Acceptance tests   
“Acme Hacker Rank v1.0”

|  |  |
| --- | --- |
| Development team | |
| **Identifier** | *G92* |
| **Members** | *Batista Núñez, Francisco; De la Flor Bonilla, Álvaro; Ferrete Molle, José Antonio; Fernández Benjumea, Carmen; Sánchez Hipona, Antonio; Salvat Pérez, Antonio* |
| **Testing team** | |
| **Identifier** |  |
| **Members** |  |
| **Indexing data** | |
| **Test designers** | *Fernández Benjumea, Carmen* |
| **Testers** |  |
| **Notes** | *Casos de uso de R7* |

**Table de contenidos**

[Caso de uso 7.1: Registrarse en el sistema como “company” or a “rookie” 3](#_Toc8040832)

[Caso de uso 7.2: Listado de “positions” y navegar por sus correspondientes “companies” 8](#_Toc8040833)

[Caso de uso 7.3: Listado de “companies” y navegar por sus correspondiente “position”. 9](#_Toc8040834)

[Caso de uso 7.4: Buscador de “positions”. 12](#_Toc8040835)

# Caso de uso 7.1: Registrarse en el sistema como “company” or a “rookie”

Descripción

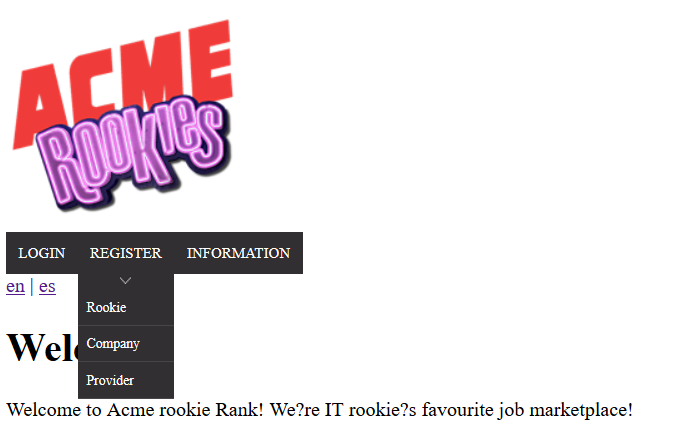
Cualquier actor que entre en la aplicación puede registrarse como “company” o “rookie”.

