**PROYECTO FINAL**

**CURSO LENGUAJES FORMALES Y AUTÓMATAS**

**Objetivos**

Demostrar las habilidades prácticas para el diseño, desarrollo e implementación de una sitio web online que permita al usuario final cargar un documento para su análisis léxico respectivo.

**Requisitos funcionales**

1. **Registro de usuario**
2. El sitio permitirá el registro de usuarios de forma unificada, es decir, un solo registro de usuario permitirá realizar compras en cualquier de los sitios adheridos al proyecto.
3. Los campos por solicitar son: correo electrónico, número de teléfono, fecha de nacimiento, nickname, password y fotografía.
4. El usuario podrá seleccionar el método de notificaciones: correo electrónico, WhatsApp o ambos.
5. El password se almacenará en cifrado en la base de datos.
6. El sitio permitirá la toma de una fotografía y realización de reconocimiento facial (recorte de rostro).
7. El sitio permitirá personalizar la fotografía del usuario mediante el uso de filtros o stickers.
8. El sitio almacenará la fotografía original (para realización de login) y la fotografía modificada para mostrar en la barra de estado del sitio.
9. El sitio enviará de forma automática una credencial/identificación en formato PDF con código QR como constancia de inscripción al sitio.
10. La constancia de inscripción debe consignar la fotografía modificada (filtros). Esta sección debe ser divertida y fresca.
11. Los usuarios creados mediante esta metodología automáticamente tendrán el rol de usuario.
12. El registro y personalización del usuario deberá ser inferior a 30 segundos. El usuario podrá cambiar password y foto

****

**Ejemplo de una tarjeta de identificación con filtro personalizado (diseñar e implementar su propio modelo :-p )**

1. **Login**
2. El usuario podrá realizar login mediante: usuario y password, usuario y fotografía (coincidencia facial), lectura de QR de su identificación.
3. El usuario podrá reiniciar su contraseña en caso necesario, se deberá notificar según lo seleccionado por el usuario en el punto anterior (registro).
4. Tiempo máximo para realizar el login deberá ser inferior a 15 segundos y 30 segundos para reiniciar su contraseña.
5. **Análisis léxico**
6. El usuario logueado exitosamente accederá a la plataforma donde podrá cargar diversos archivos en formato txt.
7. El usuario seleccionará el idioma en el que texto está escrito.
8. El sitio web permitirá seleccionar uno de 3 idiomas de la siguiente lista:
   1. Español
   2. Inglés
   3. Ruso
   4. Francés
   5. Chino
   6. Árabe
9. Una vez seleccionado el archivo y el idioma, el usuario podrá iniciar el proceso de análisis haciendo clic en el botón ”Procesar”.
10. El procesamiento consistirá en un análisis léxico del documento cargado, generando como resultado los siguiente:
    1. Cantidad total de palabras
    2. Palabras que más se repiten
    3. Palabras que menos se repiten
    4. Pronombres personales
    5. Nombres de personas
    6. Sustantivos (forma raíz)
    7. Verbos (forma raíz)
    8. Cualquier otra clasificación reconocible mediante un patrón.
11. Al final del análisis el sitio preguntará al usuario si desea recibir un reporte en formato PDF en su correo o teléfono, según el medio de notificación seleccionado en el proceso de registro.
12. **Roles**

|  |  |  |
| --- | --- | --- |
| **No.** | **ROL** | **DESCRIPCIÓN** |
| 1 | ADMIN | Gestión de usuarios, roles y parametrizaciones del sistema, perfil |
| 2 | SUPERVISOR | Dashboard de ventas, entregas, perfil |
| 4 | USUARIO | Compras, tracking, histórico, perfil |

Podrá implementar los roles y permisos que considere necesarios.

**Requisitos no funcionales**

1. El sistema de registro de usuarios deberá ser centralizado en una única base de datos, se proporciona la siguiente cadena de conexión, deben organizarse para utilizar la misma estructura de usuario, los usuarios podrán realizar compras en cualquier sitio utilizando el registro de usuario inicial.

HOST: www.server.daossytem.pro

PORT: 3301

USER: usr\_usuarios\_lf\_2025\_pf

PASSWORD: 5sr\_5s51r34s\_lf\_2025\_pf

BD: bd\_usuarios\_lf\_2025\_pf

1. Desarrollo del proyecto mediante capas: BD + Backend y Frontend.
2. El sitio web (frontend) deberá ser público, accesible mediante una URL o dominio y tener el respectivo certificado SSL para navegación segura.
3. Los códigos fuentes deberán existir en un repositorio GIT con los commits realizados por cada participante del proyecto.
4. Implementación de JWT para consumo de endpoints.
5. Implementación de re-captcha en vista de checkout.
6. Implementación y uso de Docker para virtualización.

**Integrantes:**

* 1 a 6 estudiantes máximo.
* No se permiten cambios de grupo (transfuguismo).

