

# **Diseño y construcción de un Smart Plug**

**Autor**

**Ing. Mariano Mondani**

**Director del trabajo**

**Ing. Juan Manuel Cruz (FIUBA, UTN-FRBA)**

**Jurado propuesto para el trabajo**

- **Ing. Federico Giordano Zacchigna (FIUBA)**
- **Ing Gustavo Alessandrini (INTI)**
- **Esp. Ing. Ramiro Alonso (FIUBA)**

Este plan de trabajo ha sido realizado en el marco de la asignatura Gestión de Proyectos entre abril y mayo de 2016.

## **Tabla de contenido**

[1. Avance en las tareas](#)

[2. Cumplimiento de los requerimientos](#)

[3. Gestión de riesgos](#)

<b>Revisión</b>	<b>Cambios realizados</b>	<b>Fecha</b>
1.0	Creación del documento	27/08/2016
1.1	Se agrega un párrafo antes del diagrama de Gantt	29/08/2016
1.2	Se agrega una carátula y se agregan saltos de página para separar las secciones.	30/08/2016
1.3	Modificaciones en la carátula.	31/08/2016

## 1. Avance en las tareas

1.1 Investigación y elección de frontends analógicos	1.2 Elección de módulo WiFi y estudio de las hojas de datos y notas de aplicación.	1.3 Elección y familiarización con el microcontrolador a utilizar	
\$	=	*	-
2.1 Diseño del diagrama esquemático y determinación del BOM	2.2 Ruteo del PCB y Fabricación de un prototipo	2.3 Verificación del prototipo	
\$	=	\$	++
3.1 Familiarización con FreeOSEK	3.2 Programación del driver para el frontend analógico.	3.3 Testeo del driver del front end analógico.	3.4 Programación del driver del módulo WiFi.
*	-	\$	=
3.5 Testeo del driver del módulo WiFi.	3.6 Programación de la aplicación principal e integración con los drivers.	3.7 Testeo del firmware	
\$	=	\$	+
4.1 Profundización en los aspectos de programación para Android.	4.2 Diseño de las pantallas con las que interactuará el usuario.	4.3 Programación de la aplicación.	
\$	=	\$	=
5.1 Determinación de los requerimientos estéticos y funcionales.	5.2 Diseño del gabinete.		
\$	=	\$	=

6 Escritura del manual de usuario.							
\$	=						
7.1 Escritura del informe de avance		7.2 Escritura del informe final		7.3 Presentación pública final			
*	-	\$	=	\$	=		

En subcelda inferior izquierda colocar:

- \*\* si los recursos u horas utilizadas fueron o están siendo muy inferior a lo planificado.
- \* si los recursos u horas utilizadas fueron o están siendo inferior a lo planificado.
- \$ si los recursos u horas utilizadas fueron o están siendo de acuerdo a lo planificado.
- \$\$ si los recursos u horas utilizadas fueron o están siendo superior a lo planificado.
- \$\$\$ si los recursos u horas utilizadas fueron o están siendo muy superior a lo planificado.

