Un dibujo de un animal

Descripción generada automáticamente con confianza media

Prueba técnica desarrollador fullstack

“Prueba Practica JavaScript Interna.docx”

Miguel Angel Mendigaño Arismendy.

Java developer.

Duitama-Boyacá

20-11-2022

RESUMEN

Grupo ASD fue contratado para realizar el registro de todos los seres que tienen poderes de algún tipo del universo Marvel. Por tal motivo se debe tener el un registro de cada ser, un número de indicación única, conocer si pertenece a algún grupo de súper héroes o villanos, su lugar de operación principal (el nombre de la ciudad), su condición: si está en libertad, detenido o desconocido y el tipo de súper poder: si es Mental, Poderes Mágicos, Súper Fuerza… etc, algunos seres súper poderosos como Thanos tienen más de un tipo.

Además, se debe tener registro si tienen algún vehículo y el tipo de vehículo que utilizan.

Mas información ver: https://es.wikipedia.org/wiki/Actos\_de\_Registro\_(c%C3%B3mic)

Su misión es modelar un API RESTFul1 que permita listar los mutantes, regístralos y además los vehículos, esta debería permitir buscar por todos los mutantes.

* Buscar los mutantes por: nombre y lugar de operación.
* Crear nuevos mutantes.
* Actualizar mutantes.
* Permite cambiar el nombre y el lugar de operación.
* Listar los vehículos.

Tecnologías Debe utilizar las siguientes tecnologías:

* NodeJS.
* Angular 8 o superior o ReactJS.

Motor de base de datos

Es libre de utilizar cualquier motor de base de datos relacional, pero se recomienda MYSQL o Postgres. Si decide utilizar tecnologías NoSQL se requiere justificar la selección de motor y ventajas. Reglas • El código debe estar versionado en GIT o Subversion, siguiendo algún modelo de entrega. (se sugiere GIT, basado en un flujo de rama por característica).

No debe haber excepciones sin capturar. • Las respuestas HTTP de la API deben estar estandarizadas a^4 :

* 200 Por consultas exitosas o 400 Para reportar datos faltantes al crear o actualizar un activo.
* 404 para búsquedas sin resultados.
* 500 Para errores que pasen en la capa de backend. o Todas las respuestas deben
* contener una descripción o resumen de lo ocurrido.

1. Introducción

Se realizo el desarrollo de la aplicación ASD\_APP\_Mutant con el uso de NODE JS para el backend, soportando su estructura con el frameworks propio de express, se realiza la base de datos con My SQL.

Para el desarrollo del frontend se utilizó angular 8. Con la ayuda de estilos del framework de boostraps, también se utilizaron estilo de fontowesone, para la letra y algunos iconos.

Todo el proceso se realizó en línea trabajando en la plataforma de githug para el control de versiones con el uso de git, se manejó el concepto de Branch Flow.

1.2 Objetivos

El objetivo de este desarrollo es mostrar las habilidades que poseo en el desarrollo de la aplicación, cumpliendo los estándares de clean code, basado en principios SOLID, de acuerdo a los requerimientos de la solución abordada.

La documentación de la aplicación web basada en Node js y angular, se complementando y adjuntando en la carpeta de recursos del api, a la par que se fue desarrollando dicha aplicación, se adjuntaron los querys de construcción de la base de datos y los insert para tener información en ellas y realizar los test.

La idea es mostrar a pequeña escala todas las fases de un proyecto de desarrollo desde mi punto de vista como desarrollador.

1.2.1 Objetivos específicos

Demostrar habilidades propias el desarrollo de aplicaciones web con el uso de lenguajes y herramientas

Propias.

• Estructura de código limpio

• Manejo de control de versiones

• Entendimiento de programación orientada a objetos

• Conceptos de SOLID

• Uso de javascript

• Manejo de HTTP Verbs

1.3 Modelo de base de datos

Diagrama

Descripción generada automáticamente

Se creo un total de siete tablas con el uso de prefijos comunes adecuados al tipo de datos a manejar, esto en grandes volúmenes de datos facilita las operaciones entre ellos y con una distribución final de la siguiente manera:

T\_pais: Contiene todos los atributos para la entidad país, en el cual nació, se encuentra u opera la entidad mutante.

Paiid: identificación de cada objeto

Painom: nombre del país

Paicod: indicativo de cada país

T\_rol: Contiene todos los atributos para la entidad rol que ejerce la entidad mutante.

rolid: identificación de cada objeto

rolafinidad: define la tendencia del mutante a que bando tomar.

T\_condicion: Contiene todos los atributos para la entidad condición, donde se define la situación presidiaria del mutante.

conid: identificación de cada objeto

connom: nombre de la condición en la que se encuentra

T\_vehiculo: Contiene todos los atributos para la entidad vehículo, donde se definen los tipos de vehículos que usan los mutantes.

vehid: identificación de cada objeto.

vehnom: nombre del vehículo.

vehcod: código de identificación propio del manejo de siglas, comúnmente visto en el universo Marvel.

vehactivo:el estado en el que se encuentra

vehdescripcion: donde se especifican las características del vehículo en mención.

vehimg: se adjuntan los links con las imágenes correspondientes de la entidad.

