

## Proyecto “GoMan “

Nuestro proyecto de ‘Programación Estructurada’ consiste en una aplicación de escritorio en el que un “coach” te asiste para administrar tu tiempo en un horario optimizado automáticamente por la aplicación. Con el fin de orientar al usuario a ser más productivo en su día a día.

### *Definición de requerimientos*

#### **Actores del sistema**

- El usuario que posea la aplicación en su ordenador. Los cuales podrían ser estudiantes, trabajadores, ambos o ninguno de los anteriores.

#### **Requisitos de usuario**

- 1. Debe ser posible agendar y modificar actividades.
- 2. Deben poderse agendar actividades con 1 mes de anticipación (Como máximo).
- 3. Deben poderse observar las actividades próximas.
- 4. Debe poderse ver el seguimiento detallado de sus actividades.
- 5. Deben recibirse consejos si se percibe mal progreso en el usuario.

#### **Requisitos de sistema**

- 1. El programa almacena datos de las actividades.
- 2. La información del usuario se guarda de manera local, por lo que se perderá si este decide borrarla.
- 3. Todos los datos ingresados por el usuario deben ser modificables por el mismo.
- 4. La agenda se presenta en forma de tabla únicamente para la semana actual para evitar gráficas mensuales, las semanas pasadas y futuras se muestran en forma de lista con apartados para cada actividad.
- 5. El programa lee la fecha actual del dispositivo, en caso de no encontrar una fecha, se solicitará la fecha actual.
- 6. El programa calcula un valor de rendimiento en forma de puntaje.

- 7. El programa registra días y duración de actividades para al final de la semana mostrar un promedio de horas diarias de productividad y el porcentaje de actividades realizadas.

### **Requerimientos Funcionales**

- GoMan es un programa dedicado al agendado de las actividades, el usuario podrá agendar actividades diaria, semanal y mensualmente en conjunto con la información de la actividad (Nombre, fecha, hora, importancia, duración en horas), el usuario podrá modificar la información o eliminar la actividad en el momento que este lo desee. GoMan ordenará las actividades según la importancia definida por el usuario (Muy importante, importante, pendiente). El programa confirmará con el usuario (Notificación de pregunta) que los planes agendados estén actualizados, también llevará un control del progreso y lo mantendrá informado de manera detallada (Con un apartado de puntajes y datos estadístico) y no detallada (Mediante notificaciones y carteles en la interfaz). GoMan podrá aconsejar al usuario semanalmente (Si percibe que este no está cumpliendo sus actividades correctamente) con mensajes preestablecidos para diferentes situaciones.
- 1.- Agendar actividades diaria, semanal y mensualmente en conjunto con la información de la actividad (Nombre, fecha, hora, importancia, duración en horas).
- 2.- Editar información de la actividad.
- 3.- Generar un reporte semanal detallado en base a las actividades realizadas (puntajes y datos estadísticos).
- 4.- Generar consejos para el usuario basados en su productividad final de cada semana.
- 5.- Informar interactivamente (Mediante notificaciones y carteles en la interfaz) al usuario sobre su progreso.
- 6.- Ordenar las actividades conforme a la importancia definida por el usuario.
- 7.- Retrasar y adelantar actividades conforme a decisión del usuario.
- 8.- Confirmar los planes establecidos y los cambios hechos por el usuario.

