Registro Diario de Avances

**Fecha:** 11/10/2025

**Autor:**

**Danluis Enrique Romani Mora**

**Eduardo Chaparro Huaman**

**Fabrizcio Argandoña Montalvo**

**Versión:** 1.0

# 1. Introducción

En este documento se muestra el avance de tareas respectivas al Sprint 3 con lo que también se detallan las historias de usuarios pertenecientes y el esfuerzo realizado para completar cada una de ellas, con evidencias del código utilizado, burndown chart y el análisis respectivo al rendimiento del equipo.

# 2. Lineamientos de Diseño

Desarrollar una tercera versión funcional de la aplicación que permita a los usuarios:

* Inicio de sesión y acceso de los alumnos
* Mostrar mensajes en caso de error al momento de iniciar sesión.
* La posibilidad de recuperar la contraseña.
* La visualización del test de inteligencias múl tiples.

# 3. Historias de Usuario y Tareas

| Backlog ID | Historia de Usuario | Tiempo Estimado | Fecha de Inicio | Fecha de Finalización |
| --- | --- | --- | --- | --- |
| HU-2.2 | Validez del test de Inteligencias Múltiples | 16 | 1/10/2025 | 3/10/2025 |
| HU-2.3 | Realizar test de inteligencias múltiples | 10 | 1/10/2025 | /10/2025 |
| HU-2.4 | Cálculo automático del puntaje | 4 | 1/10/2025 | /10/2025 |
| HU-2.5 | Resultado del tipo de inteligencia predominante | 2 | /10/2025 | /10/2025 |

# 4. Evidencias de Código

## 4.1 Historia de Usuario 2.2: Validez del test de inteligencias múltiples.

Archivo inicio.docente.component.html

<body>

<div class="overlay"></div>

<div class="panel-container">

<h2>Docente: </h2>

<div class="panel-header">

<button class="alumnos-button" (click)="mostrarSeccion('alumnos')">alumnos</button>

<button class="test-button" (click)="mostrarSeccion('test')">Test</button>

</div>

*<!-- Sección de alumnos -->*

<div \*ngIf="seccionActual === 'alumnos'" class="alumnos-container">

<div class="alumnos-lista">

<button \*ngFor="let alumnos of alumnos" class="alumno-button">

{{ alumnos }}

</button>

</div>

</div>

*<!-- Sección de test -->*

<div \*ngIf="seccionActual === 'test'" class="alumnos-container">

<div class="alumnos-lista">

<button class="alumno-button" (click)="irATestGardner()">Test de Gardner</button>

*<!-- Nuevo botón Información -->*

<button class="info-button" (click)="abrirModal()">Información</button>

</div>

</div>

<button class="logout-button" (click)="logout()">Cerrar Sesión</button>

</div>

*<!-- Modal -->*

<div class="modal-overlay" \*ngIf="modalAbierto">

<div class="modal">

<h3>Teorías de Gardner</h3>

<p>

La teoría de las inteligencias múltiples, propuesta por Howard Gardner,

sostiene que no existe una única inteligencia, sino varias formas de

inteligencia que las personas desarrollan en distintos grados. Estas incluyen,

entre otras: la lingüística, lógico-matemática, espacial, musical,

corporal-kinestésica, interpersonal, intrapersonal y naturalista.

</p>

<p>

En el ámbito educativo, esta teoría es fundamental porque permite reconocer

y potenciar las habilidades diversas de los estudiantes, promoviendo una

enseñanza más inclusiva, personalizada y efectiva.