T\_vehiculo: Contiene todos los atributos para la entidad mutante, es la tabla donde se manejan el grueso de datos y relaciones ya que contiene cinco llaves foráneas, donde se definen los atributos de la entidad.

mutid: identificación de cada objeto.

mutnom: nombre del vehículo.

mutapodo: código de identificación propio del manejo de siglas, comúnmente visto en el universo Marvel.

mutactivo: el estado en el que se encuentra, por trazabilidad de datos no se deben borrar de la base de datos, solo se cambia su estado para definir si son visibles o no.

conid: llave foránea.

rolid: llave foránea.

vehid: llave foránea.

paiid: llave foránea.

podmutid: llave foránea.

mutimg: se adjuntan los links con las imágenes correspondientes de la entidad.

T\_poder\_mutante: Contiene todos los atributos muchos a uno de la entidad poder a mutante, donde un mutante

Puede tener varios poderes.

podmutid: identificación de cada objeto

podid: llave foránea

mutid: llave foránea

Todos los DLL para la creación de las tablas e insert de la base de datos, se encuentran en :

ASD\_APP\_MUTANT\_Miguel\_Angel\_Mendiga-o\ASD\_APP\_api\MySql

1.4 Backend con Node JS

Basado en los conceptos de SOLID y la responsabilidad única, se manejó la estructura del

backend de acuerdo a entidades, donde ejecutan servicios para el llamado de datos a la base de

Interfaz de usuario gráfica, Texto

Descripción generada automáticamentedatos por medio de un controller, de esta manera se encuentra organizado el backend.

La ejecución del backend se da por el comando npm run dev, que ejecuta el archivo de carga index, alojado en la carpeta build.

Para la creación del backend fue necesario la instalación de librerías de express, promises-mysql y nodemon, estos facilita la creación del api desde typescript, para posteriormente ser convertido a javascript para ser ejecutado desde este, también simplifica la creación del pool de conexiones de comunicación con la base de datos, la cual para pruebas se realizo en un repositorio local, por intermedio del localhost en el puerto 3306, se usan claves genéricas para root: admin.

En el controller encontraremos la interface de la petición con la base de datos, se usó query native, para cargar el parte del proceso de operación a la base de datos.

En la carpeta de routes se encontrarán los servicios que interactuaran con el front y en la carpeta útil se encuentra los archivos de control para la listas y error, desde donde se implementaran los try catch, los cuales en el momento solo están enviando mensajes, pero con la ayuda de este archivo se pueden realizar múltiples configuraciones de respuesta.

Fuera de las carpetas encontraremos el archivo base de datos que contiene el código de interacción del pool de conexiones con la base de datos, el archivo index que contiene los datos de arranque y expedición de rutas de servicios y el archivo keys que contiene las variables correspondientes a nuestra base de datos.

1.5 Frontend.

Para el desarrollo del frontend se utilizo angular con ayuda de Bootstrap para los estilos, basado en un sistema de módulos con componentes independientes por entidad y función, se implemento un aplicativo web de multipaginado.

La estructura del proyecto es definida en el siguiente árbol:

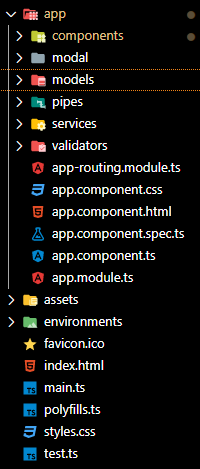


Imagen que contiene Texto

Descripción generada automáticamente

Texto, Sitio web

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente con confianza media

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente



Components: Se encuentra el bloque de archivos js, html, Css/scss y espec para cada función de cada entidad.

Modals: contiene el bloque de archivos js, html, Css/scss y espec para lo modales utilizados.

Pipes: contiene las herramientas que nos permiten modificar o transformar la información presentada en pantalla, para este caso la búsqueda.

Services: contiene los servicios que interactúan con el backend.

Validators: contiene el bloque de funciones para el manejo de datos en la interacción del cliente.

2. Control de versiones

El control de versiones se realizó desde un repositorio en github:

<https://github.com/mend1001/ASD_APP_MUTANT_Miguel_Angel_Mendiga-o.git>

Mediante el flujo de ramas, se realizo el control de versiones, para cada funcionalidad de la aplicación se realizó una rama nueva, las cuales se borraban una vez terminado el desarrollo de la misma, todos los commit se realizaron de acuerdo a las normas work flows, donde fueron específicos , detallados y en inglés, para facilitar el entendimiento a diferentes desarrolladores.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

3. Despliegue de pruebas con postman.

Se crea una colección de pruebas postman con los diferentes servicios creados, la colección completa y los resultados se guardan en la carpeta de recursos en el API.

Se realiza un breve video demostrando la operación de la prueba.