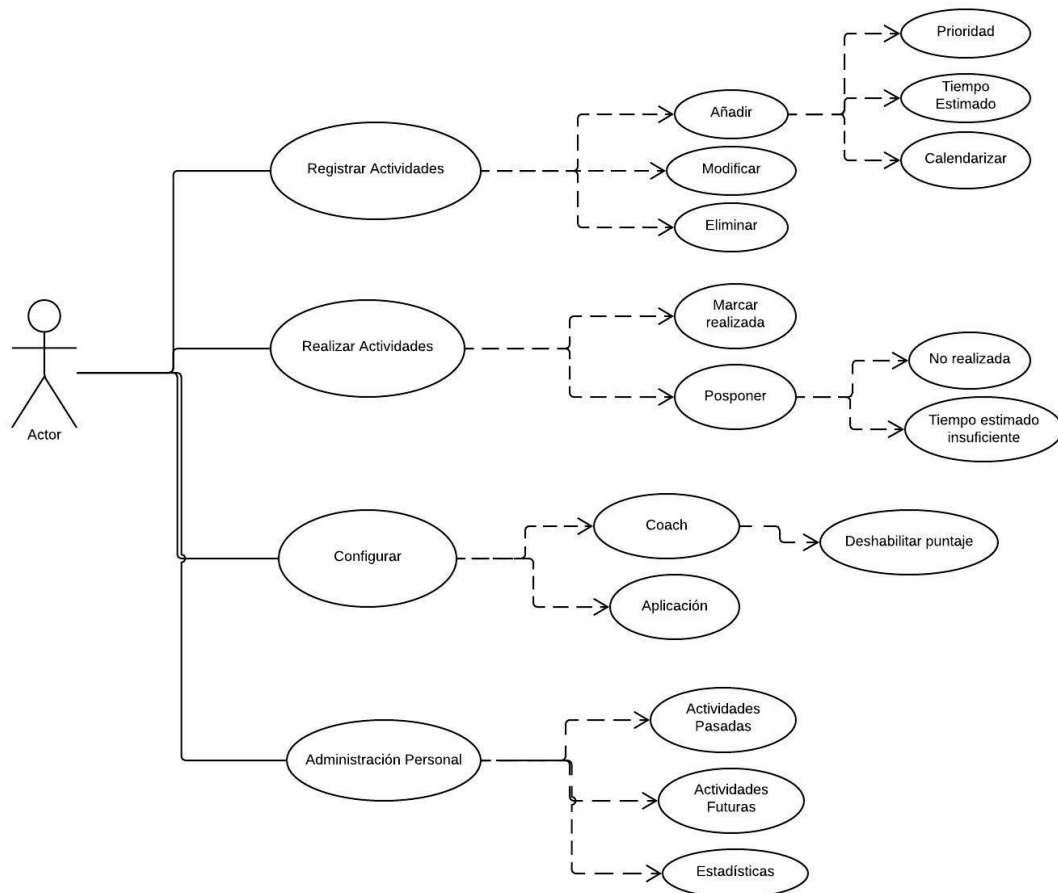
### **Requerimientos No Funcionales**

- GoMan permite al usuario controlar en su totalidad la gestión de sus actividades, únicamente se pueden modificar las actividades de la

semana actual, el programa toma la fecha del dispositivo para saber cuál es la semana actual. Las actividades se guardan y presentan en forma de tabla únicamente para la semana actual. Las actividades podrán ser retrasadas o adelantadas siempre y cuando no sean de importancia “Muy importante”, sin embargo, la fecha sigue siendo modificable por posibles errores o cambios. Para evitar malentendidos el programa confirma que aquellas actividades “Muy importantes” se mantienen con el mismo plan que cuando se agendaron, esto se hace mediante notificaciones un día antes de la fecha de la actividad. El valor de productividad en puntuación es activable y desactivable según las preferencias del usuario.

- 1.- El usuario controla en su totalidad la gestión de sus actividades.
- 2.- Las actividades de la semana actual representan el único conjunto con el que se puede interactuar (Marcar como realizadas, marcar duración extra, etc.).
- 3.- La calendarización de actividades en forma de tabla sólo está disponible para la semana actual.
- 4.- **NO** se pueden **retrasar actividades importantes**, pero la fecha sigue siendo modificable por errores o posibles cambios.
- 5.- Es posible deshabilitar el puntaje (La aplicación solo muestra datos estadísticos).

## Diagrama de Casos de Uso



## Descripción de Casos de Uso

Código: 00001

Nombre: Registrar actividades

Actor(es): Usuario

Descripción: El usuario quiere ingresar una actividad a realizar en la aplicación.

Precondiciones:

1. El usuario está consciente del tiempo estimado que le tomará realizar la actividad.
2. El usuario sabe que día y hora realizará la actividad.
3. El usuario conoce el nivel de prioridad que tiene esa actividad.

Postcondiciones:

1. El sistema almacena la actividad en la fecha indicada.

Escenario Principal:

1. El usuario ingresa el nombre, fecha, hora, tiempo estimado y nivel de prioridad de la actividad.
2. El sistema registra en la fecha y hora indicada la actividad.

Escenario Alternativo:

[Si tras el paso 1 el usuario ingresa erróneamente un dato]

2. El usuario deberá situarse en donde se registró la actividad para editarla o eliminarla.
3. Volver al paso 1 del escenario principal.

Código: 00002

Nombre: Realizar Actividades

Actor(es): El usuario.

Descripción: El usuario requiere realizar una actividad.

Precondiciones:

1. La actividad ha sido previamente registrada.
2. La fecha y hora de realización es correcta.

Postcondiciones

1. El sistema registra que la actividad ha sido realizada.

Escenario Principal:

1. El usuario ubica la actividad que desea realizar.
2. El usuario selecciona la opción de iniciar la actividad.
3. El sistema verifica que sea la fecha y hora previamente establecidas.
4. El usuario selecciona la opción para registrar la actividad como finalizada.

Escenario Alternativo:

[Si tras el paso 3, el usuario no logra terminar la actividad en el tiempo registrado]

4. El usuario selecciona la opción de posponer la hora de finalización.
5. Las actividades contiguas se atrasarán dependiendo del tiempo añadido.
6. Paso 4 del escenario principal.

Código: 00003

Nombre: Editar actividades.

Actor(es): El usuario.

Descripción: El usuario requiere modificar atributos de alguna actividad registrada.

Precondiciones:

1. La actividad debe estar previamente registrada.
2. La actividad NO puede ser de prioridad alta.

Postcondiciones

1. El usuario modifica atributos de la actividad. (fecha, hora, tiempo estimado).

Escenario Principal:

1. El usuario se ubica en sus actividades.
2. El usuario selecciona la opción de editar actividad.
3. El usuario modifica el atributo que necesite.
4. El sistema verifica el nivel de prioridad de la actividad.
5. El sistema modifica el atributo seleccionado

Escenario Alternativo:

*[En caso de que el nivel de prioridad sea alto y se desee modificar la fecha]*

5. El sistema imprime que no es posible editar la fecha de una actividad con alta prioridad.
6. El sistema da la opción de regresar a las actividades.