</p>

<div class="modal-buttons">

<button class="link-button" (click)="abrirEnlace()">Más información</button>

<button class="close-button" (click)="cerrarModal()">Cerrar</button>

</div>

</div>

</div>

</body>

archivo ts

import { Component } from '@angular/core';

import { Router } from '@angular/router';

import { FormsModule } from '@angular/forms';

import { CommonModule } from '@angular/common';

@Component({

selector: 'app-inicio-docentes',

imports: [FormsModule, CommonModule],

templateUrl: './inicio-docentes.component.html',

styleUrl: './inicio-docentes.component.css'

})

export class InicioDocentesComponent {

seccionActual: 'alumnos' | 'test' = 'alumnos';

modalAbierto = false;

alumnos = [

'Juan', 'Pedro', 'Luis', 'Enrique', 'Carlos', 'Diego', 'Fernando',

'Gabriel', 'Miguel', 'Jose', 'Maria', 'Ana', 'María', 'Laura',

'Sofía', 'Pablo', 'Javier', 'Daniel', 'Ángel', 'Lucas',

];

constructor(*private* router: Router) {}

mostrarSeccion(seccion: 'alumnos' | 'test') {

this.seccionActual = seccion;

}

irATestGardner() {

this.router.navigate(['/test-gardner']);

}

abrirModal() {

this.modalAbierto = true;

}

cerrarModal() {

this.modalAbierto = false;

}

abrirEnlace() {

window.open('https://utecno.wordpress.com/wp-content/uploads/2014/07/howard\_gardner\_-\_estructuras\_de\_la\_mente.pdf', '\_blank'); *// Cambia este enlace*

}

logout() {

localStorage.clear();

this.router.navigate(['/login']);

}

}

## 4.2 Historia de Usuario 2.3: Realizar test de inteligencias múltiples

Archivo test-gardner.component.html

<body>

<div class="overlay"></div>

<div class="panel-container">

<h2>Test de Gardner</h2>

<div class="test-layout">

*<!-- Panel izquierdo: preguntas -->*

<div class="preguntas-section">

<form (ngSubmit)="onSubmit()">

<div \*ngFor="let pregunta of preguntasPaginadas" class="pregunta-card">

<p>{{pregunta.id}}. {{pregunta.texto}}</p>

<div class="opciones">

<label \*ngFor="let opcion of [1,2,3,4,5,6]" [class.selected]="pregunta.respuesta === opcion">

<input

type="radio"

[name]="'pregunta-' + pregunta.id"

[value]="opcion"

[(ngModel)]="pregunta.respuesta"

required

/>

{{ opcion }}

</label>

</div>

</div>

*<!-- Usa el template -->*

<ng-container \*ngTemplateOutlet="navegacion"></ng-container>

<button type="submit" class="item-button">Enviar Test</button>

</form>

</div>

*<!-- Panel derecho: índice -->*

<div class="indice-section">

<h3>Preguntas</h3>

<div class="indice-lista">

<button

\*ngFor="let pregunta of preguntas"

class="indice-item"

[class.respondida]="pregunta.respuesta !== null"

[class.activa]="pregunta.id >= inicioIndice && pregunta.id < inicioIndice + preguntasPorPagina"

(click)="irAPagina(pregunta.id)">

{{ pregunta.id }}

</button>

</div>

*<!-- Usa el mismo template -->*

<ng-container \*ngTemplateOutlet="navegacion"></ng-container>

</div>

</div>

</div>

*<!-- Template reutilizable -->*

<ng-template #navegacion>

<div class="navegacion">

<div class="nav-izquierda">

<button

type="button"

class="nav-button"

[class.oculto]="paginaActual <= 1"

(click)="anteriorPagina()"

>

Anterior

</button>

</div>

<div class="nav-derecha">

<button

type="button"

class="nav-button"

[class.oculto]="paginaActual >= totalPaginas"

(click)="siguientePagina()"

>

Siguiente

</button>

</div>

</div>

</ng-template>

</body>

ts:

import { Component } from '@angular/core';

import { Router } from '@angular/router';

import { CommonModule } from '@angular/common';

import { FormsModule } from '@angular/forms';

import Swal from 'sweetalert2';

@Component({

selector: 'app-test-gardner',

standalone: true,

imports: [CommonModule, FormsModule],

templateUrl: './test-gardner.component.html',

styleUrls: ['./test-gardner.component.css']

})

export class TestGardnerComponent {

preguntasPorPagina = 5;

paginaActual = 1;

preguntas = [

{ id: 1, texto: 'Me gusta resolver problemas matemáticos o lógicos.', respuesta: null },

{ id: 2, texto: 'Disfruto participar en actividades musicales.', respuesta: null },