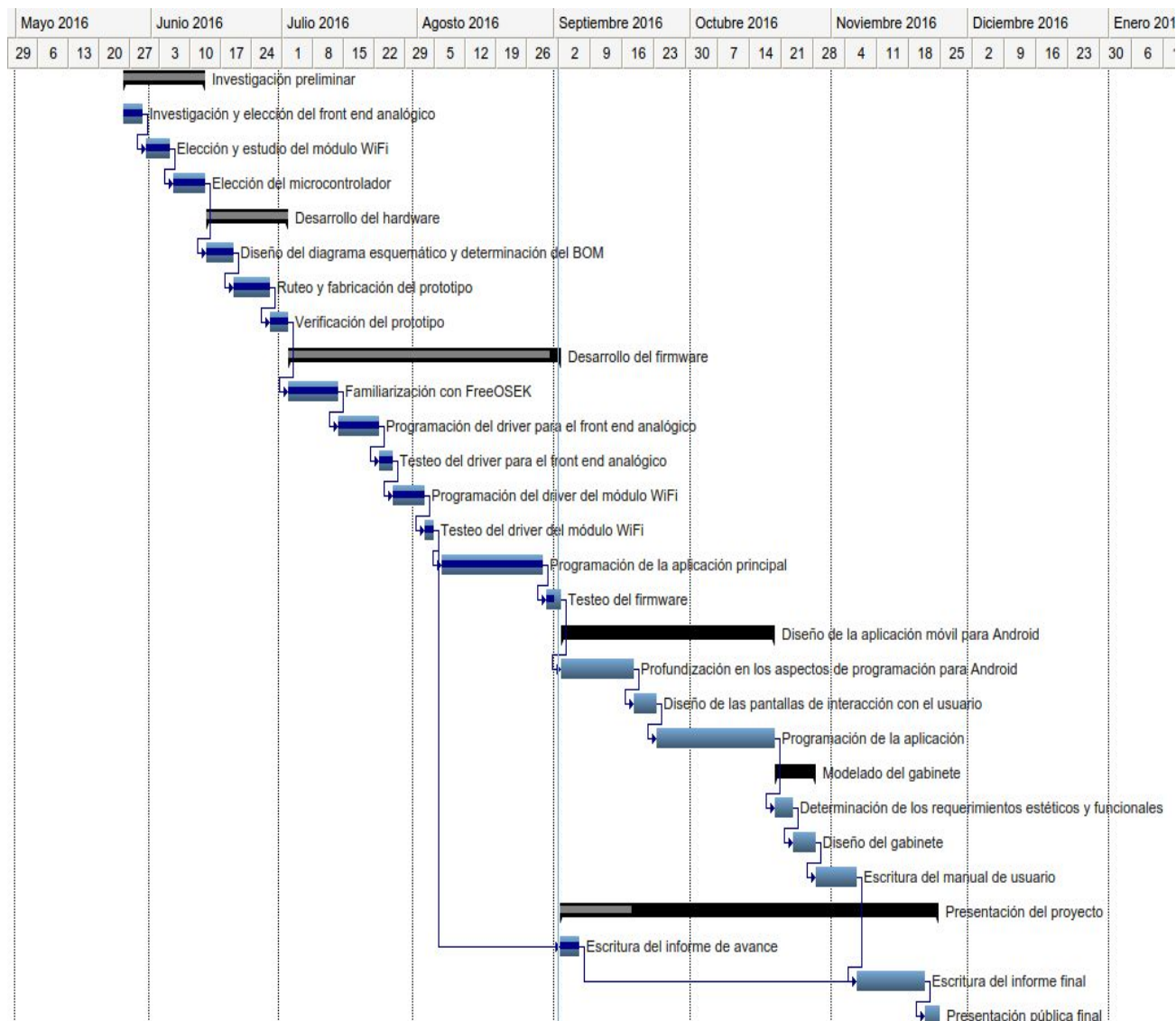
En subcelda inferior derecha colocar:

- -- si la tarea se ejecutó o se está ejecutando mucho más rápido de lo previsto
- - si la tarea se ejecutó o se está ejecutando más rápido de lo previsto
- = si la tarea se ejecutó o se está ejecutando en el tiempo previsto.
- + si la tarea se ejecutó o se está ejecutando con demoras.
- ++ si la tarea se ejecutó o se está ejecutando con demoras muy significativas.



A continuación se muestra el diagrama de Gantt del proyecto, en el cual se actualizaron las fechas de acuerdo al estado actual de situación. Se puede estimar que, de no mediar inconvenientes, se podrán cumplir con los plazos establecidos.