**Configuración técnica**

* Disponer de un servidor o servicio de hosting para la publicación del sistema.
* El código fuente generado debe ser claro, conciso, escalable y mantenible.
* El código fuente debe de ser resguardado en repositorio Github de forma privada. Debe de agregar a la cuenta [jcordone1@miumg.edu.gt](mailto:jcordone1@miumg.edu.gt) al proyecto para verificar el desarrollo colaborativo
* Debe cumplir con buenas prácticas de desarrollo de software (indentado de código, nombres descriptivos, controles de errores, comentarios útiles, segregación de responsabilidades, patrones de diseño, etc…).
* El código fuente debe cumplir con buenas prácticas de seguridad y desarrollo seguro basadas en OWASP.
* El código fuente debe de cumplir con encriptación para los datos sensibles (Contraseñas, token de seguridad, información sensible, etc…)
* La interfaz gráfica debe de ser compatible con los siguientes navegadores:
  + - Google Chrome
    - Firefox
    - Microsoft Edge
* Para el desarrollo web, debe de aplicar el principio que se conoce en UI/UX como regla 60-30-10

**Explicación:**

* + 60% (color dominante): Se usa para el fondo o elementos principales que dan uniformidad al diseño.
  + 30% (color secundario): Complementa a la dominante y genera contraste visual.
  + 10% (color de acento): Resalta elementos clave (botones, llamadas a la acción, enlaces importantes).
* El desarrollo web debe de ser amigable y responsiva (Adaptarse a computadoras de escritorio, laptop, table y dispositivos móviles Android y IOS)
* Las tecnologías para utilizar en el desarrollo de software del proyecto deben de ser las siguientes:
  + Frontend
    - Nuxt.js versión 3
    - Vue.js versión 3
    - Vuetify versión 3
    - Html
    - CSS
    - Javascript
    - Typescript
* Backend
  + .NET
  + Net core 8
  + C#
* Base de datos
* MYSQL o Oracle o Microsoft SQL server
* Diagramado y modelado
* UML
* Herramientas tecnológicas
* Postman
* Bizagi
* Star UML
* Pueden utilizar lenguajes y/o tecnologías equivalentes.
* Los grupos de trabajo deben de entregar el código fuente del software con sus notas, observaciones, versiones diagramas de clases, diagramas de flujo, casos de uso expresados en UML.
* La interfaz de usuario de la aplicación debe estar expresada en idioma español.
* Los grupos deberán de entregar diagrama entidad relación
* Los grupos deberán de entregar el diagrama de arquitectura identificando cada servidor mediante IP y puertos utilizados.
* Los grupos deberán de realizar pruebas unitarias, caja blanca y caja negra a lo desarrollado y evidenciarlo.
* El desarrollo deberá de guardar bitácoras generadas por el usuario las cuales sean auditables en caso de ser requeridas (Login de usuario (Quien, donde, cuando, desde donde “IP”)).
* Debe de cifrar toda la comunicación entre microservicios mediante el uso de protocolos de seguridad TLS (https)
* Debe de validar todos los valores de entrada previos a ser procesados en cada endpoint.
* Para cada endpoint debe de sanitizar los valores de entrada previos a ser procesados en cada endpoint para prevención de algún tipo de ataque informático (SQL Inyection, cross-site Scripting (XSS)-persistente).
* Deberá de resguardar y no exponer todas las variables de entorno usadas en los servidores y configuraciones propias de los proyectos backend y frontend.
* Debe de identificar los errores del microservicio Backend y Frontend.
* Los errores deben de ser claros y precisos, no deberá de mostrar información sensible (Errores de base de datos, integral referencial, etc.)

**Requisitos de entrega**

1. Grupos de 4 personas máximo.
   1. Deberán de realizar
      1. Presentación Ejecutiva del Proyecto en Power Point. Deberá contener las funcionalidades principales del proyecto. En idioma inglés, al igual que la exposición.
         1. Metodologías implementadas
         2. Diseño de arquitectura de software
         3. Diseño diagrama entidad Relación
         4. Diagramado y modelado UML
         5. Costos de fabricación en implementación
         6. Demostración funcional de proyecto
      2. Documentos técnicos
         1. Código fuente
         2. Script SQL de base de datos
         3. Guía para compilar frontend (Desde clonar en github hasta la compilación)
         4. Guía para compilar Backend (Desde clonar en github hasta la compilación)
         5. Guía para consumo de APIS ('POST','GET','DELETE','PUT')
         6. Collections POSTMAN
         7. Diagrama Entidad Relación
         8. Archivo de análisis
         9. Diagramas de clases
         10. Diagramas de flujos
         11. Casos de usos expresados en UML
         12. Evidencia pruebas
         13. Diagrama de arquitectura
      3. Manual de usuario (Cliente)
      4. Video explicativo que abarque aspectos técnicos: conexión a base de datos, diagramas, estructura frontend, estructura backend (5 min. máximo) en idioma inglés, cada integrante deberá de aparecer en el video, adjuntar el link del video en la presentación ejecutiva.

**CALENDARIO**

|  |  |  |  |
| --- | --- | --- | --- |
| No. | FECHA | DESCRIPCIÓN | PUNTEO |
| 1 | 13/09/2025 | Registro de usuario y login. Sitio local | 5 |
| 2 | 27/09/2025 | Compras, Dashboard y entregas.  Sitio público. | 5 |
| 3 | 04/10/2025 | Simulacro y pruebas. | 5 |