{ id: 3, texto: 'Prefiero aprender a través de imágenes, diagramas o mapas mentales.', respuesta: null },

{ id: 4, texto: 'Puedo expresar mis ideas fácilmente con palabras.', respuesta: null },

{ id: 5, texto: 'Me resulta fácil trabajar en grupo.', respuesta: null },

{ id: 6, texto: 'Me gusta observar la naturaleza y aprender de ella.', respuesta: null },

{ id: 7, texto: 'Soy consciente de mis emociones y cómo influyen en mí.', respuesta: null },

{ id: 8, texto: 'Me gusta moverme y aprender haciendo actividades físicas.', respuesta: null },

{ id: 9, texto: 'Disfruto resolver acertijos y desafíos.', respuesta: null },

{ id: 10, texto: 'Me inspiro al escuchar música.', respuesta: null },

{ id: 11, texto: 'Me gusta leer libros.', respuesta: null },

{ id: 12, texto: 'Me gusta jugar videojuegos.', respuesta: null },

{ id: 13, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 14, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 15, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 16, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 17, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 18, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 19, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 20, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

{ id: 21, texto: 'Me gusta jugar a los juegos de mesa.', respuesta: null },

];

constructor(*private* router: Router) {}

get totalPaginas() {

*return* Math.ceil(this.preguntas.length / this.preguntasPorPagina);

}

get inicioIndice() {

*return* (this.paginaActual - 1) \* this.preguntasPorPagina + 1;

}

get preguntasPaginadas() {

*const* inicio = (this.paginaActual - 1) \* this.preguntasPorPagina;

*return* this.preguntas.slice(inicio, inicio + this.preguntasPorPagina);

}

siguientePagina() {

if (this.paginaActual < this.totalPaginas) this.paginaActual++;

}

anteriorPagina() {

if (this.paginaActual > 1) this.paginaActual--;

}

irAPagina(id: number) {

this.paginaActual = Math.ceil(id / this.preguntasPorPagina);

}

onSubmit() {

*const* incompletas = this.preguntas.filter(p => p.respuesta === null);

if (incompletas.length > 0) {

Swal.fire({

icon: 'warning',

title: 'Faltan respuestas',

text: 'Por favor responde todas las preguntas antes de enviar.',

});

*return*;

}

console.log('Respuestas del test:', this.preguntas);

Swal.fire({

icon: 'success',

title: 'Respuestas guardadas',

text: 'El resultado se está calculando...',

timer: 2500,

showConfirmButton: false

}).then(() => {

this.router.navigate(['/inicio-alumnos']);

});

}

}

## 4.4 Historia de Usuario 2.4: Cálculo automático del puntaje

Archivo

## 

## 4.5 Historia de Usuario 2.5: Resultado del tipo de inteligencia predominante

Archivo perfil-alumno.component.html

<body>

<div class="overlay"></div>

<div class="panel-container">

<h2>Perfil del Alumno</h2>

<div class="panel-header">

<button class="perfil-button" [class.active]="seccionActual==='perfil'" (click)="mostrarSeccion('perfil')">Perfil</button>

<button class="cursos-button" [class.active]="seccionActual==='cursos'" (click)="mostrarSeccion('cursos')">Cursos</button>

<button class="test-button" [class.active]="seccionActual==='inteligencias'" (click)="mostrarSeccion('inteligencias')">Inteligencias</button>

</div>

*<!-- Sección Perfil -->*

<div \*ngIf="seccionActual === 'perfil'" class="perfil-container">

<img [src]="getFotoPerfil()" alt="Foto de perfil" class="foto-perfil">

<div class="datos-alumno">

<p><strong>Nombre:</strong> {{ alumno.nombre }}</p>

<p><strong>Edad:</strong> {{ alumno.edad }} años</p>

</div>

</div>

*<!-- Sección Cursos -->*

<div \*ngIf="seccionActual === 'cursos'" class="cursos-container">

<div class="cursos-lista">

<button \*ngFor="let curso of cursos" class="curso-button">

{{ curso }}

</button>

</div>

</div>

*<!-- Sección Inteligencias -->*

<div \*ngIf="seccionActual === 'inteligencias'" class="inteligencias-container">

<div \*ngFor="let intel of inteligencias" class="inteligencia-item">

<span class="tipo">{{ intel.tipo }}</span>

<div class="barra">

<div class="progreso" [style.width.%]="intel.nivel">{{ intel.nivel }}%</div>

</div>

</div>

</div>

<button class="logout-button" (click)="logout()">Cerrar Sesión</button>

</div>

</body>

ts :

import { Component } from '@angular/core';

import { CommonModule } from '@angular/common';

import { Router } from '@angular/router';

