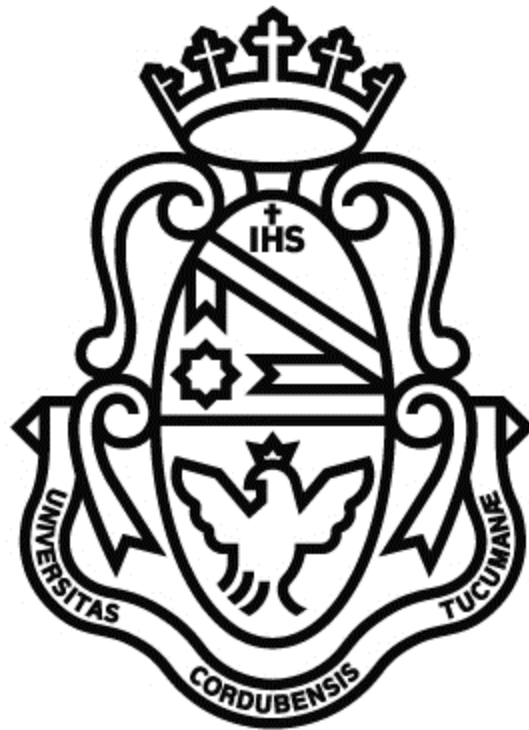


Universidad Nacional de Córdoba

Facultad de Ciencias Exactas, Físicas y naturales



Sistemas de Computación

Diseño de un sistema sensor de temperatura

Martínez Franco Leonardo

32 572 434

2014

ÍNDICE

Aspectos Generales del proyecto	3
Dominio del Problema	3
Requerimientos Funcionales	3
Requerimientos no Funcionales	3
Análisis	4
LEL – Léxico Extendido del Lenguaje.....	4
Diagrama de Casos de uso	5
Diagrama de Casos de Uso	5
Actores.....	5
Escenarios	6
Diseño	11
CRC Cards.....	11
Diagrama de clases	12
Diagrama de clases del Sistema.....	12
Diagrama de Secuencia.....	13
UI: Interfaz de Usuario.....	14
Bibliografía.....	17

ASPECTOS GENERALES DEL PROYECTO

El objetivo del trabajo es diseñar una aplicación compuesta por HW y software.

El HW está encargado de censar un valor de temperatura ambiente, y el software, montado sobre una arquitectura, mostrará el resultado (el valor medido).

Además, le permitirá al usuario establecer un valor de temperatura límite.

El Hardware hará sonar una alarma si la temperatura medida excede este valor límite. La aplicación debe notificarle al usuario de esta situación.

DOMINIO DEL PROBLEMA

Se requiere una aplicación que permita monitorear un dispositivo de hardware que mide la temperatura de una habitación.

Se podrá definir desde la aplicación un valor de temperatura límite. Si se excede este valor, el Hardware hará sonar una alarma.

El hardware almacena este valor de temperatura límite, y controla que la temperatura medida no supere este valor. Esto es así por cuestiones de seguridad. Ya que, en caso que la aplicación no pueda comunicarse con el hardware en algún momento, éste pueda seguir funcionando normalmente.

La aplicación es, entonces, un gestor del hardware, y provee al usuario de una interfaz gráfica que informa la evolución de temperatura de la habitación.

Si la alarma se activó, sólo deja de sonar cuando el valor de temperatura desciende hasta ser inferior a la de umbral.

REQUERIMIENTOS FUNCIONALES

- Controlar mediante software el funcionamiento de un dispositivo de hardware (que mide temperatura).
- Poder establecer un valor de temperatura límite.
- Notificar al usuario cuando esa temperatura sea superada.
- Mostrar un gráfico de variaciones de la temperatura en el tiempo.

REQUERIMIENTOS NO FUNCIONALES

- Tiempo entre que se realiza la medición y se muestra en pantalla no debe superar el segundo.
- Interfaz que se maneja con uso del mouse y se ingresan valores por teclado.
- La aplicación de la interfaz gráfica funciona en Linux y Windows (sobre una arquitectura x86).

