

Manual de Usuario - SpeakUp!

Leonardo M. Bojórquez Valdez

14 de Noviembre de 2025

INDICE

- 1. INTRODUCCIÓN**
- 2. INSTALACIÓN DEL SISTEMA**
- 3. INSTALACIÓN DE NOTEPAD++**
- 4. BASE DE DATOS**
- 5. CREACIÓN DE TABLAS DE LA BASE DE DATOS**
- 6. CODIGOS**
- 7. MANTENIMIENTO DEL SISTEMA**
- 8. IMPACTO AMBIENTAL**
- 9. CONCLUSIÓN**

INTRODUCCIÓN

SpeakUp! es una plataforma educativa diseñada para apoyar a niñas y niños de primaria en su aprendizaje del idioma inglés mediante actividades interactivas, juegos dinámicos y módulos adaptados a cada grado escolar.

El sistema nació con una idea simple pero poderosa: *aprender jugando siempre se siente más ligero*. SpeakUp! combina colores cálidos, animaciones suaves y mecánicas amigables para que cada usuario pueda explorar, practicar y mejorar su comprensión del idioma sin sentirse presionado. Aquí, el aprendizaje fluye como si fuera un juego más en su día.

Este manual guía paso a paso el uso del sistema: desde el registro y el inicio de sesión, hasta la navegación por los cursos, perfil de usuario y actividades. También se incluyen instrucciones de instalación, medidas de seguridad y las bases para el mantenimiento del sistema.

Cada sección contiene un espacio para colocar capturas de pantalla, facilitando la integración visual del proceso real dentro de tu proyecto.



INSTALACIÓN DEL SISTEMA

Para ejecutar SpeakUp! en una computadora local, es necesario instalar un entorno que incluya Apache, PHP y MySQL. En este manual se utilizará WAMP, una herramienta sencilla y confiable para proyectos educativos y de desarrollo.

A continuación se describe el proceso completo para instalar WAMP desde cero.

Descargar WampServer

1. Ingresa al sitio oficial: <https://www.wampserver.com/en/>
2. Dirígete a la sección Downloads.
3. Selecciona la versión que corresponda a tu sistema operativo:
 - **WampServer 64 bits (x64)** — recomendado para la mayoría de equipos actuales.
 - **WampServer 32 bits (x86)** — solo si tu equipo es antiguo.
4. Haz clic en **Download Direct Link** para comenzar la descarga.



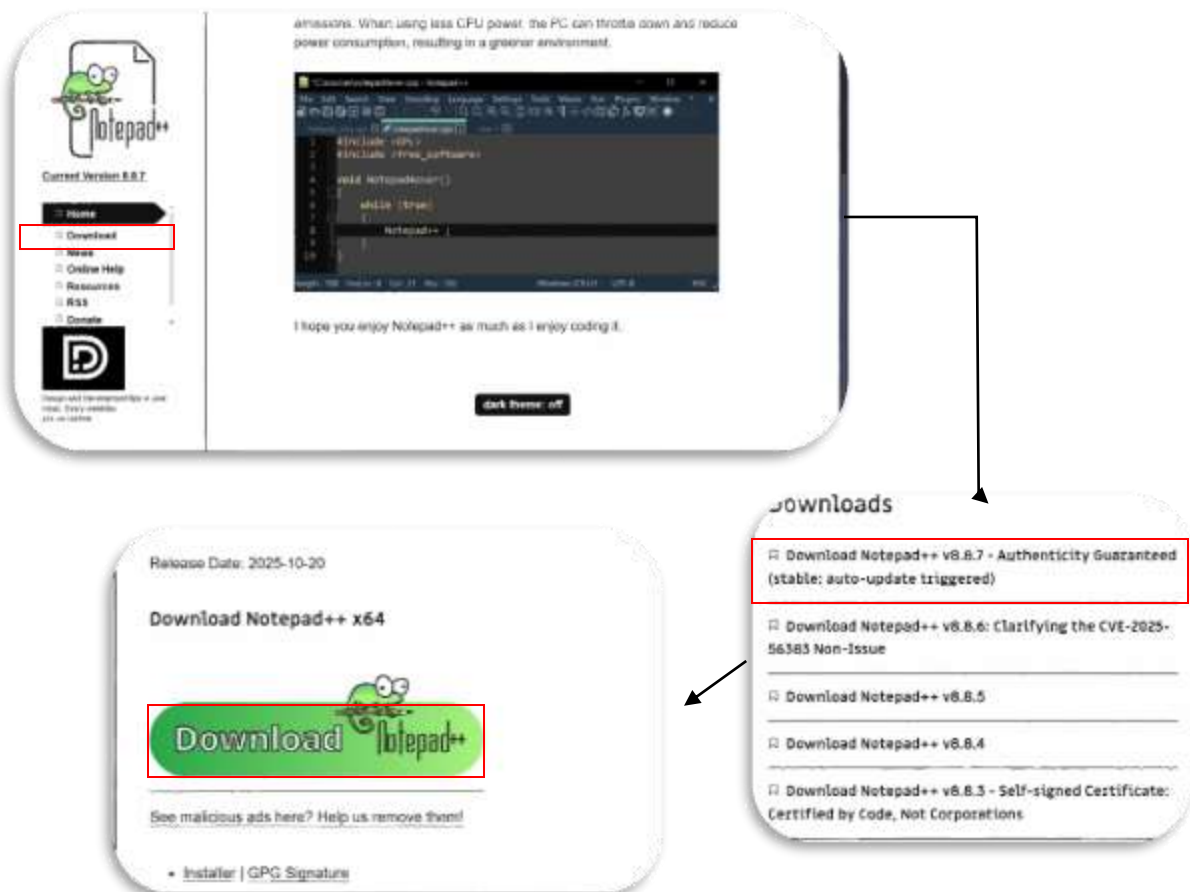
INSTALACIÓN DE NOTEPAD++

Notepad será una pieza clave en el desarrollo de “SpeakUp!”, ya que será el motor donde se guardaran y modificaran todos los códigos que conforman a “SpeakUp!”

A continuación se describe el proceso completo para instalar Notepad++ desde cero.

Descargar Notepad++

1. Abre tu navegador y ve al sitio oficial de Notepad++:
2. Dirígete a la sección Downloads.
3. Selecciona la versión que corresponda a tu sistema operativo:
 - Notepad++ Installer 64-bit (x64) — recomendado para la mayoría de equipos actuales.
 - Notepad++ Installer 32-bit (x86) — solo si tu equipo es más antiguo.
4. Haz clic en el botón de descarga para obtener el archivo instalador .exe



Ejecutar el Instalador

1. Una vez descargado el archivo, ábrelo con doble clic.
 2. Si aparece una ventana de seguridad preguntando si deseas permitir cambios en el dispositivo, selecciona Sí.
 3. Elige el idioma de instalación (puedes seleccionar Español).
 4. Presiona Next para avanzar entre las pantallas del instalado
 5. Configuración de la Instalación
-
1. Ubicación del programa:
 - Deja la ruta predeterminada, a menos que necesites instalarlo en otra carpeta por razones específicas.
 2. Componentes:
 - Deja activados los componentes esenciales (Plugins, Auto-Completion, Themes).
 3. Haz clic en Next y luego en Install para iniciar la instalación.

Finalizar

1. Espera a que el instalador complete el proceso.
2. Selecciona Finish para cerrar el instalador.
3. Notepad++ se abrirá por primera vez, listo para ser usado.

BASE DE DATOS

Para que el sistema pueda almacenar información (usuarios, mensajes, registros, etc.), es necesario crear una base de datos. En este apartado se describe el procedimiento para crearla utilizando php MyAdmin, una herramienta visual incluida en la mayoría de los entornos locales con MySQL.

Acceder a php MyAdmin

1. Asegúrate de que tu servidor local esté ejecutándose.
 - Si usas WAMP, el ícono debe mostrarse verde.
2. Abre tu navegador web.
3. Escribe en la barra de dirección:

`http://localhost/phpmyadmin`

4. Presiona Enter te pedirá una contraseña y un usuario, el usuario es “root” y no tiene contraseña, estos te servirán para para ingresar a la herramienta de administración de bases de datos.

Crear una Nueva Base de Datos

Dentro de phpMyAdmin, dirígete al menú superior y selecciona la pestaña Databases (Bases de datos).

1. En el campo Database name, escribe el nombre de tu base de datos.
 - Se recomienda usar un nombre corto, claro y sin espacios. Ejemplo:
 - `speakup_db` - (Sera el nombre que se usa en los códigos, por lo que se recomienda usarlo)
2. Haz clic en Create (Crear) para finalizar el proceso.

Confirmación

Si el procedimiento fue exitoso, phpMyAdmin mostrará un mensaje indicando que la base de datos se creó correctamente. A partir de este momento, ya puedes crear tablas y almacenar información dentro de ella.

CREACIÓN DE TABLAS DE LA BASE DE DATOS

Una vez creada la base de datos, es necesario generar las tablas que almacenarán la información principal del sistema. En este proyecto se utilizan dos tablas fundamentales:

- usuarios: Contiene los datos generales de los usuarios registrados.
- puntajes: Guarda los resultados obtenidos por cada usuario en las diferentes actividades del sistema.

Las siguientes instrucciones describen el procedimiento para crear cada tabla mediante sentencias SQL.

Tabla: usuarios

La tabla usuarios almacena los datos de identificación, contacto y registro de cada usuario.

Campo	Tipo / Descripción
id	Identificador único del usuario (PRIMARY KEY).
first_name	Nombre(s) del usuario.
last_name_p	Apellido paterno.
last_name_m	Apellido materno.
age	Edad del usuario.
location	Ciudad o región de procedencia.
email	Correo electrónico (único por usuario).
phone	Número telefónico.
gender	Género del usuario (male, female, other).
password	Contraseña encriptada.
created_at	Fecha y hora de registro.
grade	Nivel, grupo o grado académico.

En la parte de abajo te viene un botón llamado consola, ahí puedes pegar este código y crear las tablas con los datos ya mencionados:

```
CREATE TABLE usuarios (  
id INT AUTO_INCREMENT PRIMARY KEY,  
first_name VARCHAR(100) NOT NULL,  
last_name_p VARCHAR(100) NOT NULL,  
last_name_m VARCHAR(100) NOT NULL,  
age INT,  
location VARCHAR(150),  
email VARCHAR(120) UNIQUE NOT NULL,  
phone VARCHAR(20),  
gender ENUM('male','female','other') DEFAULT 'other',  
password VARCHAR(255) NOT NULL,  
created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
grade VARCHAR(50)  
);
```

Tabla: puntajes

La tabla puntajes almacena los resultados obtenidos por cada usuario. Cada fila está vinculada a un registro en la tabla usuarios mediante el campo user_id.

Los puntajes están organizados en las siguientes categorías:

- Writing: writing1, writing2, writing3, writing4
- Listening: listening1, listening2, listening3, listening4
- Grammar: grammar1, grammar2, grammar3, grammar4
- Juegos: juego1, juego2, juego3, juego4
- Trivia: trivia1, trivia2, trivia3, trivia4

En la parte de abajo te viene un botón llamado consola, ahí puedes pegar este código y crear las tablas con los datos ya mencionados:

```
CREATE TABLE puntajes (  
  puntaje_id INT AUTO_INCREMENT PRIMARY KEY,  
  user_id INT NOT NULL,  
  writing1 INT DEFAULT 0,  
  writing2 INT DEFAULT 0,  
  writing3 INT DEFAULT 0,  
  writing4 INT DEFAULT 0,  
  
  listening1 INT DEFAULT 0,  
  listening2 INT DEFAULT 0,  
  listening3 INT DEFAULT 0,  
  listening4 INT DEFAULT 0,  
  
  grammar1 INT DEFAULT 0,  
  grammar2 INT DEFAULT 0,  
  grammar3 INT DEFAULT 0,  
  grammar4 INT DEFAULT 0,  
  
  juego1 INT DEFAULT 0,  
  juego2 INT DEFAULT 0,  
  juego3 INT DEFAULT 0,  
  juego4 INT DEFAULT 0,  
  
  trivia1 INT DEFAULT 0,  
  trivia2 INT DEFAULT 0,  
  trivia3 INT DEFAULT 0,  
  trivia4 INT DEFAULT 0,  
  
  FOREIGN KEY (user_id) REFERENCES usuarios(id)  
  ON DELETE CASCADE  
  ON UPDATE CASCADE  
);
```

CODIGOS

A continuación se mostrara un apartado demasiado largo, pero explícitamente necesario para el funcionamiento correcto de “SpeakUp!” Por lo que se mostrara todos los códigos necesarios para el funcionamiento, también se mostrara el nombre con el que fueron guardados, este es estrictamente necesario usarlo si quiere correr el programa correctamente:

login.php

```
<?php
session_start();
$error = "";
if ($_SERVER["REQUEST_METHOD"] === "POST") {
    include("conexion.php");
    $email = $_POST["email"] ?? "";
    $password = $_POST["password"] ?? "";
    if (empty($email) || empty($password)) {
        $error = "Rellena todos los campos.";
    } else {
        $stmt = $conn->prepare("SELECT * FROM usuarios WHERE email = ?");
        $stmt->bind_param("s", $email);
        $stmt->execute();
        $resultado = $stmt->get_result();
        if ($resultado->num_rows === 0) {
            $error = "El correo no está registrado.";
        } else {
            $usuario = $resultado->fetch_assoc();
            if (!password_verify($password, $usuario["password"])) {
                $error = "Contraseña incorrecta.";
            } else {
                $_SESSION['user_id'] = $usuario['id'];
                $_SESSION['first_name'] = $usuario['first_name'];
                $_SESSION['last_name_p'] = $usuario['last_name_p'];
                $_SESSION['last_name_m'] = $usuario['last_name_m'];

                header("Location: Principalphp.php");
                exit;
            }
        }
    }
}
?>

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>SpeakUp! - Iniciar sesión</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    display: flex;
    justify-content: center;
    align-items: flex-start;
    min-height: 100vh;
    animation: fadeIn 0.8s ease-in-out;
```

```

    color: #3b2f00;
  }
  @keyframes fadeIn { from { opacity: 0; transform: translateY(-20px); } to { opacity: 1; transform: translateY(0); } }

.header {
  position: absolute;
  top: 20px;
  right: 30px;
  z-index: 100;
}
.header img { width: 140px; transition: transform 0.3s ease; }
.header img:hover { transform: scale(1.08) rotate(4deg); }

.main-container {
  display: flex;
  background: rgba(255, 255, 210, 0.95);
  border-radius: 20px;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.25);
  overflow: hidden;
  margin-top: 30px;
  animation: slideUp 0.8s ease;
}
@keyframes slideUp { from { opacity: 0; transform: translateY(50px); } to { opacity: 1; transform: translateY(0); } }

.left-panel {
  flex: 1;
  background: linear-gradient(135deg, #FFD600, #FFB300);
  color: #4b2f00;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  padding: 40px;
  text-align: center;
}
.left-panel .logo-circle {
  width: 180px;
  height: 180px;
  background: white;
  border-radius: 50%;
  display: flex;
  justify-content: center;
  align-items: center;
  margin-bottom: 20px;
  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
}
.left-panel .logo-circle img {
  width: 140px;
  height: auto;
  border-radius: 50%;
  transition: transform 0.3s ease;
}
.left-panel .logo-circle img:hover { transform: scale(1.08); }
.left-panel h2 { font-size: 26px; margin-bottom: 15px; }
.left-panel p { font-size: 15px; line-height: 1.6; max-width: 300px; }

.right-panel {
  flex: 1;
  padding: 40px 35px;
  text-align: center;
}
.right-panel .form-logo {
  width: 120px;
  margin-bottom: 15px;
}
h1 { color: #4b2f00; margin-bottom: 25px; font-size: 24px; }
label { display: block; text-align: left; margin-top: 10px; margin-bottom: 5px; font-weight: 600; }
input[type="email"], input[type="password"] {
  width: 100%; padding: 12px; border-radius: 8px; border: 2px solid #FFD857; outline: none;
  margin-bottom: 15px; background: #FFF9D9;
}

