

## 1. Introducción

Un inversionista posee un conjunto de bienes **inmuebles** y desea gestionar la compra, venta y alquiler de las propiedades que dispone.

Los inmuebles son todos aquellos bienes considerados “bienes raíces” por tener de común la circunstancia de estar íntimamente ligados al suelo, unidos de forma inseparable, física o jurídicamente al terreno. Algunos bienes raíces son: casas, fincas, lotes, centros comerciales, edificios, etc.

Para el inversionista es indispensable gestionar el inventario de propiedades que tiene a su haber para responder oportunamente ante la posibilidad de concretar nuevos negocios. En este caso, el inversionista de bienes raíces solo hace negocio con casas.

## 2. Objetivos del Proyecto

- a) Realizar un prototipo no funcional con validación de la entrada de datos. El prototipo debe corresponder a una aplicación de consola (stand alone) con GUI.
- b) Ejercitar la toma de decisiones que conlleva el ejercicio de diseño a partir de la necesidad planteada (Gestión de bienes inmuebles).
- c) Aplicar el MVC incluyendo capa de integración y aspecto de seguridad.
- d) Aplicar el patrón adaptador para integración con terceros.
- e) Aplicar el patrón creacional.
- f) Aplicar el patrón decorador.
- g) Aplicar el patrón observador.
- h) Aplicar el patrón estrategia.
- i) Considerar aspectos de seguridad (autenticación y autorización).
- j) Considerar aspectos de “bitacoreo”.
- k) Documentar el diseño de software de la necesidad planteada.

### 1. El contexto y las funcionalidades esperadas

Inicialmente, el tipo de propiedades que negocia el inversionista son casas solamente.

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

Todas las propiedades tienen un “número de finca” que las identifica de forma única en el inventario, área del terreno, valor fiscal y la ubicación: Provincia, Cantón, Distrito y dirección exacta.

Para el registro de propiedades, también deberá tomarse en consideración la cantidad de niveles, el color, cuántos espacios tiene la cochera para parqueo y año de construcción de la propiedad.

Cada vez que se registra una propiedad se debe adjuntar una fotografía que corresponda a la propiedad en cuestión. La fotografía debe quedar almacenada en el sistema de archivos relativo a donde se ejecuta la tarea y dicha ruta debe estar contenida en el documento XML respectivo. Asegúrese que la ruta donde almacena las imágenes no sea una ruta **absoluta**, que sea una ruta **relativa** al contexto de ejecución de su tarea programada.

Debe ser posible registrar un cliente interesado, los datos que serán requeridos para su registro son: identificación, número telefónico, nombre y primer apellido, dirección de correo y estado civil. El estado civil debería **seleccionarse** de una lista pre-configurada con los diferentes valores que corresponden al estado civil de una persona.

Un cliente interesado es aquel cliente que dice tener interés en comprar o alquilar alguna de las propiedades del inversionista. El cliente interesado pasará a ser cliente activo de forma automática inmediatamente después de haber firmado un contrato de compra o alquiler. Lo anterior puede significar que se deba modificar el XML donde se almacenan los clientes, si un cliente tiene el estatus de “interesado” deberá pasar a estatus de “activo” en dicho XML.

Para registrar la venta o alquiler de una propiedad se debe **seleccionar** al cliente que firmará el contrato, digitar el monto de la transacción de la venta o alquiler y la fecha de firma del contrato. Una vez que el inversionista vende una propiedad no podrá realizar nuevas negociaciones sobre dicho inmueble. Cada vez que ocurre una venta de una propiedad debe calcularse un porcentaje de comisión, que para el caso de la venta de una casa, corresponde al 2,5% del valor fiscal. En caso de ser un contrato de alquiler, se debe digitar el monto que será cobrado mensualmente por concepto de mantenimiento y el monto que será dado en garantía por el alquiler. Todos los contratos, sin excepción deberán tener un código de 6 dígitos que cumpla con el siguiente formato:

Código de contrato de Venta: VEN####

Código de contrato de Alquiler: ALQ####

Nota: los símbolos ##### corresponden a valores numéricos que deben ser generados de forma automática por el sistema, debe generarse un número de cuatro dígitos consecutivo.

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

Debe ser posible registrar pagos por concepto de alquiler y cuotas de mantenimiento, para ello se debe **seleccionar tanto** el cliente que tiene alquilada la propiedad, como la propiedad alquilada y digitar la fecha de pago, el monto del pago (que incluye alquiler y cuota de mantenimiento) y el nombre de la persona que realiza el pago. Note que la persona que realiza el pago puede ser distinta de la persona que tiene alquilada la propiedad.