<https://youtu.be/Cp85nHMm0Xk>

2.2 Gestión de la calidad

En este apartado la empresa contratante pretende que la empresa contratada se comprometa a que cumpla una serie de requisitos de traspaso de información entre ambas facilitando así la comprensión de en lo que se está trabajando, como se está evolucionando, los posibles problemas que se están encontrando para estar mejor informado del desarrollo de la aplicación.

2.2.1 Comunicación y seguimiento del proyecto

En la sección de comunicación y seguimiento de proyecto se establecerían los mecanismos y tiempos de comunicación entre empresa contratante y empresa contratada incluyendo una serie de puntos que deben de cumplir relacionados con este tema como son:

• Los documentos descritos en las diferentes fases del proyecto como son el análisis funcional y el análisis técnico, tendrán que estar previamente validados.

• Informes de situación periódicos que incluyen:

o Planificación del proyecto actualizada desde el último informe de situación que se envió hasta el momento, para detectar desvíos sobre la planificación inicial.

o Documento describiendo las incidencias que se han dado desde el último informe de situación y como han repercutido en la planificación del proyecto.

o Documento describiendo los posibles riesgos, responsables e impacto que puedan tener en la planificación del proyecto.

o Comentarios del consultor respecto a la situación actual.

• Actas de reunión

2.3 Visión actual del proyecto

En el siguiente apartado se trata, lo que el cliente está demandando y necesita que cumpla su aplicación ya sea por legislación, posicionamiento, marketing… Lo que se trata de documentar aquí es que se conoce lo que están pidiendo y se entiende que tiene un marco regulatorio y de posicionamiento el cual se debe conocer.

2.3.1 Descripción de la necesidad a cubrir

Con el fin de evaluar uno de los criterios que tiene la empresa para escoger a la empresa que realizará los desarrollos cada empresa tiene que expresar que ha quedado claro lo que el cliente necesita y por esta razón se les explica lo que nosotros hemos entendido que ellos quieren.

Se expone a continuación una descripción de lo que el cliente demanda.

Se pretende el diseño de una aplicación web en la que los usuarios puedan ver, organizar y apuntarse a partidos de una multitud de deportes. Esto se escogerá con tres combos que hay en la parte inicial y mostraran los deportes disponibles, las zonas y los horarios disponibles para la búsqueda. Para realizar dicha búsqueda se deberá estar logado por lo que si no se está y se quiere buscar deberá saltar un pop up indicando que se logue. En dicho pop up deberá contener un enlace para el caso en el que el usuario se olvide de la contraseña y un botón para poder permanecer logado sin necesidad de tener que logarse siempre cada que vez que se entre en la aplicación. Habrá una parte anónima y otra parte en la que el usuario tendrá que logarse. Cada usuario poseerá un área personal en el que tendrá un histórico y los datos de su perfil. Cada usuario podrá crear los partidos configurándolos acorde a las búsquedas de la home. Deberá contener una mínima estructura que sea igual en todas las partes del portal. Deberá contener una cabecera o header y un pie de página o footer en los cuales figure en el caso del header el logo de la aplicación una opción para poder cambiar de idioma, un link para registrarse y un link para poder acceder con tu usuario y contraseña. En el footer tendrá que contener el copyright con la marca y los enlaces tanto a la home como a los términos y condiciones y privacidad del sitio. En la página de los partidos se deberá mostrar en una tabla. Los colores del portal deberán ser de los mismos del corporativo.

La aplicación debe estar construida con una tecnología moderna y escalable y a la par ser responsive para la visualización desde todo tipo de dispositivos y para distintos tamaños de pantalla. Debe ser segura puesto que poseerá una plataforma de pago y adaptarse a los estándares de desarrollo mundiales.

2.3.2 Descripción de contenidos por página

En las siguientes tablas se muestran los contenidos que la empresa EMPRESA\_CONTRANTANTE\_1 solicita a la empresa JMSOLUTIONSL contratada que quiere que contenga tanto el orden como la forma de los elementos.

1.) PÁGINA INICIAL

Tabla en la que la empresa solicitante declara los elementos que tiene que tener cada página, en este caso la página de resultado de búsqueda. Los elementos se nombran en el orden que deben aparecer.

PAGINA

ZONA

ELEMENTO

Características elemento

del

Home

cabecera

Logo de la aplicación

A la izquierda y del tamaño del de la cabecera.

Selector de idioma

Debe ser un

desplegable con tres

opciones de idioma,

español inglés y

portugués. Situado en

la parte derecha

Enlace para

Un enlace con icono de persona el cual llevara a la página de registro.

registrarse

Enlace para logarse

En la parte derecha del todo. Contendrá un icono de acceso y mostrará una ventana

modal para poder

poner el usuario y la

contraseña

Pie de página

Párrafo con el

En la parte izquierda y en gris

copyright de la

empresa

Enlace a la página principal

Primero de

los 3

enlaces en azul

Enlace a política de privacidad

En segundo lugar

Enlace a términos y condiciones

En tercer lugar

contenido

Combo 1 verde

Mostrará un párrafo que indica que es lo que muestra en este caso el deporte y los deportes seleccionables

Combo 2 naranja

Mostrará un párrafo que indica que es lo que muestra En este caso la zona y las zonas seleccionables

Combo 3 azul

Mostrará un párrafo que indica que es lo que muestra En este caso el horario y los horarios disponibles

Botón buscar

Botón azul para buscar

Tabla 2: Descripción de los contenidos que se deben mostrar en la página inicial a petición de

EMPRESA\_CONTRANTANTE\_1.

