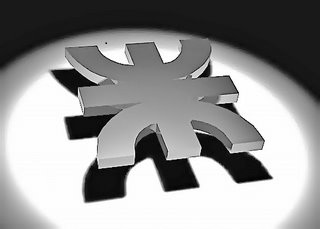
**Universidad Tecnológica Nacional**



**Facultad Regional**

**Córdoba**

**Cátedra: Proyecto Final**

**Producto: “AlquilaCosas.com.ar”**

Carpeta Número 3

|  |
| --- |
| **Grupo N° 1:**  **Atala, Jorge Leg. 47746**  **Cardozo, Damián Leg. 51762**  **Giagante, Ignacio Leg. 48679**  **Pintos, Javier Leg. 47401**  **Polo, Araceli Leg. 43740** |

# Introducción

El propósito de la siguiente carpeta es presentar el Modelo de Requerimiento y Modelo de Diseño, desarrollado para el producto “Alquilar Cosas.com.ar”.

El objetivo del Modelo de Requerimiento, es guiar el desarrollo hacia el sistema correcto. Esto se consigue mediante una buena descripción de los requisitos del sistema, es decir, las condiciones o capacidades que el sistema debe cumplir. Esta descripción debe ser clara y precisa para permitir llegar a un acuerdo entre el cliente y los desarrolladores sobre que debe y no debe hacer el sistema.

Para lograr esto es importante que el cliente sea capaz de leer y comprender los requerimientos, por tal motivo se debe utilizar el lenguaje del cliente para describir los mismos.

El propósito de llevar a cabo el diseño son: adquirir una comprensión en profundidad de los aspectos relacionados con los requerimientos funcionales y restricciones relacionadas con los lenguajes de programación, crear una puto de partida apropiado para las actividades de implementación.

# Modelo de Requerimiento

2.1 Definición del Sistema de Información

**Objetivo:**

Desarrollar un producto que permita brindar información, tanto a usuarios particulares como empresas, sobre productos de diferentes rubros a alquilar; y que permita ponerse en contacto con el dueño del mismo para alquilarlo.

**Límites:**

Desde que un usuario se registra hasta que se concreta un alquiler y las partes se califican mutuamente.

*Nota: se entiende por calificar al proceso en el cual las partes involucradas dejan sus comentarios de la experiencia y se dan un puntaje (ej. Podría ser numérico del 1 al 10) de acuerdo si la misma fue positiva o negativa.*

**Alcances:**

|  |  |  |
| --- | --- | --- |
| ID | NOMBRE DEL CASO DE USO | OBJETIVO |
| 01 | Registrar usuario |  |
| 02 | Modificar usuario |  |
| 03 | Registrar categoría |  |
| 04 | Modificar categoría |  |
| 05 | Eliminar categoría |  |
| 06 | Consultar publicaciones |  |
| 07 | Registrar publicación de un producto |  |
| 08 | Modificar publicación de un producto |  |
| 09 | Suspender publicación de un producto |  |
| 10 | Registrar destacación de una publicación |  |
| 11 | Eliminar destacación de una publicación |  |
| 12 | Registrar pedido de alquiler de un producto |  |
| 13 | Cancelar pedido de alquiler de un producto |  |
| 14 | Registrar confirmación de pedido de alquiler |  |
| 15 | Modificar estado de alquiler de un/os producto/s |  |
| 16 | Modificar alquiler de un/os producto/s |  |
| 17 | Registrar calificación de un usuario hacia otro |  |
| 18 | Registrar pregunta de un usuario hacia otro |  |
| 19 | Registrar respuesta de pregunta hacia otro usuario |  |
| 20 | Registrar denuncia sobre publicación inadecuada |  |
| 21 | Registrar denuncia de comentario inadecuada |  |
| 22 | Verificación de denuncias |  |
| 23 | Registrar la suspensión de un usuario |  |
| 24 | Registrar réplica de un usuario a otro |  |
| 25 | Registrar publicitante |  |
| 26 | Registrar anuncio publicitario |  |
| 27 | Modificar estado anuncio publicitario |  |
| 28 | Administrar anuncios publicitarios |  |
| 29 | Emitir informe sobre cantidad de usuarios inscriptos en un determinado tiempo |  |
| 30 | Emitir informe sobre productos publicados en un determinado tiempo |  |
| 31 | Emitir informe sobre alquileres realizados en un determinado tiempo |  |

2.2 Modelo de Casos de Uso del Sistema de Información

**Definición de Iteraciones:**

*Primera iteración:*

Color de Referencia:

La primera iteración permitirá ejecutar el siguiente circuito lógico:

* Permitir al cliente registrarse como usuario del sistema
* Darle posibilidad al cliente de que consulte las publicaciones activas que ese encuentren en el sistema.
* Permitir a un usuario del sistema publicar un producto.
* Permitir a un usuario del sistema realizar preguntas sobre una publicación en particular.

Lista de casos de usos intervinientes agrupados por paquetes:

**Paquete Usuarios**

01-0Registrar usuario

02- Modificar usuario

**Paquete Denuncias y Comentarios**

18- Registrar pregunta de un usuario hacia otro

19- Registrar respuesta de pregunta hacia otro usuario

**Paquete Publicidad**

06- Consultar publicaciones

07- Registrar publicación de un producto

08- Modificar publicación de un producto

**Paquete Administración**

03-Registrar categoría

04- Modificar categoría

05- Eliminar categoría

*Segunda iteración:*

Color de Referencia:

La segunda iteración permitirá ejecutar el siguiente circuito lógico:

* Se adiciona la posibilidad de al publicar un producto el usuario pueda destacar su publicación.
* Permitir a un usuario concretar un alquiler.
* Permitir a los usuarios que participan de un alquiler, al finalizar el mismo calificarse mutuamente.
* Permitir a los usuarios participantes de un alquiler, ingresar réplicas a las calificaciones publicadas.

