

Control de documento

Nombre del proyecto	RP COACHING
Cierre de iteración I2	17-28/10/2022
Generador por	Magnolia Hernández Aguilera
Aprobado por	Jesús Adrián López Luevanos
Alcance de la distribución del documento	Control interno para todo el proyecto.

Índice

Sobre este documento	3
Resumen de la Iteración	4
Identificación.....	4
Hitos especiales.....	4
Artefactos y evaluación.....	5
Riesgos y problemas	6
Notas y observaciones	6
Asignación de recursos	7
Anexos.....	8
Glosario de términos.....	9
Significado de los elementos de la notación gráfica	11
Estereotipado UML utilizado	11
OSignificado de los elementos No UML	11

Sobre este documento

La calidad se logra por medio de la revisión constante de las actividades que conducen desde la idea al producto. Al momento del cierre de una iteración es buen momento para hacer un alto, y evaluar lo logrado, los problemas encontrados y los retos a enfrentar.

El presente documento marca el final de la iteración I7-28/10/2022, y contiene una evaluación de los artefactos y actividades realizadas durante la misma.

Se recogen también las impresiones y observaciones hechas durante el desarrollo de la iteración, así como el esfuerzo invertido en cada una de las disciplinas involucradas.

Resumen de la Iteración

Identificación

Código de iteración: I2, E1, C1, T2...

Se suele usar la siguiente convención: I, E, C, T por la inicial de la fase a la que pertenece la iteración: Inicio, Elaboración, Construcción o Transición.

Se sigue con un número o correlativo que cuenta desde uno.

Fecha de inicio y cierre es auto explicativo. Lo mismo con los comentarios, de haberlos.]

Código de la iteración	Fase a la que pertenece	Fecha de inicio	Fecha de cierre	Comentarios
I7	Inicio	22/10/2022	27/10/2022	Se presento una pequeña falta de organización por parte del equipo, pero se logró solucionar con una reunión intensiva en la cual se vieron puntos importantes.

Hitos especiales

Factor	Metrica	Calificacion	Comentario
Correccion	Compleitud	2	La metrica de completitud nos indica que tan completo esta nuestro proyecto y nos facilita medir el grado en que satisface sus especificaciones y ver si realmente se alcanzaron los objetivos del usuario.
Fiabilidad	Exactitud	2	Elejimos la exactitud como metrica ya que nos puede ayudar a medir de manera precisa que tan bien lleva acabo sus funciones esperadas.
Eficiencia	Eficiencia en la ejecucion	1	En este caso se elijio medir la eficiencia en la ejecucion para poder determinar la cantidad de recursos que gasta la computadora y el codigo del programa.
Integridad	Seguridad	1	En integridad es correcto medir la seguridad para controlar el acceso al software o a los datos al personal no autorizado
Facilidad de uso	Facilidad de operación	2	Enfocados en la facilidad de operación podemos encontrar el esfuerzo requerido para aprender a usar e interpretar la informacion de entrada y salida que nuestra aplicación ofrece.
Facilidad de mantenimiento	Tolerancia de errores	2	Dependiendo de la tolerancia de errores de nuestra aplicación se vera reflejado el esfuerzo requerido para localizar y arreglar nuestro programa.
Facilidad de pruebas	Facilidad de auditorias	2	En este factor la metrica de facilidad de auditorias nos ayudara a tener en optimas condiciones la aplicación y asi reducir el esfuerzo cuando se necesite probar el programa.
Flexibilidad	Modularidad	1	Tener una aplicación con modularidad nos ayudara a que sea mas flexible y reduira el esfuerzo que se necesite para modificar el sistema operativo
Portabilidad	Facilidad de expansion	1	En la portabilidad se busca transferir tanto software como hardware de un sistema a otro por eso mismo usaremos la facilidad de expansion como metrica ayudandonos a ver que tan portable puede ser nuestra aplicación y que cantidad de esfuerzo requiere.
Reusabilidad	Consistencia	3	Usando la consistencia como metrica podremos darnos cuenta que tan consistente es nuestra aplicación y asi saber que partes de nuestro programa puede ser reutilizada en nuevos proyectos.
Facilidad de operabilidad		1	
		18	

Artefactos y evaluación

Artefacto	Meta (%)	Comentarios
(EJ01) Instalación del entorno software para el desarrollo de la aplicación móvil	100%	
(P01) Generar pantalla de carga de la aplicación en el entorno de desarrollo	100%	

Artefacto	Aspecto a evaluar	Evaluación	Comentarios
(EJ01)	Instalación del entorno software para el desarrollo de la aplicación	<ul style="list-style-type: none"> Revisión de los requerimientos mínimos de hardware para la instalación del entorno de desarrollo. 	
T03	Generar pantalla de carga de la aplicación en el entorno de desarrollo	<ul style="list-style-type: none"> La realización de la pantalla de carga contiene pocos elementos. El previo diseño de la pantalla facilitó el trabajo en el acomodo de los elementos. 	