ANÁLISIS

LEL – LÉXICO EXTENDIDO DEL LENGUAJE

*Modelo contextual que permite capturar el lenguaje de un dominio, para tener un mejor entendimiento de éste.
Permite entender el lenguaje del dominio.*

<i>Sinónimo</i>	Temperatura, valor censado por el HW
<i>Noción</i>	Valor que mide el HW
<i>Impacto</i>	Información que comparten el HW y el SW
	La alarma suena cuando supera un valor límite
	El HW se encuentra censando este valor constantemente
	El software muestra este valor al usuario

Entrada LEL para “Temperatura”

<i>Sinónimo</i>	Comunicador con HW
<i>Noción</i>	Encargado de comunicar el sistema y el HW
<i>Impacto</i>	Recibe temperaturas del HW
	Envía temperatura límite al HW

Entrada LEL para “Comunicador con HW”

DIAGRAMA DE CASOS DE USO

Partiendo del análisis de requerimientos, definimos el siguiente diagrama de casos de uso que se muestra en la figura 1.

DIAGRAMA DE CASOS DE USO

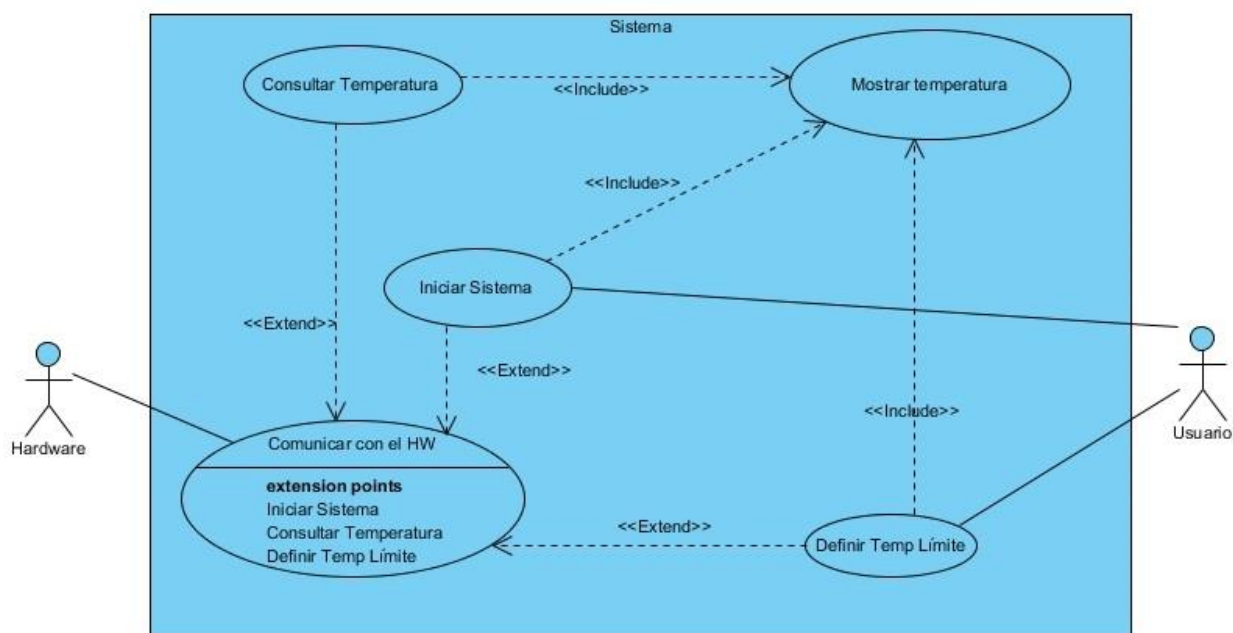


Figura 1. Diagrama de casos de uso del sistema

ACTORES

Definimos al actor principal Usuario. Consideramos al Hardware como actor externo, ya que el sistema interactuará con él.

- Usuario
 - Inicia el sistema.
 - Define una temperatura límite.
- Hardware
 - El sistema le consulta la temperatura ambiente.

ESCENARIOS

COMUNICAR CON EL HARDWARE

OBJETIVO

El sistema envía y recibe información del Hardware.

CONTEXTO

El sistema ha sido iniciado por el usuario.

RECURSOS

Recurso de comunicación Sistema / Hardware.

ACTORES

Información a comunicar

EPISODIOS

Secuencia de inicialización de comunicación.

1. El sistema intenta comunicarse con el Hardware.
2. El sistema espera la notificación del HW.
 - a. Si no llega notificación (la comunicación falla), el sistema vuelve a intentar comunicarse.
 - b. Si la comunicación es exitosa, el sistema recibe una notificación del HW.

Rama 1: El sistema envía el valor de temperatura límite al Hardware.

3. El sistema indica al HW que le enviará el valor de temperatura límite.
4. El sistema envía valor de temperatura límite.
5. El sistema recibe notificación del HW.
 - a. Si no llega notificación (la comunicación falla), el sistema vuelve al paso 1.
 - b. Si la comunicación es exitosa, el sistema recibe una notificación del HW.

Rama 2: El sistema recibe el valor de temperatura censado.

3. El sistema solicita al HW el valor de temperatura actual.
4. El sistema espera el valor.
 - a. Si no hay comunicación del HW, el sistema vuelve al paso 1.
5. El sistema almacena el valor de temperatura actual.

Rama 3: El sistema recibe el valor de temperatura límite.

3. El sistema solicita al HW que le enviará el valor de temperatura límite.
4. El sistema espera el valor.
 - a. Si no hay comunicación del HW, el sistema vuelve al paso 1.
5. El sistema almacena el valor de temperatura límite.

INICIAR SISTEMA

OBJETIVO

El sistema establece comunicación con el Hardware y le muestra al usuario la interfaz gráfica.

CONTEXTO

El sistema ha sido iniciado por el usuario.

La aplicación se puede comunicar con el hardware.

RECURSOS

Recurso de comunicación Sistema / Hardware.

Dispositivo de salida (display).

ACTORES

Temperatura actual.

Temperatura límite.

EPISODIOS

1. El usuario inicia la aplicación.
2. El sistema solicita al HW la medición de temperatura actual (**extiende** de Comunicar con HW, Rama 2).
3. El sistema solicita al HW el valor de temperatura límite que tiene establecida (**extiende** de Comunicar con HW, Rama 3).
4. El sistema inicializa las variables de temperatura.
5. El sistema muestra la interfaz gráfica al usuario (**incluye** Mostrar Temperatura).

CONSULTAR TEMPERATURA

OBJETIVO

El sistema solicita al hardware que le entregue el valor de temperatura que está midiendo actualmente.

CONTEXTO

El sistema ha sido iniciado por el usuario.

La aplicación se puede comunicar con el hardware.

RECURSOS

Dato de temperatura medida.

ACTORES

Comunicador con el HW.

EPISODIOS

1. El sistema notifica al HW que requiere un valor de temperatura (**extiende** Consultar Temperatura, Rama 2).
2. El sistema compara la temperatura medida con la de umbral.
 - a. Si la temperatura actual es inferior a la de límite, el sistema no realiza ninguna otra acción.
 - b. Si la temperatura actual es superior a la de límite, el sistema notifica al usuario que la temperatura excedió el umbral.
3. El sistema Muestra la temperatura modificada al usuario.

DEFINIR TEMP LÍMITE

OBJETIVO

Establece un valor de temperatura límite. Cada valor de temperatura entregado al sistema por el HW será comparado con éste.

CONTEXTO

El sistema ha sido iniciado por el usuario.

La aplicación se puede comunicar con el hardware.

RECURSOS

Valor de temperatura actual.

Valor de temperatura límite.

Dispositivos de entrada.

ACTORES

Temperatura límite.

Comunicador con el HW.

EPISODIOS

1. El usuario solicita al sistema el ingreso de la temperatura límite.
2. El sistema acepta el pedido.
3. El sistema solicita al usuario el ingreso de un valor de temperatura.
4. El usuario ingresa un valor de temperatura.
5. El usuario acepta su acción.
6. El sistema valida el valor ingresado.
 - a. Si el valor ingresado no es válido, el sistema notifica al usuario.
 - i. El sistema rechaza el pedido de ingreso del usuario. Vuelve al paso 1.
7. Si el valor ingresado es válido, el sistema envía el valor de Temperatura límite al HW (**extiende** comunicar con HW, Rama 1).
8. El sistema actualiza su valor de temperatura límite.
9. El sistema indica al usuario que el valor ha sido ingresado exitosamente mediante una notificación.
10. El sistema muestra la temperatura modificada al usuario (**incluye** Mostrar Temperatura).

MOSTRAR TEMPERATURA

OBJETIVO

A través de la interfaz gráfica, el sistema le entrega al usuario los resultados de la evolución de la temperatura en el tiempo de forma gráfica.

CONTEXTO

El sistema ha sido iniciado por el usuario.

RECURSOS

Dispositivos de salida (pantalla).

ACTORES

Temperatura Actual. Temperatura límite.

EPISODIOS

1. El sistema muestra los valores de temperatura actual y límite en la pantalla.
2. El sistema actualiza el histograma de temperaturas.

DISEÑO

CRC CARDS

Estas tarjetas identifican posibles clases para diseñar una solución orientada a objetos. Cada tarjeta representa un objeto del mundo real.

Para analizar el sistema desde una perspectiva de objetos, se comenzó trabajando con tarjetas CRC, para poder identificar candidatos a clases.

Temperatura

Temperatura Límite

Comparador

Consultar Temperatura

Comunicador HW

Comparar Temperatura

Temperatura, Temperatura Límite

Comunicador HW

Obtener Temperatura actual

Temperatura

Obtener Temperatura Límite

Temperatura Límite

Comunicar Temperatura Límite

DIAGRAMA DE CLASES

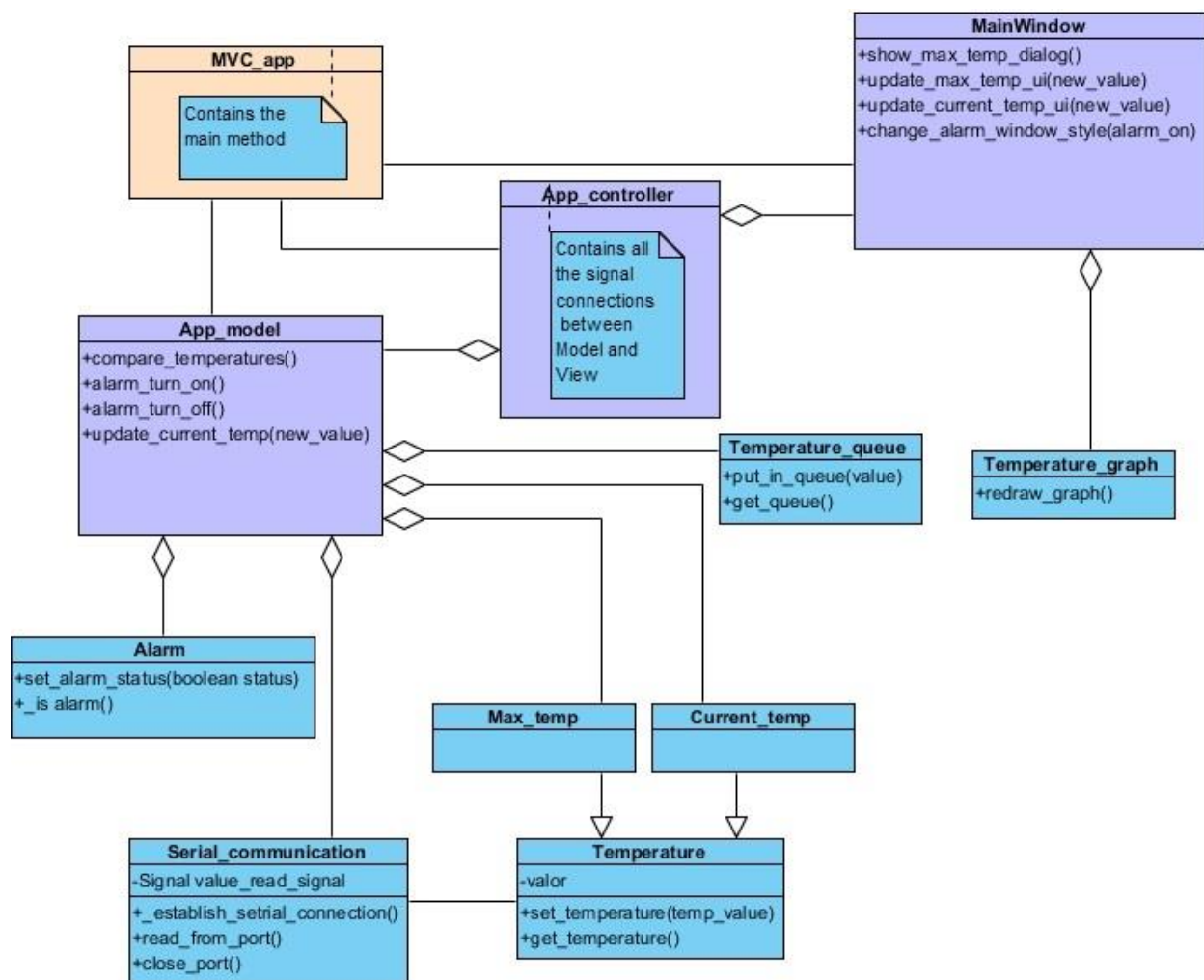
Describimos el sistema en relación a las clases y sus interacciones.

Cabe destacar que en la etapa de diseño se partió de un diagrama de clases, para luego representar el sistema con un diagrama de clases. Pero que se ha pasado de un diagrama a otro, volviendo para refinar uno y otro a medida que se diseñaba.

La premisa de creación del proyecto era utilizar el patrón MVC, ya que así podemos fácilmente independizar la lógica del programa con la interfaz de usuario. En este sentido, se logró que toda la comunicación y la interacción entre la Lógica y la Vista se realice por medio del Controller, así, si en el futuro se requiere actualizar la UI, la lógica del programa no deba modificarse.

DIAGRAMA DE CLASES DEL SISTEMA

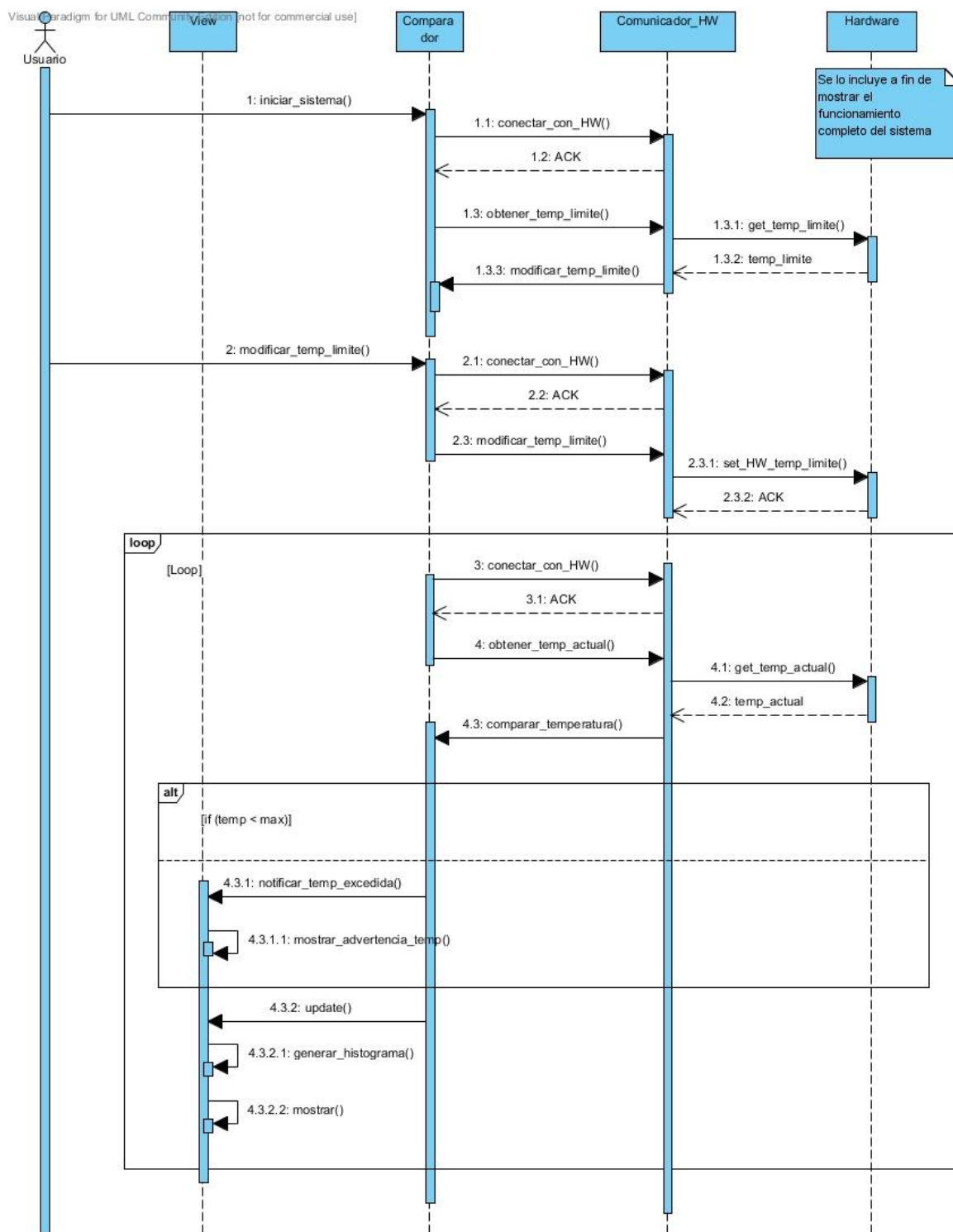
Se muestra el diagrama de clases con los métodos y atributos más importantes.