Nombre	Duración	% Completado	Inicio	Fin	Predecesoras
<input checked="" type="checkbox"/> <b>Investigación preliminar</b>	65h	100%	<b>23/05/2016</b>	<b>10/06/2016</b>	
Investigación y elección del front end analógico	20h	100%	23/05/2016	27/05/2016	
Elección y estudio del módulo WiFi	20h	100%	28/05/2016	02/06/2016	2
Elección del microcontrolador	25h	100%	03/06/2016	10/06/2016	3
<input checked="" type="checkbox"/> <b>Desarrollo del hardware</b>	15d	100%	<b>10/06/2016</b>	<b>28/06/2016</b>	
Diseño del diagrama esquemático y determinación del BOM	20h	100%	10/06/2016	16/06/2016	4
Ruteo y fabricación del prototipo	30h	100%	16/06/2016	24/06/2016	6
Verificación del prototipo	10h	100%	24/06/2016	28/06/2016	7
<input checked="" type="checkbox"/> <b>Desarrollo del firmware</b>	52.5d	96%	<b>28/06/2016</b>	<b>27/08/2016</b>	
Familiarización con FreeOSEK	40h	100%	28/06/2016	09/07/2016	8
Programación del driver para el front end analógico	30h	100%	09/07/2016	18/07/2016	10
Testeo del driver para el front end analógico	10h	100%	18/07/2016	21/07/2016	11
Programación del driver del módulo WiFi	25h	100%	21/07/2016	28/07/2016	12
Testeo del driver del módulo WiFi	10h	100%	28/07/2016	30/07/2016	13
Programación de la aplicación principal	80h	100%	01/08/2016	23/08/2016	14
Testeo del firmware	15h	50%	24/08/2016	27/08/2016	15
<input checked="" type="checkbox"/> <b>Diseño de la aplicación móvil para Android</b>	40d	0%	<b>27/08/2016</b>	<b>13/10/2016</b>	
Profundización en los aspectos de programación para Android	50h	0%	27/08/2016	12/09/2016	16
Diseño de las pantallas de interacción con el usuario	20h	0%	12/09/2016	17/09/2016	18
Programación de la aplicación	90h	0%	17/09/2016	13/10/2016	19
<input checked="" type="checkbox"/> <b>Modelado del gabinete</b>	7.5d	0%	<b>13/10/2016</b>	<b>22/10/2016</b>	
Determinación de los requerimientos estéticos y funcionales	10h	0%	13/10/2016	17/10/2016	20
Diseño del gabinete	20h	0%	17/10/2016	22/10/2016	22
Escritura del manual de usuario	30h	0%	22/10/2016	31/10/2016	23
<input checked="" type="checkbox"/> <b>Presentación del proyecto</b>	72d	19%	<b>27/08/2016</b>	<b>18/11/2016</b>	
Escritura del informe de avance	15h	100%	27/08/2016	31/08/2016	14
Escritura del informe final	50h	0%	31/10/2016	15/11/2016	24,26
Presentación pública final	15h	0%	15/11/2016	18/11/2016	27



## 2. Cumplimiento de los requerimientos

Req #1.1: Operar con cargas de 220 V, 50 Hz y hasta 5A. El encendido y apagado de la carga se realizará con un relay mecánico.
Req #1.2: Utilizar un front-end analógico monolítico para realizar el análisis de los parámetros eléctricos.
Req #1.3: Conectarse a la red hogareña mediante un módulo WiFi.
Req #1.4: Ofrecer una interfaz de usuario sencilla a nivel de hardware. Debe existir un único pulsador que permita conectar el Smart Plug a la red.
Req #1.5: Indicar el estado de funcionamiento del Smart Plug mediante un único led bicolor.
Req #2.1: Comunicación con el front-end analógico y obtención de los parámetros eléctricos.
Req #2.2: Conexión del módulo WiFi a la red mediante WPS (WiFi Protected Setup).
Req #2.3: Cada módulo debe tener un registro de la hora y la fecha actual, obtenida a través de un servidor NTP.
Req #2.4: Establecer un protocolo de comunicación entre los Smart Plugs y la aplicación móvil para intercambiar datos y configuraciones.
Req #2.5: Implementar el software en lenguaje C utilizando técnicas propias de los lenguajes orientados a objetos (herencia, polimorfismo, etc).
<b>Req #2.6: Conexión del módulo WiFi a la red mediante un modo Soft-AP.</b>
Req #3.1: Escanear la red en busca de los Smart Plugs presentes.
Req #3.2: Encender o apagar cada Smart Plug.
Req #3.3: Establecer un nombre para cada Smart Plug.
Req #3.4: Programación horaria para el encendido y el apagado de la carga. Se pueden definir horarios distintos para cada día de la semana.
Req #3.5: Visualización de los valores actuales de potencia activa, tensión y energía consumida.
Req #3.6: Visualización, mediante gráficos, de la energía consumida por la carga en las últimas 24 hs, los últimos 30 días, etc.

### **3. Gestión de riesgos**

Riesgo #1: pérdida del prototipo del Smart Plug.
Riesgo #2: imposibilidad de conseguir componentes electrónicos clave.
Riesgo #3: retraso en la fabricación del PCB.
Riesgo #4: aparición en el mercado de un producto de similares características y menos precio.
Riesgo #5: el proyecto queda suspendido frente a otros proyectos más prioritarios.