Riesgos y problemas

Riesgo: al inicio del sprint no se tenía claridad en las actividades que se iban a desarrollar en la semana, ya que los miembros del equipo estaban algo dispersos.

Solución: hacer una reunión con todo el equipo involucrado en el desarrollo de la aplicación y hacer revisión del cronograma que se programó cuando se inició el proyecto, así como asignar las actividades correspondientes a cada miembro del equipo.

Notas y observaciones

El desarrollo del sprint no se vio tan afectado por la falta de claridad en las actividades, pero en los miembros del equipo se vio una actitud de individualismo.

Asignación de recursos

Rol	Horas-Hombre	Desempeñado por	Observaciones
Analista	56	Magnolia Hernández Aguilera	
Verificador	56	Jesús Rodolfo Terán Cuellar	
Documentador	56	Jesús Adrián López Luevanos	

Anexos

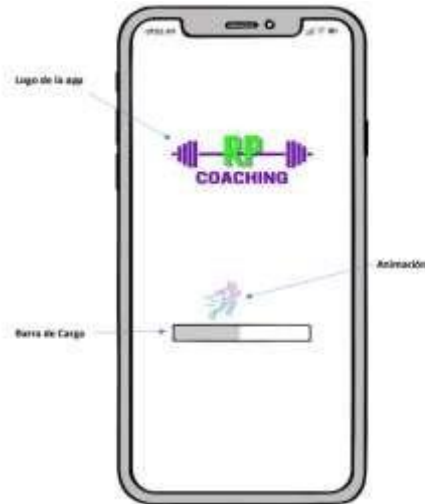
(EJ01)

	Requisitos mínimos	Requisitos recomendados
Sistema operativo	Windows 7(SP1+), windows 10. versiones de 64 bits	Windows 11 64 bit
CPU	Arquitectura x64 con instrucciones sse2 (Intel Core 2 Duo, Extreme o Quad, Intel Core i3, AMD Sempron, Phenom I y II y Athlon II)	Arquitectura x64 con instrucciones sse2 (Intel core i5, i7, AMD FX Y Ryzen)
API de gráficos	DX 10 Y 11	DX 12
Requisitos adicionales	Drivers de hardware compatibles	Drivers de hardware compatibles
Gráficos	Tarjeta de video de 512 mb (GTX 650 en adelante)	RTX 3070(para proyectos 3D)
RAM	Desde 4Gb	16Gb
Espacio en disco	10Gb	10gb

Retroalimentación: el objetivo final de realizar la actividad durante el sprint es para poder saber los requerimientos y características que deberían tener nuestros equipos para poder hacer la instalación y la ejecución de manera correcta del entorno de desarrollo seleccionado.

T03

Plantilla de pantalla de carga.



Código de la barra de carga:

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4  using UnityEngine.SceneManagement;
5  using UnityEngine.UI;
6
7  public class CargadorNivel : MonoBehaviour
8  {
9      public GameObject PantallaDeCarga;
10     public Slider Slider;
11
12     public void CargarNivel(int NumeroDeEscena)
13     {
14         StartCoroutine(CargarAsync(NumeroDeEscena));
15     }
16
17     IEnumerator CargarAsync(int NumeroDeEscena)
18     {
19         AsyncOperation Operacion = SceneManager.LoadSceneAsync(NumeroDeEscena);
20
21         PantallaDeCarga.SetActive(true);
22
23         while (!Operacion.isDone)
24         {
25             float Progreso = Mathf.Clamp01(Operacion.progress / .9f);
26             Slider.value = Progreso;
27
28             yield return null;
29         }
30     }
31 }

```

Glosario de términos

Drivers:

Un controlador o driver es el software que controla un dispositivo en un PC; por ejemplo, una tarjeta de vídeo o de sonido. Los controladores actúan como “puentes” entre las aplicaciones y los dispositivos, encargándose de que ambos interactúen.

SSE:

(Streaming SIMD Extensions) es una extensión al grupo de instrucciones MMX para procesadores Pentium III, introducida por Intel en febrero de 1999.

Significado de los elementos de la notación gráfica

Estereotipado UML utilizado

0Significado de los elementos No UML