Precondiciones

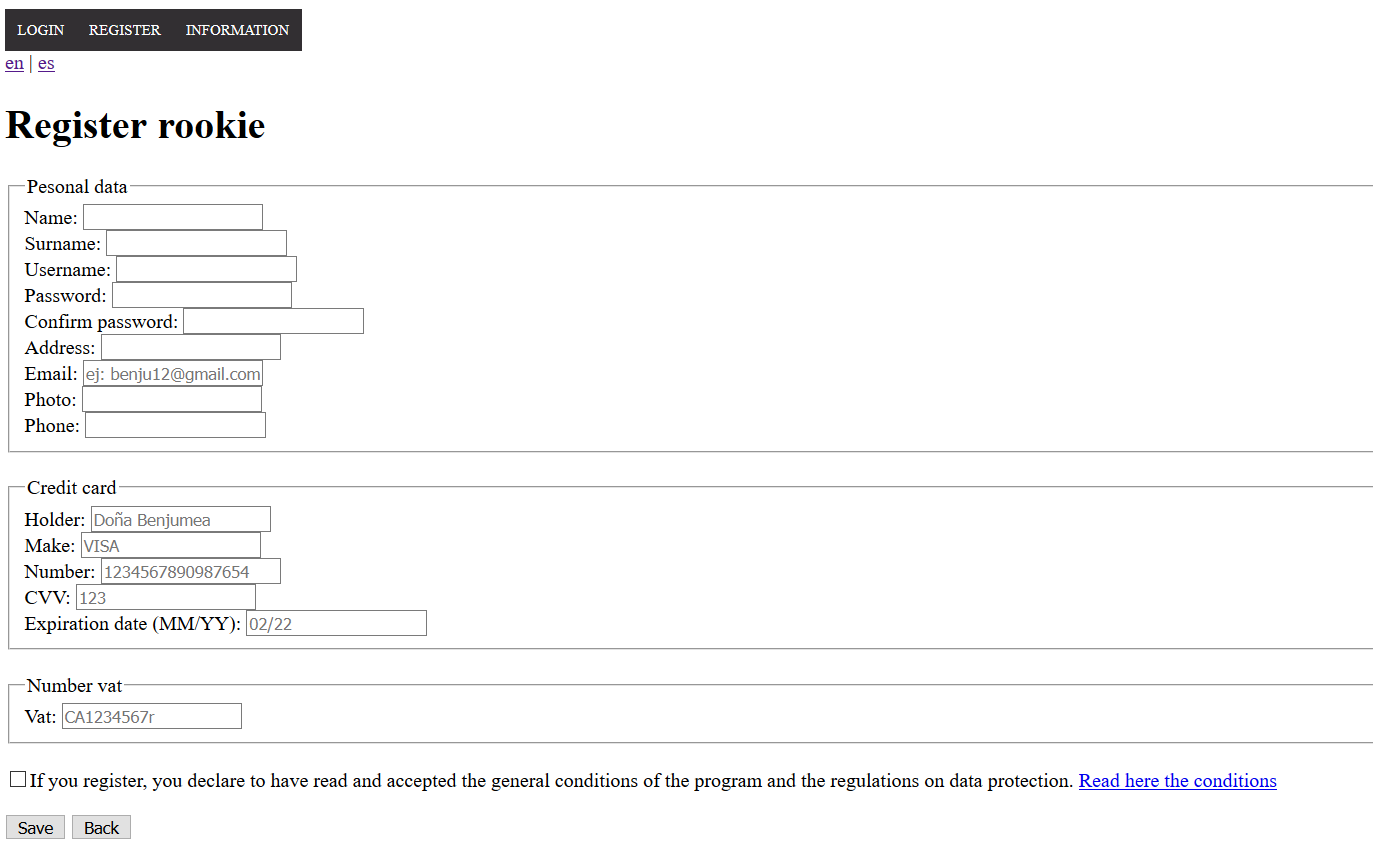
Hay que ejecutar PopulateDatabase.java

Acceso

Homepage > Register > Company // Homepage > Register > Rookie



A continuación, se mostrará el formulario de registro como rookie (todas las pruebas se realizarán a partir de dicho actor)



Tests

|  |  |
| --- | --- |
| Test 1 | |
| **Description** | *Debemos poder ver todos los datos que debe rellenar un actor para poder registrarse en la aplicación.* |
| **Expected** | *En el formulario deben aparecer los siguientes campos: nombre, apellidos, nombre de usuarios, contraseña, confirmar contraseña, dirección, email, foto, teléfono, propietario, marca, número, fecha, vat y términos y condiciones. Para el registro de company también debe aparecer el nombre comercial.* |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 2 | |
| **Description** | *Si mandamos el formulario en blanco deben aparecer los siguientes errores.* |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 3 | |
| **Description** | *Si rellenamos el formulario con un nombre de usuario que ya exista o con una contraseña o confirmar contraseña que no sean iguales deben aparecer los siguientes errores.* |
| **Expected** | *Usuario: rookie Contraseña: rookie1234 Confirmar contraseña: rookie* |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 4 | |
| **Description** | *Registramos a un actor como “rookie” con los siguientes datos y después pulsamos el botón de guardar:*  *Contraseña y confirmar contraseña es carfer.* |
| **Expected** | *Debe registrar al actor correctamente en el sistema.* |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 5 | |
| **Description** | *Después de registrar al actor comprobaremos que dicho actor puede entrar sin problemas a la aplicación.*  Homepage > Login |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

# Caso de uso 7.2: Listado de “positions” y navegar por sus correspondientes “companies”

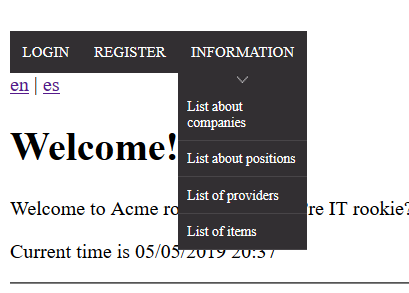
Descripción

Cualquier actor que entre en la aplicación puede ver un listado con todas las “positions” y navegar por sus correspondientes “companies”.

Precondiciones

Hay que ejecutar PopulateDatabase.java

Acceso

Homepage > Information > List about positions

Tests

|  |  |
| --- | --- |
| Test 1 | |
| **Description** | Se mostrará un listado con todas las “positions” que se encuentren en mondo final y no estén canceladas junto a su “company”.  Haciendo clic en el ticker una “position” se mostrará dicha “position”.  Haciendo clic en el show de una “position” se mostrará su “company”. |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

# Caso de uso 7.3: Listado de “companies” y navegar por sus correspondiente “position”.

Descripción

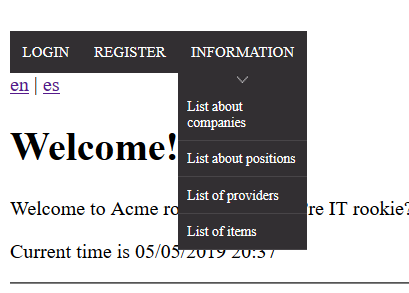
Cualquier actor que entre en la aplicación puede ver un listado con todas las “companies” y navegar por sus correspondientes “positions”.