Debe ser posible registrar un mantenimiento a una propiedad alquilada. No debe ser posible registrar mantenimientos a propiedades que hayan sido vendidas. Para registrar un mantenimiento se debe **seleccionar** la propiedad que será sujeta del mantenimiento. Los datos que interesan registrar son: tipo de mantenimiento que deberá ser: preventivo o correctivo, magnitud del mantenimiento: leve, moderado o grande, inversión realizada, fecha del mantenimiento y nombre del responsable del mantenimiento. Tanto el tipo de mantenimiento, como la magnitud del mantenimiento deben **seleccionarse** desde listas pre-construidas.

Debe ser posible consultar la lista de todas las propiedades alquiladas. Para cada propiedad alquilada se deben presentar los datos del cliente, los datos del contrato, todos los datos de la propiedad, los datos de todos los pagos de alquiler y cuotas de mantenimiento realizados y los datos de los mantenimientos realizados sobre la propiedad. Su equipo debe decidir cómo permitirán la consulta de los detalles de cada propiedad, si se presentan todos los datos antes mencionados o si se van presentando según el usuario solicita consultar más y más detalles sobre una propiedad.

Debe ser posible consultar la lista de todas las propiedades vendidas. Para cada propiedad vendida debe presentar los datos del cliente, los datos del contrato y todos los datos de la propiedad (incluyendo el monto correspondiente a la comisión). El equipo de trabajo deberá seleccionar una estrategia para presentar todo lo solicitado por esta funcionalidad.

Debe ser posible conocer la lista de las “n” propiedades compradas por el inversionista de forma más reciente. Para cada propiedad se debe presentar toda la información que se ingresó al momento de registrar la propiedad (lo presentado debe incluir la foto que se utilizó al momento del registro de cada propiedad).

Debe ser posible consultar la lista de todos clientes interesados. Para cada cliente se debe presentar toda la información almacenada al momento del registro. Debe ser posible conocer además la lista de clientes activos, que son aquellos que han firmado al menos un contrato de venta o alquiler de propiedad. Para cada cliente será necesario conocer toda la información almacenada al momento del registro así como toda la información de propiedades que ha adquirido o que ha alquilado.

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

Debe ser posible consultar la información de los mantenimientos realizados sobre una propiedad en particular. Es importante conocer toda la información de cada mantenimiento. Para consultar la información de todos los mantenimientos realizados sobre una propiedad deberá **seleccionarse** la propiedad deseada.

Debe ser posible consultar toda la información de los pagos por concepto de alquiler y cuota de mantenimiento para una propiedad en particular. Para consultar la información de todos los pagos realizados sobre una propiedad deberá **seleccionarse** la propiedad deseada.

Debe ser posible consultar el detalle de un contrato de venta/alquiler de una propiedad. Para consultar el contrato de venta/alquiler de una propiedad deberá **seleccionarse** la propiedad deseada.

IMPORTANTE: DESPUES DE CADA REGISTRO EN EL SISTEMA, SE DEBE PRESENTAR COMO RESULTADO DEL REGISTRO TODA LA INFORMACIÓN OBTENIDA DEL USUARIO Y UNA INDICACIÓN QUE EL REGISTRO (en el XML respectivo) SE HA LLEVADO A CABO DE FORMA EXITOSA.

Todos los datos generados a partir de la ejecución de la tarea programada deben ser almacenados en memoria secundaria (Para cada caso: clientes, compra de propiedad, venta de propiedad, registro de funcionalidad, etc, debe crearse un documento XML). Lo anterior significa que deberá existir un documento XML que almacene a todos los clientes, un documento XML que almacene todas las compras, un documento XML que almacene todas las ventas, y así para cada funcionalidad de esta tarea programada. El equipo de trabajo debe decidir una estructura de documento XML que considere adecuada para almacenar toda la información solicitada por cada funcionalidad. El nombre de los documentos XML los decide el equipo de trabajo.

Con respecto a almacenar fotografías, lo usual es almacenar en el documento XML el *path* relativo (o ruta ) donde está la imagen almacenada en el sistema de archivos. Su equipo debe decidir las reglas sobre el tipo de imágenes que soporta la tarea y sobre los límites de tamaño y “peso” de las imágenes de ser necesario.