2.) PÁGINA DE RESULTADOS DE BUSQUEDA

Tabla en la que la empresa solicitante declara los elementos que tiene que tener cada página, en este caso la página de resultado de búsqueda. Los elementos se nombran en el orden que deben aparecer.

PÁGINA

ZONA

ELEMENTO

Características elemento

del

Home

cabecera

Logo de la aplicación

A la izquierda y del tamaño del de la cabecera.

Selector de idioma

Debe ser un

desplegable con tres

opciones de idioma,

español inglés y

portugués. Situado en

la parte derecha

Enlace con el nombre del usuario logado

Llevará

a

las

propiedades del

perfil.

Enlace para salir de la parte logada

Con icono de salir. Preguntará al usuario si desea salir con un pop up

Pie de página

Párrafo

con

el

En la parte izquierda y en gris

copyright de la

empresa

Enlace a la página principal

Primero de

los

3

enlaces en azul

Enlace a política de privacidad

En segundo lugar

Enlace a términos y condiciones

En tercer lugar

contenido

Texto

Mostrará un título que indica que son los partidos disponibles

Tabla

Mostrará una tabla en el que se muestre el

id del partido el

deporte seleccionado

el lugar el campo la

fecha y el número de

integrantes. Dicha

tabla deberá contener

un botón para

apuntarse y se deberá

negar poder apuntarse

si el número de

integrantes ya está

lleno.

Botón crear partido

Botón azul encima de la tabla

Tabla 3: Descripción de los contenidos que se deben mostrar en la página de resultados a petición de

EMPRESA\_CONTRANTANTE\_1

3.) PÁGINA DE CREAR PARTIDO

Tabla en la que la empresa solicitante declara los elementos que tiene que tener cada página, en este caso la página de crear partido. Los elementos se nombran en el orden que deben aparecer.

PÁGINA

ZONA

ELEMENTO

Características elemento

del

Home

cabecera

Logo de la aplicación

A la izquierda y del tamaño del de la cabecera.

Selector de idioma

Debe ser un

desplegable con tres

opciones de idioma,

español inglés y

portugués. Situado en

la parte derecha

Enlace con el nombre del usuario logado

Llevará a

las

propiedades del

perfil.

Enlace para salir de la parte logada

Con icono de salir. Preguntará al usuario si desea salir con un pop up

Pie de página

Párrafo con el

En la parte izquierda y en gris

copyright de la

empresa

Enlace a la página principal

Primero de los

3

enlaces en azul.

Enlace a política de privacidad

En segundo lugar.

Enlace a términos y condiciones

En tercer lugar.

contenido

Formulario para crear un partido

Deberá contener una etiqueta en el lado izquierdo y el campo en el lado derecho.

Los campos son

nombre deporte,

lugar, fecha

integrantes y

comentarios.

Etiqueta

Etiqueta que indica registro.

Tabla 4: Descripción de los contenidos que se deben mostrar en la página de crear partido a petición de

EMPRESA\_CONTRATANTE\_1

4.) PAGINA DE REGISTRARSE

Tabla en la que la empresa solicitante declara los elementos que tiene que tener cada página, en este caso la página de registrarse. Los elementos se nombran en el orden que deben aparecer.

PÁGINA

ZONA

ELEMENTO

Características elemento

del

Home

cabecera

Logo de la aplicación

A la izquierda y del tamaño del de la cabecera.

Selector de idioma

Debe ser un

desplegable con tres

opciones de idioma,

español inglés y

portugués. Situado en

la parte derecha

Enlace con el nombre del usuario logado

Llevará a

las

propiedades del

perfil.

Enlace para salir de la parte logada

Con icono de salir. Preguntará al usuario si desea salir con un pop up

Pie de página

Párrafo con el

En la parte izquierda y en gris

copyright de la

empresa

Enlace a la página principal

Primero de

los 3

enlaces en azul

Enlace a política de privacidad

En segundo lugar

Enlace a términos y condiciones

En tercer lugar

contenido

Formulario para

Deberá contener una

registrarse etiqueta en el lado

izquierdo y el campo en el lado derecho.

Los campos son,

nombre, apellidos,

fecha de nacimiento,

email, contraseña y población.

etiqueta

Etiqueta que indica registro

Tabla 5: Descripción de los contenidos que se deben mostrar en la página de registro a petición de

EMPRESA\_CONTRATANTE\_1

2.3.3 Requisitos Globales

En la parte de requisitos globales se muestran una serie de requisitos que deben ser estándar en todas las partes del aplicativo siendo funciones comunes por ejemplo temas de posicionamiento SEO, accesibilidad etc.