MVC_app es la clase encargada de instancias los objetos Model, View y Controller, y de conectarlos.

DIAGRAMA DE SECUENCIA

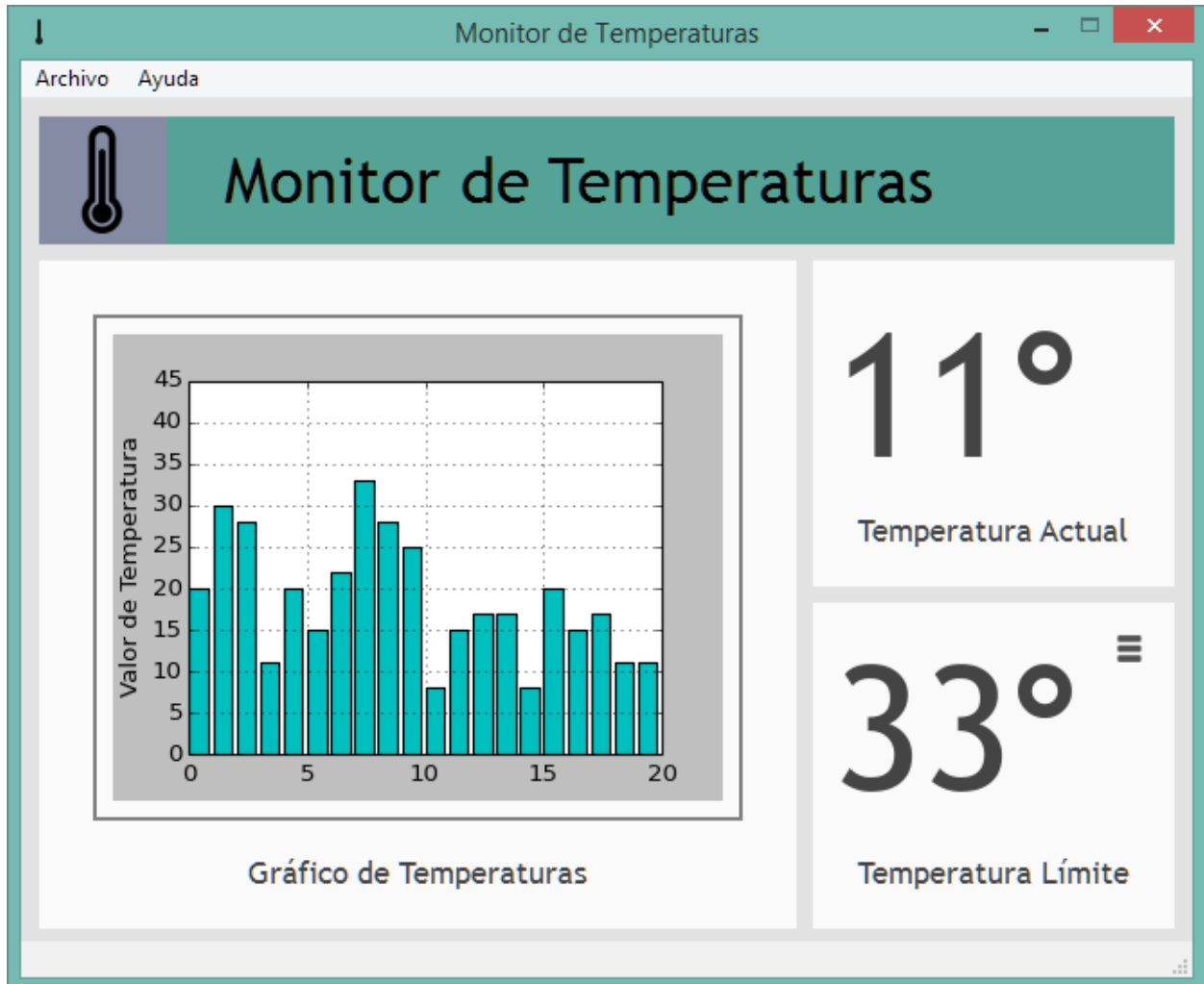
El diagrama de secuencia muestra la interacción y la secuencia de las interacciones de los distintos elementos del sistema.



