
Especificación de requisitos de software

Proyecto:

EnterARTE

Instrucciones para el uso de este formato

Este formato es una plantilla tipo para documentos de requisitos del software.

Está basado y es conforme con el estándar IEEE Std 830-1998.

Los textos en color azul son indicaciones que deben eliminarse y, en su caso, sustituirse por los contenidos descritos en cada apartado.

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. Calidad.
2023		Grupo enterARTE	

Contenido

<u>FICHA DEL DOCUMENTO</u>	3
<u>CONTENIDO</u>	4
<u>1 INTRODUCCIÓN</u>	6
<u>1.1 Propósito</u>	6
<u>1.2 Alcance</u>	6
<u>1.3 Personal involucrado</u>	6
<u>1.4 Definiciones, acrónimos y abreviaturas</u>	7
<u>1.5 Referencias</u>	7
<u>1.6 Resumen</u>	7
<u>2 DESCRIPCIÓN GENERAL</u>	7
<u>2.1 Perspectiva del producto</u>	7
<u>2.3 Características de los usuarios</u>	8
<u>2.4 Restricciones</u>	9
<u>3 REQUISITOS ESPECÍFICOS</u>	9
<u>3.2 Product Backlog</u>	9
<u>3.2.1 Product Backlog</u>	9
<u>3.3 Sprints</u>	11
<u>3.3.1 Sprint 0</u>	11
<u>3.3.2 Sprint 1</u>	12
<u>3.3.3 Sprint 2</u>	13
<u>3.3.4 Sprint 3</u>	14
<u>3.3.5 Sprint 4</u>	15

1 Introducción

1.1 Propósito

1. El presente documento tiene como propósito definir las especificaciones funcionales para el desarrollo de un sistema de información web.
Por medio de este documento definiremos los requerimientos que cumplirá la aplicación web "enterARTE", consistente en gestionar la información y la compra de entradas de espacios y eventos culturales.
2. El documento está dirigido a desarrolladores que se ocupen del análisis, diseño e implementación del sitio web.

1.2 Alcance

1. El sitio web llevará el nombre de enterARTE.
2. El mismo permitirá recopilar una agenda cultural completa y ordenada para un sector geográfico determinado (Argentina), que incluya información detallada de espacios culturales y eventos de producción independiente, así como el posteo de dichos eventos por parte de los organizadores
3. El sitio contará con un carrito de compras que permitirá a los usuarios interesados comprar entradas para cualquier evento al que deseen asistir.
4. El sistema será desarrollado como una aplicación web para múltiples dispositivos (responsive design) para cumplir con el objetivo de la automatización de las operaciones (registro de eventos y espacios culturales y adquisición de entradas).

1.3 Personal involucrado

Nombre	Karen Pigni
Rol	Analista, desarrolladora, programadora
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	karen.pigni.estudio@gmail.com

Nombre	Francisco Gemolotto
Rol	Analista, desarrollador, programador
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	franciscogemoo@gmail.com

Nombre	Tayhana Villarreal
Rol	Analista, desarrolladora, programadora
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	villarrealtay@gmail.com

Nombre	Julian Meneses
Rol	Analista, desarrollador, programador
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	julian.meneses11@gmail.com

Nombre	Gerardo Antonio Castellanos Torres
Rol	Analista, desarrollador, programador
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	gcastelt@gmail.com

Nombre	Soledad Orrego
Rol	Analista, desarrollador, programador
Categoría Profesional	Informática
Responsabilidad	Análisis de información, diseño y programación del SIS-I
Información de contacto	ms.orrego21@gmail.com

Aclaración: El rol de SCRUM MASTER irá rotando a lo largo del proyecto.

1.4 Definiciones, acrónimos y abreviaturas

Nombre	Descripción
Usuario	Persona que usará el sistema para gestionar procesos
SIS-I	Sistema de Información Web para la Gestión de Procesos Administrativos y Académicos
ERS	Especificación de Requisitos Software
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
FTP	Protocolo de Transferencia de Archivos
Moodle	Aula Virtual
EC	Espacio Cultural

1.5 Referencias

Título del Documento	Referencia
Standard IEEE 830 - 1998	Este documento es una guía para definir los requerimientos que cumplirá el sitio web según las necesidades de nuestra organización.

