

Bootcamp Java Developer

Fase 2 - Java Web Developer
Módulo 14

Roles

¿Qué rol cumple cada uno?

Diseñador UI/UX

Realiza estudios y prototipos del desarrollo.

Lo hace pensando **en el usuario y debe generar tests**, acciones de estudio, así como **prototipos** pensados para cada dispositivo y contexto.

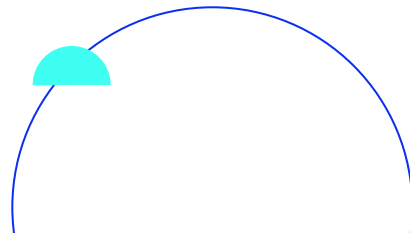
Cuenta con **conocimientos de UX, accesibilidad y diseño** suficientes para poder realizar tal tarea.



Maquetador

Arma el diseño previamente generado y prototipado por el **diseñador UX**.

Muchas veces, debemos trabajar con herramientas de diseño para poder modificar elementos específicos para cada parte del desarrollo pero **no es nuestra tarea diseñar o prototipar**.



Desarrollador Front

Se encarga del trabajo con lenguajes como **HTML, CSS y JavaScript** al igual que el maquetador, pero **profundiza en el trabajo con frameworks** de JavaScript tales como **Vue.js, Angular, React.js, Node.js, y MongoDB**.

En muchos desarrollos la tarea de **maquetador la desarrolla quien también trabaja en el desarrollo front**, pero al ser tareas bastante complejas y específicas lo ideal es que si queremos lograr un proyecto de calidad, se separen las tareas.

Si el **desarrollo y la maquetación lo hace la misma persona**, en el presupuesto freelance se debe segmentar el trabajo, ya que no es el mismo. **El costo debe constar por separado y también los plazos de entrega.**



Comienzo

¿Qué es un prototipo?

Comienzo del desarrollo

Para empezar un proyecto que puede ser una interfaz de cualquier aplicación, desarrollo o un sitio web, necesitamos ya sea nosotros o quien nos da la tarea de realizar el maquetado, **trabajar sobre un prototipo.**

Toda actividad tiene primero un *brainstorming*, y luego se realiza un bosquejo de lo que será finalmente nuestra idea realizada.

En este caso, nuestro proceso, si es que nosotros mismos lo iniciamos, comenzará con un **mapa del sitio**. Esto no es más ni menos que una lista de páginas accesibles para los usuarios.



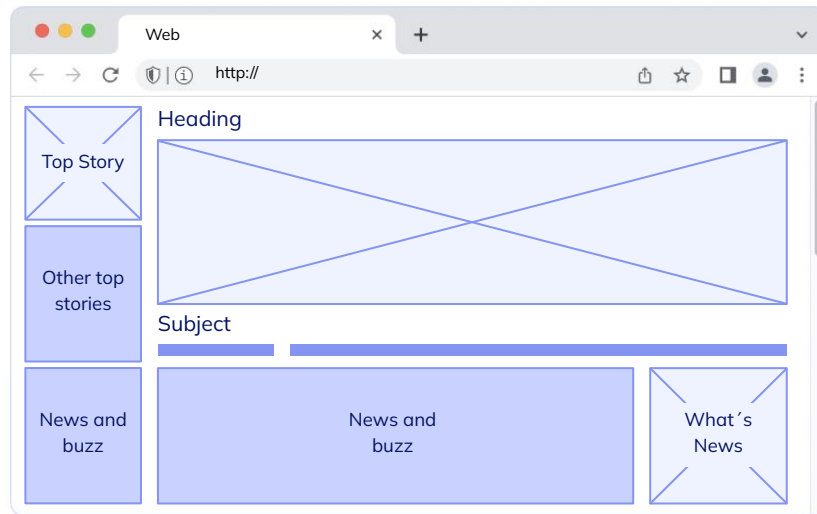
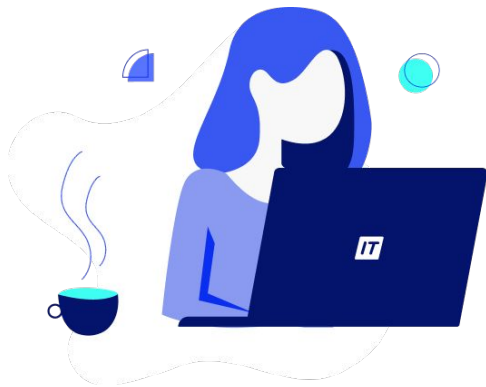
Mapa del sitio