Precondiciones

Hay que ejecutar PopulateDatabase.java

Access

Homepage > Information > List about companies



Tests

|  |  |
| --- | --- |
| Test 1 | |
| **Description** | Se mostrará un listado con todas las “companies” junto a su “position”. |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 2 | |
| **Description** | Haciendo clic en el nombre comercial de una “company” se mostrará dicha “company”.  Haciendo clic en el show de una “company” se mostrará sus “positions”. |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

# Caso de uso 7.4: Buscador de “positions”.

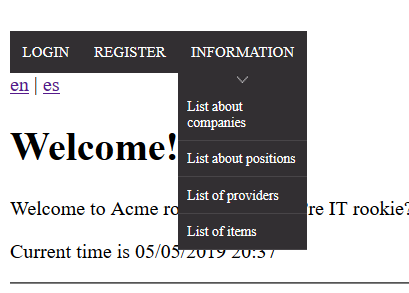
Description

Cualquier actor que entre en la aplicación puede ver un listado con todas las “positions” y navegar por sus correspondientes “companies”. Además de poder buscar un “positions”.

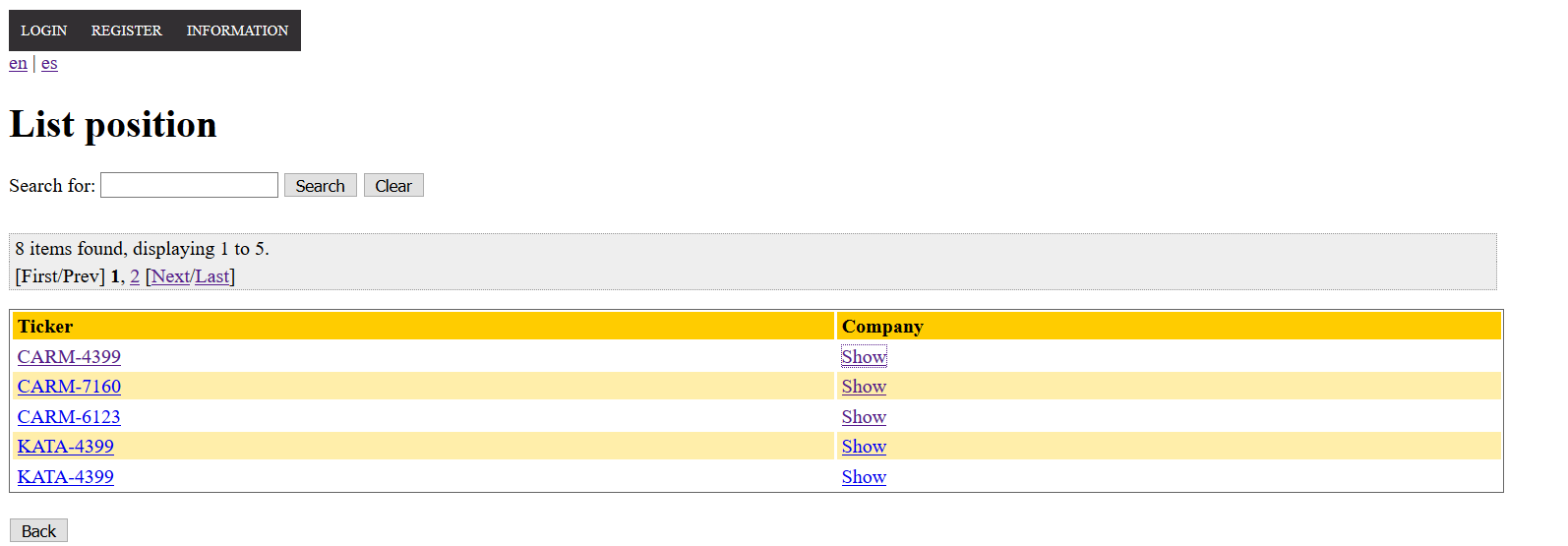
Preconditions

Hay que ejecutar PopulateDatabase.java

Access

Homepage > Information > List about positions

En la parte superior del listado encontramos un pequeño buscador, el cual busca por el título, la descripción, perfil, habilidad, tecnología o por nombre de una “company”.



Tests

|  |  |
| --- | --- |
| Test 1 | |
| **Description** | Si introducimos en el buscador car, mostrará las “positions” que contenga dicha palabra en algunos de los parámetros descritos anteriormente. |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |

|  |  |
| --- | --- |
| Test 2 | |
| **Description** | Si introducimos una palabra que no la posea ninguna “position”, no mostrará nada. |
| **Expected** |  |
| **Outcome** |  |
| **Notes** |  |