**SPRINT 2 GESTIÓN DE USUARIOS.**

**HISTORIAS DE USUARIO SPRINT 2**

|  |  |  |
| --- | --- | --- |
| **Historia de usuario** | | |
| **Identificador:** HU05 | **Usuario:** Administrador | |
| **Nombre de la historia:** Visualizar usuarios en una tabla desde el panel administrativo | | |
| **Prioridad:** Alta (Alta/Media/Baja) | | **Riesgo en Desarrollo:** Bajo |
| **Puntos Estimados:** 3 h | | **Iteración Asignada:** 2 |
| **Responsabilidad de programación:** Alex Vallejo | | |
| **Modificación de la historia de usuario:** NA | | |
| **Descripción:** Como administrador, quiero visualizar todos los usuarios registrados en una tabla interactiva, para tener una vista general de la información y gestionar sus datos fácilmente. | | |
| **Criterios de aceptación:**   * La tabla debe mostrar: nombre, apellido, correo, rol y estado. * Debe ser responsive y ordenable. * Debe consultar los datos desde el backend vía API. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuario** | | |
| **Identificador:** HU06 | **Usuario:** Administrador | |
| **Nombre de la historia:** Crear un nuevo usuario desde el panel administrativo | | |
| **Prioridad:** Alta (Alta/Media/Baja) | | **Riesgo en Desarrollo:** Medio |
| **Puntos Estimados:** 4 h | | **Iteración Asignada:** 2 |
| **Responsabilidad de programación:** Alex Vallejo | | |
| **Modificación de la historia de usuario:** NA | | |
| **Descripción:** Como administrador, quiero poder registrar un nuevo usuario desde el panel, asignando su rol correspondiente, para tener control sobre el acceso al sistema. | | |
| **Criterios de aceptación:**   * El formulario debe permitir seleccionar el rol. * Validación de campos obligatorios. * Mensaje de confirmación al guardar. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuario** | | |
| **Identificador:** HU07 | **Usuario:** Administrador | |
| **Nombre de la historia:** Editar información de un usuario desde la tabla | | |
| **Prioridad:** Media (Alta/Media/Baja) | | **Riesgo en Desarrollo:** Medio |
| **Puntos Estimados:** 4 h | | **Iteración Asignada:** 2 |
| **Responsabilidad de programación:** Alex Vallejo | | |
| **Modificación de la historia de usuario:** NA | | |
| **Descripción:** Como administrador, quiero editar los datos de los usuarios existentes, para mantener actualizada su información o modificar su rol si es necesario. | | |
| **Criterios de aceptación:**   * Se debe abrir un formulario al hacer clic en icono de editar. * Todos los campos deben ser modificables, excepto el username. * Debe actualizarse en backend y reflejarse en frontend. | | |

|  |  |  |
| --- | --- | --- |
| **Historia de usuario** | | |
| **Identificador:** HU08 | **Usuario:** Administrador | |
| **Nombre de la historia:** Eliminar un usuario desde la tabla | | |
| **Prioridad:** Media (Alta/Media/Baja) | | **Riesgo en Desarrollo:** Medio |
| **Puntos Estimados:** 2 h | | **Iteración Asignada:** 2 |
| **Responsabilidad de programación:** Alex Vallejo | | |
| **Modificación de la historia de usuario:** NA | | |
| **Descripción:** Como administrador, quiero eliminar usuarios del sistema si ya no deben tener acceso, para mantener segura la plataforma. | | |
| **Criterios de aceptación:**   * El icono de Eliminar debe solicitar confirmación previa. * El usuario eliminado debe desaparecer de la tabla. * Backend debe retornar mensaje de éxito o error. | | |

**HISTORIAS TÉCNICAS SPRINT 2**

|  |  |
| --- | --- |
| **Historia Técnica** | |
| **Identificador:** HT06 | |
| **Nombre:** Crear endpoint para listar todos los usuarios | |
| **Prioridad:** Alta | **Iteración:** 2 |
| **Puntos Estimados:** 3 | |
| **Descripción:**  Diseñar e implementar un endpoint que retorne una lista paginada de todos los usuarios registrados, con los campos requeridos por el frontend para mostrar en tabla: nombre, apellido, correo, rol y estado. | |
| **Criterios de Aceptación:**   * Debe devolver resultados en formato JSON. * Debe ser accesible solo por usuarios autenticados con rol ADMIN. * Debe permitir paginación y búsqueda por nombre o correo. | |

|  |  |
| --- | --- |
| **Historia Técnica** | |
| **Identificador:** HT07 | |
| **Nombre:** Crear formulario de nuevo usuario en el frontend | |
| **Prioridad:** Alta | **Iteración:** 2 |
| **Puntos Estimados:** 4 h | |
| **Descripción:**  Desarrollar en React un formulario de creación de usuario, visible solo para el administrador, que permita ingresar los datos básicos y seleccionar el rol. | |
| **Criterios de Aceptación:**   * Validación de campos requeridos y formato de email. * Rol seleccionable entre CLIENTE, ADMIN y TRANSPORTISTA. * Envío exitoso debe actualizar la tabla de usuarios. | |

|  |  |
| --- | --- |
| **Historia Técnica** | |
| **Identificador:** HT08 | |
| **Nombre:** Implementar endpoint para crear usuarios desde el admin | |
| **Prioridad:** Alta | **Iteración:** 2 |
| **Puntos Estimados:** 3 | |
| **Descripción:**  Crear una API protegida que permita al administrador registrar nuevos usuarios con cualquier rol, validando duplicidad de correo y username | |
| **Criterios de Aceptación:**   * El endpoint debe estar protegido (solo rol ADMIN). * El rol debe ser obligatorio. * Retornar mensaje de éxito o error detallado. | |

|  |  |
| --- | --- |
| **Historia Técnica** | |
| **Identificador:** HT09 | |
| **Nombre:** Habilitar edición de usuarios en el frontend | |
| **Prioridad:** Media | **Iteración:** 2 |
| **Puntos Estimados:** 4 | |
| **Descripción:**  Agregar funcionalidad de edición dentro de la tabla de usuarios, que permita abrir un modal/formulario y actualizar los datos del usuario. | |
| **Criterios de Aceptación:**   * El campo username no debe ser editable. * Al guardar, debe actualizar la información sin recargar la página. * Mostrar notificaciones de éxito o error. | |

|  |  |
| --- | --- |
| **Historia Técnica** | |
| **Identificador:** HT10 | |
| **Nombre:** Implementar eliminación de usuarios | |
| **Prioridad:** Media | **Iteración:** 2 |
| **Puntos Estimados:** 2 | |
| **Descripción:**  Crear en frontend y backend la funcionalidad para eliminar usuarios del sistema desde la tabla de administración, con confirmación previa del administrador. | |
| **Criterios de Aceptación:**   * Al hacer clic en “Eliminar” debe aparecer un mensaje de confirmación. * El backend debe borrar el usuario y responder con estado 200. * La tabla debe actualizarse automáticamente al eliminar. | |