El mapa del sitio es fundamental para evitar problemas con el cliente y **determinar sobre cuántas páginas estaremos trabajando.**



Wireframe

El *wireframe* es el **prototipo más básico** para trabajar y nos permite ubicar los elementos que formarán parte de nuestra interfaz.



Figma

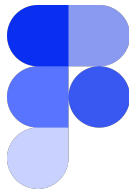
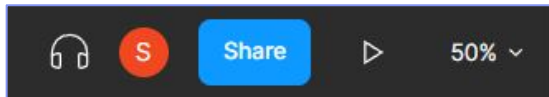
Figma es un programa que nos permite trabajar de manera sencilla con **prototipos y diferentes esquemas incluso del mundo Agile**.

En el caso de los maquettadores y/o desarrolladores debemos conocer los elementos básicos del mismo para **potenciar así nuestro trabajo colaborativo**.

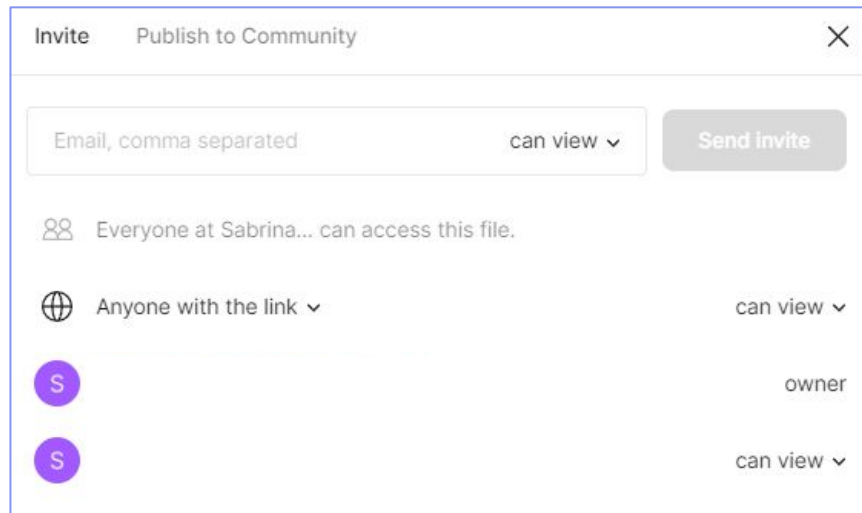
Para eso te invitamos a **crear una cuenta gratuita en [Figma.com](https://www.figma.com)**.

Luego, puedes bajar la app de escritorio o elegir el trabajo online.

Para acceder a un diseño o prototipo o cualquier elemento creado en Figma solo debes obtener el link de trabajo y setear las características de edición/lectura según sea deseado.

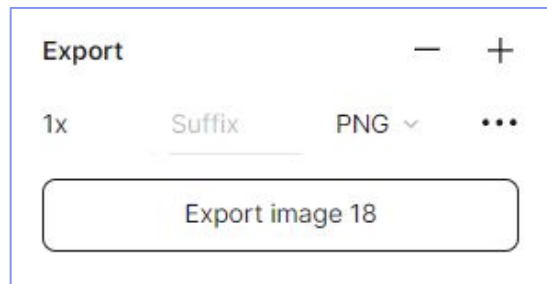
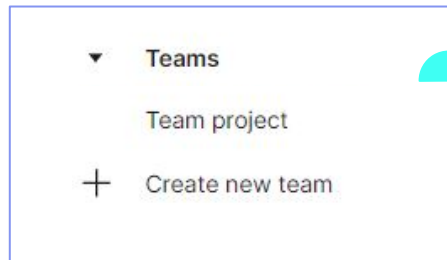


También podemos **crear equipos y agregar miembros** aunque, en la versión gratuita, esta facilidad se ve limitada a tres proyectos por equipo.