```

```

    }
    input:focus { box-shadow: 0 0 8px rgba(255,200,70,0.3); }

    button {
        width: 100%;
        margin-top: 20px;
        padding: 12px;
        background: linear-gradient(135deg, #FFD600, #FFB300);
        color: #4b2f00;
        border: none;
        border-radius: 10px;
        font-weight: bold;
        font-size: 15px;
        cursor: pointer;
        transition: 0.3s;
    }
    button:hover {
        transform: scale(1.05);
        box-shadow: 0 5px 15px rgba(0,0,0,0.2);
    }

    .links { margin-top: 15px; }
    .links button {
        background: none;
        border: none;
        color: #4b2f00;
        text-decoration: underline;
        cursor: pointer;
        font-size: 14px;
    }
    .links button:hover { color: #FFD600; }

    .toast {
        position: fixed;
        top: 20px;
        right: 20px;
        background: #FFD600;
        color: #4b2f00;
        padding: 14px 22px;
        border-radius: 12px;
        font-weight: 700;
        box-shadow: 0 6px 20px rgba(0,0,0,0.2);
        opacity: 0;
        transform: translateY(-20px);
        transition: opacity 0.4s ease, transform 0.4s ease;
        z-index: 9999;
    }
    .toast.show {
        opacity: 1;
        transform: translateY(0);
    }
    @media (max-width: 800px) {
        .main-container { flex-direction: column; width: 90%; }
        .left-panel { padding: 30px; }
        .right-panel { padding: 25px; }
    }
</style>
</head>
<body>
<div class="header">
    
</div>
<div class="main-container">
    <div class="left-panel">
        <div class="logo-circle">
            
        </div>
        <h2>¡Aprende inglés con confianza!</h2>
        <p>SpeakUp! es tu espacio para mejorar tus habilidades de comunicación en inglés de manera dinámica y divertida. Practica conversación, vocabulario y pronunciación con lecciones interactivas diseñadas para todos los niveles.</p>
    </div>

```

```

<div class="right-panel">
  
  <h1>Inicia sesión</h1>
  <form method="POST">
    <label for="email">Correo electrónico:</label>
    <input type="email" id="email" name="email" required value="<?= htmlspecialchars($_POST['email'] ?? '') ?>">
    <label for="password">Contraseña:</label>
    <input type="password" id="password" name="password" required>
    <button type="submit">Entrar</button>
    <div class="links">
      <button type="button" onclick="window.location.href='speakup_register.php'">
        ¿No tienes cuenta? ¡Crea una!
      </button><br>
      <button type="button" onclick="window.location.href='contraseña_olvidada.php'">
        ¿Olvidaste tu contraseña?
      </button>
    </div>
  </form>
</div>
</div>
<div id="toast" class="toast"></div>
<script>
function showToast(message) {
  const toast = document.getElementById('toast');
  toast.textContent = message;
  toast.classList.add('show');
  setTimeout(() => {
    toast.classList.remove('show');
  }, 2500);
}
<?php if(!empty($error)): ?>
  showToast("<?= $error ?>");
<?php endif; ?>
</script>
</body>
</html>

```

speakup_register.php

```
<?php
$host = 'localhost';
$db = 'speakup_db';
$user = 'root';
$pass = '';
$charset = 'utf8mb4';
$dsn = "mysql:host=$host;dbname=$db;charset=$charset";
try {
    $pdo = new PDO($dsn, $user, $pass);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e) {
    die("Error de conexión: " . $e->getMessage());
}
$message = "";
$toast_type = "";
$email_error = "";
$form_data = [
    'first_name'=> "",
    'last_name_p'=> "",
    'last_name_m'=> "",
    'age'=> "",
    'location'=> "",
    'email'=> "",
    'phone'=> "",
    'gender'=> "",
    'grade'=> ""
];
if($_SERVER['REQUEST_METHOD'] === 'POST') {
    $form_data['first_name'] = trim($_POST['first_name']);
    $form_data['last_name_p'] = trim($_POST['last_name_p']);
    $form_data['last_name_m'] = trim($_POST['last_name_m']);
    $form_data['age'] = (int) $_POST['age'];
    $form_data['location'] = trim($_POST['location']);
    $form_data['email'] = trim($_POST['email']);
    $form_data['phone'] = trim($_POST['phone']);
    $form_data['gender'] = $_POST['gender'];
    $form_data['grade'] = (int) $_POST['grade'];
    $password = $_POST['password'];
    $confirm_password = $_POST['confirm_password'];
    if(strlen($password) < 6){
        $message = "La contraseña debe tener al menos 6 caracteres.";
        $toast_type = 'error';
    } elseif($password !== $confirm_password){
        $message = "Las contraseñas no coinciden.";
        $toast_type = 'error';
    } else {
        $password_hash = password_hash($password, PASSWORD_DEFAULT);

        $sql = "INSERT INTO usuarios
            (first_name,last_name_p,last_name_m,age,location,email,phone,gender,grade,password)
            VALUES (:first_name,:last_name_p,:last_name_m,:age,:location,:email,:phone,:gender,:grade,:password)";
        $stmt = $pdo->prepare($sql);
        try {
            $stmt->execute([
                'first_name'=>$form_data['first_name'],
                'last_name_p'=>$form_data['last_name_p'],
                'last_name_m'=>$form_data['last_name_m'],
                'age'=>$form_data['age'],
                'location'=>$form_data['location'],
                'email'=>$form_data['email'],
                'phone'=>$form_data['phone'],
                'gender'=>$form_data['gender'],
                'grade'=>$form_data['grade'],
                'password'=>$password_hash
            ]);
        } catch (PDOException $e) {
            $message = "¡Cuenta creada con éxito! Redirigiendo al login...";
            $toast_type = 'success';
        }
    }
}
```

```

        $form_data = [
            'first_name'=>'', 'last_name_p'=>'', 'last_name_m'=>'', 'age'=>'',
            'location'=>'', 'email'=>'', 'phone'=>'', 'gender'=>'', 'grade'=>''
        ];
    } catch(PDOException $e) {
        if($e->getCode() == 23000 && !empty($form_data['email'])){
            $message = "Este correo ya está registrado.";
        } else {
            $message = "Error: " . $e->getMessage();
        }
        $toast_type = 'error';
    }
}
?>

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>SpeakUp! - Crear cuenta</title>
<link rel="icon" href="Logo_SpeakUp.png">
<style>
* { box-sizing: border-box; margin:0; padding:0; }
body {
    font-family:'Poppins',sans-serif;
    background:url('FondoSpeak.png') no-repeat center center fixed;
    background-size:cover;
    display:flex;flex-direction:column;align-items:center;
    min-height:100vh;position:relative;color:#3b2f00;overflow-x:hidden;
}
body::before {
    content:'';position:absolute;inset:0;
    background:linear-gradient(180deg, rgba(255,247,205,0.3), rgba(255,243,190,0.25));
    backdrop-filter: blur(3px); z-index:0;
}
.bubble{ position:absolute;bottom:-150px;border-radius:50%;background:rgba(255,255,255,0.85);filter: drop-shadow(0 0 10px
rgba(255,240,170,0.7)); animation:floatUp 6s linear infinite; z-index:-1;}
.b1 {width:50px;height:50px;left:5%;animation-delay:0s;}
.b2 {width:30px;height:30px;left:25%;animation-delay:1s;}
.b3 {width:60px;height:60px;left:50%;animation-delay:0.5s;}
.b4 {width:40px;height:40px;left:70%;animation-delay:1.5s;}
.b5 {width:50px;height:50px;left:85%;animation-delay:2s;}
@keyframes floatUp { 0% {transform:translateY(0) scale(1);opacity:0.85;} 50% {transform:translateY(-500px)
scale(1.15);opacity:1;} 100% {transform:translateY(-1000px) scale(1);opacity:0;}}
.header{width:100%;display:flex;justify-content:flex-end;padding:18px 30px 0 30px;z-index:2;position:relative;}
.logo-wrap img{width:200px;transition: transform 0.3s, filter 0.3s;}
.logo-wrap img:hover{transform:scale(1.15) rotate(-5deg);filter:drop-shadow(0 0 25px #FFD600);}
.wrap {display:flex;justify-content:center;align-items:flex-start;padding:26px 12px 60px 12px;z-index:2;width:100%;}
.container{background:rgba(255,255,255,0.95);border-radius:28px;padding:36px 40px;width:620px;max-width:calc(100%-
32px);border:3px solid #FFD600;box-shadow:0 14px 40px rgba(255,204,60,0.25);position:relative;transition: transform 0.2s ease, box-
shadow 0.2s ease;}
.container:hover {transform:translateY(-3px);box-shadow:0 20px 50px rgba(255,200,70,0.3);}
h1 {color:#FFB300;font-size:32px;font-weight:900;margin-bottom:22px;text-align:center;text-shadow:0 0 10px
rgba(255,250,220,0.9);}
.form-grid{display:grid;grid-template-columns:1fr 1fr;gap:14px 16px;margin-bottom:10px;}
.field{display:flex;flex-direction:column;}
.field label{margin-bottom:6px;font-weight:700;color:#4b3a00;font-size:14px;}
.field input, .field select {width:100%;padding:12px;border-radius:12px;border:2px solid #FFD857;background:#FFF9D9;font-
size:15px;outline:none;}
.field input:focus, .field select:focus{box-shadow:0 6px 18px rgba(255,210,80,0.45);transform:translateY(-1px);}
.full-row{grid-column:1/-1;}
.actions{margin-top:12px;display:flex;flex-direction:column;gap:12px;}
button.primary{background:linear-gradient(135deg,#FFD600,#FFB300);color:#4b2f00;border:none;border-
radius:14px;padding:14px;font-weight:800;font-size:16px;cursor:pointer;}
button.primary:hover{transform:scale(1.05) rotate(-1deg);box-shadow:0 14px 36px rgba(255,200,70,0.34);}
button.link {background:none;border:none;color:#FFB300;text-decoration:underline;cursor:pointer;font-weight:700;}
.small-note{color:#7a6730;font-size:13px;margin-top:6px;text-align:center;}

```



```

        .toast {position:fixed;top:20px;right:20px;padding:14px 22px;border-radius:12px;font-weight:700;box-shadow:0 6px 20px
        rgba(0,0,0,0.2);opacity:0;transform:translateY(-20px);transition:opacity 0.4s ease, transform 0.4s ease;z-index:9999;}
        .toast.show{opacity:1;transform:translateY(0);}
        .toast.success{background:#C9F191;color:#2B4200;}
        .toast.error{background:#FFADAD;color:#4b2f00;}
        @media(max-width:700px){.form-grid{grid-template-columns:1fr;}.header{padding:12px 18px 0 18px;}.header
        img{width:150px;}.container{padding:26px 20px;width:calc(100%-20px);}}
    </style>
</head>
<body>

    <div class="bubble b1"></div>
    <div class="bubble b2"></div>
    <div class="bubble b3"></div>
    <div class="bubble b4"></div>
    <div class="bubble b5"></div>

    <div class="header">
        <div class="logo-wrap">
            
        </div>
    </div>

    <div class="wrap">
        <div class="container">
            <h1>✦ Crear nueva cuenta ✦</h1>
            <form id="registerForm" method="POST" autocomplete="off">
                <div class="form-grid">
                    <div class="field">
                        <label for="first_name">Nombre(s):</label>
                        <input type="text" id="first_name" name="first_name" required placeholder="Ej. Sofía" value="<?=  

htmlspecialchars($form_data['first_name']) ?>">
                    </div>
                    <div class="field">
                        <label for="last_name_p">Apellido Paterno:</label>
                        <input type="text" id="last_name_p" name="last_name_p" required placeholder="Ej. López" value="<?=  

htmlspecialchars($form_data['last_name_p']) ?>">
                    </div>
                    <div class="field">
                        <label for="last_name_m">Apellido Materno:</label>
                        <input type="text" id="last_name_m" name="last_name_m" required placeholder="Ej. Martínez" value="<?=  

htmlspecialchars($form_data['last_name_m']) ?>">
                    </div>
                    <div class="field">
                        <label for="age">Edad:</label>
                        <input type="number" id="age" name="age" min="3" max="120" required placeholder="Ej. 9" value="<?=  

htmlspecialchars($form_data['age']) ?>">
                    </div>
                    <div class="field">
                        <label for="location">Localidad:</label>
                        <input type="text" id="location" name="location" required placeholder="Ej. Ciudad de México" value="<?=  

htmlspecialchars($form_data['location']) ?>">
                    </div>
                    <div class="field full-row">
                        <label for="email">Correo electrónico:</label>
                        <input type="email" id="email" name="email" required placeholder="ejemplo@correo.com" value="<?=  

htmlspecialchars($form_data['email']) ?>">
                    </div>
                    <div class="field">
                        <label for="phone">Teléfono:</label>
                        <input type="tel" id="phone" name="phone" required placeholder="Ej. 55 1234 5678" value="<?=  

htmlspecialchars($form_data['phone']) ?>">
                    </div>
                    <div class="field">
                        <label for="gender">Género:</label>
                        <select id="gender" name="gender" required>
                            <option value="">Selecciona</option>
                            <option value="masculino" <?=$form_data['gender']=='masculino'?selected:" ?>>Masculino</option>
                            <option value="femenino" <?=$form_data['gender']=='femenino'?selected:" ?>>Femenino</option>
                            <option value="otro" <?=$form_data['gender']=='otro'?selected:" ?>>Otro</option>

```

```

        </select>
    </div>
    <div class="field">
        <label for="grade">Grado:</label>
        <select id="grade" name="grade" required>
            <option value="">Selecciona</option>
            <option value="1" <?= $form_data['grade']==1?'selected':" ?">>1° de Primaria</option>
            <option value="2" <?= $form_data['grade']==2?'selected':" ?">>2° de Primaria</option>
            <option value="3" <?= $form_data['grade']==3?'selected':" ?">>3° de Primaria</option>
            <option value="4" <?= $form_data['grade']==4?'selected':" ?">>4° de Primaria</option>
        </select>
    </div>

    <div class="field full-row">
        <label for="password">Contraseña:</label>
        <input type="password" id="password" name="password" required placeholder="Mínimo 6 caracteres"
minlength="6">
    </div>
    <div class="field full-row">
        <label for="confirm_password">Confirmar contraseña:</label>
        <input type="password" id="confirm_password" name="confirm_password" required placeholder="Repite la
contraseña" minlength="6">
    </div>
</div>
<div class="actions">
    <button type="submit" class="primary" id="createBtn">👤 Crear cuenta</button>
    <button type="button" class="link" id="backBtn">← Regresar al inicio</button>
    <div class="small-note">Al registrarte aceptas nuestras políticas de privacidad 📄</div>
</div>
</form>
</div>
</div>

<div id="toast" class="toast"></div>

<script>
function showToast(message, type='error', redirect=false){
    const toast = document.getElementById('toast');
    toast.textContent = message;
    toast.className = 'toast show '+type;
    setTimeout(()=>{
        toast.classList.remove('show');
        if(type==='success' && redirect){
            window.location.href='login.php';
        }
    },2500);
}
<?php if($message != ""): ?>
showToast("<?= $message ?>", "<?= $toast_type ?>", <?= $toast_type==='success'?true:false' ?>);
<?php endif; ?>

const clickSound = new Audio('click.mp3');
document.getElementById('backBtn').addEventListener('click', ()=>{
    clickSound.currentTime=0; clickSound.play();
    setTimeout(()=>window.location.href='login.php',200);
});
document.getElementById('registerForm').addEventListener('submit',function(e){
    clickSound.currentTime=0; clickSound.play();
});
</script>

</body>