Se menciona una lista de requisitos considerados globales para la web:

• La aplicación debe ser multi-idioma en contenidos, en Inglés, Portugués y Español siendo este último, el idioma por defecto.

• Accesibilidad, siendo de obligado cumplimiento la normativa Doble A

• Todas las páginas deben ser imprimibles.

• Integración con sistemas tratamiento de errores, Log de aplicación, configuración centralizada, SEO (Search Engine Optimization), posicionamiento en buscadores, etc.)

• Se deberá cumplir la nueva ley de protección de datos ( Ley RGPD citada en el anexo)

2.3.4 Requisitos de rendimiento

En la era digital en la que vivimos la gente consume contenidos web desde cualquier sitio y se requiere que la carga de las páginas sea dinámica y rápida. Esto hará una experiencia de usuario favorable a la aplicación. Además la velocidad de carga de la aplicación web puede ayudar al posicionamiento SEO (Search Engine Optimization). A continuación se muestra, lo que la empresa cliente pediría en cuanto a requisitos de rendimiento.

Deberá superar test de rendimiento en cuanto tiempo y velocidad de carga a una velocidad estándar.

Dichas pruebas se realizaran en el plan de pruebas y mediante el uso de la aplicación de Google llamada Page Speed test donde especifica la velocidad y la optimización dando una serie de consejos para la mejora.

2.3.5 Requisitos de usabilidad

El dueño de la aplicación quiere que dicho aplicativo sea lo más fácil de usar posible permitiendo así llegar a un número mayor de personas y para eso declara unos requisitos de usabilidad de obligado cumplimiento como por ejemplo:

La aplicación debe ser los más intuitiva posible para facilitar el uso a los usuarios, con botones y paneles grandes y pasos fácilmente identificables. El orden de las ejecuciones y de los pasos deben seguir una lógica y debe ser la misma en todo el portal.

2.3.6 Requisitos de accesibilidad

El cliente busca con este requisito pasar una serie de normas para tener certificaciones de accesibilidad y así una vez más llegar a más personas.

En la medida de lo que sea posible, se tendrá que garantizar que existe un nivel mínimo de accesibilidad en todas las páginas, para las personas que tienen necesidades especiales. Por lo que se impone que debe contener el nivel de accesibilidad doble A.

Para cumplir dicho nivel hay que seguir una serie de criterios interpuestos WCAG 2.0 (pautas de accesibilidad del contenido en la web 2.0) tales como.

Dichas pautas se pueden resumir en:

• Todo contenido que no sea texto debe contener una alternativa textual por ejemplo a las imágenes tener una alternativa de texto

• Que una persona con ceguera pueda tener una alternativa de audio

• Introducir subtítulos en todos los videos.

• El foco de situación del usuario tiene que estar visible y claramente diferenciado del lugar en el que se encuentra.

• Indicación en errores a la hora de rellenar de datos de formularios.

• Uso de colores para distinguir más fácilmente los elementos visuales.

• Toda funcionalidad debe ser accesible mediante el teclado por ejemplo con el uso del

tabulador, flechas…

• El idioma predeterminado de cada página puede ser determinado por el software

• Contraste entre elementos de la web como por ejemplo textos, logotipos y cuadros.

• Múltiples vías para llegar a una misma página dentro de la aplicación

• Navegación coherente y en el mismo orden dentro del portal

Se busca poder incorporar el sello de doble A al portal web

La siguiente imagen muestra el sello de cumplir dichos requisitos.

Ilustración 2: Sello indicador de accesibilidad según los estándares especificados en la AA

2.3.7 Requisitos de posicionamiento

La programación de dicha aplicación debe contemplar que estará colgada en una web en la que se buscará un mejor posicionamiento para la indexación en Google por lo que deberá cubrir todos los requisitos que requiera Google para su correcta indexación.

Se deberán seguir los siguientes pasos para la correcta indexación:

• Crear títulos de páginas únicos y precisos

• Utilización de etiquetas “meta”

• Utilización de etiquetas de encabezado

• Utilización de etiquetas de datos estructurados

• Creación de urls amigables y lógicas.

• Creación de sitemaps y robots.

2.4 Criterios que sigue el cliente para escoger a la empresa que desarrollará su aplicación.

En el siguiente apartado se describen los criterios que la Empresa\_contratante\_1 ha establecido para elegir a la empresa que efectuará el desarrollo y así mismo para tener un peso en futuros desarrollos.

La siguiente tabla muestra la relación de los criterios de evaluación y su peso.

Criterio

Importancia

Peso

Interpretación por parte del equipo de la propuesta e implicación mostrada en el desarrollo de la misma

Máxima

3

Entrega de ejemplos, plantillas etc

Máxima

3

Metodologia de desarrollo para proporcionar escalabilidad y sencillez

Alta

3

Descripcion minuciosa de la oferta

Alta

2

Otros desarrollos de aplicaciones similares

Alta

2

Tamaño del equipo y diferenciación de roles

Alta

2

Linea de comunicación abierta

Alta

2

Coste total cercano al previsto y ofertado

Alta

2

Tiempo total para el desarrollo cercano al previsto y ofertado

Alta

2

Experiencia aportada con la misma tecnología

Media

1

Número de proyecto aportados como aval

Media

1

Flexibilidad en cambios anunciados

Media

1

Garantía de mantenimiento

Media

1

Resolución de problemas críticos

Media

1

Tabla 6: Criterios para selección de oferta ganadora

Se hará una valoración total de cada oferta y la que sume más puntos será la que resulte ganadora.

