

  });

  backButtons.forEach((button) => {

    button.addEventListener("click", () => showPage(null));

  });

  commitSendButton.addEventListener("click", saveLetter);

  pickAnotherButton.addEventListener("click", showRandomLetter);

});

我現在想要延伸到製作點擊這兩個按鈕之後會發生什麼運作。

投遞信件：

點擊「投遞一封回憶信」，畫面會切換到書寫區。輸入框提示文字是"寫給過去、未來，或某個再也無法對話的溫柔靈魂…"

寫完信後，點擊「投入大海」，信件內容就會被匿名儲存起來（目前是存在您的瀏覽器中）。

使用者可以隨時點「返回」回到主畫面。

拾起漂流瓶：

點擊「拾起一個漂流瓶」，畫面會切換到閱讀區。

系統會隨機從所有被投遞的信件中，抽出一封顯示給使用者。我希望是有一個動畫跑完，才顯示信件。

使用者可以點擊「拾起下一個」來閱讀不同的信，或點「返回」回到主畫面。