Por otro lado, el trabajo de **detección de elementos** es muy sencillo dado que utiliza las mismas herramientas que cualquier programa de diseño. Resulta muy fácil detectar tamaños tanto tipográficos, como de márgenes o espacios en general de nuestro documento.

También, la **exportación de imágenes y/o elementos** es muy sencilla. Se hace con la opción **Export**, luego de seleccionar el elemento en cuestión.



El prototipado, así como el diseño, no son tareas ni del maquetador ni del desarrollador front, pero es muy importante estar al tanto del ABC del programa. Así podremos trabajar en equipo, con un concepto de **funcionalidad asistida**.

Poder detectar todas las partes del desarrollo, así como características del prototipo en sí, nos dará autonomía y, a la vez, estaremos inmersos en un sistema de **trabajo colaborativo** tan necesario para toda empresa.



Entender la grilla CSS

En maquetación se trabaja con una grilla de **12 columnas**. Estas **12 columnas** equivalen al **100% del viewport**.

Por ejemplo, si un elemento toma dos columnas, su `width` será de 16.66%. Se realizó el siguiente cálculo:

$$\frac{2 \text{ (las columnas que se ocupan)} \times 100}{12 \text{ (total de columnas)}} = 16.66\%$$

Veamos un ejemplo más claro donde, en el primer caso, el contenido ocupa 3 columnas (25%) y el segundo contenido ocupa 9 columnas (75%)



Grillas en CSS

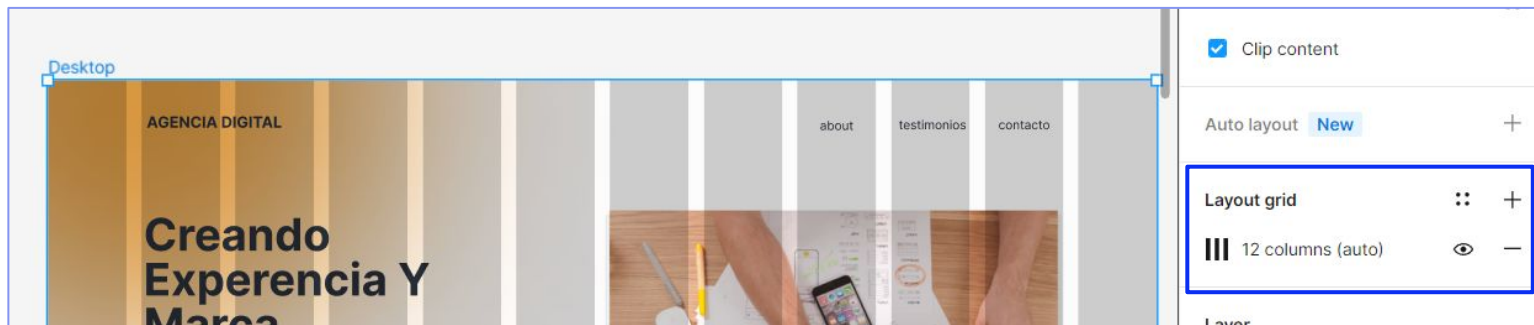
Hagamos un ejercicio de % según la cantidad de columnas, en el caso anterior `.col-3` debería ser implementado al primer elemento y `.col-9` al segundo:

```
.col-1 {width: 8.33%;}  
.col-2 {width: 16.66%;}  
.col-3 {width: 25%;}  
.col-4 {width: 33.33%;}  
.col-5 {width: 41.66%;}  
.col-6 {width: 50%;}  
.col-7 {width: 58.33%;}  
.col-8 {width: 66.66%;}  
.col-9 {width: 75%;}  
.col-10 {width: 83.33%;}  
.col-11 {width: 91.66%;}  
.col-12 {width: 100%;}
```