Lista de casos de usos intervinientes agrupados por paquetes:

**Paquete Publicación:**

10- Registrar destacación de una publicación.

**Paquete Alquiler:**

12- Registrar pedido de alquiler de un producto

14- Registrar confirmación de pedido de alquiler

15- Modificar estado de alquiler de un/os producto/s

16-Modificar alquiler de un/os producto/s

24- Registrar réplica de un usuario a otro

**Paquete Denuncias y Comentarios:**

17- Registrar calificación de un usuario hacia otro.

*Tercera iteración:*

Color de Referencia:

La tercera iteración permitirá ejecutar el siguiente circuito lógico:

* Permitir al usuario realizar denuncias de publicaciones y cometarios que cumplan con las normas establecidas.
* Permitir al usuario cancelar un pedido de alquiler que haya solicitado.
* Permitir al administrador eliminar la destacación de unapublicidad.
* Permitir al administrador revisar las denuncias cargadas por los usuario, de publicaciones y/o comentarios, de ser correcta la denuncia dar de baja la publicación y/o comentario.
* Permitir al administrador enviar una advertencia al usuario, si éste cumple con tener la cantidad máxima permitida de advertencia, proceder a suspender al usuario.

Lista de casos de usos intervinientes agrupados por paquetes:

**Paquete Publicación:**

09- Suspender publicación de un producto

11- Eliminar destacación de una publicación

**Paquete Alquiler:**

13- Cancelar pedido de alquiler de un producto

**Paquete Denuncias y Comentarios:**

20-Registrar denuncia sobre publicación inadecuada

21-Registrar denuncia de comentario inadecuada

**Paquete Administración:**

22-Verificación de denuncias

**Paquete Usuarios**

23- Registrar la suspensión de un usuario

*Cuarta iteración:*

Color de Referencia:

La cuarta iteración permitirá ejecutar el siguiente circuito lógico:

* Permitir al administrador generar diferentes reportes con información útil para la toma de decisiones.
* Permitir a un publicitante registrarse.
* Permitir al publicitante solicitar que se publique su anuncio.

Lista de casos de usos intervinientes agrupados por paquetes:

**Paquete Publicidad:**

25- Registrar publicitante

26- Registrar anuncio publicitario

27- Modificar estado anuncio publicitario

28- Administrar anuncios publicitarios

**Paquete Reportes:**

29-Emitir informe sobre cantidad de usuarios inscriptos en un determinado tiempo

30-Emitir informe sobre productos publicados en un determinado tiempo

31- Emitir informe sobre alquileres realizados en un determinado tiempo

**Diagrama de casos de uso del sistema de información por paquete:**

****

****

****

****

****

****

****

**Modelo de Objeto del Dominio:**

Diagrama de Clases

Descripción de clases:

# Modelo de Diseño

* 1. Documento de arquitectura
     1. Estandares de codificación

Herramientas

* Analisis, diseno del sistema a traves de la herramienta Enterprise Architect 7.0
* El lenguaje basico de desarrollo es Java, version 6 update 26, aplicable al desarrollo en el servidor web.
* Apache Tomcat sera la herramienta para dar soporte a la aplicacion en el servidor
* Para el desarrollo de la pagina cliente se utilizara html 5.0 y Javascript con soporte AJAX.
* Para la administracion de la base de datos se utlizara MySql.
* Para repositorio del codigo y control de versiones se utilizara Google Codes a traves de la herramienta Subversion
* Como Ide de desarrollo para todas las herramientas utilizaremos Eclipse Helios con el pluging de Spring MVC

Estilo de codificación y buenas prácticas

* const-correctness y sign-correctness
* bien estructurado (no usar continue, break, ni return adentro de loops), usar for / while acordemente
* no usar el valor de retorno de funciones para códigos de error, sino usar exception handling: hacer una jerarquía de excepciones
* No usar Tab, usar 4 espacios
* Comentarios de linea simple utilice ‘//’ para comentarios mas largos utlice ‘/\* ... \*/’
* Unix EOFs
* Utilizar Allman’s stile para codificar (llave en la siguiente linea), ejemplo:

Class Foo  
{  
 public atributo // a 4 espacios de la llave  
 …  
   
 //DOCUMENTE FUNCION DEL METODO  
 /\* nombre: Descripción de la función del método  
 \* parametro 1: descripción de lo que se espera  
 \* ...  
 \* valor de retorno: Descripción de lo que el

\* método retorna, posibles valores especiales

\*/

void metodo()  
 {                  //llave en el siguiente renglón

//NO USE LA MISMA LINEA PARA DEFINIR VARIABLES DEL MISMO TIPO

int a; //defina variables del mismo tipo

int b; //en líneas distintas

if(condicion)

{             //llave en el siguiente renglón  
 for/while(...)  
 {             //llave en el siguiente renglon  
   
 //CONDICIONALES Y BUCLES DE UNA LINEA, SIN

//LLAVES  
 if(condicion)  
 sentencia de una sola linea;  
  
 for/while(...)  
 sentencia de una sola linea;  
  
Y recuerde: codifique siempre como si la persona que va a mantener su código fuera un violento psicópata que sabe donde vive.  
  
P.D. sabemos donde vive

* + 1. Directivas

El diseno de la pagina debe respetar el siguiente formato:

