

Pauta Entregable I (Informe + Modelos + Software)  
Análisis y Diseño de Software / Fundamentos de Ingeniería de Software

Integrantes:

|  |  |  |
| --- | --- | --- |
| Nombre | Email | Teléfono |
| Felix Urtubia |  |  |
| Juan Pablo León | Juan.leonl.14@sansano.usm.cl | 99052469 |
| Jordan Esquivel |  |  |

**Elimine las siguientes dos páginas y las secciones en rojo antes de entregar el informe.**

# Pauta para Entregable/Informe I

## Objetivo

Presentar las consideraciones iniciales respecto al proyecto a desarrollar y los elementos requeridos para el Entregable I.

## Consideraciones generales

-Todos los grupos de trabajo **(3 personas)** trabajan en el mismo proyecto. Sin embargo, se espera y acepta diversidad en las formas de trabajo e implementación.

-Se ha establecido como obligatorio el uso de las siguientes herramientas para **modelado y gestión del desarrollo de software**:

* **Visual Paradigm CE**(modelado, UML)
* **Trello** (kanban)
* **Github** (issue tracker)
* **Github** (scm)

-Se evaluará **buenas prácticas** del desarrollo. Así también buenas prácticas en el uso de las herramientas mencionadas (mayores detalles en siguiente sección). Todos los proyectos deben ser *públicos* en las herramientas mencionadas.

-Se realizarán **revisiones por pares**. Los ayudantes informarán el *match* entre grupos. El resultado de esta revisión debe concluir en un reporte conciso de 1 página (máximo 2 si es necesario). El reporte de revisión debe ser entregado a no más de una semana a contar de la fecha límite para el Entregable. La evaluación de este reporte tiene una ponderación de 20% en la nota del “Proyecto”.

-Su grupo de trabajo tendrá un ayudante asignado por todo el semestre.

-Todos los diagramas, a excepción del modelo de la base de datos, deben ser realizados utilizando **UML 2.0**. Debe usar una herramienta acorde a esta consideración.

## Consideraciones sobre las herramientas

Las herramientas mencionadas serán utilizadas siguiendo buenas prácticas del desarrollo de software. En particular:

* **Trello**: la herramienta implementa un Kanban que debe tener, al menos, tres columnas: Backlog, In Progress, Done. Si usted juzga necesario mantener más columnas, puede hacerlo. Los elementos a incluir en las columnas deben ser **minimal marketeable features (MMF)**. No está permitido el uso del Kanban como issue tracker, es decir, no se aceptan ítems como “corregir defecto en código xxxxx”.
* **Issue Tracker de Github**: la herramienta implementa un **issue tracker** o gestor de incidencias. En esta herramienta deben ir todas las incidencias encontradas. Se incluye aquí elementos como “corregir acento en palabra xxxxx de página inicial”, “aplicar refactoring a código en archivo xxxx.xxxx”, “corregir defecto en algoritmo xxxx”, etc. No deben ir en esta herramienta elementos de tipo MMF.
* **Github**: la herramienta es para **gestionar las versiones de su proyecto**. Todo el código y la documentación relacionada son ítems de configuración. Es de gran importancia que usted utilice esta herramienta para **gestionar versiones y NO como backup**. Si usted tiene código que no compila o con errores, o documentación incompleta, NO debe subirla a Github.

*Tanto los ayudantes como el profesor estarán revisando periódicamente el trabajo sobre estas herramientas. El uso de estas herramientas también es evaluado.*

## Contenido del Entregable I

1. Este informe (Informe I)
2. Modelos
3. Primera parte del software.

## Contenido del Informe I (ver detalles más abajo)

1. Listado de requerimientos

## Contenido de los Modelos (ver detalles más abajo)

1. Casos de uso
2. Diagrama de secuencia del sistema
3. Modelo (ER) de base de datos

## Contenido del Software (primera parte)

1. Base de datos *funcionando* (acorde al modelo entregado en el documento)
2. Funcionalidad que será indicada por su profesor y/o ayudantes en su respectivo paralelo

## Entrega

El Entregable I completo (informe + modelos + software) debe ser subido a GitHub (ver notas más arriba) **hasta el día 11/09/2016 a las 23:55 horas**. El informe debe ser subido en formato PDF. Los modelos deben ser subidos en el archivo VPP del software Visual Paradigm. Por cada día de atraso existe una penalización de 20 puntos. El ayudante asignado a su grupo de trabajo coordinará, posterior a esta fecha, una demostración breve en horario a convenir (ayudantía).