Grillas en Figma

En Figma, podemos **ver las grillas** desde el panel de propiedades de la siguiente forma:

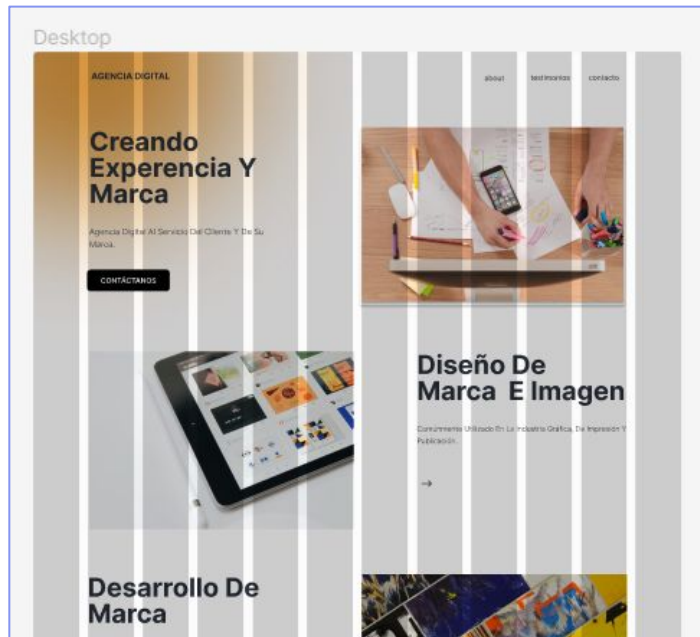


Trabajo con grillas

Seguir una grilla

Debemos seleccionar siempre **Layout grid > 12 columns > Auto** para poder lograr la grilla correcta dado que existen otros formatos pero no son los actualmente trabajados en maquetación.

Actualmente, se utiliza esta grilla para poder ubicar cada elemento en nuestro *layout*.



¿Por qué es importante entender la grilla?

El trabajo con nuestras 12 columnas nos permite **entender la proporción** que guarda cada elemento dentro del diseño general.

Entenderemos mediante este sistema el espíritu del diseño, de modo tal que, responsivo o adaptable, nuestro desarrollo siempre será exactamente igual a lo aprobado por el cliente en etapa de prototipado.



Dev Mode o Modo de Desarrollo desde Figma

Trabajo con Dev

Si bien existían herramientas o extensiones similares en Figma, actualmente se puede trabajar de forma simple (a través de un *slider*) con un **Dev Mode**. **Se asemeja a un inspector de elementos** (el que activamos con *F12* en nuestro navegador) que brinda datos muy interesantes al momento de hacer nuestro CSS.

Es importante entender que **el Dev Mode se utiliza para conocer más sobre el diseño** y brindarnos una herramienta de **ayuda en el proceso de interpretación del diseño**, pero no

con el propósito de automatizar el trabajo con CSS en un *copy/paste* mágico.

Es fundamental tener en cuenta que posee “errores”, por así llamarlos, dado que **estima ciertas propiedades o valores** que deberían implementarse pero no tiene el discernimiento necesario para interpretar al diseñador como podríamos hacerlo nosotros.