```

contraseña_olvidada.php

```
<?php
$host = 'localhost';
$db = 'speakup_db';
$user = 'root';
$pass = '';
$charset = 'utf8mb4';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";

try {
    $pdo = new PDO($dsn, $user, $pass);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch(PDOException $e) {
    die("Error de conexión: " . $e->getMessage());
}
$message = "";
$email_error = "";

if($_SERVER['REQUEST_METHOD'] === 'POST') {
    $email = trim($_POST['email']);
    $new_password = $_POST['new_password'];
    $confirm_password = $_POST['confirm_password'];

    if(strlen($new_password) < 6){
        $message = "La contraseña debe tener al menos 6 caracteres.";
    } elseif($new_password !== $confirm_password){
        $message = "Las contraseñas no coinciden.";
    } else {
        $password_hash = password_hash($new_password, PASSWORD_DEFAULT);

        $sql = "UPDATE users SET password_hash = :password_hash WHERE email = :email";
        $stmt = $pdo->prepare($sql);
        $stmt->execute(['password_hash' => $password_hash, 'email' => $email]);

        if($stmt->rowCount() > 0){
            $message = "¡Contraseña actualizada con éxito! Serás redirigido al login...";
        } else {
            $email_error = "Este correo no está registrado.";
        }
    }
}

?>

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Speak Up! - Recuperar contraseña</title>
<link rel="icon" href="Logo_SpeakUp.png">
<style>
* { box-sizing: border-box; margin: 0; padding: 0; }
body {
    font-family: 'Poppins', sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    display: flex;
    flex-direction: column;
    align-items: center;
    min-height: 100vh;
    position: relative;
    color: #3b2f00;
    overflow-x: hidden;
}
body::before {
    content: "";
    position: absolute;
    inset: 0;
    background: linear-gradient(180deg, rgba(255,247,205,0.3), rgba(255,243,190,0.25));
}
```

```

        backdrop-filter: blur(3px);
        z-index: 0;
    }
    .bubble { position: absolute; bottom: -150px; border-radius: 50%; background: rgba(255,255,255,0.85); filter: drop-shadow(0 0 10px rgba(255,240,170,0.7)); animation: floatUp 6s linear infinite; z-index: -1; }
    .b1 { width:50px;height:50px;left:5%; animation-delay:0s; }
    .b2 { width:30px;height:30px;left:25%; animation-delay:1s; }
    .b3 { width:60px;height:60px;left:50%; animation-delay:0.5s; }
    .b4 { width:40px;height:40px;left:70%; animation-delay:1.5s; }
    .b5 { width:50px;height:50px;left:85%; animation-delay:2s; }
    @keyframes floatUp { 0% { transform: translateY(0) scale(1); opacity: 0.85; } 50% { transform: translateY(-500px) scale(1.15); opacity: 1; } 100% { transform: translateY(-1000px) scale(1); opacity: 0; } }

    .header { width: 100%; display: flex; justify-content: flex-end; padding: 18px 30px 0 30px; z-index: 2; position: relative; }
    .header img { width: 200px; transition: transform 0.3s, filter 0.3s; }
    .header img:hover { transform: scale(1.15) rotate(-5deg); filter: drop-shadow(0 0 25px #FFD600); }

    .wrap { display: flex; justify-content: center; align-items: flex-start; padding: 26px 12px 60px 12px; z-index: 2; width: 100%; }
    .container { background: rgba(255,255,255,0.95); border-radius: 28px; padding: 36px 40px; width: 420px; max-width: calc(100% - 32px); border: 3px solid #FFD600; box-shadow: 0 14px 40px rgba(255,204,60,0.25); position: relative; text-align: center; transition: transform 0.2s ease, box-shadow 0.2s ease; }
    .container:hover { transform: translateY(-3px); box-shadow: 0 20px 50px rgba(255,200,70,0.3); }

    h1 { color: #FFB300; font-size: 28px; font-weight: 900; margin-bottom: 22px; text-shadow: 0 0 10px rgba(255,250,220,0.9); }

    input[type="email"], input[type="text"] {
        width: 100%; padding: 12px; border-radius: 12px; border: 2px solid #FFD857; background: #FFF9D9; font-size: 15px; outline: none; margin-bottom: 15px;
    }
    input:focus { box-shadow: 0 6px 18px rgba(255,210,80,0.45); transform: translateY(-1px); }

    button { width: 100%; padding: 14px; margin-top: 5px; background: linear-gradient(135deg, #FFD600, #FFB300); color: #4b2f00; border: none; border-radius: 14px; font-weight: 800; font-size: 16px; cursor: pointer; transition: transform .2s ease, box-shadow .2s ease; }
    button:hover { transform: scale(1.05) rotate(-1deg); box-shadow: 0 14px 36px rgba(255,200,70,0.34); }

    .text-button { background: none; border: none; color: #FFB300; text-decoration: underline; cursor: pointer; font-weight: 700; margin-top: 10px; }
    .text-button:hover { color: #FFD600; }

    .error-msg { color: red; font-size: 13px; margin-bottom: 6px; }

    .toast { position: fixed; top: 20px; right: 20px; background: #FFD600; color: #4b2f00; padding: 14px 22px; border-radius: 12px; font-weight: 700; box-shadow: 0 6px 20px rgba(0,0,0,0.2); opacity: 0; transform: translateY(-20px); transition: opacity 0.4s ease, transform 0.4s ease; z-index: 9999; }
    .toast.show { opacity: 1; transform: translateY(0); }

    @media (max-width: 500px) { .container { width: 90%; padding: 25px; } .header img { width: 150px; } }
</style>
</head>
<body>

<div class="bubble b1"></div>
<div class="bubble b2"></div>
<div class="bubble b3"></div>
<div class="bubble b4"></div>
<div class="bubble b5"></div>

<div class="header">
    
</div>

<div class="wrap">
    <div class="container">
        <h1>🔑 Recuperar contraseña</h1>

        <form method="POST">
            <label for="email">Correo registrado:</label>
            <input type="email" id="email" name="email" required value="<?= isset($email) ? htmlspecialchars($email) : ">">
            <?php if($email_error != ""): ?>
                <div class="error-msg"><?=$email_error?></div>
            </?php

```

```

<?php endif; ?>

<label for="new_password">Nueva contraseña:</label>
<input type="text" id="new_password" name="new_password" required minlength="6">

<label for="confirm_password">Confirmar contraseña:</label>
<input type="text" id="confirm_password" name="confirm_password" required minlength="6">

<button type="submit">Actualizar contraseña</button>
<button type="button" class="text-button" onclick="window.location.href='login.php'">Regresar al inicio</button>
</form>
</div>
</div>

<div id="toast" class="toast"></div>

<script>
function showToast(message) {
    const toast = document.getElementById('toast');
    toast.textContent = message;
    toast.classList.add('show');
    setTimeout(() => { toast.classList.remove('show'); }, 2000);
}

<?php if($message != ""): ?>
showToast("<?=$message ?>");
setTimeout() => { window.location.href='login.php'; }, 2000);
<?php endif; ?>
</script>

</body>
</html>

```

principalphp.php

```
<?php
$lifetime = 60 * 60 * 24 * 365 * 10;
ini_set('session.gc_maxlifetime', $lifetime);
session_set_cookie_params([
    'lifetime' => $lifetime,
    'path' => '/',
    'secure' => false,
    'httponly' => true,
    'samesite' => 'Lax'
]);

session_start();
session_regenerate_id(true);
include 'db.php';
$user_id = $_SESSION['user_id'] ?? null;

$user = null;
if ($user_id) {
    $sql = "SELECT first_name, last_name_p, last_name_m, grade FROM usuarios WHERE id = ?";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("i", $user_id);
    $stmt->execute();
    $result = $stmt->get_result();
    $user = $result->fetch_assoc();
}

$first_name = $user['first_name'] ?? 'Usuario';
$last_name_p = $user['last_name_p'] ?? '';
$grade = $user['grade'] ?? 0;

$full_name = $user ? htmlspecialchars($user['first_name'] . " " . $user['last_name_p']): 'Usuario';

switch((int)$grade){
    case 1: $course_link = "cursos_1ro.php"; break;
    case 2: $course_link = "cursos_2do.php"; break;
    case 3: $course_link = "cursos_3ro.php"; break;
    case 4: $course_link = "cursos_4to.php"; break;
    default: $course_link = "cursos.php";
}
?>

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Speak Up! - Inicio</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    color: #4b2f00;
    overflow-x: hidden;
}

@keyframes fadeInUp {
    from { opacity: 0; transform: translateY(30px); }
    to { opacity: 1; transform: translateY(0); }
}
@keyframes float {
    0%, 100% { transform: translateY(0px); }
    50% { transform: translateY(-8px); }
}
@keyframes shine {
    0% { background-position: -200px 0; }
```

```

    100% { background-position: 200px 0; }
  }
  @keyframes bubbleFloat {
    0% { transform: translateY(0) scale(1); opacity: 0.8; }
    50% { opacity: 1; }
    100% { transform: translateY(-500px) scale(1.2); opacity: 0; }
  }

.navbar {
  display: flex;
  justify-content: space-between;
  align-items: center;
  background-color: #ffffffee;
  padding: 15px 30px;
  position: sticky;
  top: 0;
  z-index: 100;
  box-shadow: 0 4px 12px rgba(0,0,0,0.1);
  backdrop-filter: blur(8px);
  border-bottom: 3px solid #FFD600;
}

.navbar img { height: 50px; }
.nav-links { display: flex; align-items: center; gap: 18px; }
.nav-links button {
  background: none;
  border: none;
  font-weight: 600;
  cursor: pointer;
  font-size: 15px;
  transition: 0.3s;
  color: #4b2f00;
}

.nav-links button:hover {
  color: #FFB300;
  transform: scale(1.05);
  text-shadow: 0 0 10px #FFD600aa;
}

.nav-profile {
  width: 45px; height: 45px;
  border-radius: 50%; overflow: hidden; cursor: pointer;
  border: 2px solid #FFD600;
  box-shadow: 0 0 10px #FFD60088;
  transition: 0.3s;
}

.nav-profile:hover { transform: scale(1.1); }

.hero {
  position: relative;
  text-align: center;
  color: #4b2f00;
  height: 370px;
  background: linear-gradient(135deg, #FFD600cc, #FFB300cc), url('FondoHero.jpg') center/cover no-repeat;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  border-bottom-left-radius: 40px;
  border-bottom-right-radius: 40px;
  padding: 0 20px;
  box-shadow: 0 6px 20px rgba(0,0,0,0.15);
  animation: fadeInUp 1s ease forwards;
  overflow: hidden;
}

.bubble {
  position: absolute;
  bottom: -100px;
  border-radius: 50%;

```

```

    opacity: 0.7;
    background: radial-gradient(circle, rgba(255,255,255,0.8) 0%, rgba(255,214,0,0.6) 70%);
    animation: bubbleFloat 10s linear infinite;
}
.bubble:nth-child(1) { left: 10%; width: 40px; height: 40px; animation-delay: 1s; }
.bubble:nth-child(2) { left: 25%; width: 25px; height: 25px; animation-delay: 3s; }
.bubble:nth-child(3) { left: 45%; width: 50px; height: 50px; animation-delay: 2s; }
.bubble:nth-child(4) { left: 65%; width: 35px; height: 35px; animation-delay: 4s; }
.bubble:nth-child(5) { left: 80%; width: 60px; height: 60px; animation-delay: 0s; }

.hero::after {
  content: "";
  position: absolute;
  top: 0; left: -100%;
  width: 50%; height: 100%;
  background: linear-gradient(120deg, transparent 0%, rgba(255,255,255,0.6) 50%, transparent 100%);
  animation: shine 5s infinite linear;
}
.hero h1 {
  font-size: 40px;
  margin-bottom: 10px;
  text-shadow: 2px 2px 8px rgba(75,47,0,0.4);
  animation: float 5s ease-in-out infinite;
}
.hero p {
  font-size: 19px;
  max-width: 600px;
  text-shadow: 1px 1px 6px rgba(75,47,0,0.4);
}
.hero button {
  margin-top: 25px;
  padding: 14px 32px;
  border-radius: 14px;
  border: none;
  background: linear-gradient(135deg, #FFD600, #FFB300);
  color: #4b2f00;
  font-weight: bold;
  font-size: 17px;
  cursor: pointer;
  transition: 0.3s;
  box-shadow: 0 4px 12px rgba(0,0,0,0.2);
  position: relative;
  overflow: hidden;
}
.hero button::before {
  content: "";
  position: absolute;
  top: 0; left: -75px;
  width: 50px; height: 100%;
  background: linear-gradient(120deg, transparent, rgba(255,255,255,0.6), transparent);
  transform: skewX(-20deg);
  transition: 0.6s;
}
.hero button:hover::before { left: 120%; }
.hero button:hover {
  transform: scale(1.08);
  box-shadow: 0 0 20px #FFD600aa;
}

.table-section { padding: 50px 20px; text-align: center; animation: fadeInUp 1s ease forwards; }
.table-section h2 { font-size: 28px; margin-bottom: 30px; }
.grades-table {
  margin: 0 auto; width: 75%;
  display: grid; grid-template-columns: repeat(2, 1fr);
  grid-template-rows: repeat(2, 200px);
  gap: 25px;
}
.grade-card {

```



```

background: #ffffffcc; backdrop-filter: blur(10px);
border-radius: 22px; display: flex;
justify-content: center; align-items: center;
font-size: 24px; font-weight: bold;
cursor: pointer; position: relative;
overflow: hidden; transition: all 0.3s ease;
box-shadow: 0 5px 15px rgba(0,0,0,0.1);
}
.grade-card:hover { transform: scale(1.05); box-shadow: 0 0 20px #FFD600aa; }
.grade-card img {
position: absolute; top: 0; left: 0;
width: 100%; height: 100%; object-fit: cover;
border-radius: 22px; opacity: 0;
transition: opacity 0.4s ease-in-out; z-index: 1;
}
.grade-card span { position: relative; z-index: 2; pointer-events: none; }
.grade-card:hover img { opacity: 1; }

.why-section {
padding: 50px 20px; background: #fff8cc;
text-align: center; animation: fadeInUp 1s ease forwards;
}
.why-section h2 { font-size: 28px; margin-bottom: 25px; }
.why-cards { display: flex; justify-content: center; flex-wrap: wrap; gap: 25px; }
.why-card {
background: #ffffffcc; backdrop-filter: blur(10px);
border-radius: 20px; padding: 25px;
width: 250px; box-shadow: 0 5px 15px rgba(0,0,0,0.1);
transition: all 0.3s; animation: float 6s ease-in-out infinite;
}
.why-card:hover { transform: translateY(-5px) scale(1.03); box-shadow: 0 0 20px #FFD600aa; }

footer {
background: linear-gradient(135deg, #FFD600, #FFB300);
color: #4b2f00; padding: 30px 20px;
text-align: center; border-top-left-radius: 30px;
border-top-right-radius: 30px; box-shadow: 0 -4px 15px rgba(0,0,0,0.1);
}
footer a { color: #4b2f00; margin: 0 10px; font-size: 20px; transition: 0.2s; }
footer a:hover { transform: scale(1.2); text-shadow: 0 0 10px #FFB300; }
footer p { margin-top: 15px; font-size: 14px; }

@media(max-width:900px){
.grades-table { grid-template-columns: 1fr; width: 90%; }
.navbar { flex-direction: column; gap: 10px; }
.hero h1 { font-size: 30px; }
}
</style>
</head>
<body>

<div class="navbar">

<div class="nav-links">
<button onclick="window.location.href='principalphp.php'">Inicio</button>
<button onclick="window.location.href='mi_perfil.php'">Mi perfil</button>
<button onclick="window.location.href='cursos.php'">Cursos</button>
<button onclick="window.location.href='mi_progreso.php'">Mi Progreso</button>
<button onclick="window.location.href='contacto.php'">Contacto</button>
<button onclick="window.location.href='login.php'">Cerrar sesión</button>
<div class="nav-profile" onclick="window.location.href='mi_perfil.php'">

</div>
</div>
</div>

<div class="hero">