1.6 Resumen

Este documento consta de tres secciones. En la primera sección se realiza una introducción al mismo y se proporciona una visión general de la especificación de recursos del sistema. En la segunda sección del documento se realiza una descripción general del sistema, con el fin de conocer las principales funciones que éste debe realizar, los datos asociados y los factores, restricciones, supuestos y dependencias que afectan al desarrollo, sin entrar en excesivos detalles. Por último, la tercera sección del documento es aquella en la que se definen detalladamente los requisitos que debe satisfacer el sistema.

2 Descripción general

2.1 Perspectiva del producto

El proyecto busca llegar a todas las personas interesadas difundir y/o en disfrutar de algún evento, ofreciendo todas las propuestas disponibles de forma libre y gratuita. Creemos que el arte es una herramienta fundamental para la inclusión social y un eslabón imprescindible de la cultura popular.

Propósito: Buscamos generar un espacio en el que tod@s l@s artistas puedan dar a conocer sus producciones o eventos para poder, de forma sencilla, conectarse con sus potenciales espectadores. Para esto desarrollamos un sistema ágil de compra y venta de entradas, que permitirá a l@s usuarios/as acceder desde cualquier lugar y en todo momento a visualizar la agenda cultural de su ciudad.

2.2 Características de los usuarios

Tipo de usuario	Administrador
Formación	Manejo de herramientas informáticas
Actividades	Control, actualización y manejo del sistema en general.

Tipo de usuario	Usuario registrado
Formación	Manejo de entornos web
Actividades	Posibilidad de publicar su espacio y evento. Seguimiento de la publicación. Posibilidad de reservar y/o comprar entradas para el evento. Recibe notificaciones. Observa e indaga información de eventos Posibilidad de guardar preferencias (ubicación, categorías, forma de pago, etc)

Tipo de usuario	Visitante
Formación	Manejo de entornos web
Actividades	Observa e indaga información de eventos y espacios culturales.

2.3 Restricciones

- Interfaz para ser usada con internet tanto desde escritorio como dispositivo móvil.

- Uso de Dominio (A determinar más adelante).
- Lenguajes y tecnologías en uso: HTML5, CSS3, BOOTSTRAP, ANGULAR, TYPESCRIPT; PYTHON, SQL, Base de datos.
- Los servidores deben ser capaces de atender consultas concurrentemente.
- El sistema se diseñará según un modelo cliente/servidor.
- El sistema tendrá un diseño e implementación sencilla con el fin de que sea amigable para el usuario.

3 Requisitos específicos

Product Backlog

- #US1: Como usuario me gustaría poder cancelar mi compra en un plazo de 48hs**
- #US2: Como enterSpace nos gustaría que la entrada a los eventos sean código QR o alfanumérico.**
- #US3: Como usuario me gustaría seleccionar mi asiento si es necesario.**
- #US4: Como usuario me gustaría acceder a mi historial de compras**
- #US5: Como usuario quiero poder ver las entradas que seleccione antes de efectuar la compra.**
- #US6: Como usuario quiero ver la suma total de lo que voy a comprar**
- #US7: Como usuario me gustaría saber si mi compra fue realizada con éxito**
- #US8: Como usuario, me gustaría poder eliminar todas las entradas que se encuentren en el carrito de compras sin tener que eliminarlas de una en una en caso de que lo requiera.**
- #US9: Como usuario quiero poder comprar una o más entradas a eventos culturales**
- #US10: Como usuario quiero poder editar los productos cargados en el carrito de compras**
- #US11: Como usuario quiero navegar por una página intuitiva para tener la mejor experiencia de usuario.**
- #US12: Como equipo enterarte nos gustaría poder recibir donaciones de los US.**
- #US13: Como usuario quiero recibir mi entrada por email.**

TK:

- #TK1: Generar un botón, luego de aceptar el pago de: "cancelar compra" con un texto aclarando que solo se puede cancelar dentro de las próximas 48hs.**
- #TK2: Generar un código único alfanumérico para cada entrada.**
- #TK3: Generar una vista interactiva para que el usuario seleccione su asiento**
- #TK4: Generar una vista del registro de compras del usuario.**
- #TK5: Listar las entradas seleccionadas en el carrito.**
- #TK6: Realizar de forma dinámica la suma de los productos y plasmarlos en el carrito.**
- #TK7: Mostrar un aviso cuando la compra se realizó con éxito.**

- #TK8: Mostrar un botón con la opción de vaciar carrito
- #TK9: Crear login con autenticación mediante Json Web Token.
- #TK10: Agregar al carrito la opción de editar los productos cargados.
- #TK11: Registrar los requerimientos en la documentación IEEE830. (subirlo dentro de una carpeta con nombre "Documentación" en GitHub en la rama main)
- #TK12: Generar un botón en la página para recibir "donaciones".
- #TK13: Generar la colección con email y enviar las entradas por este medio.
- #TK14: Como equipo "enterSPACE" queremos continuar ampliando nuestro proyecto de acuerdo a los requerimientos del cliente.
- #TK15: Revisar el mapa de sitio con el nuevo objetivo.
- #TK16: Realizar el diseño Lógico de la Normalización y el Modelo Relacional.
- #TK17: Realizar el diseño conceptual del DER: Diagrama Entidad-Relación.
- #TK18: Crear el proyecto Angular y estructura carpeta de github e Incorporar la navegabilidad de la aplicación mediante Routing con Angular.
- #TK19: Diseñar y desarrollar módulo Home/Dashboard – Angular
- #TK20: Diseñar y desarrollar Modulo Login/Registro - Angular (HTML y CSS)
- #TK21: Crear módulos y componentes para la tienda virtual/ecommerce.
- #TK22: Realizar el diseño Físico que es el script legible de la DB en MySQL.
- #TK23: Crear un Diagrama de Clases y Casos de Uso para facilitar el modelado en POO y Gestión de proyecto.
- #TK24: Crear e inyectar servicio del módulo cart
- #TK25: Crear e inyectar servicio del componente about
- #TK26: Establecer el sistema de usuarios en Django-Admin (registro, login y roles)
- #TK27: Crear un ABM de productos/servicios utilizando el panel de control de Django.
- #TK28: Crear e inyectar servicio del módulo event
- #TK29: Crear e inyectar servicio del componente create-event
- #TK30: Crear e inyectar servicio del componente home
- #TK31: Crear e inyectar servicio del módulo auth.
- #TK32: Establecer el modelo de base de datos del primer sprint en Django usando clases.
- #TK33: Crear e inyectar servicio del componente profile-user.
- #TK34: Incorporar formularios reactivos de Registro y Login para la autenticación y formularios para la manipulación (Alta) de servicios/productos/suscripción, según el caso con sus respectivas validaciones.
- #TK35: Diseñar los formularios, asegurándose de que cumplan con los requisitos mencionados en detalles. Se puede utilizar Bootstrap para los estilos y colores semánticos.
- #TK36: Trabajar en la conexión con el backend para el registro, inicio de sesión y manipulación de datos.
- #TK37: Definir la API REST y los endpoints necesarios para el registro y el inicio de sesión de usuarios. Utilizar un marco de desarrollo como Django para acelerar el proceso.
- #TK38: Registro / Alta de al menos 3 productos/servicios/suscripción (solo desde el rol Administrador, mediante su panel de control).
- #TK39: Desarrollar el tablero de control. Los datos serán consumidos desde la Base de datos MySQL.
- #TK40: Crear Tablero de control para usuarios.
- #TK41: Simular Pago con API Mercado Pago o similar.
- #TK42: Finalizar estructura del front-end.

#TK43: Crear Tablero de control administrador CRUD básico funcional de producto y cliente.

#TK44: Demo final en video con participación de todo el equipo describiendo todas las funcionalidades.

#TK45: Publicar el sitio en un servidor.

Sprints.

