

## Historia de Usuario 4

### Descripción conceptual

<b>Módulo</b>	<i>Textfield de nickname</i>
<b>Descripción de la(s) funcionalidad(es) requerida(s):</b>	<i>El usuario debe usar un campo de texto donde escribir su nombre o nickname de preferencia, el cual será usado posteriormente para identificarlo durante la partida.</i>

### Backend

<b>URL</b> <b>localhost://nickname</b>	<b>Método</b> <b>POST</b>	<b>Código http</b> <b>200 (éxito), 400 (error en la entrada)</b>
<i>Esta historia de usuario permite al usuario enviar su nombre o nickname a través de un campo de texto. Si el nombre es válido, se retorna un código 200 y se almacena en la base de datos. Si el nombre no cumple con los requisitos (por ejemplo, está vacío o contiene caracteres no permitidos), se retorna un código 400 con un mensaje de error.</i>		
<i>{ "nickname": "NombreDelUsuario" }</i>	<i>200: { "status": "success", "message": "Nickname guardado correctamente" } 400: { "status": "error", "message": "El nickname no puede estar vacío o contiene caracteres no permitidos" }</i>	

### Frontend

#### **Interacción esperada:**

*El usuario ingresa a la pantalla de inicio de sesión o configuración de perfil.  
Encuentra un campo de texto etiquetado como "Nombre".  
El usuario escribe su nombre o nickname de preferencia en el campo de texto.  
El usuario hace clic en un botón "Buscar partida" para enviar la información.*

#### **Flujo visual y eventos:**

*El campo de texto está vacío inicialmente.  
El usuario escribe su nombre o nickname.  
Si el campo está vacío o contiene caracteres no permitidos, se muestra un mensaje de error debajo del campo de texto.  
Si el nombre es válido, al hacer clic en "Buscar partida", se muestra un mensaje de éxito y el sistema redirige al usuario a la siguiente pantalla*

#### **Mockups/Prototipos:**



**Notas Adicionales:**

## Historia de Usuario 5

### Descripción conceptual

Módulo	Modo de juego Normal
Descripción de la(s) funcionalidad(es) requerida(s):	Esta es la jugabilidad de Twordle, aquí los usuarios se enfrentan a adivinar la palabra correcta antes que su contrincante.

### Backend

URL localhost:8080/juego	Método POST	Código http 200
Caso de uso técnico: Este endpoint inicia una nueva partida en el modo Normal. Retorna un código 200 si la partida se inicia correctamente, junto con un identificador único de la partida. Si hay un error, retorna un código 400 con un mensaje de error.		
{ "player1": "id_jugador1" }		{ "status": "success", "data": { "gameId": "12345" } }

### Frontend

#### Interacción esperada:

El usuario selecciona el modo de juego "Normal" desde el menú principal.  
 El usuario ingresa una palabra en el campo de texto y presiona el botón "Enviar".  
 El sistema valida la palabra y muestra si es correcta o incorrecta, junto con una pista visual (por ejemplo, colores que indican letras correctas o incorrectas).  
 El juego continúa hasta que adivina la palabra correcta o se agotan los intentos.

#### Flujo visual y eventos:

Al iniciar la partida, se muestra un contador de intentos restantes.  
 Cuando el usuario envía una palabra, se muestra un spinner mientras el sistema valida la palabra.  
 Si la palabra es incorrecta, se muestra un mensaje de error y se actualiza el contador de intentos.  
 Si la palabra es correcta, se muestra un mensaje de victoria y se finaliza la partida.

### Mockups/Prototipos:



## Historia de Usuario 6

### Descripción conceptual

Módulo	Opción de Nueva Partida
Descripción de la(s) funcionalidad(es) requerida(s):	Al finalizar la partida, el usuario tiene la opción de buscar una nueva partida.

### Backend

URL localhost:8080/juego	Método POST	Código http 200 400
Este endpoint inicia una nueva partida en el modo Normal. Retorna un código 200 si la partida se inicia correctamente, junto con un identificador único de la partida. Si hay un error, retorna un código 400 con un mensaje de error.		
{ "player1": "id_jugador1" }	{ "status": "success", "data": { "gameId": "12345" } }	

### Frontend

### Interacción esperada:

El usuario ha completado una partida en el modo Normal (ha adivinado la palabra correcta o ha agotado sus intentos).

El sistema muestra un mensaje de resultado:

Si el usuario adivinó la palabra, se muestra: "¡Ganaste! Has adivinado la palabra correcta."

Si el usuario no adivinó la palabra, se muestra: "¡Perdiste! La palabra correcta era: [palabra oculta]."

Junto al mensaje de resultado, se muestra un botón con la opción: "Jugar de nuevo".

### Flujo visual y eventos:

*Pantalla de resultado: Muestra el mensaje de victoria o derrota.*

*Incluye un botón "Jugar de nuevo" que permite al usuario iniciar una nueva partida*

*Al hacer clic en "Jugar de nuevo", el sistema muestra un breve mensaje de carga (por ejemplo, un spinner con el texto "Preparando nueva partida...").*

*Una vez que la nueva partida está lista, el sistema muestra la interfaz de juego reiniciada.*

*Interfaz de juego reiniciada:*

*El contador de intentos se restablece.*

### **Mockups/Prototipos:**