```

```

<div class="bubble"></div><div class="bubble"></div><div class="bubble"></div>
<div class="bubble"></div><div class="bubble"></div>

<h1>¡Bienvenid@, <?= htmlspecialchars($full_name); ?>!</h1>
<p>Aprende inglés jugando y divirtiéndote con nuestros cursos gratuitos de Speak Up! para primaria.</p>
<button onclick="window.location.href='<?= $course_link ?>'>Explora los cursos</button>
</div>

<div class="table-section">
<h2>¿Qué grado eres?</h2>
<div class="grades-table">
<div class="grade-card" onclick="window.location.href='cursos_1ro.php'">
<span>1ro de Primaria</span>
</div>
<div class="grade-card" onclick="window.location.href='cursos_2do.php'">
<span>2do de Primaria</span>
</div>
<div class="grade-card" onclick="window.location.href='cursos_3ro.php'">
<span>3ro de Primaria</span>
</div>
<div class="grade-card" onclick="window.location.href='cursos_4to.php'">
<span>4to de Primaria</span>
</div>
</div>
</div>

<div class="why-section">
<h2>¿Por qué aprender con Speak Up!</h2>
<div class="why-cards">
<div class="why-card">
<h3>🆓 Gratis y divertido</h3>
<p>Todos nuestros cursos son gratuitos y diseñados para que los niños disfruten aprendiendo inglés.</p>
</div>
<div class="why-card">
<h3>🎮 Aprende jugando</h3>
<p>Lecciones basadas en actividades lúdicas que hacen que cada palabra cobre vida.</p>
</div>
<div class="why-card">
<h3>📏 Lecciones cortas, grandes resultados</h3>
<p>Diseñadas para mantener la atención y el ritmo de aprendizaje ideal para niños.</p>
</div>
</div>
</div>

<footer>
<p>Síguenos en nuestras redes:</p>
<a href="#">📷</a>
<a href="#">📺</a>
<a href="#">📞</a>
<a href="#">🌐</a>
<p>© 2025 Speak Up!. Todos los derechos reservados.</p>
</footer>
</body>
</html>

```

cursos_1ro.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$sql = "SELECT first_name FROM usuarios WHERE id = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$user = $result->fetch_assoc();

$first_name = $user['first_name'];

$messages = [
    "List@ para el siguiente reto, $first_name?",
    "¿Cuál es nuestro siguiente desafío, $first_name?",
    "¡Vamos a aprender algo nuevo, $first_name!",
    "¡Preparad@ para el próximo reto, $first_name?"
];
$random_msg = $messages[array_rand($messages)];
?>
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>1er Grado - Cursos de Inglés</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    color: #4b2f00;
}

.navbar {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #ffffff;
    padding: 10px 30px;
    position: sticky;
    top: 0;
    z-index: 100;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}

.navbar img.logo { height: 50px; }

.navbar button {
    background: linear-gradient(145deg, #FFD600 0%, #FFB300 100%);
    border: none;
    border-radius: 20px;
    padding: 6px 14px;
    cursor: pointer;
    font-weight: 700;
    font-size: 14px;
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 4px 8px rgba(0,0,0,0.2);
    transition: all 0.3s ease;
}

.navbar button:hover {
```

```

    transform: translateY(-3px) scale(1.05);
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 8px 15px rgba(0,0,0,0.25);
  }

.hero {
  text-align: center;
  margin: 30px 20px;
}

.hero h1 {
  font-size: 48px;
  font-weight: 900;
  color: #FFB300;
  text-shadow: 2px 2px 10px rgba(75,47,0,0.5);
  margin-bottom: 15px;
}

.hero p {
  font-size: 22px;
  margin-bottom: 40px;
}

@keyframes riseUp {
  0% { transform: translateY(40px); opacity: 0; }
  100% { transform: translateY(0); opacity: 1; }
}

.courses-grid {
  display: grid;
  grid-template-columns: repeat(2, 1fr);
  grid-template-rows: repeat(2, 180px);
  gap: 25px;
  padding: 0 20px 50px 20px;
}

.course-card {
  position: relative;
  border-radius: 30px;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  font-size: 20px;
  font-weight: 800;
  cursor: pointer;
  color: #fff;
  transition: transform 0.4s, box-shadow 0.4s, background 0.4s;
  text-align: center;
  padding: 12px 18px;
  letter-spacing: 1px;
  box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 6px 15px rgba(0,0,0,0.2);
  opacity: 0;
  animation: riseUp 0.6s forwards;
}

.course-card span {
  font-size: 16px;
  font-weight: 500;
  margin-top: 8px;
}

.course-card:nth-child(1) { animation-delay: 0.1s; background: linear-gradient(145deg, #FFD600, #FFB300); }
.course-card:nth-child(2) { animation-delay: 0.3s; background: linear-gradient(145deg, #FFB300, #FFD600); }
.course-card:nth-child(3) { animation-delay: 0.5s; background: linear-gradient(145deg, #FFC95C, #FF9F1C); }
.course-card:nth-child(4) { animation-delay: 0.7s; background: linear-gradient(145deg, #FFDA6C, #FFD166); }

.course-card:hover {
  transform: translateY(-8px) scale(1.07);
  box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 12px 25px rgba(0,0,0,0.3);
}

```

```

@media(max-width:700px){
  .hero h1 { font-size:36px; }
  .hero p { font-size:18px; }
  .course-card { font-size:18px; height:140px; padding:10px 14px; }
  .courses-grid { grid-template-columns: 1fr; grid-template-rows: repeat(4,140px); }
}
</style>
</head>
<body>

<div class="navbar">
  
  <button onclick="window.location.href=principalphp.php">Volver al inicio</button>
</div>

<div class="hero">
  <h1>1er Grado</h1>
  <p><?php echo htmlspecialchars($random_msg); ?></p>
</div>

<div class="courses-grid">
  <div class="course-card" onclick="window.location.href='Writing1.php'">
    Writing
    <span>Escritura</span>
  </div>
  <div class="course-card" onclick="window.location.href='Listening1.php'">
    Listening
    <span>Escucha</span>
  </div>
  <div class="course-card" onclick="window.location.href='Grammar1.php'">
    Grammar
    <span>Gramática</span>
  </div>
  <div class="course-card" onclick="window.location.href='juegos1.php'">
    Games
    <span>Juegos</span>
  </div>
</div>

</body>
</html>

```

juegos1.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$sql = "SELECT first_name FROM usuarios WHERE id = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$user = $result->fetch_assoc();

$first_name = $user['first_name'];

$messages = [
    "List@ para el siguiente reto, $first_name?",
    "¿Cuál es nuestro siguiente desafío, $first_name?",
    "¡Vamos a aprender algo nuevo, $first_name!",
    "¡Preparad@ para el próximo reto, $first_name?"
];
$random_msg = $messages[array_rand($messages)];
?>
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>1er Grado - Cursos de Inglés</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    color: #4b2f00;
}

.navbar {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #ffffff;
    padding: 10px 30px;
    position: sticky;
    top: 0;
    z-index: 100;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}

.navbar img.logo { height: 50px; }

.navbar button {
    background: linear-gradient(145deg, #FFD600 0%, #FFB300 100%);
    border: none;
    border-radius: 20px;
    padding: 6px 14px; /* más delgado */
    cursor: pointer;
    font-weight: 700;
    font-size: 14px;
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 4px 8px rgba(0,0,0,0.2);
    transition: all 0.3s ease;
}

.navbar button:hover {
```

```

    transform: translateY(-3px) scale(1.05);
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 8px 15px rgba(0,0,0,0.25);
}

.hero {
    text-align: center;
    margin: 30px 20px;
}

.hero h1 {
    font-size: 48px;
    font-weight: 900;
    color: #FFB300;
    text-shadow: 2px 2px 10px rgba(75,47,0,0.5);
    margin-bottom: 15px;
}

.hero p {
    font-size: 22px;
    margin-bottom: 40px;
}

@keyframes riseUp {
    0% { transform: translateY(40px); opacity: 0; }
    100% { transform: translateY(0); opacity: 1; }
}

.courses-grid {
    display: grid;
    grid-template-columns: repeat(2, 1fr);
    grid-template-rows: repeat(3, 140px);
    gap: 25px;
    padding: 0 20px 50px 20px;
}

.course-card {
    position: relative;
    border-radius: 30px;
    display: flex;
    justify-content: center;
    align-items: center;
    font-size: 20px;
    font-weight: 800;
    cursor: pointer;
    color: #fff;
    transition: transform 0.4s, box-shadow 0.4s, background 0.4s;
    text-align: center;
    padding: 12px 18px; /* más delgado */
    letter-spacing: 1px;
    box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 6px 15px rgba(0,0,0,0.2);
    opacity: 0;
    animation: riseUp 0.6s forwards;
}

.course-card:nth-child(1) { animation-delay: 0.1s; background: linear-gradient(145deg, #FFD600, #FFB300); }
.course-card:nth-child(2) { animation-delay: 0.3s; background: linear-gradient(145deg, #FFB300, #FFD600); }
.course-card:nth-child(3) { animation-delay: 0.5s; background: linear-gradient(145deg, #FFC95C, #FF9F1C); }
.course-card:nth-child(4) { animation-delay: 0.7s; background: linear-gradient(145deg, #FFDA6C, #FFD166); }
.course-card:nth-child(5) { animation-delay: 0.9s; background: linear-gradient(145deg, #FFE27A, #FFD166); }
.course-card:nth-child(6) { animation-delay: 1.1s; background: linear-gradient(145deg, #FFD966, #FFB347); }

.course-card:hover {
    transform: translateY(-8px) scale(1.07);
    box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 12px 25px rgba(0,0,0,0.3);
}

@media(max-width:700px){
    .hero h1 { font-size:36px; }
    .hero p { font-size:18px; }
    .course-card { font-size:18px; height:120px; padding:10px 14px; }
}

```

```

        .courses-grid { grid-template-columns: 1fr; grid-template-rows: repeat(6,120px); }
    }
</style>
</head>
<body>

<div class="navbar">
    
    <button onclick="window.location.href='cursos_1ro.php'">Volver a los cursos</button>
</div>

<div class="hero">
    <h1>1er Grado</h1>
    <p><?php echo htmlspecialchars($random_msg); ?></p>
</div>

<div class="courses-grid">
    <div class="course-card" onclick="window.location.href='memorama1.php'">Memorama</div>
    <div class="course-card" onclick="window.location.href='trivial.php'">Trivia</div>
    <div class="course-card" onclick="window.location.href='colores1.php'">Colores</div>
    <div class="course-card" onclick="window.location.href='vocales1.php'">Vocales</div>
    <div class="course-card" onclick="window.location.href='numeros1.php'">Números</div>
    <div class="course-card" onclick="window.location.href='animales1.php'">Animales</div>
</div>

</body>
</html>

```


Writing1.php

```
<?php
session_start();
include 'db.php';
if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];

// Obtener nombre
$stmt = $conn->prepare("SELECT first_name FROM usuarios WHERE id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$user = $res->fetch_assoc();
$first_name = $user['first_name'];

// Obtener puntaje actual
$stmt2 = $conn->prepare("SELECT writing1 FROM puntajes WHERE user_id=?");
$stmt2->bind_param("i", $user_id);
$stmt2->execute();
$res2 = $stmt2->get_result();
$scoreData = $res2->fetch_assoc();
$currentScore = $scoreData['writing1'];

$questions = [
    ['sentence' => 'Good (☀️)', 'answer' => 'morning'],
    ['sentence' => 'Good (🌙)', 'answer' => 'night'],
    ['sentence' => 'Thank (🙏)', 'answer' => 'you'],
    ['sentence' => 'I (❤️) you', 'answer' => 'love'],
    ['sentence' => 'Happy (🎂)', 'answer' => 'birthday'],
    ['sentence' => 'A (🐶) says woof!', 'answer' => 'dog'],
    ['sentence' => 'A (🐱) says meow!', 'answer' => 'cat'],
    ['sentence' => 'A (🚗) goes vroom!', 'answer' => 'car'],
    ['sentence' => 'A (☀️) a day', 'answer' => 'sunny'],
    ['sentence' => 'Have a (😊) day!', 'answer' => 'nice']
];
shuffle($questions);

shuffle($questions);
?>
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<title>Writing 1 - Speak Up!</title>
<link href="https://fonts.googleapis.com/css2?family=Comic+Neue:wght@300;700&display=swap" rel="stylesheet">
<style>
#restart-container a {
    background: #FFB300;
    padding: 14px 26px;
    border-radius: 15px;
    font-weight: 700;
    color: #4b2f00;
    text-decoration: none;
    display: inline-block;
    margin-top: 30px;
    font-size: 20px;
    box-shadow: 0 4px 0 #d89a00;
    transition: all 0.2s ease-in-out;
}
#restart-container a:hover {
    transform: scale(1.08);
    box-shadow: 0 6px 10px rgba(0,0,0,0.2);
    background: #FFD54F;
}
```

```

        body {font-family:'Comic Neue',sans-serif;background:linear-gradient(to top,#FFF9D9,#FFE57F);margin:0;padding:20px;text-align:center;overflow:hidden;}
        .bubble {position:absolute;bottom:-150px;border-radius:50%;background:rgba(255,255,255,0.6);animation:floatUp 6s linear infinite;}
        @keyframes floatUp {0% {transform:translateY(0);opacity:0.7;} 50% {transform:translateY(-500px);opacity:1;} 100% {transform:translateY(-1000px);opacity:0;}}
        .b1 {width:50px;height:50px;left:5%;animation-delay:0s;} .b2 {width:30px;height:30px;left:25%;animation-delay:1s;}
        .b3 {width:60px;height:60px;left:50%;animation-delay:0.5s;} .b4 {width:40px;height:40px;left:70%;animation-delay:1.5s;}
        .b5 {width:50px;height:50px;left:85%;animation-delay:2s;}
        #word-box {font-size:45px;margin:30px 0;}
        input {padding:10px;font-size:18px;border-radius:8px;border:2px solid #FFD600;text-align:center;transition:box-shadow 0.3s;}
        button {padding:10px 20px;font-size:18px;background:#FFD600;border:none;border-radius:10px;cursor:pointer;margin-left:10px;}
        .progress-container {width:50%;background:#eee;margin:20px auto;border-radius:10px;}
        .progress-bar {height:20px;width:0%;background:#FFEB3B;border-radius:10px;transition:width 0.5s;}
        .toast {position:absolute;top:80px;left:50%;transform:translateX(-50%);background:#FFB300;padding:12px 20px;border-radius:12px;font-weight:700;color:#4b2f00;opacity:0;transition:opacity 0.5s,transform 0.5s;pointer-events:none;}
        .toast.show {opacity:1;transform:translateX(-50%) translateY(-10px);}
        #confetti-canvas {position:absolute;top:0;left:0;width:100%;height:100%;pointer-events:none;}
        #timer {font-size:20px;font-weight:bold;background:#FFB300;padding:8px 15px;border-radius:10px;display:inline-block;margin-top:20px;}
        #start-overlay {position:fixed;top:0;left:0;width:100%;height:100%;background:rgba(255,243,172,0.92);display:flex;flex-direction:column;justify-content:center;align-items:center;z-index:9999;text-align:center;color:#4b2f00;}
        #start-overlay h1 {font-size:36px;font-weight:700;animation:bounce 1.5s infinite;padding:10px 25px;border-radius:20px;background:#FFB300;box-shadow:0 0 18px #FFDD67;border:3px solid #F7C843;}
        @keyframes bounce {0%,100% {transform:translateY(0);} 50% {transform:translateY(-6px);}}

        /* Nueva animación del countdown */
        #countdown {
            position: fixed;
            top: 0; left: 0;
            width: 100%; height: 100%;
            display: flex;
            justify-content: center;
            align-items: center;
            background: rgba(255, 243, 172, 0.97);
            z-index: 9998;
            font-size: 200px;
            font-weight: 900;
            color: #4b2f00;
            text-shadow: 0 0 25px #FFD600, 0 0 45px #FFB300;
            animation: fadeIn 0.3s ease-in-out;
        }
        @keyframes fadeIn {
            from { opacity: 0; }
            to { opacity: 1; }
        }
        #count-number {
            animation: pop 0.6s ease-in-out;
        }
        @keyframes pop {
            0% { transform: scale(0); opacity: 0; }
            50% { transform: scale(1.4); opacity: 1; }
            100% { transform: scale(1); opacity: 1; }
        }
    </style>
</head>
<body>

<h2>Puntaje actual: <b><?=$currentScore?> pts</b></h2>

<div id="start-overlay">
    
    <h1>Writing 1 - Speak Up!</h1>
    <button id="start-btn" style="margin-top:20px;padding:14px 28px;background:#FFB300;border:none;border-radius:15px;font-size:20px;cursor:pointer;font-weight:bold;color:#4b2f00;">¡ Empezar reto!</button>
    <a href="cursos_1ro.php" style="margin-top:18px;background:#FFB300;padding:10px 20px;border-radius:12px;font-weight:700;color:#4b2f00;text-decoration:none;display:inline-block;box-shadow:0 4px 0 #d89a00;transition:0.2s;">➡ Regresar al inicio</a>
</div>