N° de sprint	00
Sprint Backlog	<ul style="list-style-type: none"> Definir Scrum Master inicial y registrar cada integrante faltante dentro del repositorio a trabajar. Revisar documentación IEEE830 si está completa y retomar desde allí para esta nueva etapa seguir documentando los avances. Revisar la estructura web Semántica, responsive (RWD) <p>ANÁLISIS:</p> <ul style="list-style-type: none"> Definir requerimientos para el nuevo módulo a desarrollar e-commerce (colocarlos en el Product Backlog del Project), a su vez revisar si han cumplimentado todos los requerimientos previos, realizando mejoras del mismo. Plantear Historias de Usuarios y Tareas dependientes de las US para incorporarlas en el repositorio remoto GitHub. (Issues y Milestones). Definir tareas dentro de las Historias de Usuario (GITHUB).
Responsabilidades	<p>Francisco Germolotto</p> <p>Karen Pigni</p> <p>Tayhana Villarreal</p> <p>Julián Meneses</p> <p>Fernando Rosales</p> <p>Soledad Orrego</p> <p>Martín Sanchez</p> <p>Gerardo Antonio Castellanos Torres</p>
Calendario	Fecha Inicio = 10/04/2023 - Fecha de Fin = 21/04/2023
Inconvenientes: Conocimientos técnicos y específicos en un lapso de tiempo corto designado para realizar el proyecto.	

N° de sprint	01
Sprint Backlog	<p>Se define nuevo Scrum Master y se procede a la nueva división de tareas:</p> <p>Desde el Front:</p> <ul style="list-style-type: none"> • Convertir los archivos .html del proyecto previo, en una SPA en Angular con módulos con sus componentes correspondientes y crear módulos y componentes para la tienda virtual o ecommerce. • Incorporar la navegabilidad de la aplicación mediante Routing con Angular. <p>Diseño del Backend:</p> <p>Subir en la carpeta documentación las tablas actualizadas de la base de datos:</p> <ul style="list-style-type: none"> • Diseño conceptual (DER). • Diseño Lógico que consta de la Normalización y el Modelo Relacional. • Diseño Físico que es el script legible de la DB en MySQL. • Crear un Diagrama de Clases y Casos de Uso para facilitar el modelado en POO. <p>Por último, se decide proceder con la gestión del proyecto utilizando un flujo de trabajo de GitFlow, donde el equipo puede trabajar de manera más eficiente en la implementación de nuevas funcionalidades y corrección de errores (nuevas ramas creadas), mientras se mantiene una versión estable y funcional del software en la rama principal.</p>
Responsabilidades	<p>Francisco Germolotto</p> <p>Karen Pigni</p> <p>Tayhana Villarreal</p> <p>Julián Meneses</p> <p>Soledad Orrego</p> <p>Gerardo Antonio Castellanos Torres</p>
Calendario	Fecha Inicio = 24/04/2023 - Fecha de Fin = 05/05/2023
Inconvenientes: Conocimientos técnicos y específicos en un lapso de tiempo corto designado para realizar el proyecto.	

N° de sprint	02
Sprint Backlog	<p>Se define nuevo Scrum Master y se procede a la nueva división de tareas:</p> <p>Front:</p> <ul style="list-style-type: none"> ● Crear servicios que consuman un API Rest de prueba. Para realizar las peticiones http, utilizar el servicio de HttpClient. ● Inyectar a los componentes los correspondientes servicios a fin de que los componentes puedan acceder a los datos y mostrarlos en pantalla. Incluye enlace de datos (Data binding). ● Incorporar la programación reactiva (asincrónica) mediante observables para la comunicación entre servicios y componentes. <p>Backend:</p> <ul style="list-style-type: none"> ● Establecer el modelo de base de datos del primer sprint en Django usando clases. ● Establecer el sistema de usuarios en Django-Admin (registro, login y roles) ● Crear un ABM de productos/servicios utilizando el panel de control de Django
Responsabilidades	<p>Francisco Germolotto</p> <p>Karen Pigni</p> <p>Tayhana Villarreal</p> <p>Julián Meneses</p> <p>Soledad Orrego</p> <p>Gerardo Antonio Castellanos Torres</p>
Calendario	Fecha Inicio = 08/05/2023 - Fecha de Fin = 19/05/2023
Inconvenientes: Conocimientos técnicos y específicos en un lapso de tiempo corto designado para realizar el proyecto.	