Trabajo con Dev

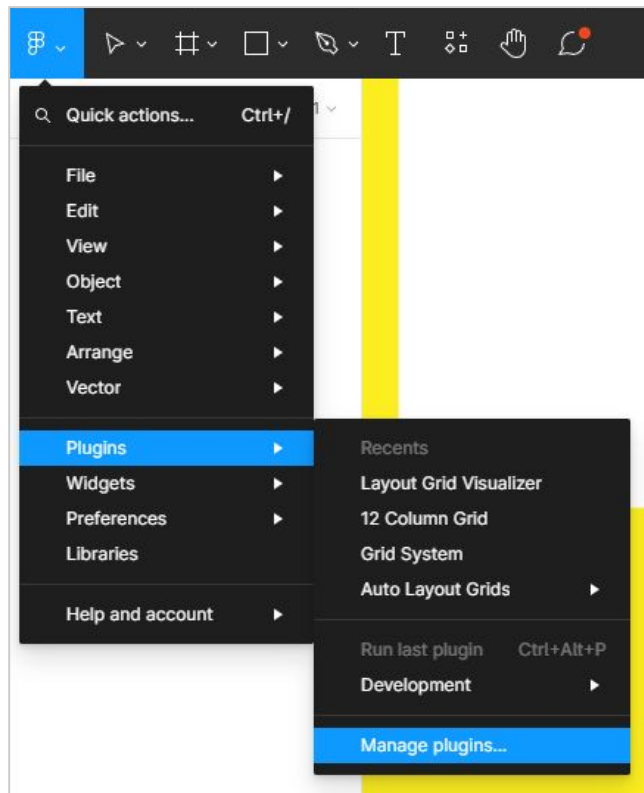


Uso de plugins en Figma

Existen *plugins* para trabajar en Figma. Solo debemos ir al menú del lado izquierdo y buscar la opción *plugins*.

En este ejemplo, mostraremos cómo buscar un *plugin* llamado **Layout Grid Visualizer**. Este plugin permite también generar grillas en nuestros diseños.

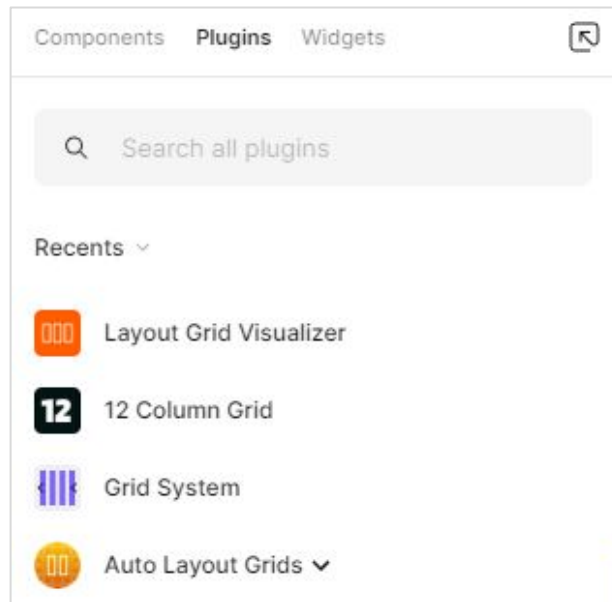
Para encontrar un *plugin*, buscaremos en el menú la opción **Plugin** y luego seleccionaremos **Manage Plugins**.



Escribiremos el nombre del *plugin* que estemos buscando, lo seleccionaremos y podremos activarlo (*run*) en nuestro Figma para utilizarlo.

Nota: revisa los tutoriales oficiales de Figma para conocer más sobre sus herramientas y usos. También, te sugerimos realizar el curso de [UI: Interfaz de Usuario](#) e [Introducción a UX](#) si deseas profundizar más en el diseño de Interfaces Web.

Link oficial: [Figma - YouTube](#)



Revisión

- Repasar los conceptos vistos teniendo en cuenta que el **rol de diseño** está asignado al **Diseñador UX/UI no al maquettador del sitio**.
- Sí es importante que el maquettador o desarrollador tenga **conocimientos de UX** pero no es el rol específico para el desarrollo de diseño de la interfaz ni su prototipado.



**¡Sigamos
trabajando!**