El sacar la mayor puntuación en esta tabla significa ganar la oferta y tener el desarrollo por lo que es fundamental que se presente unos buenos requisitos previos.

3. Valoración inicial del proyecto por parte de JMSOLUTIONSL

Este capítulo muestra como documentar la primera estimación que se realizará con el objetivo de presentar la oferta. En esta, se estima tanto costes como necesidades de tiempo y personal basándose en otras experiencias. Esto junto con la toma de requisitos debe de ser los primeros documentos que se envíen al cliente.

En la valoración inicial, se trata de indicar aproximadamente y sin muchas especificaciones cuanto se tardaría en el desarrollo y con qué coste aproximadamente se llevaría a cabo mostrando así que se tiene una idea global del proyecto, lo cual es buen síntoma para dar ver que se ha entendido. Esto no es nada vinculante pero tiene que ser una aproximación real a lo que se debe entregar después. Hay que justificar mínimamente el porqué de cada tiempo.

Se realiza una primera valoración tomando como referencia pequeños desarrollos de prueba para la medición de tiempo y esfuerzos. La siguiente descripción es una primera valoración inicial para trasladar al cliente, dando una aproximación del coste, de las fases del proyecto y de la duración del mismo. La especificación de todos los valores se realizará en el capítulo de plan de ejecución. Esta primera valoración nos permite ver a nuestra empresa si realmente somos competitivos y podemos optar a la oferta que presenta la empresa del cliente. Si hemos optado por definir un coste aproximado de unos 30.000€ si la empresa cliente diera como objetivo estar por debajo de esto, a nosotros como empresa a priori no nos compensaría pelear por este proyecto. Habría que estudiar muchos factores más, como un posible acuerdo de mantenimiento o posibles futuros desarrollos en los que nos permitiría ganar quizás lo que perdiéramos en el desarrollo.

• Duración aproximada de los trabajos: 17 semanas

• Fases contempladas: Análisis, diseño, maquetación, desarrollo, despliegue y formación.

• Recursos necesarios y coste:

o Jefe de proyecto: 10.000 €

o Equipo para su desarrollo : entre 20.000 €

La planificación temporal de los trabajos a realizar sería la siguiente:

Tabla 7: Planificación inicial por parte de la empresa JMSOLUTIONSL especificando las partes

La planificación anterior se fundamenta en los siguientes puntos:

Fase

Tareas

Experiencias anteriores

Análisis y diseño técnico

·Análisis funcional detallado.

Esta fase nos llevó unos dos 2 semanas.

·Diseño técnico del sistema.

·Validación por todas las áreas implicadas.

Desarrollo

·Construcción y maquetación de la aplicación.

En otros proyectos de igual envergadura hemos utilizado unas 10 semanas

Implantación y despliegues

·Instalación en todos los

En todas las aplicaciones

entornos desarrolladas recientemente

·Pruebas de validación.

·Test de intrusión.

·Puesta en producción. este período ha sido mayor en

la mayor parte de los casos.(2 semanas)

Tabla 8: Explicación a la tabla 6 del porqué de la duración de cada fase

4. Equipo de trabajo y construcción para la realización del proyecto

Con el fin de la realización del proyecto y de cumplir con las exigencias que muestra el cliente o bien satisfacer los acuerdos llegados con este mismo, hay que constituir el equipo de trabajo, para mandárselo al cliente ya que es un punto de exigencia por el cual valoran positivamente la división en roles y la distinción de estos mismos en distintas personas del equipo. Se trata de mostrar que se posee un equipo variado y experimentado en el que se lleva a cabo la división por tareas y que cada integrante tiene su rol. Es importante que la empresa cliente vea esta división de roles para la suma de puntos según sus criterios. Hay que saber con antelación que equipo va a formar el desarrollo aun que luego se puede hacer cambios previa información al cliente. Además no solo está en que el equipo se variado y divido en roles sino que se busca mostrar la experiencia de cada uno por lo que también se identifica muchas veces la experiencia de cada integrantes llegando a aportar CV si fuera necesario. Todo va encaminado por parte de la empresa a asegurarse que el desarrollo se hace con calidad y que todo saldrá según lo previsto.

La siguiente información habría que enviársela al cliente o bien se presentaría en una de las reuniones semanales que estipulan.

Para la realización del proyecto se necesitaran un grupo de trabajo formado por un director del proyecto, un jefe de proyecto, dos programadores, un maquetado y un analista.