**Sí es requerida la robustez de su solución! Todos los datos solicitados son requeridos y se debe proveer la validación correspondiente para cada tipo de dato. Esto es, si se solicita un número debe validarse que el usuario escriba un número, si se solicita un texto, debe validarse que el usuario haya escrito algo en el campo de texto, si se ingresa una dirección de correo, debe validarse el formato de la dirección de correo, si se solicita seleccionar un ítem de una lista debe validarse que el usuario haya seleccionado un ítem de la lista, etc, etc.**

La aplicación debe permitir la ejecución de las siguientes funcionalidades al inversionista:

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

Nota: Es importante considerar que esta solución computacional debe considerar la necesidad de contar con 2 tipos de usuario (inicialmente), el inversionista y la persona que juega el rol de secretaria). El inversionista, deberá poder acceder a todas las funcionalidades y la persona que realiza las labores de secretaria solo podrá acceder a las funcionalidades que son estrictamente de consulta de información. Puede ser que el inversionista solicite eventualmente otro tipo de perfiles de usuario y también podría solicitar asignar permisos diferenciados sobre las diferentes funcionalidades.

Adicionalmente todas las funcionalidades deben quedar registradas en una bitácora, el equipo de trabajo deberá determinar cuáles son los elementos que permitirán realizar una adecuada trazabilidad de las acciones del usuario. Inicialmente, la bitácora deberá almacenarse en un motor de base de datos relacional. Es importante que el tiempo de agregar un registro a la bitácora NO deberá ser “cobrado” al usuario cada vez que ejecute una funcionalidad.

Finalmente, todas las consultas que presenten valores monetarios deberán tener, además, la capacidad de presentar (en pantalla) los mismos montos dolarizados. Para ello será necesario acceder al web service del BCCR para determinar el tipo de cambio que corresponda. Preste atención que la dolarización de montos solo corresponde para efectos de presentación, dado que todos los montos registrados son en moneda local.

- 1) Registrar un cliente interesado en el negocio de bienes raíces (debe solicitar toda la información indicada previamente en esta especificación).  
NOTAS IMPORTANTES QUE DEBEN SER CONSIDERADAS PARA ESTA FUNCIONALIDAD Y PARA TODAS LAS FUNCIONALIDADES:
  - a. Todo lo que diga: “**debe seleccionarse**”, implica que debe existir un componente de lista desplegable en la interfaz gráfica y el usuario deberá seleccionar el valor que le interesa.
  - b. Cada ejecución de la funcionalidad deberá agregar un nuevo registro en el documento XML respectivo.
  - c. Todos los datos solicitados, e.g. el correo electrónico, números, textos, deben ser validados conforme a expresiones regulares. En el documento XML no deben registrarse valores que no hayan sido validados desde la GUI.
- 2) Registrar la compra de una propiedad (una nueva propiedad se agrega al inventario de propiedades del inversionista).
  - a. Debe ser posible registrar cada uno de los tipos de propiedades y su correspondiente información según se indicó previamente en esta especificación.
- 3) Registrar la venta de una propiedad (contrato de venta).

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

- a. Para registrar la venta de una propiedad debe ser posible SELECCIONAR/CONSULTAR la propiedad que será vendida, obtener la información de dicha propiedad y después registrar la venta según se indicó previamente en esta especificación. La consulta de la propiedad puede hacerse por número de finca, o bien seleccionando la propiedad de una lista de propiedades.
- 4) Registrar una propiedad como alquilada (contrato de alquiler)
  - a. Para registrar el alquiler de una propiedad debe ser posible SELECCIONAR/CONSULTAR la propiedad que será alquilada, obtener la información de dicha propiedad y después registrar el alquiler según se indicó previamente en esta especificación. La consulta de la propiedad puede hacerse por número de finca, o bien seleccionando la propiedad de una lista de propiedades.
- 5) Registrar un pago producto de un alquiler y cuota de mantenimiento de una propiedad
  - a. Para registrar el pago de alquiler y cuota de mantenimiento de una propiedad debe ser posible SELECCIONAR/CONSULTAR la propiedad en cuestión, obtener la información de dicha propiedad y después registrar el pago del alquiler y cuota de mantenimiento según se indicó previamente en esta especificación. La consulta de la propiedad puede hacerse por número de finca, o bien seleccionando la propiedad de una lista de propiedades.
- 6) Registrar un mantenimiento a una propiedad.
  - a. Para registrar un mantenimiento de una propiedad debe ser posible SELECCIONAR/CONSULTAR la propiedad en cuestión, obtener la información de dicha propiedad y después registrar la información del mantenimiento según se indicó previamente en esta especificación. La consulta de la propiedad puede hacerse por número de finca, o bien seleccionando la propiedad de una lista de propiedades.
- 7) Consultar las propiedades alquiladas
  - a. Debe ser posible consultar toda la lista de propiedades alquiladas (el listado debe mostrar el tipo de propiedad, “número de finca”, área del terreno, valor fiscal, y fecha del contrato de alquiler.
  - b. A partir de dicha lista debe ser posible SELECCIONAR una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 8) Consultar las propiedades vendidas