```

```

<div id="count-down"><span id="count-number">3</span></div>

<div id="timer">Tiempo: 0s</div>
<div class="bubble b1"></div><div class="bubble b2"></div><div class="bubble b3"></div><div class="bubble b4"></div><div
class="bubble b5"></div>

<h2 style="margin-top:50px;">Completa la frase en inglés...</h2>
<div id="word-box"></div>
<input type="text" id="answer-input" autocomplete="off" placeholder="Escribe la palabra faltante">
<p id="hint"></p>
<button id="submit-btn">Enviar</button>

<div class="progress-container"><div class="progress-bar" id="progress-bar"></div></div>
<canvas id="confetti-canvas"></canvas>
<div id="restart-container" style="display:none;"><a href="cursos_1ro.php">Volver a cursos</a></div>
<div class="toast" id="toast"></div>

<script>
const questions = <?= json_encode($questions) ?>;
let current = 0, startTime, timerInterval, correctCount = 0, errors = 0;
const answerInput = document.getElementById('answer-input');
const progressBar = document.getElementById('progress-bar');
const toast = document.getElementById('toast');
const motivacion = ["¡Buen trabajo!", "¡Sigue así!", "¡Lo estás haciendo genial!", "¡Perfecto!", "¡Excelente!"];
const hints = ["Casi!!", "empieza con ", "Más cerca!!", "se escribe: ", "Ya casi!!", "sigue así: "];

/* Sonidos 🎵 */
function sonidoBien() {
  const ctx = new (window.AudioContext || window.webkitAudioContext)();
  const o = ctx.createOscillator(), g = ctx.createGain();
  o.frequency.value = 700; o.type = "triangle";
  o.connect(g); g.connect(ctx.destination);
  o.start(); g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
  o.stop(ctx.currentTime + 0.4);
}
function sonidoMal() {
  const ctx = new (window.AudioContext || window.webkitAudioContext)();
  const o = ctx.createOscillator(), g = ctx.createGain();
  o.frequency.value = 200; o.type = "square";
  o.connect(g); g.connect(ctx.destination);
  o.start(); g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
  o.stop(ctx.currentTime + 0.4);
}

function updateTimer() {
  const now = Date.now();
  const elapsed = Math.floor((now - startTime) / 1000);
  document.getElementById('timer').textContent = `Tiempo: ${elapsed}s`;
}
function showToast(msg) {
  toast.textContent = msg;
  toast.classList.add('show');
  setTimeout(() => toast.classList.remove('show'), 1500);
}
function showConfetti() {
  const canvas = document.getElementById('confetti-canvas');
  const ctx = canvas.getContext('2d');
  canvas.width = window.innerWidth;
  canvas.height = window.innerHeight;

  const colors = ['#FFF176', '#FFEB3B', '#FFD600', '#FFB300', '#FF59D', '#F4FF81', '#C6FF00'];
  const confs = [];
  for(let i = 0; i < 150; i++){
    confs.push({
      x: Math.random() * canvas.width,
      y: Math.random() * canvas.height,
      dx: Math.random() * 2 - 1,
      dy: Math.random() * 3 + 1,
      r: Math.random() * 6 + 3,
      color: colors[Math.floor(Math.random() * colors.length)],
    });
  }
}

```

```

        alpha: Math.random() * 0.8 + 0.4
    });
}
function animate() {
    ctx.clearRect(0, 0, canvas.width, canvas.height);
    confs.forEach(c => {
        c.x += c.dx; c.y += c.dy;
        if (c.y > canvas.height) c.y = -10;
        if (c.x > canvas.width) c.x = 0;
        if (c.x < 0) c.x = canvas.width;
        ctx.beginPath();
        ctx.arc(c.x, c.y, c.r, 0, 2 * Math.PI);
        ctx.fillStyle = c.color;
        ctx.globalAlpha = c.alpha;
        ctx.fill();
        ctx.globalAlpha = 1;
    });
    requestAnimationFrame(animate);
}
animate();
}

function showQuestion() {
    if (current >= questions.length) return;
    const q = questions[current];
    document.getElementById('word-box').textContent = q.sentence;
    answerInput.value = ''; errors = 0;
    document.getElementById('hint').textContent = '';
}

function updateProgress() {
    progressBar.style.width = `${(correctCount / questions.length) * 100}%`;
}

/* Countdown */
document.getElementById('start-btn').onclick = () => {
    document.getElementById('start-overlay').style.display = 'none';
    let count = 3;
    const cd = document.getElementById('countdown');
    cd.style.display = 'flex'; cd.textContent = count;
    const interval = setInterval(() => {
        cd.style.animation = 'none'; void cd.offsetWidth; cd.style.animation = 'pop 0.6s ease-in-out';
        count--; cd.textContent = count > 0 ? count : "GO!!";
        if (count < 0) {
            clearInterval(interval);
            cd.style.display = 'none';
            startTime = Date.now();
            timerInterval = setInterval(updateTimer, 1000);
            showQuestion();
        }
    }, 1000);
};

document.getElementById('submit-btn').onclick = () => {
    if (current >= questions.length) return;
    const q = questions[current];
    const ans = answerInput.value.toLowerCase().trim();

    if (ans === q.answer) {
        sonidoBien();
        answerInput.style.boxShadow = "0 0 15px 4px #4CAF50";
        setTimeout(() => answerInput.style.boxShadow = "", 1000);
        correctCount++;
        updateProgress();
        showToast(motivacion[Math.floor(Math.random() * motivacion.length)]);
        current++;

        if (current >= questions.length) {
            clearInterval(timerInterval);
            showConfetti();
            const tiempo = Math.floor((Date.now() - startTime) / 1000);

```

```

const puntaje = Math.max(100 - Math.floor(tiempo / 2), 10);
fetch("guardar_writing1.php", {
  method: "POST",
  headers: { "Content-Type": "application/x-www-form-urlencoded" },
  body: "puntaje=" + puntaje
});
document.getElementById('restart-container').style.display = 'block';
showToast(`¡Felicidades ${puntaje} pts! Tiempo: ${tiempo}s 🎉`);
} else {
  showQuestion();
}
} else {
  sonidoMal();
  const revealed = q.answer.substr(0, Math.min(errors + 1, q.answer.length));
  document.getElementById('hint').textContent = `Pista: ${revealed}...`;
  errors++;
}
};
</script>

</body>
</html>

```

memorama1.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$stmt = $conn->prepare("SELECT first_name FROM usuarios WHERE id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$user = $res->fetch_assoc();
$first_name = $user['first_name'];
$stmt2 = $conn->prepare("SELECT juego1 FROM puntajes WHERE user_id=?");
$stmt2->bind_param("i", $user_id);
$stmt2->execute();
$res2 = $stmt2->get_result();
$scoreData = $res2->fetch_assoc();
$currentScore = $scoreData ? $scoreData['juego1'] : 0;
$allPairs = [
    ["House", "Casa"], ["Car", "Carro"], ["Book", "Libro"], ["Milk", "Leche"],
    ["Table", "Mesa"], ["Chair", "Silla"], ["Sun", "Sol"], ["Moon", "Luna"]
];

shuffle($allPairs);
$gamePairs = array_slice($allPairs, 0, 8);

$cards = [];
foreach ($gamePairs as $pair) {
    $cards[] = ["text"=>$pair[0], "pair"=>$pair[1]];
    $cards[] = ["text"=>$pair[1], "pair"=>$pair[0]];
}
shuffle($cards);
?>
<!DOCTYPE html>
<html lang="es">
<head>
<link href="https://fonts.googleapis.com/css2?family=Comic+Neue:wght@300;700&display=swap" rel="stylesheet">
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Memorama Speak Up!</title>
<link rel="icon" href="Logo_SpeakUp.png">
<style>
@keyframes pop {
    0% { transform: scale(0); opacity: 0; }
    50% { transform: scale(1.4); opacity: 1; }
    100% { transform: scale(1); opacity: 1; }
}

#countdown {
    position: fixed;
    top: 0; left: 0;
    width: 100%; height: 100%;
    display: none;
    justify-content: center;
    align-items: center;
    background: rgba(255, 243, 172, 0.97);
    z-index: 9998;
    font-size: 200px;
    font-weight: 900;
    color: #4b2f00;
    text-shadow: 0 0 25px #FFD600, 0 0 45px #FFB300;
    animation: fadeIn 0.3s ease-in-out;
}

#count-number {
```

```

    animation: pop 0.6s ease-in-out;
  }

  @keyframes pop {
    0% { transform: scale(0); opacity: 0; }
    50% { transform: scale(1.4); opacity: 1; }
    100% { transform: scale(1); opacity: 1; }
  }

  @keyframes fadeIn {
    from { opacity: 0; }
    to { opacity: 1; }
  }

  body {
    font-family: 'Poppins', sans-serif;
    background: linear-gradient(to top, #FFF9D9, #FFE57F);
    display: flex; flex-direction: column; align-items: center; justify-content: flex-start;
    min-height: 100vh; margin: 0; padding: 20px; color: #4b2f00; overflow-x: hidden;
    position: relative;
  }

  .bubble { position: absolute; bottom: -150px; border-radius: 50%; background: rgba(255, 255, 255, 0.6); animation: floatUp 6s linear infinite; }
  @keyframes floatUp { 0% { transform: translateY(0); opacity: 0.7; } 50% { transform: translateY(-500px); opacity: 1; } 100% { transform: translateY(-1000px); opacity: 0; } }
  .b1 { width: 50px; height: 50px; left: 5%; animation-delay: 0s; }
  .b2 { width: 30px; height: 30px; left: 25%; animation-delay: 1s; }
  .b3 { width: 60px; height: 60px; left: 50%; animation-delay: 0.5s; }
  .b4 { width: 40px; height: 40px; left: 70%; animation-delay: 1.5s; }
  .b5 { width: 50px; height: 50px; left: 85%; animation-delay: 2s; }

  #logo-container {
    position: relative;
    display: flex;
    flex-direction: column;
    align-items: center;
    z-index: 10;
    margin-top: 140px;
  }
  #logo {
    width: 160px;
    cursor: pointer;
    transition: 0.3s;
  }
  .toast {
    position: absolute;
    bottom: 100%;
    margin-bottom: 25px;
    background: #FFB300;
    padding: 12px 20px;
    border-radius: 12px;
    font-weight: 700;
    color: #4b2f00;
    opacity: 0;
    transition: opacity 0.5s, transform 0.5s;
    pointer-events: none;
    white-space: nowrap;
  }
  .toast.show {
    opacity: 1;
    transform: translateX(-50%) translateY(-10px);
  }

  .card-container { display: grid; grid-template-columns: repeat(4, 120px); grid-gap: 15px; justify-content: center; margin-top: 30px; }
  .card { width: 120px; height: 80px; background: #FFD600; display: flex; align-items: center; justify-content: center; border-radius: 12px; cursor: pointer; perspective: 600px; position: relative; }
  .card-inner { position: absolute; width: 100%; height: 100%; transition: transform 0.6s; transform-style: preserve-3d; border-radius: 12px; }
  .card.flipped .card-inner { transform: rotateY(180deg); }
  .card-front, .card-back { position: absolute; width: 100%; height: 100%; backface-visibility: hidden; display: flex; align-items: center; justify-content: center; border-radius: 12px; font-weight: bold; font-size: 16px; }

```

```

.card-front{background:#FFD600;color:#fff;}
.card-back{background:#FFF9D9;transform:rotateY(180deg);}
.card.correct{box-shadow:0 0 15px #4CAF50;border:3px solid #4CAF50;}
.card.wrong{box-shadow:0 0 15px #f44336;border:3px solid #f44336;animation:shake 0.3s;}
@keyframes shake{0%{transform:translateX(0);}25%{transform:translateX(-5px);}50%{transform:translateX(5px);}75%{transform:translateX(-5px);}100%{transform:translateX(0);}
}
</style>
</head>
<body>

<div id="start-overlay" style="
    position:fixed;top:0;left:0;width:100%;height:100%;
    background:rgba(255,243,172,0.92);
    display:flex;flex-direction:column;justify-content:center;align-items:center;
    z-index:9999;text-align:center;color:#4b2f00;">

    <h1 style="
        font-size:36px;
        font-weight:700;
        animation:bounce 1.5s infinite;
        padding:10px 25px;
        border-radius:20px;
        background:#FFB300;
        box-shadow:0 0 18px #FFDD67;
        border:3px solid #F7C843;
    ">
        Memorama 1 ros
    </h1>

    <button id="start-btn" style="
        margin-top:20px;padding:14px 28px;background:#FFB300;
        border:none;border-radius:15px;font-size:20px;
        cursor:pointer;font-weight:bold;color:#4b2f00;">
        ¡Empezar reto!
    </button>

    <a href="curso_1 ro.php" style="
        margin-top:18px;
        background:#FFB300;
        padding:10px 20px;
        border-radius:12px;
        font-weight:700;
        color:#4b2f00;
        text-decoration:none;
        display:inline-block;
        box-shadow:0 4px 0 #d89a00;
        transition:0.2s;
    ">⬅ Regresar al inicio</a>

</div>

<style>
@keyframes bounce {
    0%,100% { transform:translateY(0); }
    50% { transform:translateY(-6px); }
}
</style>

<div id="countdown">
    <span id="count-number">3</span>
</div>

<div id="timer" style="
    font-size:22px;font-weight:bold;
    background:#FFB300;padding:8px 18px;border-radius:12px;margin-top:20px;
">Tiempo: 0s</div>

```



```

<div class="bubble b1"></div>
<div class="bubble b2"></div>
<div class="bubble b3"></div>
<div class="bubble b4"></div>
<div class="bubble b5"></div>

<div id="logo-container">
  
  <div id="toast" class="toast"></div>
</div>

<h2 style="margin-top:20px; background:#FFB300; padding:10px 20px; border-radius:12px;">
  Puntaje actual: <b><?=$currentScore?> pts</b>
</h2>

<div class="card-container">
<?php foreach($cards as $i=>$card): ?>
<div class="card" data-text="<?= htmlspecialchars($card['text']) ?>" data-pair="<?= htmlspecialchars($card['pair']) ?>">
  <div class="card-inner">
    <div class="card-front">?</div>
    <div class="card-back"><?= htmlspecialchars($card['text']) ?></div>
  </div>
</div>
<?php endforeach; ?>
</div>
<canvas id="confetti-canvas"
  style="position:absolute;top:0;left:0;width:100%;height:100%;pointer-events:none;">
</canvas>

<div id="restart-container" style="display:none;margin-top:20px;">
  <a href="curso_1ro.php" style="padding:12px 25px;background:#FFB300;color:#4b2f00;border-radius:12px;font-weight:bold;text-decoration:none;">Volver a los cursos</a>
</div>

<script>
const firstName = "<?= htmlspecialchars($first_name) ?>";
const cards = document.querySelectorAll('.card');
let flipped = [];
let lock = true;
let pairsFound = 0;
const totalPairs = <?= count($gamePairs) ?>;

let startTime = null;
let timerInterval;

function updateTimer() {
  let now = Date.now();
  let elapsed = Math.floor((now - startTime) / 1000);
  document.getElementById('timer').textContent = `Tiempo: ${elapsed}s`;
}

document.getElementById('start-btn').onclick = () => {
  document.getElementById('start-overlay').style.display = 'none';
  const countdown = document.getElementById('countdown');
  const countNum = document.getElementById('count-number');
  countdown.style.display = 'flex';

  let count = 3;
  countNum.textContent = count;

  const interval = setInterval(() => {
    count--;
    if (count > 0) {
      countNum.textContent = count;
      countNum.style.animation = 'none'; void countNum.offsetWidth; // reinicia la animación
      countNum.style.animation = 'pop 0.6s ease-in-out';
    } else if (count === 0) {
      countNum.textContent = '¡Ya!';
      countNum.style.animation = 'none'; void countNum.offsetWidth;
    }
  }, 1000);
}

```

