***Kambista qa challenge 👋!***

Para levantar el app tuve que instalar Node y luego Express, pero aún así no se levantaban correctamente los servicios, me daba un error “Cannot GET /” cuando intentaba ingresar a la ruta del servidor *localhost:4000*



Finalmente luego de mover unas cuantas cosas empezó a aparecer, pero el archivo *index.html* estaba intentando obtener el archivo *main.js* de una ruta equivocada, así que para hacerlo funcional modifiqué ese *index.html* y le borré la carpeta a la cual apuntaba (ya que están en la misma ruta) y empezó a funcionar.



No se si esa sería la solución, pero al menos pude empezar a ver las respuestas al momento de interactuar con la página de login:



IDENTIFICA LAS TAREAS Y/O CASOS DE PRUEBA QUE ESTARÍAN ENLAZADAS A LAS REQUERIMIENTOS PROPUESTOS.

A primera vista se puede notar que los requerimientos están un poco incompletos, ya que no se tienen mocks o diseños que puedan complementar de forma gráfica como se deberían ver las páginas de Login y Register.

Al mismo tiempo, no existen diagramas que soporten el flujo de las pantallas (caso de uso, transición de estados, flujo, etc).

Al no haber documentación funcional, ni tampoco documentación técnica o de la arquitectura, lo primero que se realizará será una prueba de humo para validar que existe al menos comunicación entre el front y el back, la cual se hizo en el proceso dei instalación. Luego de eso se procederá a realizar pruebas exploratorias, en la cual se identificarán los flujos principales y se escribirán los primeros casos de prueba, básados en los requerimientos definidos.

Luego de hacer las pruebas de humo se pudo observar que la funcionalidad de login no ha sido implementada revisando el código se encontró que existe un usuario registrado y debería mostrar el mensaje ‘LOGIN’, sin embargo no lo hace.

Cada vez que un usuario intenta registrarse se muestran los mismos mensajes, no se almacenan los usuarios en ninguna base de datos.

Hasta este punto se debería rechazar las dos funcionalidades, ya que no hay forma de validar si un usuario se registró o se logueo en el aplicativo. Pero ya que el front al menos está validando los campos, se avanzará con esa validación básada en los requerimientos escritos por el Scrum Master.

Los casos de prueba se expresarán en una tabla de decisiones:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Condiciones** | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Correo Vacío | X | X | - | - | - | - | - |
| Password Vacío | X | - | X | - | - | - | - |
| Usuario contiene @ | - | - | X |  |  | X | X |
| Password < 5 chars | - | - | - | X | - | X | - |
| Password >= 5 chars | - | X | - | - | X | - | X |
| **Resultado Obtenido** |  |  |  |  |  |  |  |
| *REQUIRED* | X | X | X |  |  |  |  |
| *INVALID* |  |  |  | X | X | X |  |
| *SAVED* |  |  |  |  |  |  | X |

Al realizar las pruebas se obtuvieron los siguientes resultados:

Los casos 1 al 6 fueron correctos, el mensaje de error se mostró satisfactoriamente:





El caso 7 resultó incorrecto, debido a que el app está requiriendo que el password tenga al menos 7 caracteres y la especificación dice 5:



Los casos de prueba y el resultado de la ejecución deben cargarse en una herramienta de administración de casos de prueba como TestRail, TestLink o la herramienta que tengan a la mano, independientemente de eso, los requerimientos o user stories deben ser reabiertos, explicando el motivo, en este caso, todo lo que se ha explicado hasta el momento.

IMPLEMENTA LOS TIPOS DE PRUEBAS QUE CONSIDERES NECESARIOS ( UNITARIAS, SERVICIOS, FUNCIONALES , MANUALES, ETC), DE TAL MANERA QUE SE PUEDA EVALUAR LOS REQUERIMIENTOS SOLICITADOS.

Los tipos de prueba que se recomiendan para estos requrimientos son los siguientes:

* Unitarias: Deben ser realizadas por el equipo de desarrollo antes de commitear los cambios al branch principal y antes de pasarlos al ambiente de pruebas.
* API o Servicios: Se pueden realizar de forma manual utilizando algun aplicativo como POSTMAN o incluso automatizando este proceso con la misma herramienta.
* Manuales Funcionales: Directamente en el Web App una vez esté en el ambiente de pruebas, se validarán todos los criterios de aceptación y a su vez el UI / UX, animaciones, luego se debe llegar a un acuerdo para el alcance con relación a: responsiveness, cross browser testing, mobile browser testing, ya que no se puede probar en todo, debe haber un criterio de finalización.
* Automatización: Ya que es una funcionalidad muy repetitiva, se puede automatizar con un data driven framework.
* Performance: Se debe tener en consideración la cantidad de usuarios concurrentes y tiempos de respuesta aceptados, de tal forma que se prioricen pruebas de carga, volumen, stress.
* Usabilidad: Permitirá validar que tan sencillo es utilizar y navegar a través de la funcionalidad.
* Seguridad: Ya que se trata con data personal de los usuarios y mas aún cuando hay dinero de por medio.
* Accesibilidad: \*Solo si es aprobado.

COMENTAR SOBRE EL FLUJO Y LOS AMBIENTES DE TRABAJO Y SUGIERE MEJORAS.

El fujo de trabajo me parece correecto, se espera que las pruebas que las pruebas unitarias y de integración se realicen en los ambientes de desarrollo y quizás alguna otra prueba de validación rápida que se pueda solicitar por parte del equipo de desarrollo, luego que eso, las pruebas de sistema, pruebas funcionales deben hacerse en QA y si no se tiene un ambiente especial para las pruebas de aceptación, estás también deberían hacerse en el ambiente de QA.

En cada uno de los ambientes o procesos debería incluirse un set de pruebas automatizadas que incluyan smoke y regression, de tal forma que se detecte de forma temprana si algún cambio produjo algúna falla, por ejemplo, luego de que el desarrollador commitee y se mande a compilar una versión, el mismo job de CI debería disparar un set de pruebas.

Ahora, para las pruebas de performance, se debería considerar un ambiente o un servidor independiente, de tal forma que no afecten los resultados.

Finalmente para pasar a producción se debe tener un proceso claro y estandarizado que incluya la documentación y las aprobaciones necesarias, artefactos, configuraciones, luego de pasar a prod deberían agregar que se tienen que incluir las pruebas de sanidad y validación en ese ambiente, al parecer no están incluídas en el diagrama.

SUGERIR EN QUÉ AMBIENTE Y EN QUÉ MOMENTO DEL FLUJO DE TRABAJO SE DEBEN EJECUTAR CADA TIPO DE PRUEBAS.

Está incluido en el punto anterior