Código: 00004

Nombre: Retroalimentación Semanal

Actor(es): El usuario.

Descripción: El sistema deberá realizar una retroalimentación del desempeño del usuario al momento de realizar sus actividades en la semana.

Precondiciones:

1. Deberán haber pasado 7 días desde la descarga de la aplicación, o bien, desde la última retroalimentación.

Postcondiciones

1. El sistema le dará al usuario una retroalimentación representada por un puntaje dependiendo de que tan productivo fue en la semana.

Escenario Principal:

1. El usuario inicia la aplicación el séptimo día.
2. El sistema identifica las actividades que pospuso durante la semana (-)
3. El sistema identifica las actividades que se realizaron en tiempo (+)
4. El sistema identifica el tiempo destinado a realizar las actividades de la semana,
5. El sistema imprime una puntuación y datos estadísticos como promedio de horas productivas al día, horas totales realizando actividades productivas, etc.

**Entidades de información utilizadas**

- Para obtener la información necesaria para realizar nuestro proyecto y los requerimientos nos basamos en el funcionamiento de aplicaciones similares.

*Definición de Estándares de codificación*

**Estándar de codificación**

- Realizar la declaración de variables línea por línea.
- El nombrado de variables deberá ser con sustantivos relacionados con el valor que tome o acción que realice esa variable. El nombre podrá ser de más de una palabra, pero se escribirá sin espacios.
- deberán ser sustantivos relacionados con el valor que devuelva o la acción que realice ese método. El nombre podrá ser de más de una palabra, pero se escribirá sin espacios.
- En los métodos, la llave de apertura deberá aparecer en la línea siguiente a la declaración.
- La llave de clausura del método debe comenzar una línea, alineado verticalmente con la llave de apertura.
- Cada línea debe contener una sentencia.
- Se utilizará el “nombrado PasCal” para variables, clases y métodos donde la primera letra de cada palabra deberá ser mayúscula.

**Plantilla de Documentación**

- La plantilla elegida para documentar el proyecto es la de “Documentación XML”, la cual genera un archivo de documentación xml en tiempo de compilación.

**Herramienta para la documentación automática**

- La herramienta que decidimos utilizar es “Sandcastle” el cual es un generador de documentación de Microsoft. Produce automáticamente la documentación del código de estilo MSDN a partir de la información de reflexión de los ensamblados .NET y los comentarios de la documentación XML que se encuentran en el código fuente de estos ensamblajes.

### *Proceso de Desarrollo*

#### **Métodos y Herramientas**

- El lenguaje que se utilizará en este proyecto es C#.
- El IDE que utilizaremos es Visual Studio y para compilar se requiere del framework .NET
- Se usará una herramienta llamada “Telerik” para las interfaces.
- La plataforma es será Windows (winforms).

#### **Procesos de comunicación, planeación, bitácoras.**

- Un integrante del equipo irá realizando una bitácora describiendo las actividades realizadas en la fecha correspondiente.
- El desarrollo será principalmente de forma no presencial.
- Se realizarán dos reuniones por semana para verificar avances.

#### **Métricas**

- Se decidió destinar a algunos roles relacionados al proceso de desarrollo a los integrantes del equipo, donde cada integrante será sublíder de su respectiva etapa.

Amaury Morales: Arquitecto de diseño/Coder

Jimmy Ojeda: Pruebas/Coder

Alejandro Torre: Calidad/Coder

Jorge Pinto: Coder

Donde todos nos dedicaremos a desarrollar el proyecto pero cada quien tendrá actividades específicas para lograr que el proyecto tenga la máxima calidad posible y así lograr dividir equitativamente las tareas.

#### **Calendario**

- Marzo 1-10: Etapa de diseño y planeación de estructura.
- Marzo 11-18: Desarrollo de estructura.



- Marzo 19-31: Desarrollo de funciones principales.
- Abril 1-20: Desarrollo de interfaz y funciones secundarias.
- Abril 20-Mayo 5: Etapa de Pruebas y calidad.

Nos apoyaremos de la herramienta Trello para dividir las tareas en cada etapa y para tener un **control** sobre las actividades que se realicen

### **División de Actividades**

Como se pudo apreciar en el apartado del calendario, se cuenta con 5 etapas generales en todo el proceso de desarrollo; por lo que cada etapa representa un 20% de un 100% total del proyecto (ignorando el nivel de esfuerzo requerido en cada etapa).

Las actividades se tienen que dividir equitativamente con respecto a su nivel de esfuerzo requerido, por lo que en cada etapa habrá una serie de actividades con su respectivo puntaje (del 1 al 10 aproximadamente, dependiendo del nivel de esfuerzo/dificultad) donde cada integrante deberá tener una suma total de 25 puntos de actividades al final del proceso de desarrollo. Esto con el objetivo de dividir las actividades lo más equitativamente posible dependiendo del nivel de dificultad.

### **Repositorio**

<https://www.github.com/alextorrer/GoMan>