```

        countNum.style.animation = 'pop 0.6s ease-in-out';
    } else {
        clearInterval(interval);
        countdown.style.display = 'none';
        lock = false;
        startTime = Date.now();
        timerInterval = setInterval(updateTimer, 1000);
    }
}, 1000);
};

const motivacion = ["¡Muy bien!", "¡Excelente!", "¡Sigue así!", "¡Perfecto!", "¡Genial!"];

function showToast(msg) {
    const toast = document.getElementById('toast');
    toast.textContent = msg;
    toast.classList.add('show');
    setTimeout(() => toast.classList.remove('show'), 1500);
}

function showConfetti() {
    const canvas = document.getElementById('confetti-canvas');
    const ctx = canvas.getContext('2d');
    canvas.width = window.innerWidth;
    canvas.height = window.innerHeight;

    const confs = [];
    for(let i = 0; i < 150; i++){
        confs.push({
            x: Math.random() * canvas.width,
            y: Math.random() * canvas.height,
            dx: Math.random() * 2 - 1,
            dy: Math.random() * 3 + 2,
            r: Math.random() * 6 + 2,
            color: `hsl(${Math.random()*50+40},100%,60%)`
        });
    }

    function animate() {
        ctx.clearRect(0, 0, canvas.width, canvas.height);
        confs.forEach(c => {
            c.x += c.dx;
            c.y += c.dy;
            if(c.y > canvas.height) c.y = -10;
            if(c.x > canvas.width) c.x = 0;
            if(c.x < 0) c.x = canvas.width;
            ctx.fillStyle = c.color;
            ctx.beginPath();
            ctx.arc(c.x, c.y, c.r, 0, 2 * Math.PI);
            ctx.fill();
        });
        requestAnimationFrame(animate);
    }
    animate();
}

function sonidoBien() {
    const ctx = new (window.AudioContext || window.webkitAudioContext)();
    const o = ctx.createOscillator();
    const g = ctx.createGain();
    o.frequency.value = 700;
    o.type = "triangle";
    o.connect(g);
    g.connect(ctx.destination);
    o.start();
    g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
    o.stop(ctx.currentTime + 0.4);
}

function sonidoMal() {

```

```

const ctx = new (window.AudioContext || window.webkitAudioContext)();
const o = ctx.createOscillator();
const g = ctx.createGain();
o.frequency.value = 200;
o.type = "square";
o.connect(g);
g.connect(ctx.destination);
o.start();
g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
o.stop(ctx.currentTime + 0.4);
}

cards.forEach(card=>{
  card.addEventListener('click',()=>{
    if(lock || card.classList.contains('flipped')) return;
    card.classList.add('flipped');
    flipped.push(card);

    if(flipped.length===2){
      lock=true;
      const [c1,c2]=flipped;
      if(c1.dataset.text===c2.dataset.pair || c2.dataset.text===c1.dataset.pair){
        c1.classList.add('correct');
        c2.classList.add('correct');
        sonidoBien();
        pairsFound++;
        showToast(motivacion[Math.floor(Math.random()*motivacion.length)]);
        flipped=[];
        lock=false;

        if(pairsFound=== totalPairs){
          clearInterval(timerInterval);

          let tiempo = Math.floor((Date.now() - startTime) / 1000);
          let puntaje = Math.max(100 - Math.floor(tiempo / 2), 10);

          fetch("guardar_memorama_1ro.php", {
            method: "POST",
            headers: { "Content-Type": "application/x-www-form-urlencoded" },
            body: "puntaje=" + puntaje
          });

          showConfetti();
          document.getElementById('restart-container').style.display = 'block';
          showToast(`¡Felicidades ${firstName}! Tiempo: ${tiempo}s • Puntaje: ${puntaje} 🎉`);
        }
      }
    } else {
      sonidoMal();
      setTimeout(()=>{
        c1.classList.remove('flipped');
        c2.classList.remove('flipped');
        flipped=[];
        lock=false;
      },800);
    }
  });
});
</script>

</body>
</html>

```

guardar_writing1.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    echo "No autorizado";
    exit();
}

$user_id = $_SESSION['user_id'];
if (!isset($_POST['puntaje'])) {
    echo "No hay puntaje";
    exit();
}

$puntaje = (int)$_POST['puntaje'];
$stmt = $conn->prepare("SELECT writing1 FROM puntajes WHERE user_id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$data = $res->fetch_assoc();

if($data){
    if($puntaje > $data['writing1']){
        $stmt2 = $conn->prepare("UPDATE puntajes SET writing1=? WHERE user_id=?");
        $stmt2->bind_param("ii", $puntaje, $user_id);
        $stmt2->execute();
    }
} else {
    $stmt2 = $conn->prepare("INSERT INTO puntajes (user_id, writing1) VALUES(?, ?)");
    $stmt2->bind_param("ii", $user_id, $puntaje);
    $stmt2->execute();
}

echo "ok";
```

guardar_memoranda_1ro.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) exit('No logueado');

$user_id = $_SESSION['user_id'];
$puntaje = isset($_POST['puntaje']) ? (int)$_POST['puntaje'] : 0;

$stmt = $conn->prepare("UPDATE puntajes SET juego1 = GREATEST(juego1, ?) WHERE user_id=?");
$stmt->bind_param("ii", $puntaje, $user_id);
$stmt->execute();

if ($stmt->affected_rows === 0) {
    $stmt = $conn->prepare("INSERT INTO puntajes (user_id, juego1) VALUES (?, ?)");
    $stmt->bind_param("ii", $user_id, $puntaje);
    $stmt->execute();
}

echo 'OK';
?>
```

cursos_2do.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$sql = "SELECT first_name FROM usuarios WHERE id = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$user = $result->fetch_assoc();

$first_name = $user['first_name'];

$messages = [
    "List @ para el siguiente reto, $first_name?",
    "¿Cuál es nuestro siguiente desafío, $first_name?",
    "¡Vamos a aprender algo nuevo, $first_name!",
    "¿Preparad @ para el próximo reto, $first_name?"
];
$random_msg = $messages[array_rand($messages)];
?>

<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>2do Grado - Cursos de Inglés</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    color: #4b2f00;
}

.navbar {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #ffffff;
    padding: 10px 30px;
    position: sticky;
    top: 0;
    z-index: 100;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}

.navbar img.logo { height: 50px; }

.navbar button {
    background: linear-gradient(145deg, #FFD600 0%, #FFB300 100%);
    border: none;
    border-radius: 20px;
    padding: 6px 14px;
    cursor: pointer;
    font-weight: 700;
    font-size: 14px;
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 4px 8px rgba(0,0,0,0.2);
    transition: all 0.3s ease;
}
```

```

.navbar button:hover {
  transform: translateY(-3px) scale(1.05);
  box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 8px 15px rgba(0,0,0,0.25);
}

/* Título centrado */
.hero {
  text-align: center;
  margin: 30px 20px;
}

.hero h1 {
  font-size: 48px;
  font-weight: 900;
  color: #FFB300;
  text-shadow: 2px 2px 10px rgba(75,47,0,0.5);
  margin-bottom: 15px;
}

.hero p {
  font-size: 22px;
  margin-bottom: 40px;
}

@keyframes riseUp {
  0% { transform: translateY(40px); opacity: 0; }
  100% { transform: translateY(0); opacity: 1; }
}

.courses-grid {
  display: grid;
  grid-template-columns: repeat(2, 1fr);
  grid-template-rows: repeat(2, 180px);
  gap: 25px;
  padding: 0 20px 50px 20px;
}

.course-card {
  position: relative;
  border-radius: 30px;
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  font-size: 20px;
  font-weight: 800;
  cursor: pointer;
  color: #fff;
  transition: transform 0.4s, box-shadow 0.4s, background 0.4s;
  text-align: center;
  padding: 12px 18px;
  letter-spacing: 1px;
  box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 6px 15px rgba(0,0,0,0.2);
  opacity: 0;
  animation: riseUp 0.6s forwards;
}

.course-card span {
  font-size: 16px;
  font-weight: 500;
  margin-top: 8px;
}

.course-card:nth-child(1) { animation-delay: 0.1s; background: linear-gradient(145deg, #FFD600, #FFB300); }
.course-card:nth-child(2) { animation-delay: 0.3s; background: linear-gradient(145deg, #FFB300, #FFD600); }
.course-card:nth-child(3) { animation-delay: 0.5s; background: linear-gradient(145deg, #FFC95C, #FF9F1C); }
.course-card:nth-child(4) { animation-delay: 0.7s; background: linear-gradient(145deg, #FFDA6C, #FFD166); }

.course-card:hover {

```

```

transform: translateY(-8px) scale(1.07);
box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 12px 25px rgba(0,0,0,0.3);
}

@media(max-width:700px){
.hero h1 { font-size:36px; }
.hero p { font-size:18px; }
.course-card { font-size:18px; height:140px; padding:10px 14px; }
.courses-grid { grid-template-columns: 1fr; grid-template-rows: repeat(4,140px); }
}
</style>
</head>
<body>

<div class="navbar">
  
  <button onclick="window.location.href='principalphp.php'">Volver al inicio</button>
</div>

<div class="hero">
  <h1>2do Grado</h1>
  <p><?php echo htmlspecialchars($random_msg); ?></p>
</div>

<div class="courses-grid">
  <div class="course-card" onclick="window.location.href='Writing2.php'">
    Writing
    <span>Escritura</span>
  </div>
  <div class="course-card" onclick="window.location.href='Listening2.php'">
    Listening
    <span>Escucha</span>
  </div>
  <div class="course-card" onclick="window.location.href='Grammar2.php'">
    Grammar
    <span>Gramática</span>
  </div>
  <div class="course-card" onclick="window.location.href='juegos2.php'">
    Games
    <span>Juegos</span>
  </div>
</div>

</body>
</html>

```


juegos2.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$sql = "SELECT first_name FROM usuarios WHERE id = ?";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$user = $result->fetch_assoc();

$first_name = $user['first_name'];

$messages = [
    "List@ para el siguiente reto, $first_name?",
    "¿Cuál es nuestro siguiente desafío, $first_name?",
    "¡Vamos a descubrir cosas nuevas, $first_name!",
    "¡Preparad@ para el próximo juego, $first_name?"
];
$random_msg = $messages[array_rand($messages)];
?>
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>2do Grado - Cursos de Inglés</title>
<style>
body {
    margin: 0;
    font-family: "Poppins", sans-serif;
    background: url('FondoSpeak.png') no-repeat center center fixed;
    background-size: cover;
    color: #4b2f00;
}

.navbar {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #ffffff;
    padding: 10px 30px;
    position: sticky;
    top: 0;
    z-index: 100;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}

.navbar img.logo { height: 50px; }

.navbar button {
    background: linear-gradient(145deg, #FFD600 0%, #FFB300 100%);
    border: none;
    border-radius: 20px;
    padding: 6px 14px;
    cursor: pointer;
    font-weight: 700;
    font-size: 14px;
    box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 4px 8px rgba(0,0,0,0.2);
    transition: all 0.3s ease;
}

.navbar button:hover {
```

```

transform: translateY(-3px) scale(1.05);
box-shadow: inset 0 -3px 0 rgba(0,0,0,0.15), 0 8px 15px rgba(0,0,0,0.25);
}

/* Título centrado */
.hero {
  text-align: center;
  margin: 30px 20px;
}

.hero h1 {
  font-size: 48px;
  font-weight: 900;
  color: #FFB300;
  text-shadow: 2px 2px 10px rgba(75,47,0,0.5);
  margin-bottom: 15px;
}

.hero p {
  font-size: 22px;
  margin-bottom: 40px;
}

@keyframes riseUp {
  0% { transform: translateY(40px); opacity: 0; }
  100% { transform: translateY(0); opacity: 1; }
}

.courses-grid {
  display: grid;
  grid-template-columns: repeat(2, 1fr);
  grid-template-rows: repeat(3, 140px);
  gap: 25px;
  padding: 0 20px 50px 20px;
}

.course-card {
  position: relative;
  border-radius: 30px;
  display: flex;
  justify-content: center;
  align-items: center;
  font-size: 20px;
  font-weight: 800;
  cursor: pointer;
  color: #fff;
  transition: transform 0.4s, box-shadow 0.4s, background 0.4s;
  text-align: center;
  padding: 12px 18px;
  letter-spacing: 1px;
  box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 6px 15px rgba(0,0,0,0.2);
  opacity: 0;
  animation: riseUp 0.6s forwards;
}

.course-card:nth-child(1) { animation-delay: 0.1s; background: linear-gradient(145deg, #FFD600, #FFB300); }
.course-card:nth-child(2) { animation-delay: 0.3s; background: linear-gradient(145deg, #FFB300, #FFD600); }
.course-card:nth-child(3) { animation-delay: 0.5s; background: linear-gradient(145deg, #FFC95C, #FF9F1C); }
.course-card:nth-child(4) { animation-delay: 0.7s; background: linear-gradient(145deg, #FFDA6C, #FFD166); }
.course-card:nth-child(5) { animation-delay: 0.9s; background: linear-gradient(145deg, #FFE27A, #FFD166); }
.course-card:nth-child(6) { animation-delay: 1.1s; background: linear-gradient(145deg, #FFD966, #FFB347); }

.course-card:hover {
  transform: translateY(-8px) scale(1.07);
  box-shadow: inset 0 -4px 0 rgba(0,0,0,0.1), 0 12px 25px rgba(0,0,0,0.3);
}

@media(max-width:700px){
  .hero h1 { font-size:36px; }
  .hero p { font-size:18px; }
}

```

```

        .course-card { font-size:18px; height:120px; padding:10px 14px; }
        .courses-grid { grid-template-columns: 1fr; grid-template-rows: repeat(6,120px); }
    }
</style>
</head>
<body>

<div class="navbar">
    
    <button onclick="window.location.href='cursos_2do.php'">Volver a los cursos</button>
</div>

<div class="hero">
    <h1>2do Grado</h1>
    <p><?php echo htmlspecialchars($random_msg); ?></p>
</div>

<div class="courses-grid">
    <div class="course-card" onclick="window.location.href='memorama2.php'">Memorama</div>
    <div class="course-card" onclick="window.location.href='trivia2.php'">Trivia</div>
    <div class="course-card" onclick="window.location.href='colores2.php'">Colores</div>
    <div class="course-card" onclick="window.location.href='vocales2.php'">Vocales</div>
    <div class="course-card" onclick="window.location.href='numeros2.php'">Números</div>
    <div class="course-card" onclick="window.location.href='animales2.php'">Animales</div>
</div>

</body>
</html>

```

Writing1.php

```

<?php
session_start();
include 'db.php';
if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];

$stmt = $conn->prepare("SELECT first_name FROM usuarios WHERE id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$user = $res->fetch_assoc();
$first_name = $user['first_name'];

$stmt2 = $conn->prepare("SELECT writing1 FROM puntajes WHERE user_id=?");
$stmt2->bind_param("i", $user_id);
$stmt2->execute();
$res2 = $stmt2->get_result();
$scoreData = $res2->fetch_assoc();
$currentScore = $scoreData ? $scoreData['writing1'] : 0;
$questions = [
    ['sentence' => 'The (☺) is round', 'answer' => 'ball'],
    ['sentence' => 'The sky is (☁)', 'answer' => 'blue'],
    ['sentence' => 'I can (🏃‍♂️)', 'answer' => 'run'],
    ['sentence' => 'I see a (🦋)', 'answer' => 'butterfly'],
    ['sentence' => 'I like (🍏)', 'answer' => 'apples'],
    ['sentence' => 'A cow gives (🐄)', 'answer' => 'milk'],
    ['sentence' => 'The sun is (☀️)', 'answer' => 'happy'],
    ['sentence' => 'I have (👁️)', 'answer' => 'eyes'],
    ['sentence' => 'This is my (👩🏫)', 'answer' => 'teacher'],
    ['sentence' => 'The grass is (🌱)', 'answer' => 'green']
];
shuffle($questions);