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

- a. Debe ser posible consultar toda la lista de propiedades vendidas (el listado debe mostrar el tipo de propiedad, “número de finca”, área del terreno, valor fiscal, la fecha del contrato de venta.
  - b. A partir de dicha lista debe ser posible **SELECCIONAR** una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 9) Consultar las propiedades compradas recientemente
  - a. Debe ser posible conocer la lista de las **n** propiedades compradas recientemente. El inversionista deberá decidir cuántas propiedades desea ver en la lista. La lista deberá incluir el tipo de propiedad, área del terreno, valor fiscal y la provincia.
  - b. A partir de la lista debe ser posible **seleccionar** una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 10) Consultar la lista de clientes interesados (prospectos)
  - a. Debe ser posible conocer la lista de clientes prospectos. La lista deberá incluir la identificación, el nombre y estado civil.
  - b. A partir de la lista debe ser posible seleccionar un cliente en particular y conocer toda la información según se indicó previamente en esta especificación.
- 11) Consultar la lista de clientes (clientes que cuentan con un contrato de venta o alquiler de propiedad)
  - a. Debe ser posible conocer la lista de clientes activos. La lista deberá incluir la identificación, el nombre y estado civil.
  - b. A partir de la lista debe ser posible seleccionar un cliente en particular y conocer toda la información según se indicó previamente en esta especificación.
- 12) Consultar los mantenimientos realizados sobre una propiedad
  - a. Se debe presentar una lista de propiedades alquiladas. La lista deberá incluir el tipo de propiedad, área del terreno, valor fiscal y la provincia.
  - b. A partir de la lista debe ser posible seleccionar una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 13) Consultar los pagos por concepto de alquiler y cuota de mantenimiento
  - a. Se debe presentar una lista de propiedades alquiladas. La lista deberá incluir el tipo de propiedad, área del terreno, valor fiscal y la provincia.
  - b. A partir de la lista debe ser posible seleccionar una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.

## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, luis.chavarria@gmail.com

---

- 14) Consultar el detalle de un contrato de venta de propiedad
  - a. Se debe presentar una lista de propiedades vendidas. La lista deberá incluir el tipo de propiedad, área del terreno, valor fiscal y la provincia.
  - b. A partir de la lista debe ser posible seleccionar una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 15) Consultar el detalle de un contrato de alquiler de propiedad
  - a. Se debe presentar una lista de propiedades alquiladas. La lista deberá incluir el tipo de propiedad, área del terreno, valor fiscal y la provincia.
  - b. A partir de la lista debe ser posible seleccionar una propiedad en particular y conocer toda la información según se indicó previamente en esta especificación.
- 16) Debe ser posible generar un documento PDF con toda la información generada a partir de la ejecución de la funcionalidad 8 (propiedades vendidas). Eventualmente el inversionista podría solicitar la generación de otro tipo de documentos.
- 17) Deber ser posible generar un documento PDF con toda la información generada a partir de la ejecución de la funcionalidad 10 (clientes interesados). Eventualmente el inversionista podría solicitar la generación de otro tipo de documentos.

Tanto para la funcionalidad 16 como 17, el equipo de trabajo debe decidir el formato del documento PDF que será generado, además deben decidir en nombre que tendrá cada documento PDF que será generado a partir de ambas funcionalidades.

En el equipo de trabajo deberán tomar decisiones sobre el flujo de eventos de cada funcionalidad en caso que los datos ingresados no correspondan con lo esperado. La retroalimentación al usuario debe ser muy clara.

De último momento el inversionista ha solicitado que todos los reportes en formato PDF tengan la capacidad de ser creados en idioma español y en idioma inglés (Las primeras páginas del PDF corresponden al idioma español y las siguientes páginas a la correspondiente traducción al inglés). Esto por cuanto se ha dado cuenta, que algunos de sus clientes, solo conocen el idioma inglés. Para la traducción deberá utilizarse el servicio de traducción automática de google. Puede ser que, dependiendo del tipo de clientes, sea necesario utilizar otros servicios de traducción en línea para otros idiomas.