El director del proyecto que junto con el jefe de proyecto se encarga de estar en contacto con el cliente para los temas económicos y posibles dudas por parte de este. El jefe de proyecto es la persona que controla el progreso de la aplicación y la que se encarga de ajustar tiempos y esfuerzos. Es el encargado de informar al director de proyecto de porcentaje evolucionado y de las posibles incidencias que vayan saliendo.

Los programadores son los encargados del completo desarrollo back-end de la aplicación. Los maquetadores serán los encargados de la parte del front-end y de ajustar los estilos y la parte visual de la aplicación y aportando diseños de imágenes botones etc.

El analista es el encargado de formar la arquitectura y detectar posibles fallos de funcionamiento. Es el encargado de comprobar todas las casuísticas de la aplicación y a la par desarrollar la documentación tanto técnica como funcional para el cliente.

Ilustración 3: Equipo de trabajo

PERFIL

DESCRIPCIÓN

Director de proyecto

Su objetivo es dar un soporte a todas las partes del proyecto. Es un perfil con una experiencia alta en proyectos de igual alcance. Asiste a las reuniones propuestas para el control y seguimiento.

Jefe de proyecto

Es el responsable de la ejecución y del cumplimiento tanto de tiempos como objetivos del proyecto. Deberá coordinar al equipo de trabajo, coordinarse con el cliente,… Será el encargado de revisar el transcurso del proyecto y las incidencias ocurridas con el fin de garantizar el cumplimiento de tiempos y objetivos.

Analista Funcional

Tendrá una gran experiencia y conocimiento en dicho sector, y estará coordinado y en comunicación con el Jefe de Proyecto en todo momento. Es el encargado de todas las pruebas y validaciones

Analista Programador

Experto técnico en desarrollo de aplicaciones con tecnología .NET Será el encargado de la realización del diseño técnico-funcional en código fuente.

Diseñador / Maquetador

Experto técnico en desarrollo de aplicaciones en el aspecto visual y renderizado.

Tabla 9: Descripción de los perfiles necesitados para el desarrollo

El equipo de JMSOLUTIONSL destinado a desarrollo de aplicaciones dispone de los recursos hardware, software y de soporte necesarios para llevar acabo los servicios para el desarrollo de dicha aplicación con la mayor eficacia posible.

A continuación se detallan las tarifas correspondientes a los perfiles contemplados:

PUESTO

PRECIO/HORA

HORAS TRABAJADAS

TOTAL

Jefe de Proyecto

35 €/h

560 horas

19600€

Desarrollador senior 1

30 €/h

560 horas

16800€

Desarrollador senior 2

30 €/h

560 horas

16800€

Analista

25 €/h

120 horas

3000€

Maquetar/diseñador

25 €/h

120 horas

3000€

TOTAL

1920

59200€

Tabla 10: Precios según puesto

Los precios ofertados no incluyen el IVA.

Forma de Pago. El último día de cada mes se emitirá una factura correspondiente a los trabajos realizados durante el mes. El vencimiento de las facturas será a 30 días desde su fecha de emisión.

5. Plan de ejecución de desarrollo de la aplicación por parte de

JMSOLUTIONSL

Junto con el equipo de trabajo, es fundamental hacer un buen diagrama de desarrollo del proyecto indicando en cada momento la duración de las tareas que se están realizando. Esto no solo es importante para el cliente sino para la empresa que desarrolla puesto que le permite tener controlados los tiempos y el orden del desarrollo.

En el siguiente plan de ejecución se muestra el orden en el que se ha desarrollado las tareas y la duración de las mismas.

En el cual se puede observar que la mayor parte se ha dedicado al desarrollo. Pero también cabe destacar que la parte de análisis y diseño previa es muy importante ocupando así casi un tercio del proyecto. Una fase de análisis y diseño efectiva ayudará a una mejor construcción y planificación. Permite resolver previamente futuras dudas de diseño y programación permitiendo anticiparse a futuras eventualidades.

Dicho plan se le entregará al cliente como prueba de que se tiene todo previsto y para poder ir haciendo un seguimiento de dicha aplicación. También es una media de justificación de tiempo por el que se declara en cada momento lo que se está haciendo y el porqué de que haya tardado un tiempo u otro.

23

6. Descripción de la solución técnica propuesta

En este capítulo se describe como tiene que ser la presentación de la documentación técnica que se le manda al cliente. En dicha documentación se debe exponer el modelo de programación que se va a realizar, detallando cada componente al mínimo para que sea validado por este. El objetivo de este apartado es poder adelantarse a posibles problemas, fallos, errores… pudiendo así ahorrar costes y tiempos.

Esta es una documentación que a priori podemos pensar que el cliente como sector no técnico que es, puede no interesarle pero prestan especial atención puesto que muchas veces poseen licencias de algunas librerías en concreto que quieren que uses, o justo lo contrario no quieren que uses algún tipo de tecnología simplemente porque no está a la moda o no es lo demandado hoy. Un ejemplo de esto, es la demanda por parte de las empresas de desarrollar todo utilizando Bootstrap o Angular que como se puso de moda, las empresas requerían estos desarrollos expresamente. Esta documentación también es requerida ya que en un futuro si quisieran mejorar la aplicación por ejemplo pues sería parte de la documentación que ellos entregarían a la siguiente empresa que desarrollara para que siguieran con el mismo orden y la misma nomenclatura del modelo.

6.1 Arquitectura

MVC es un modelo de programación que está de moda actualmente y está disponible en ASP.NET. Las aplicaciones MVC se caracterizan por una fuerte separación de la capa lógica del código de acceso a datos y la interfaz de usuario en Modelos, Controladores y Vistas.

Ilustración 4: Arquitectura MVC

6.1.1.Modelo

Nuestro sitio web presenta información y contenido sobre objetos. Todas las validaciones de todos los campos deben hacerse aquí y no en el controlador. Por ejemplo las validación de los input, ya sea para controlar el formato, la dimensión o simplemente que no este vacío debe hacerse aquí. Se adjunta en los anexos un ejemplo de validación por modelo.

El nombre de las propiedades de las clases han seguido la taxonomía de CLASEX+NOMBRE DE VARIABLE. Es fundamental seguir la misma nomenclatura siempre ya que nos permite identificar donde estamos y a que clase pertenece el elemento en cuestión.

A continuación definimos La arquitectura de nuestro modelo que ha seguido las siguientes clases con sus propiedades.

6.1.1.1. Clase Usuario

Para la clase Usuario se han establecido las siguientes propiedades que debe contener, especificando el nombre interno, el tipo de variable que se utiliza y la longitud máxima de esta última.

Campo Nombre de Variable Tipo Longitud Max

Id UserID int 8 carácteres

Nombre UserName String 20 carácteres

Apellidos UserSurname String 40 carácteres

Email UserEmail String 40 carácteres

Teléfono UserPhoneNumber String 9 caracteres

Domicilio UserAddress String 40 carácteres

Puntuación UserPoint Int 4 carácteres

Ultima Conexión UserLastSession Date

Token UserToken String 16 carácteres

Contraseña UserPassword String 8 carácteres

Fecha de nacimiento UserBirh Date

Partidos UserPartidos Partidos

Zona de juego UserZone Zona

Tabla 11: Propiedades utilizadas para la clase Usuario

6.1.1.2. Clase Partido

Para la clase Partido se han establecido las siguientes propiedades que debe contener, especificando el nombre interno, el tipo de variable que se utiliza y la longitud máxima de esta

última.

Campo Nombre de Variable Tipo Longitud Max

Id MatchID int 6 carácteres

Fecha MatchDate Date 8 carácteres

Lugar MatchPlace String 20 carácteres

Hora MatchTime Date

Integrantes MatchPlayerNumber int 40 carácteres

Administrador MatchUserCreator String 9 caracteres

Deporte MatchType String 40 carácteres

Zona MatchZone Int 4 carácteres

Ultima Conexión UserLastSession Date

Tabla 12: Propiedades utilizadas para la clase Partido

6.1.1.3. Clase Deporte

Para la clase Deporte se han establecido los siguientes campos que debe contener especificando el nombre interno, el tipo de variable que se utiliza y la longitud máxima de esta ultima

Campo Nombre de Variable Tipo Longitud Max

Id SportID int 6 carácteres

Nombre SportName String 8 carácteres

Integrantes MatchPlayerNumber int 40 carácteres

Tabla 13: Propiedades utilizadas para la clase Deporte

6.1.1.4. Clase Sesión

Para la clase Sesion se han establecido los siguientes campos que debe contener especificando el nombre interno, el tipo de variable que se utiliza y la longitud máxima de esta última

Campo Nombre de Variable Tipo Longitud Max

Id SessionID int 16carácteres

Fecha SessionDate date Tipo estándar

Tabla 14: Propiedades utilizadas para la clase Sesión

6.1.2.Vista

Las vistas contendrán código HTML para que el navegador lo pueda mostrar. Contendrán los modelos y representarán los objetos mostrando así la información. Dicho código HTML se mezcla con C# para dar archivos CSHTML. La vista solo mostrar código de interfaz de usuario dejando la lógica al modelo y al controlador. Se ha usado Razor como motor de vistas en el que nos permite separar la sintaxis de servidor de la del framewor de asp.net MVC.

Están dividas en nuestra arquitectura dependiendo de si son vistas parciales pertenecientes a un controlador o compartidas

Ilustración 5: Arquitectura de ficheros de la Vista

6.1.3.Controlador

Es el que una vez echa todas las comprobación de cara a la interfaz de usuario se encarga de hacer la lógica de nuestro programa, es decir, inserción y extracción en bases de datos, operación lógicas etc.

MVC (Model, View, Controller) será nuestro patrón de diseño, pero nuestra arquitectura se dividirá por capas.

Tenemos 3 controladores

• HomeController : El encargado del inicio

• AccountController : el que controla la sesión y datos del usuario

• MatchController : el encargado de temas relacionados con la creación e ilustración de partidos

Realmente el homeController no es el encargado del inicio por la nomenclatura que lleve como podía pasar en HTML donde el que carga por defecto es el index.html sino que se ha definido en la configuración que ese será el controlador encargado del