?>
<!DOCTYPE html>
<html lang="es">
<head>
<meta charset="UTF-8">
<title>Writing 1 - Speak Up!</title>
<link href="https://fonts.googleapis.com/css2?family=Comic+Neue:wght@300;700&display=swap" rel="stylesheet">
<style>
#restart-container a {
    background: #FFB300;
    padding: 14px 26px;
    border-radius: 15px;
    font-weight: 700;
    color: #4b2f00;
    text-decoration: none;
    display: inline-block;
    margin-top: 30px;
    font-size: 20px;
    box-shadow: 0 4px 0 #d89a00;
    transition: all 0.2s ease-in-out;
}
#restart-container a: hover {
    transform: scale(1.08);
    box-shadow: 0 6px 10px rgba(0,0,0,0.2);
    background: #FFD54F;
}

body {font-family: 'Comic Neue', sans-serif; background: linear-gradient(to top, #FFF9D9, #FFE57F); margin: 0; padding: 20px; text-align: center; overflow: hidden;}

```

```

.bubble{position:absolute;bottom:-150px;border-radius:50%;background:rgba(255,255,255,0.6);animation:floatUp 6s linear
infinite;}
@keyframes floatUp {0% {transform:translateY(0);opacity:0.7;}50% {transform:translateY(-
500px);opacity:1;}100% {transform:translateY(-1000px);opacity:0;}}
.b1 {width:50px;height:50px;left:5%;animation-delay:0s;} .b2 {width:30px;height:30px;left:25%;animation-delay:1s;}
.b3 {width:60px;height:60px;left:50%;animation-delay:0.5s;} .b4 {width:40px;height:40px;left:70%;animation-delay:1.5s;}
.b5 {width:50px;height:50px;left:85%;animation-delay:2s;}
#word-box {font-size:45px;margin:30px 0;}
input {padding:10px;font-size:18px;border-radius:8px;border:2px solid #FFD600;text-align:center;transition:box-shadow 0.3s;}
button {padding:10px 20px;font-size:18px;background:#FFD600;border:none;border-radius:10px;cursor:pointer;margin-left:10px;}
.progress-container{width:50%;background:#eee;margin:20px auto;border-radius:10px;}
.progress-bar {height:20px;width:0%;background:#FFEB3B;border-radius:10px;transition:width 0.5s;}
.toast {position:absolute;top:80px;left:50%;transform:translateX(-50%);background:#FFB300;padding:12px 20px;border-
radius:12px;font-weight:700;color:#4b2f00;opacity:0;transition:opacity 0.5s,transform 0.5s:pointer-events:none;}
.toast.show{opacity:1;transform:translateX(-50%) translateY(-10px);}
#confetti-canvas {position:absolute;top:0;left:0;width:100%;height:100%;pointer-events:none;}
#timer {font-size:20px;font-weight:bold;background:#FFB300;padding:8px 15px;border-radius:10px;display:inline-block;margin-
top:20px;}
#start-overlay{position:fixed;top:0;left:0;width:100%;height:100%;background:rgba(255,243,172,0.92);display:flex;flex-
direction:column;justify-content:center;align-items:center;z-index:9999;text-align:center;color:#4b2f00;}
#start-overlay h1 {font-size:36px;font-weight:700;animation:bounce 1.5s infinite;padding:10px 25px;border-
radius:20px;background:#FFB300;box-shadow:0 0 18px #FFDD67;border:3px solid #F7C843;}
@keyframes bounce {0%,100% {transform:translateY(0);}50% {transform:translateY(-6px);}}

#countdown {
  position: fixed;
  top: 0; left: 0;
  width: 100%; height: 100%;
  display: flex;
  justify-content: center;
  align-items: center;
  background: rgba(255, 243, 172, 0.97);
  z-index: 9998;
  font-size: 200px;
  font-weight: 900;
  color: #4b2f00;
  text-shadow: 0 0 25px #FFD600, 0 0 45px #FFB300;
  animation: fadeIn 0.3s ease-in-out;
}
@keyframes fadeIn {
  from { opacity: 0; }
  to { opacity: 1; }
}
#count-number {
  animation: pop 0.6s ease-in-out;
}
@keyframes pop {
  0% { transform: scale(0); opacity: 0; }
  50% { transform: scale(1.4); opacity: 1; }
  100% { transform: scale(1); opacity: 1; }
}
</style>
</head>
<body>

<h2>Puntaje actual: <b><?=$currentScore ?> pts</b></h2>

<div id="start-overlay">
  
  <h1>Writing 2 - Speak Up!</h1>
  <button id="start-btn" style="margin-top:20px;padding:14px 28px;background:#FFB300;border:none;border-radius:15px;font-
size:20px;cursor:pointer;font-weight:bold;color:#4b2f00;">¡Empezar reto!</button>
  <a href="cursos_2do.php" style="margin-top:18px;background:#FFB300;padding:10px 20px;border-radius:12px;font-
weight:700;color:#4b2f00;text-decoration:none;display:inline-block;box-shadow:0 4px 0 #d89a00;transition:0.2s;">⬅ Regresar al inicio</a>
</div>

<div id="countdown"><span id="count-number">3</span></div>

<div id="timer">Tiempo: 0s</div>

```

```

<div class="bubble b1"></div><div class="bubble b2"></div><div class="bubble b3"></div><div class="bubble b4"></div><div
class="bubble b5"></div>

```

```

<h2 style="margin-top:50px;">Completa la frase en inglés...</h2>
<div id="word-box"></div>
<input type="text" id="answer-input" autocomplete="off" placeholder="Escribe la palabra faltante">
<p id="hint"></p>
<button id="submit-btn">Enviar</button>

<div class="progress-container"><div class="progress-bar" id="progress-bar"></div></div>
<canvas id="confetti-canvas"></canvas>
<div id="restart-container" style="display:none;"><a href="cursos_2do.php">Volver a cursos</a></div>
<div class="toast" id="toast"></div>

```

```

<script>
const questions = <?=json_encode($questions)?>;
let current = 0, startTime, timerInterval, correctCount = 0, errors = 0;
const answerInput = document.getElementById('answer-input');
const progressBar = document.getElementById('progress-bar');
const toast = document.getElementById('toast');
const motivacion = ["¡Buen trabajo!", "¡Sigue así!", "¡Lo estás haciendo genial!", "¡Perfecto!", "¡Excelente!"];
const hints = ["Casi!!, empieza con ", "Más cerca!!, se escribe: ", "Ya casi!!, sigue así: "];

```

```

function sonidoBien() {
  const ctx = new (window.AudioContext || window.webkitAudioContext)();
  const o = ctx.createOscillator(), g = ctx.createGain();
  o.frequency.value = 700; o.type = "triangle";
  o.connect(g); g.connect(ctx.destination);
  o.start(); g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
  o.stop(ctx.currentTime + 0.4);
}
function sonidoMal() {
  const ctx = new (window.AudioContext || window.webkitAudioContext)();
  const o = ctx.createOscillator(), g = ctx.createGain();
  o.frequency.value = 200; o.type = "square";
  o.connect(g); g.connect(ctx.destination);
  o.start(); g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
  o.stop(ctx.currentTime + 0.4);
}

```

```

function updateTimer() {
  const now = Date.now();
  const elapsed = Math.floor((now - startTime) / 1000);
  document.getElementById('timer').textContent = `Tiempo: ${elapsed}s`;
}

```

```

function showToast(msg) {
  toast.textContent = msg;
  toast.classList.add('show');
  setTimeout(() => toast.classList.remove('show'), 1500);
}

```

```

function showConfetti() {
  const canvas = document.getElementById('confetti-canvas');
  const ctx = canvas.getContext('2d');
  canvas.width = window.innerWidth;
  canvas.height = window.innerHeight;

```

```

const colors = ['#FFF176', '#FFEB3B', '#FFD600', '#FFB300', '#FF5959', '#F44336', '#C62828'];
const confs = [];
for(let i = 0; i < 150; i++){
  confs.push({
    x: Math.random() * canvas.width,
    y: Math.random() * canvas.height,
    dx: Math.random() * 2 - 1,
    dy: Math.random() * 3 + 1,
    r: Math.random() * 6 + 3,
    color: colors[Math.floor(Math.random() * colors.length)],
    alpha: Math.random() * 0.8 + 0.4
  });
}
function animate() {

```

```

    ctx.clearRect(0, 0, canvas.width, canvas.height);
    confs.forEach(c => {
        c.x += c.dx; c.y += c.dy;
        if (c.y > canvas.height) c.y = -10;
        if (c.x > canvas.width) c.x = 0;
        if (c.x < 0) c.x = canvas.width;
        ctx.beginPath();
        ctx.arc(c.x, c.y, c.r, 0, 2 * Math.PI);
        ctx.fillStyle = c.color;
        ctx.globalAlpha = c.alpha;
        ctx.fill();
        ctx.globalAlpha = 1;
    });
    requestAnimationFrame(animate);
}
animate();
}

function showQuestion(){
    if(current >= questions.length) return;
    const q = questions[current];
    document.getElementById('word-box').textContent = q.sentence;
    answerInput.value = ""; errors = 0;
    document.getElementById('hint').textContent = "";
}

function updateProgress(){
    progressBar.style.width = `${(correctCount/questions.length)*100}%`;
}

document.getElementById('start-btn').onclick = () => {
    document.getElementById('start-overlay').style.display = 'none';
    let count = 3;
    const cd = document.getElementById('countdown');
    cd.style.display = 'flex'; cd.textContent = count;
    const interval = setInterval(() => {
        cd.style.animation = 'none'; void cd.offsetWidth; cd.style.animation = 'pop 0.6s ease-in-out';
        count--; cd.textContent = count > 0 ? count : "¡Ya!";
        if(count < 0){
            clearInterval(interval);
            cd.style.display = 'none';
            startTime = Date.now();
            timerInterval = setInterval(updateTimer, 1000);
            showQuestion();
        }
    }, 1000);
};

document.getElementById('submit-btn').onclick = () => {
    if (current >= questions.length) return;
    const q = questions[current];
    const ans = answerInput.value.toLowerCase().trim();

    if (ans === q.answer) {
        sonidoBien();
        answerInput.style.boxShadow = "0 0 15px 4px #4CAF50";
        setTimeout(() => answerInput.style.boxShadow = "", 1000);
        correctCount++;
        updateProgress();
        showToast(motivacion[Math.floor(Math.random() * motivacion.length)]);
        current++;
    }

    if (current >= questions.length) {
        clearInterval(timerInterval);
        showConfetti();
        const tiempo = Math.floor((Date.now() - startTime) / 1000);
        const puntaje = Math.max(100 - Math.floor(tiempo / 2), 10);
        fetch("guardar_writing1.php", {
            method: "POST",
            headers: { "Content-Type": "application/x-www-form-urlencoded" },
            body: "puntaje=" + puntaje
        });
    }
}

```

```

    });
    document.getElementById('restart-container').style.display = 'block';
    showToast(`¡Felicidades ${puntaje} pts! Tiempo: ${tiempo}s 🎉`);
  } else {
    showQuestion();
  }
} else {
  sonidoMal();
  const revealed = q.answer.substr(0, Math.min(errors + 1, q.answer.length));
  document.getElementById('hint').textContent = `Pista: ${revealed}...`;
  errors++;
}
};
</script>

</body>
</html>

```


Memorama2.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

$user_id = $_SESSION['user_id'];
$stmt = $conn->prepare("SELECT first_name FROM usuarios WHERE id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$user = $res->fetch_assoc();
$first_name = $user['first_name'];
$stmt2 = $conn->prepare("SELECT juego2 FROM puntajes WHERE user_id=?");
$stmt2->bind_param("i", $user_id);
$stmt2->execute();
$res2 = $stmt2->get_result();
$scoreData = $res2->fetch_assoc();
$currentScore = $scoreData ? $scoreData['juego2'] : 0;
$allPairs = [
    ["Bread", "Pan"], ["Rice", "Arroz"], ["Juice", "Jugo"], ["Water", "Agua"],
    ["Pencil", "Lápiz"], ["Eraser", "Goma"], ["Window", "Ventana"], ["Door", "Puerta"],
    ["Shirt", "Camisa"], ["Shoes", "Zapatos"], ["Cup", "Taza"], ["Plate", "Plato"]
];

shuffle($allPairs);
$gamePairs = array_slice($allPairs, 0, 8);

$cards = [];
foreach($gamePairs as $pair) {
    $cards[] = ["text"=>$pair[0], "pair"=>$pair[1]];
    $cards[] = ["text"=>$pair[1], "pair"=>$pair[0]];
}
shuffle($cards);
?>
<!DOCTYPE html>
<html lang="es">
<head>
<link href="https://fonts.googleapis.com/css2?family=Comic+Neue:wght@300;700&display=swap" rel="stylesheet">
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Memorama Speak Up!</title>
<link rel="icon" href="Logo_SpeakUp.png">
<style>
@keyframes pop {
    0% { transform: scale(0); opacity:0; }
    50% { transform: scale(1.4); opacity:1; }
    100% { transform: scale(1); opacity:1; }
}

#countdown {
    position: fixed;
    top: 0; left: 0;
    width: 100%; height: 100%;
    display: none;
    justify-content: center;
    align-items: center;
    background: rgba(255, 243, 172, 0.97);
    z-index: 9998;
    font-size: 200px;
    font-weight: 900;
    color: #4b2f00;
    text-shadow: 0 0 25px #FFD600, 0 0 45px #FFB300;
    animation: fadeIn 0.3s ease-in-out;
}
```

```

#count-number {
  animation: pop 0.6s ease-in-out;
}

@keyframes pop {
  0% { transform: scale(0); opacity: 0; }
  50% { transform: scale(1.4); opacity: 1; }
  100% { transform: scale(1); opacity: 1; }
}

@keyframes fadeIn {
  from { opacity: 0; }
  to { opacity: 1; }
}

body {
  font-family: 'Poppins', sans-serif;
  background: linear-gradient(to top, #FFF9D9, #FFE57F);
  display: flex; flex-direction: column; align-items: center; justify-content: flex-start;
  min-height: 100vh; margin: 0; padding: 20px; color: #4b2f00; overflow-x: hidden;
  position: relative;
}

.bubble { position: absolute; bottom: -150px; border-radius: 50%; background: rgba(255, 255, 255, 0.6); animation: floatUp 6s linear infinite; }
@keyframes floatUp { 0% { transform: translateY(0); opacity: 0.7; } 50% { transform: translateY(-500px); opacity: 1; } 100% { transform: translateY(-1000px); opacity: 0; } }
.b1 { width: 50px; height: 50px; left: 5%; animation-delay: 0s; }
.b2 { width: 30px; height: 30px; left: 25%; animation-delay: 1s; }
.b3 { width: 60px; height: 60px; left: 50%; animation-delay: 0.5s; }
.b4 { width: 40px; height: 40px; left: 70%; animation-delay: 1.5s; }
.b5 { width: 50px; height: 50px; left: 85%; animation-delay: 2s; }

#logo-container {
  position: relative;
  display: flex;
  flex-direction: column;
  align-items: center;
  z-index: 10;
  margin-top: 140px;
}

#logo {
  width: 160px;
  cursor: pointer;
  transition: 0.3s;
}

.toast {
  position: absolute;
  bottom: 100%;
  margin-bottom: 25px;
  background: #FFB300;
  padding: 12px 20px;
  border-radius: 12px;
  font-weight: 700;
  color: #4b2f00;
  opacity: 0;
  transition: opacity 0.5s, transform 0.5s;
  pointer-events: none;
  white-space: nowrap;
}

.toast.show {
  opacity: 1;
  transform: translateX(-50%) translateY(-10px);
}

.card-container { display: grid; grid-template-columns: repeat(4, 120px); grid-gap: 15px; justify-content: center; margin-top: 30px; }
.card { width: 120px; height: 80px; background: #FFD600; display: flex; align-items: center; justify-content: center; border-radius: 12px; cursor: pointer; perspective: 600px; position: relative; }
.card-inner { position: absolute; width: 100%; height: 100%; transition: transform 0.6s; transform-style: preserve-3d; border-radius: 12px; }
.card.flipped .card-inner { transform: rotateY(180deg); }