@Component({

selector: 'app-perfil-alumno',

standalone: true,

imports: [CommonModule],

templateUrl: './perfil-alumno.component.html',

styleUrls: ['./perfil-alumno.component.css']

})

export class PerfilAlumnoComponent {

constructor (*private* router: Router) { }

seccionActual: string = 'perfil';

alumno = {

nombre: 'Juan Pérez',

genero: 'M', *// 'M' o 'F'*

edad: 16,

};

inteligencias = [

{ tipo: 'Lingüística', nivel: 85 },

{ tipo: 'Lógico-Matemática', nivel: 90 },

{ tipo: 'Espacial', nivel: 70 },

{ tipo: 'Corporal-Kinestésica', nivel: 60 },

{ tipo: 'Musical', nivel: 50 },

{ tipo: 'Interpersonal', nivel: 80 },

{ tipo: 'Intrapersonal', nivel: 75 },

{ tipo: 'Naturalista', nivel: 65 }

];

cursos = ['Matemática I', 'Programación', 'Física', 'Inglés', 'Estadística'];

mostrarSeccion(seccion: string) {

this.seccionActual = seccion;

}

logout() {

console.log('Sesión cerrada');

this.router.navigate(['/login']);

}

getFotoPerfil(): string {

*return* this.alumno.genero === 'F'

? 'https://cdn-icons-png.flaticon.com/512/201/201634.png'

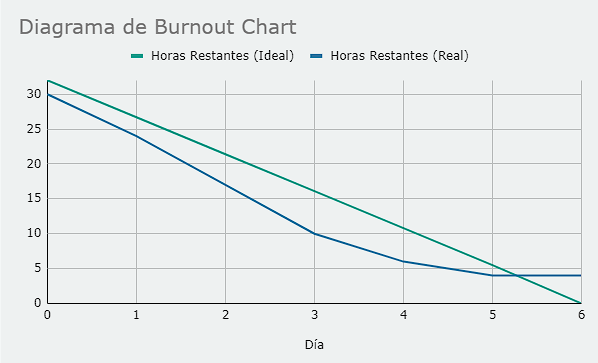
: 'https://cdn-icons-png.flaticon.com/512/201/201623.png';

}

}

# 5. Burndown Chart

| Backlog ID | Historia de Usuario | Tiempo Estimado | 08/10/2025 | 09/10/2025 | 10/10/2025 | 11/10/2025 | 13/10/2025 | 14/10/2025 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Dia1** | **Dia2** | **Dia3** | **Dia4** | **Dia5** | **Dia6** |
| HU-2.2 | Validez del test de Inteligencias Múltiples | 16 | 2 | 4 | 4 | 3 | 3 |  |
| HU-2.3 | Realizar test de inteligencias múltiples | 10 |  | 2 | 2 | 3 | 1 | 1 |
| HU-2.4 | Cálculo automático del puntaje | 4 |  |  |  |  |  | 1 |
| HU-2.5 | Resultado del tipo de inteligencia predominante | 2 |  |  | 1 | 1 |  |  |
| Tiempo de trabajo | | 32 | 30 | 24 | 17 | 10 | 6 | 4 |



# 6. Análisis del Sprint

* Esta vez iniciamos con 2 horas iniciales extra debido al rendimiento del sprint 3 donde se avanzaron previo la historia de usuario 2.2
* Esta es el primer sprint donde nos encontramos con una dificultad para avanzar con respecto a las Historias de usuario 2.3 y 2.4 donde se intento refactorizar estos varias veces en busca de

# 7. Conclusiones