Importante: el archivo VPP debe contener todos los modelos, es decir, es uno por proyecto/equipo.

# Listado de requerimientos

|  |  |  |
| --- | --- | --- |
| Id | Requerimiento funcional | Obligatoriedad |
| FR1 | El administrador debe poder cargar bases de datos | Obligatorio |
| FR2 | El administrador debe poder cargar la encuesta ya subida en SurveyMonkey. | Obligatorio |
| FR3 | El administrador debe poder revisar el historial de actividad de cada usuario | Obligatorio |
| FR4 | El administrador debe poder crear, ver, editar y eliminar usuarios | Obligatorio |
| FR5 | El administrador debe poder revisar el historial de llamadas | Obligatorio |
| FR6 | El administrador debe poder eliminar contactos cuyo estado sea el adecuado para la eliminación | Obligatorio |
| FR7 | El administrador puede crear, ver, editar y eliminar proyectos | Obligatorio |
| FR8 | El administrador debe poder asignar usuarios para que trabajen en un proyecto | Obligatorio |
| FR9 | El administrador debe poder asignar bases de datos de contactos a proyectos | Obligatorio |
| FR10 | El usuario debe poder realizar llamadas a contactos | Obligatorio |
| FR11 | El usuario debe poder rellenar la encuesta con la información que proporciona el contacto durante la llamada | Obligatorio |
| FR12 | El usuario debe poder actualizar el estado del contacto | Obligatorio |
| FR13 | El sistema debe poder registrar las llamada en un historial | Obligatorio |
| FR14 | El sistema debe poder grabar las llamadas y guardar estas grabaciones. | Obligatorio |

|  |  |  |
| --- | --- | --- |
| Id | Requerimiento no funcional | Id de FR asociado |
| NFR1 | Las bases de datos deben seguir un formato estándar | FR1 |
| NFR2 | El sistema debe permitir el volver a llamar a un contacto en caso de que la llamada termine inoportunamente | FR10 |
| NFR3 | El sistema no debe mostrar contactos que ya hayan contestado la encuesta a los usuarios. | FR10 |
| NFR4 | Dos usuarios no deben tener que entrevistar un mismo contacto al mismo tiempo. | FR10 |

# 2- Casos de uso

Casos de uso breves:

1.- Cargar base de datos.

2.- Cargar encuesta.

3.- Crear usuario.

4. -Editar usuario.

5.- Eliminar usuario.

6.- Realizar llamada.

7.- Revisar estado de usuario.

8.- Revisar historial de llamadas.

Casos de uso expandidos:

1.- Cargar base de datos.

2.- Crear usuario.

3.- Eliminar contacto.

4.- Realizar llamada.

# 3- Diagramas de secuencia del sistema

IMPORTANTE: Los diagramas de secuencia deben ser entregados en el archivo de modelo de Visual Paradigm CE.

Elija 2 de los casos de uso en formato expandido y construya diagramas de secuencia **del sistema** para ellos. Recuerde notar claramente cursos alternativos y repeticiones.

Para todos los efectos, considere al sistema como caja negra (i.e., no descomponga el sistema en componentes).

# 4- Modelo relacional de la base de datos

IMPORTANTE: El modelo de la base de datos debe ser entregado en el archivo de modelo de Visual Paradigm CE.

Presente una versión inicial lo más completa posible del modelo relacional para la base de datos que utilizará su sistema.