```

```

.card-front, .card-back{position:absolute;width:100%;height:100%;backface-visibility:hidden;display:flex;align-items:center;justify-
content:center;border-radius:12px;font-weight:bold;font-size:16px;}
.card-front{background:#FFD600;color:#fff;}
.card-back{background:#FF9900;transform:rotateY(180deg);}
.card.correct{box-shadow:0 0 15px #4CAF50; border:3px solid #4CAF50;}
.card.wrong{box-shadow:0 0 15px #f44336; border:3px solid #f44336; animation:shake 0.3s;}
@keyframes shake{0%{transform:translateX(0);}25%{transform:translateX(-
5px);}50%{transform:translateX(5px);}75%{transform:translateX(-5px);}100%{transform:translateX(0);}}
#restart-container a {
  background: #FFB300;
  padding: 14px 26px;
  border-radius: 15px;
  font-weight: 700;
  color: #4b2f00;
  text-decoration: none;
  display: inline-block;
  margin-top: 30px;
  font-size: 20px;
  box-shadow: 0 4px 0 #d89a00;
  transition: all 0.2s ease-in-out;
}
#restart-container a:hover {
  transform: scale(1.08);
  box-shadow: 0 6px 10px rgba(0,0,0,0.2);
  background: #FFD54F;
}

</style>
</head>
<body>

<div id="start-overlay" style="
  position:fixed;top:0;left:0; width:100%; height:100%;
  background:rgba(255, 243, 172,0.92);
  display:flex; flex-direction:column; justify-content:center; align-items:center;
  z-index:9999; text-align:center; color:#4b2f00;">

  <h1 style="
    font-size:36px;
    font-weight:700;
    animation:bounce 1.5s infinite;
    padding:10px 25px;
    border-radius:20px;
    background:#FFB300;
    box-shadow:0 0 18px #FFDD67;
    border:3px solid #F7C843;
  ">
    Memorama 2dos
  </h1>

  <button id="start-btn" style="
    margin-top:20px; padding:14px 28px; background:#FFB300;
    border:none; border-radius:15px; font-size:20px;
    cursor:pointer; font-weight:bold; color:#4b2f00;">
    ¡Empezar reto!
  </button>

  <a href="curso_2do.php" style="
    margin-top:18px;
    background:#FFB300;
    padding:10px 20px;
    border-radius:12px;
    font-weight:700;
    color:#4b2f00;
    text-decoration:none;
    display:inline-block;
    box-shadow:0 4px 0 #d89a00;

```

```

        transition:0.2s;
">⬅ Regresar al inicio</a>

</div>

<style>
@keyframes bounce {
    0%,100% { transform:translateY(0); }
    50% { transform:translateY(-6px); }
}
</style>

<div id="count down">
    <span id="count-number">3</span>
</div>

<div id="timer" style="
    font-size:22px; font-weight:bold;
    background:#FFB300; padding:8px 18px; border-radius:12px; margin-top:20px;
">Tiempo: 0s</div>

<div class="bubble b1"></div>
<div class="bubble b2"></div>
<div class="bubble b3"></div>
<div class="bubble b4"></div>
<div class="bubble b5"></div>

<div id="logo-container">
    
    <div id="toast" class="toast"></div>
</div>

<h2 style="margin-top:20px; background:#FFB300; padding:10px 20px; border-radius:12px;">
    Puntaje actual: <b><?=$currentScore ?> pts</b>
</h2>

<div class="card-container">
<?php foreach($cards as $i=>$card): ?>
<div class="card" data-text="<?= htmlspecialchars($card['text']) ?>" data-pair="<?= htmlspecialchars($card['pair']) ?>">
    <div class="card-inner">
        <div class="card-front"></div>
        <div class="card-back"><?= htmlspecialchars($card['text']) ?></div>
    </div>
</div>
<?php endforeach; ?>
</div>
<canvas id="confetti-canvas" style="position:absolute;top:0;left:0;width:100%;height:100%;pointer-events:none;"></canvas>

<div id="restart-container" style="display:none;margin-top:20px;">
    <a href="curso_2do.php" style="padding:12px 25px;background:#FFB300;color:#4b2f00;border-radius:12px;font-weight:bold;text-decoration:none;">Volver a los cursos</a>
</div>

<script>
const firstName = "<?= htmlspecialchars($first_name) ?>";
const cards = document.querySelectorAll('.card');
let flipped = [];
let lock = true;
let pairsFound = 0;
const totalPairs = <?= count($gamePairs) ?>;

let startTime = null;
let timerInterval;

function updateTimer(){
    let now = Date.now();
    let elapsed = Math.floor((now - startTime) / 1000);
    document.getElementById('timer').textContent = `Tiempo: ${elapsed}s`;

```

```

}

document.getElementById('start-btn').onclick = () => {
  document.getElementById('start-overlay').style.display = 'none';
  const countdown = document.getElementById('countdown');
  const countNum = document.getElementById('count-number');
  countdown.style.display = 'flex';

  let count = 3;
  countNum.textContent = count;

  const interval = setInterval(() => {
    count--;
    if (count > 0) {
      countNum.textContent = count;
      countNum.style.animation = 'none'; void countNum.offsetWidth; // reinicia la animación
      countNum.style.animation = 'pop 0.6s ease-in-out';
    } else if (count === 0) {
      countNum.textContent = '¡Ya!';
      countNum.style.animation = 'none'; void countNum.offsetWidth;
      countNum.style.animation = 'pop 0.6s ease-in-out';
    } else {
      clearInterval(interval);
      countdown.style.display = 'none';
      lock = false;
      startTime = Date.now();
      timerInterval = setInterval(updateTimer, 1000);
    }
  }, 1000);
};

```

```

const motivacion = ["¡Muy bien!", "¡Excelente!", "¡Sigue así!", "¡Perfecto!", "¡Genial!"];

```

```

function showToast(msg){
  const toast = document.getElementById('toast');
  toast.textContent = msg;
  toast.classList.add('show');
  setTimeout(()=>toast.classList.remove('show'),1500);
}

```

```

function showConfetti(){
  const canvas = document.getElementById('confetti-canvas');
  const ctx = canvas.getContext('2d');
  canvas.width = window.innerWidth;
  canvas.height = window.innerHeight;

```

```

  const confs = [];
  for(let i = 0; i < 150; i++){
    confs.push({
      x: Math.random() * canvas.width,
      y: Math.random() * canvas.height,
      dx: Math.random() * 2 - 1,
      dy: Math.random() * 3 + 2,
      r: Math.random() * 6 + 2,
      color: `hsl(${Math.random()*50+40},100%,60%)`
    });
  }
}

```

```

function animate(){
  ctx.clearRect(0, 0, canvas.width, canvas.height);
  confs.forEach(c => {
    c.x += c.dx;
    c.y += c.dy;
    if(c.y > canvas.height) c.y = -10;
    if(c.x > canvas.width) c.x = 0;
    if(c.x < 0) c.x = canvas.width;
    ctx.fillStyle = c.color;
    ctx.beginPath();

```

```

        ctx.arc(c.x, c.y, c.r, 0, 2 * Math.PI);
        ctx.fill();
    });
    requestAnimationFrame(animate);
}
animate();
}

function sonidoBien() {
    const ctx = new (window.AudioContext || window.webkitAudioContext)();
    const o = ctx.createOscillator();
    const g = ctx.createGain();
    o.frequency.value = 700;
    o.type = "triangle";
    o.connect(g);
    g.connect(ctx.destination);
    o.start();
    g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
    o.stop(ctx.currentTime + 0.4);
}

function sonidoMal() {
    const ctx = new (window.AudioContext || window.webkitAudioContext)();
    const o = ctx.createOscillator();
    const g = ctx.createGain();
    o.frequency.value = 200;
    o.type = "square";
    o.connect(g);
    g.connect(ctx.destination);
    o.start();
    g.gain.exponentialRampToValueAtTime(0.0001, ctx.currentTime + 0.4);
    o.stop(ctx.currentTime + 0.4);
}

cards.forEach(card=>{
    card.addEventListener('click', ()=>{
        if(!lock || card.classList.contains('flipped')) return;
        card.classList.add('flipped');
        flipped.push(card);

        if(flipped.length===2){
            lock=true;
            const [c1,c2]=flipped;
            if(c1.dataset.text===c2.dataset.pair || c2.dataset.text===c1.dataset.pair){
                c1.classList.add('correct');
                c2.classList.add('correct');
                sonidoBien();
                pairsFound++;
                showToast(motivacion[Math.floor(Math.random()*motivacion.length)]);
                flipped=[];
                lock=false;

                if(pairsFound===totalPairs){
                    clearInterval(timerInterval);

                    let tiempo = Math.floor((Date.now() - startTime) / 1000);
                    let puntaje = Math.max(100 - Math.floor(tiempo / 2), 10);

                    fetch("guardar_memorama_2do.php", {
                        method: "POST",
                        headers: { "Content-Type": "application/x-www-form-urlencoded" },
                        body: "puntaje=" + puntaje
                    });

                    showConfetti();
                    document.getElementById('restart-container').style.display = 'block';
                    showToast(`¡Felicidades ${firstName}! Tiempo: ${tiempo}s • Puntaje: ${puntaje} 🎉`);
                }
            } else {

```

```
sonidoMal();
setTimeout(()=>{
  c1.classList.remove('flipped');
  c2.classList.remove('flipped');
  flipped=[];
  lock=false;
},800);
}

}
});
</script>

</body>
</html>
```

guardar_writing2.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) {
    echo "No autorizado";
    exit();
}

$user_id = $_SESSION['user_id'];

if (!isset($_POST['puntaje'])) {
    echo "No hay puntaje";
    exit();
}

$puntaje = (int)$_POST['puntaje'];

$stmt = $conn->prepare("SELECT writing2 FROM puntajes WHERE user_id=?");
$stmt->bind_param("i", $user_id);
$stmt->execute();
$res = $stmt->get_result();
$data = $res->fetch_assoc();

if ($data) {
    if ($puntaje > $data['writing2']) {
        $stmt2 = $conn->prepare("UPDATE puntajes SET writing2=? WHERE user_id=?");
        $stmt2->bind_param("ii", $puntaje, $user_id);
        $stmt2->execute();
    }
} else {
    $stmt2 = $conn->prepare("INSERT INTO puntajes (user_id, writing2) VALUES (?, ?)");
    $stmt2->bind_param("ii", $user_id, $puntaje);
    $stmt2->execute();
}

echo "ok";
```


guardar_memorama_2do.php

```
<?php
session_start();
include 'db.php';

if (!isset($_SESSION['user_id'])) exit('No logueado');

$user_id = $_SESSION['user_id'];
$puntaje = isset($_POST['puntaje']) ? (int)$_POST['puntaje'] : 0;

$stmt = $conn->prepare("UPDATE puntajes SET juego2 = GREATEST(juego2, ?) WHERE user_id=?");
$stmt->bind_param("ii", $puntaje, $user_id);
$stmt->execute();

if ($stmt->affected_rows === 0) {
    $stmt = $conn->prepare("INSERT INTO puntajes (user_id, juego2) VALUES (?, ?)");
    $stmt->bind_param("ii", $user_id, $puntaje);
    $stmt->execute();
}

echo 'OK';
?>
```

MANTENIMIENTO DEL SISTEMA

El mantenimiento del sistema SpeakUp! garantiza su funcionamiento correcto, seguro y estable con el paso del tiempo. Consta de acciones preventivas y correctivas que permiten conservar la integridad de la base de datos, los archivos del proyecto y el rendimiento general de la plataforma.

Este proceso es especialmente importante cuando se ejecuta en equipos locales mediante WAMP, así como cuando se administran usuarios, puntajes y actividades interactivas.

A continuación, se describen las tareas recomendadas.

✂ 7.1 Mantenimiento Preventivo

Estas acciones ayudan a evitar fallos futuros y asegurar que el sistema siga siendo funcional:

✓ Copias de seguridad de la base de datos

Se recomienda realizar respaldos periódicos de `speakup_db`, que contiene las tablas usuarios y puntajes.

Pasos sugeridos:

1. Abrir el navegador y acceder a:
`http://localhost/phpmyadmin`
2. Seleccionar la base de datos `speakup_db`.
3. Entrar en la pestaña Exportar.
4. Seleccionar el formato SQL.
5. Descargar y almacenar el archivo respaldado en un lugar seguro.

✦ Frecuencia sugerida: semanal si el sistema está en uso activo.

✓ Verificación del servidor WAMP

Para evitar caídas del sistema por fallos en Apache o MySQL:

1. Revisar que el ícono de WAMP se encuentre verde.
2. En caso de estar naranja/rojo, reiniciar los servicios:

WAMP → Restart All Services

✓ Limpieza de archivos temporales

Algunas actividades del sistema generan imágenes, archivos subidos o datos no permanentes. Es recomendable revisar periódicamente las carpetas:

■ /www/SpeakUp/uploads/

■ /www/SpeakUp/temp/ (si aplica)

Eliminar archivos que ya no sean necesarios ayuda a evitar pérdida de espacio y errores.

Mantenimiento Correctivo

Se aplica cuando el sistema presenta errores visibles, como:

- Fallas al iniciar sesión
- Problemas al cargar actividades o juegos
- Datos incorrectos en puntajes o perfiles
- Advertencias o errores en PHP

Acciones recomendadas

Problema frecuente	Solución recomendada
Error de conexión a la base de datos	Verificar archivo db.php (host, usuario, contraseña, nombre de BDD)
Apache no inicia	Verificar si otro programa usa el puerto 80 (Skype, IIS, etc.)
MySQL detenido	Reiniciar servicios desde WAMP
Datos corruptos o perdidos	Restaurar respaldo SQL de la base speakup_db
Error 500 o pantalla en blanco	Activar modo debug en PHP o revisar sintaxis

Actualización del sistema

Cuando se agreguen nuevas actividades, estilos o funciones, se recomienda:

1. Guardar una copia del proyecto actual.
2. Editar los archivos PHP desde Notepad++ u otro editor.
3. Probar cambios en un entorno local antes de sustituir el sistema en uso.
4. Verificar nuevamente el correcto funcionamiento de:
 - Registro e inicio de sesión
 - Juegos y módulos por grado
 - Guardado de puntajes

IMPACTO AMBIENTAL

El uso de la plataforma SpeakUp! contribuye a la reducción del impacto ambiental al disminuir la necesidad de materiales impresos en el proceso de enseñanza-aprendizaje del idioma inglés. Al trasladar ejercicios, evaluaciones y actividades a un entorno digital, se reduce el consumo de papel, tinta y otros recursos físicos que generan residuos difíciles de reciclar.

Asimismo, el sistema opera en equipos de cómputo convencionales, lo que permite su implementación sin la adquisición constante de dispositivos nuevos, favoreciendo la reutilización de recursos tecnológicos existentes.

Beneficios ambientales:

- Disminución del uso de papel y materiales de impresión.
- Reducción de residuos derivados de actividades escolares tradicionales.
- Aprovechamiento de equipos informáticos ya existentes.
- Fomento del aprendizaje digital como alternativa más sustentable.

Para reducir el impacto energético derivado del uso de computadoras y servidores locales se recomienda:

- Activar configuraciones de ahorro de energía en los dispositivos.
- Apagar los equipos cuando el sistema no esté en uso.
- Promover la utilización responsable de los recursos digitales.

SpeakUp! representa una alternativa educativa que puede contribuir a la reducción del impacto ambiental y apoyar prácticas más responsables dentro del ámbito escolar.

CONCLUSIÓN

SpeakUp! es una plataforma educativa en proceso que esta enfocada para fortalecer el aprendizaje del idioma inglés en estudiantes de nivel primaria mediante actividades digitales, juegos y contenido interactivo. Su propuesta busca que el estudio del idioma sea más dinámico, accesible y motivador para los usuarios.

Este manual proporciona los elementos necesarios para la instalación, gestión y mantenimiento del sistema, así como para la administración de su base de datos y el funcionamiento general de las actividades educativas que contiene. Con ello se garantiza que la plataforma pueda operar correctamente y mantenerse en condiciones óptimas con el paso del tiempo.

Además de su aporte académico, el sistema contribuye a reducir el uso de materiales impresos, lo que representa una alternativa más responsable con el medio ambiente dentro de los procesos educativos.

En conclusión, SpeakUp! constituye una herramienta útil para apoyar el aprendizaje del idioma inglés en un entorno digital y actual, facilitando una experiencia educativa más dinámica y promoviendo el uso consciente de los recursos tecnológicos. Su aplicación y posible crecimiento futuro dependerán del uso, adaptación y evolución que se le otorgue en las instituciones educativas o espacios de aprendizaje donde sea implementado.