UI: INTERFAZ DE USUARIO

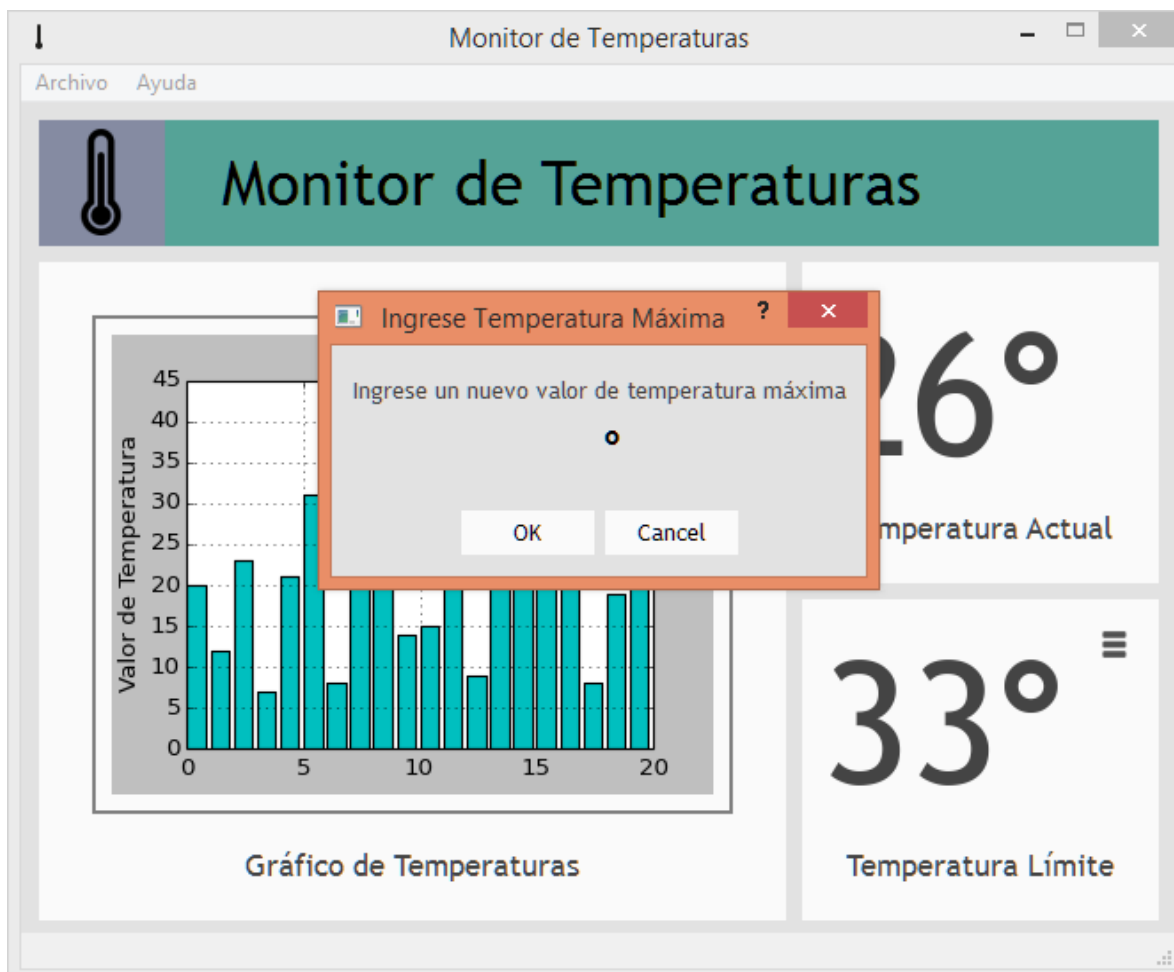
Se definió la interfaz gráfica del usuario de la siguiente forma:

La pantalla le muestra al usuario un resumen histórico de la temperatura.



Desde archivo, o desde el botón al lado del valor de Temperatura Límite, el usuario puede definir un nuevo valor de Temperatura límite.

Esto hace que aparezca una nueva ventana.