N° de sprint	03
Sprint Backlog	<p>Se define nuevo Scrum Master y se procede a la nueva división de tareas:</p> <p>Front:</p> <ul style="list-style-type: none"> Incorporar formularios reactivos de Registro y Login para la autenticación y formularios para la manipulación (Alta) de servicios/productos/suscripción, según el caso con sus respectivas validaciones. Diseñar los formularios, asegurándose de que cumplan con los requisitos mencionados. Pueden utilizar Bootstrap para los estilos y colores semánticos. <p><u>Detallar en la Issue Diseño de los formularios:</u></p> <ul style="list-style-type: none"> Deberán estar conformados por input types acorde al dato que se va a ingresar (https://developer.mozilla.org/es/docs/Web/HTML/Element/input) deberán informar al usuario mediante mensajes personalizados (Validaciones) deberán esperar los colores semánticos que propone Bootstrap, https://getbootstrap.com/docs/5.0/utilities/borders/ Poí ej. <p></p> <p></p> <ul style="list-style-type: none"> Trabajar en la conexión con el backend para el registro, inicio de sesión y manipulación de datos. <p><u>Detallar en la Issue:</u></p> <ul style="list-style-type: none"> Conexión con el backend para el registro e inicio de sesión. Conexión con backend para almacenar los datos de los productos/servicios y detalles de producto/servicios en la DB a través del modelo en Django. Opcional: Agregar rutas protegidas a la navegación en base al usuario autenticado/autorizado. Ej. para acceder al tablero de control o bien, para acceder al registro de productos/servicios. <p>Backend:</p> <ul style="list-style-type: none"> Definir la API REST y los endpoints necesarios para el registro y el inicio de sesión de usuarios. Utilizar un marco de desarrollo como Django para acelerar el proceso. Registro / Alta de al menos 3 productos/servicios/suscripción (solo desde el rol Administrador, mediante su panel de control).

Especificación de requisitos de software

	<ul style="list-style-type: none"> Desarrollar el tablero de control. Los datos serán consumidos desde la Base de datos MySQL. <p><u>Detallar en la Issue:</u></p> <ul style="list-style-type: none"> Tablero de control (Gestión) o Dashboard con módulos planteados en la maqueta del FrontEnd según proyecto: El rol ADMIN pueda ver la lista de usuarios registrados y el listado de los productos/servicios/suscripción según corresponda. El rol USUARIO registrado solo que pueda ver su Dashboard para editar su perfil. <p>OPCIONAL: Token validación del login con JWT.</p>
Responsabilidades	<p>Francisco Germolotto</p> <p>Karen Pigni</p> <p>Tayhana Villarreal</p> <p>Julián Meneses</p> <p>Soledad Orrego</p> <p>Gerardo Antonio Castellanos Torres</p>
Calendario	Fecha Inicio = 22/05/2023 - Fecha de Fin = 04/06/2023
Inconvenientes: Conocimientos técnicos y específicos en un lapso de tiempo corto designado para realizar el proyecto.	

N° de sprint	04
Sprint Backlog	<p>Requerimientos Incremento del producto: Funcionalidades completas (front + back integrado)</p> <ul style="list-style-type: none"> Tablero de control para usuarios: Mostrar catálogo (mínimo 3 productos o servicios o suscripción). Seleccionar para comprar. Simular Pago con API Mercado Pago o similar https://www.mercadopago.com.ar/developers/es/docs#online-payments Tablero de control administrador CRUD básico funcional de producto y cliente. Actualizar documentación IEEE830 - tablero Kanban (project) y Wiki. Actualizado todo el proyecto en GitHub como respaldo Demo final en video con participación de todo el equipo describiendo todas las funcionalidades. OPCIONAL: Sitio publicado en Servidor

Responsabilidades	Francisco Germolotto Karen Pigni Tayhana Villarreal Julián Meneses Soledad Orrego Gerardo Antonio Castellanos Torres
Calendario	Fecha Inicio = 05/06/2023 - Fecha de Fin = 18/06/2023
Inconvenientes: Conocimientos técnicos y específicos en un lapso de tiempo corto designado para realizar el proyecto.	