### Puntos a ser evaluados

Su diseño no solo debe responder a los requerimientos actuales, sino que además debe permitir la extensión de nuevas funcionalidades. Su diseño debe tener detalles de implementación, esto es, cada atributo debe indicar su modificador de acceso, su nombre, su tipo, su valor por omisión, etc. Cada método debe tener su modificador de acceso, su tipo de retorno, su nombre y para cada parámetro, el tipo y el nombre del parámetro. Las relaciones entre clases deben contar con su respectivo rol (cuando aplique), la cardinalidad y la navegabilidad.

Su equipo de trabajo debe entregar lo siguiente (todo debe ser “empaquetado” y enviar la respuesta a través del TecDigital):

1. Un prototipo NO funcional para una vista *stand alone* con GUI (El prototipo NO funcional debe ofrecer validación de la entrada de datos). El prototipo DEBE **garantizar la cobertura** de todos los requerimientos planteados en esta especificación. Su equipo debe seleccionar alguna herramienta que favorezca la creación de prototipos NO funcionales con validación de la entrada de datos.
2. Desarrolle un documento donde **cada** funcionalidad (con sus respondientes validaciones) está asociada con un *screenshot* o *screenshots* que garanticen la cobertura de **cada** funcionalidad solicitada. Este documento es vital para asegurar la completitud de su prototipo NO funcional.
3. Diseño de la solución usando la notación de UML (el documento de diseño debe convertirlo en un PDF). El diseño debe responder a los requerimientos y anticipar la extensión de los mismos. El diseño debe utilizar la versión 2.0 de UML (El libro *Learning UML 2.0* ya fue provisto por el profesor).
4. Un documento que haga constar cómo el diseño cumple con los principios vistos en clases (S.O.L.I.D.) y los **n** patrones considerados en este ejercicio.
  - a. Cada cumplimiento de principio y patrón de diseño debe ir acompañado de los siguientes elementos:
    - i. Screenshots con partes del diseño UML donde se cumple el principio/patrón.
    - ii. Explicación de la estrategias o estrategias tomadas en consideración para el cumplimiento del principio/patrón.

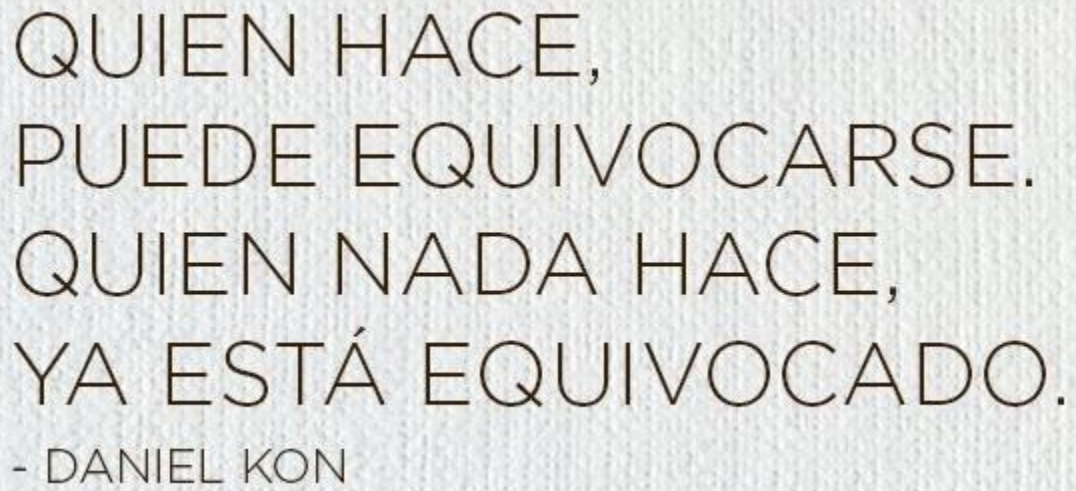
## Proyecto #2, Diseño de Software, Escuela de computación

I Semestre de 2015

Profesor: Luis Javier Chavarría Sánchez, [luis.chavarria@gmail.com](mailto:luis.chavarria@gmail.com)

---

**Para reflexionar:**



QUIEN HACE,  
PUEDE EQUIVOCARSE.  
QUIEN NADA HACE,  
YA ESTÁ EQUIVOCADO.  
- DANIEL KON