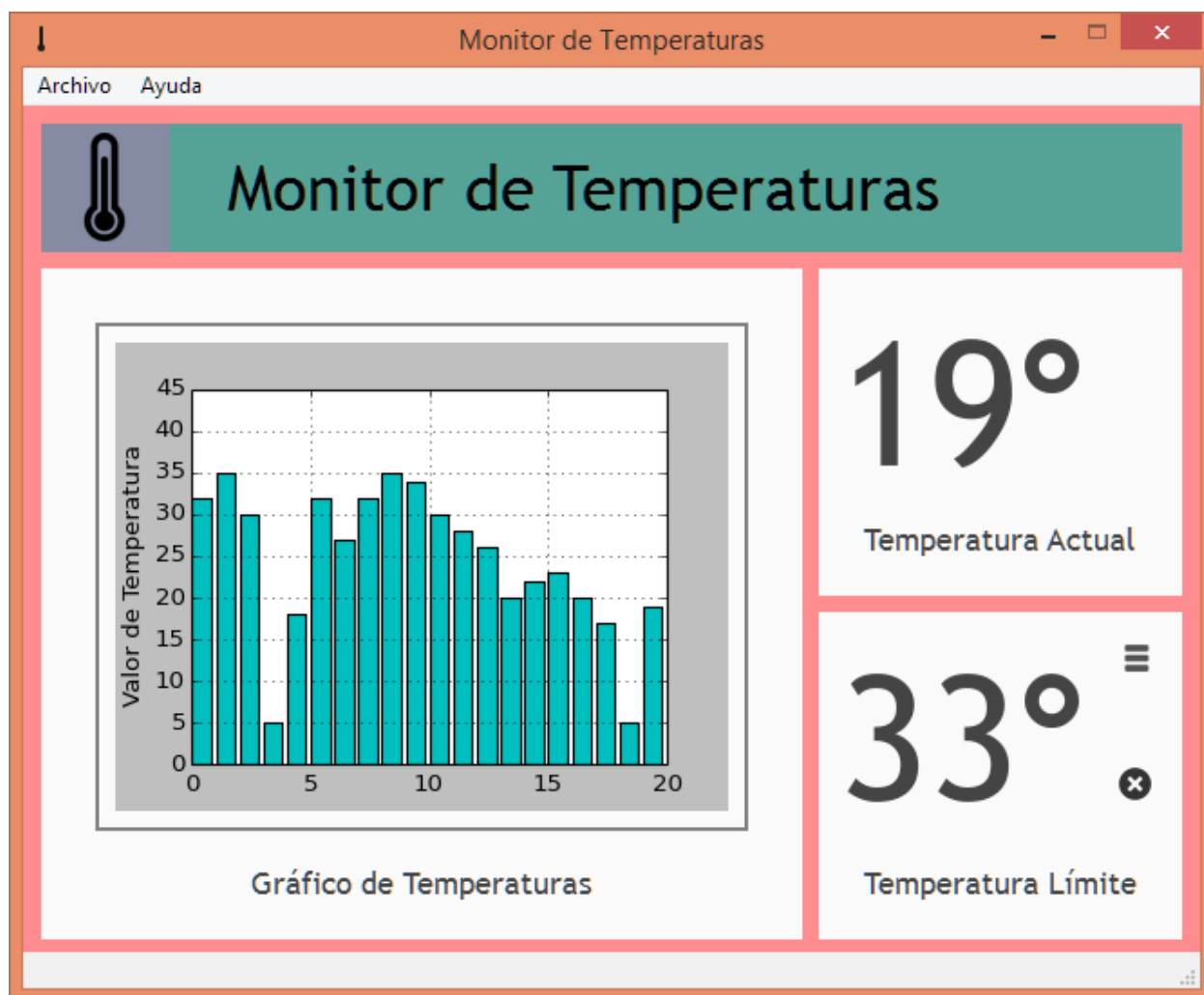


En caso de haber ingresado un valor incorrecto, el sistema no acepta el valor y deshabilita el botón. El usuario sólo puede corregir el valor o Cancelar la acción.

Los valores incorrectos son letras y caracteres especiales, y cualquier combinación que los incluya. El programa sólo acepta valores de dos dígitos (por ejemplo, 55, 03, 5, -1, -8).

Cuando la alarma se activa, la ventana principal se actualiza. Se establece un fondo de color rojo.

Un botón al lado del valor de temperatura límite aparece, permitiendo al usuario desactivar la alarma.



La alarma volverá a activarse cuando se vuelva a llegar al valor de temperatura límite.

Para abandonar la aplicación, el usuario puede hacerlo desde el botón de cerrar ventana o desde Archivo -> Salir de la Aplicación.

BIBLIOGRAFÍA

STORIES, USE CASES Y SCENARIOS

http://en.wikipedia.org/wiki/Use_case

<http://www.cloudforestdesign.com/2011/04/25/introduction-user-stories-user-personas-use-cases-whats-the-difference/>

http://www.karonaconsulting.com/downloads/UseCases_IncludesAndExtends.pdf

PATRONES DE DISEÑO

<http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>

http://en.wikipedia.org/wiki/Observer_pattern

APLICACIONES UTILIZADAS

[Visual Paradigm UML Tool](#)

[Balsamiq Mockups](#)