

DIVISIÓN DE ESTUDIOS PROFESIONALES PARA EJECUTIVOS

INGENIERÍA DE SISTEMAS

***Proyecto Bastanteo de Poderes - PeruBank***

Integrantes:

Alcantara, Jorge

Bruno, Harold

Espiritu, Hans

Villayzan Chancafe, Gabriel

GitHub:

https://github.com/hariki01/PeruBank

PROGRAMACIÓN ORIENTADA A OBJETOS

Profesor:

Jose Amadeo Martin Diaz Diaz

Lima, 28 de Noviembre de 2011

INDICE

[**1.** **INTRODUCCIÓN** 3](#_Toc310876790)

[**2.** **CASO PROPUESTO** 3](#_Toc310876791)

[**3.** **HISTORIAS DE USUARIO** 4](#_Toc310876792)

[**4.** **ESPECIFICACIONES TÉCNICAS** 6](#_Toc310876793)

[**5.** **GESTIÓN DEL CÓDIGO** 6](#_Toc310876794)

1. **INTRODUCCIÓN**

Con esta documentación se pretende explicar y detallar las especificaciones del trabajo desarrollado para atender los requerimientos del proyecto *Bastanteo de Poderes*, así como para respaldar la metodología basada en Test que se empleó a los largo del proceso.

Primero, que con esta metodología se desarrolla primero el test, para luego, en base a este, desarrollar el código necesario para que se cumpla el requerimiento del test. Esto facilita el desarrollo de un proyecto grupal como este, en el que se dividen los requerimientos y se van validando granularmente.

En ese sentido, analizamos las historias de usuario detalladas en los requerimientos del proyecto y luego pasamos a definir los test necesarios para que estas especificaciones se cumplan a cabalidad, fragmentándolas según se considere necesario y definiendo las entidades conforme lo necesite el test para arrojar la respuesta esperada.

Luego de haber conceptualizado los requerimientos en código, en forma de test, viene la etapa de implementación del código para hacer funcionar los test. Y finalmente, cerrando el ciclo de desarrollo de este proyecto, se Refactoriza el código, simplificándolo, reduciéndolo, organizándolo, pero manteniendo las funcionalidades logradas.

1. **CASO PROPUESTO**

El caso propuesto es el desarrollo de un aplicativo que permita al cliente realizar el proceso de bastanteo de poderes, donde los usuarios puedan:

* Registrar clientes
* Inscribir representantes
* Agregar poderes
* Bastantear poderes
* Verificar poderes
* Consultar poderes
* Controlar usuarios

1. **HISTORIAS DE USUARIO**

Se describen los criterios bajos los cuales se puede dar por cumplido un requerimiento, para cada uno de estos requerimientos se establecen criterios que nuestros Test deben cumplir para dar por concluida una funcionalidad.

Estas se listan a continuación:

1. Registrar clientes.- Como supervisor deseo registrar clientes de modo que los abogados puedan bastantear sus poderes bancarios.

|  |  |
| --- | --- |
| Cuando:  Registro un cliente debo indicar su RUC, su código de cliente interno (8 dígitos), razón social, fecha de inicio de operaciones, tipo de empresa (SAA, SAC, SRL, etc.) y cantidad de empleados. | Espero:  - Confirmación del registro OK.  - Mensaje de error si no se ingresan los datos requeridos.  - Mensaje de error si el cliente ya existe (RUC o código de cliente repetido). |

2. Inscribir representantes.- Como abogado deseo inscribir los representantes de cada cliente de modo que luego puedan tener los poderes bastanteados.

|  |  |
| --- | --- |
| Cuando:  Inscribo un representante debo indicar su tipo y número de documento, sus nombres y apellidos, cargo en la empresa y un grupo de bastanteo (A, B, C, etc.). | Espero:  - Confirmación de la inscripción Ok.  - Mensaje de error si no se ingresan los datos requeridos.  - Mensaje de error si ya es representante de la empresa (según tipo y número de documento). |

3. Agregar poderes.- Como supervisor deseo agregar el listado de poderes para que los abogados puedan asignarlos a los representantes según bastanteo.

|  |  |
| --- | --- |
| Cuando:  Agrego un poder debo indicar su código corto de consulta (ej.: CHCO, EFRE), el nombre (ej.: Cobro de Cheques, Retiro en Efectivo) y el tipo producto/servicio (ej: Activo, Pasivo). | Espero:  - Confirmación de la agregación Ok.  - Mensaje de error si no se ingresan los datos requeridos.  - Mensaje de error si ya se agregó el poder (código corto). |

4. Bastantear poderes.- Como abogado deseo poder anotar el resultado del bastanteo de modo que el personal de agencia pueda consultarlo antes de cada operación.

|  |  |
| --- | --- |
| Cuando:  Bastanteo debo indicar qué poderes puede realizar cada grupo de bastanteo de cada cliente. Además, para esta asignación debo indicar el tipo de intervención (ej.: A Sola Firma, En Conjunto Con) y con qué otro(s) grupo(s) de bastanteo si corresponde (combinación), así como el importe y la fecha de vencimiento (normalmente 6 meses). | Espero:  - Mensaje de confirmación Ok.  - Mensaje de error si ya se anotó el mismo poder para el mismo grupo de bastanteo, en el mismo cliente, considerando el mismo tipo de intervención y combinación. |

5. Verificar poderes.- Como personal de agencia deseo poder consultar si un representante tiene poder para una operación o no de manera que pueda realizarla.

|  |  |
| --- | --- |
| Cuando:  Consulto debo indicar el RUC del cliente, la identificación del representante, el código de poder (o elegirlo de una lista), y el importe. | Espero:  - Mensaje indicando que el representante tiene poder y cómo.  - Mensaje indicando que el poder está dado pero ya venció.  - Mensaje indicando que el representante no tiene poder para realizar la operación indicada o el importe es superior. |

6. Consultar poderes.- Debo poder consultar los bastanteos registrados.

|  |  |
| --- | --- |
| Cuando:  Consulto debo indicar el RUC del cliente, la identificación del representante, el grupo de bastanteo o tipo de intervención. | Espero:  - Lista de registros de bastanteo que se tienen según los criterios de búsqueda ingresados.  - Mensaje indicando que no hay resultados según criterios. |

7. Controlar usuarios.- Como administrador deseo poder controlar a los usuarios de modo que al ingresar puedan acceder a sus funcionalidades ya sea como supervisor, como abogado o empleado.

|  |  |
| --- | --- |
| Cuando:  Doy de alta un usuario debo indicar qué rol tiene asignado. Cada usuario debe identificarse por una clave al ingresar. | Espero:  - Mensaje de bienvenida indicando el rol que tiene así como las opciones que le corresponde.  - Mensaje de error si no se autentica correctamente. |

1. **ESPECIFICACIONES TÉCNICAS**

Lenguaje y Versión: El aplicativo fue desarrollado en JAVA v1.6.

Tests: Se utiliza la librería JUnit 4 para los Test.

IDE: SpringSource Tool Suite 2.8.0

1. **GESTIÓN DEL CÓDIGO**

El desarrollo del aplicativo se realizó con el soporte de la herramienta colaborativa *GitHUB*, que permite trabajar en equipo de una manera distribuida.

El repositorio principal <https://github.com/hariki01/PeruBank> es administrado por uno de los integrantes del grupo, mientras que el resto hace *fork* al proyecto, creando clones que permiten alterar el código y subirlo a su repositorio clonado, para que luego el administrador del proyecto junte el código en 1 solo.

El historial de cambios se puede mostrar de la siguiente manera, gráficamente:

