**Universidad ORT Uruguay**

**Facultad de Ingeniería**

**Escuela de Tecnología**

**OBLIGATORIO PROGRAMACION 1**

**DOCUMENTO DE ANÁLISIS**

[FOTO ESTUDIANTE]

**Gerardo Silva - 146172**

[FOTO ESTUDIANTE]

**Agustín**

**[Grupo]**

**Docente: [Nombre Apellido]**

**[Carrera]**

**[Fecha de entrega del documento (DD-MM-YYYY)]**

Índice

[1. Descripción general del problema a resolver 3](#_Toc514014266)

[1.1. Tipos de usuario del sistema 3](#_Toc514014267)

[1.2. Listado de funcionalidades 3](#_Toc514014268)

[2. Detalle de Funcionalidades 4](#_Toc514014269)

[2.1 F01 – Nombre de Funcionalidad 1 4](#_Toc514014270)

[2.1.1 Acceso 4](#_Toc514014271)

[2.1.2 Descripción 4](#_Toc514014272)

[2.1.3 Interfaz de usuario 4](#_Toc514014273)

[2.1.4 Validaciones 4](#_Toc514014274)

1. Descripción general del problema a resolver

[Aquí se espera que se presente una descripción general del problema que el obligatorio indica resolver. Se debe describir la idea general sin detallar ninguna funcionalidad]

/// Se requiere la creación de una aplicacion en HTML y JavaScript de un sistema en el que se puedan ingresar 4 tipos de usuarios, en el cual los usuarios puedan interactuar con una colección de inmuebles de tal forma que según privilegios puedan agregar/quitar/modificar y generar reservas.

|  |  |
| --- | --- |
|  |  |

* 1. Tipos de usuario del sistema

[Presentar un listado de los tipos de usuario del sistema]

//Visitante

//Huésped

//Anfitrión

//Administrador

* 1. Listado de funcionalidades

[Se debe presentar una lista de las funcionalidades del obligatorio]

// - Mostrar muebles listados con sus datos y calificaciones - (Visitante)

// - Mostrar muebles listados con sus datos y calificaciones y botón ver mas (Huésped)

//Mostrar muebles por criterio de propiedad (Anfitrión)

//Habilitación o des-habilitación de inmueble(Anfitrión)

//Registro de inmueble(Anfitrión)

//Ordenar muebles según precio de mayor a menor o viceversa y por calificación(Huésped, Visitante)

//Ordenar sus muebles según precio de mayor a menor o viceversa y por calificación(Anfitrion)

//Obtener reporte de muebles (Administrador)

//Acceso por medio de usuario y contraseña (Huésped, Administrador, Anfitrión)

//Registro usuario (Visitante)

//Dar alta de anfitrión (Administrador)

//Cerrar sesión (Huésped, Administrador, Anfitrión)

//Seleccionar moneda (Huésped, Visitante, Anfitrión)

//Cambiar cotización (Administrador)

//Realizar reserva de inmueble (Huésped)

//Consultar reserva (Huésped)

//Realizar calificación (Huésped)

[Cada una de las funcionalidades deberá incluir: Numero – Titulo – Usuarios que pueden acceder]

[Las funcionalidades deberán estar ordenadas por tipo de usuario.]

[

Ejemplo:

F01 – Nombre Funcionalidad 1 – Usuario/s: Administrador

F02 – Nombre Funcionalidad 2 – Usuario/s: UsuarioNormal

]

1. Detalle de Funcionalidades

A continuación, se presenta el detalle de cada una de las funcionalidades a resolver en el obligatorio. Las funcionalidades están ordenadas por tipo de usuario.

* 1. F01 – Nombre de Funcionalidad 1
     1. Acceso

[Se deberá indicar que tipo de usuario puede acceder a esta funcionalidad. Si hay mas de un usuario que puede acceder se deberá indicar]

* + 1. Descripción

[Se deberá describir el objetivo de la funcionalidad]

* + 1. Interfaz de usuario

[Incluir diseño preliminar del formulario que se le presentará al usuario]

[Para este documento no es necesario presentar obligatoriamente captura de pantalla del diseño en HTML, puede sustituirse por dibujo a mano alzada o utilizando alguna herramienta de dibujo]

* + 1. Validaciones

[Se deberá indicar que se debe validar en cada campo del formulario y cuál será el mensaje de error /sugerencia en cada caso]

[Incluir también que datos son obligatorios]