

ADAPTATION DES INTERFACES

TUTORIAL - BOOTSTRAP POLYMER

INSTRUCTOR: ANNE MARIE DERY & CHRISTIAN BREL
benjaminclaquin@gmail.com levansuu@polytech.unice.fr

1. Introduction

Ce document est un tutoriel, détaillant le fonctionnement des frameworks **bootstrap** et **polymer**. Il a pour objectif de permettre, à un développeur novice, d'utiliser ces outils pour réaliser un site qui servira d'exemple tout au long de ce tuto. Il doit également fournir un comparatif de ces technologies, une étude de leur adaptabilité sur plusieurs supports ainsi qu'un bilan des points positifs et négatifs de ces technologies (notamment vis à vis de l'adaptabilité).

Dans un premier temps, nous détaillerons l'exemple sur lequel nous travaillerons. En second lieu, nous fournirons un descriptif des outils bootstrap puis polymer. Par la suite, nous nous intéresserons à l'intégration de ces technologies. Enfin, nous proposerons un comparatif synthétique de ces frameworks.

[Introduction](#)

[L'exemple](#)

[Bootstrap](#)

[Introduction](#)

[Comment réaliser l'exemple ?](#)

[Comment tester sur les différents supports, quels sont les résultats.](#)

[Polymer](#)

[Introduction](#)

[Tutorial : Comment réaliser l'exemple](#)

[Comment tester sur les différents supports, quels sont les résultats](#)

[Polymer avec des composants qui embarquent bootstrap](#)

[Comment réaliser l'exemple](#)

[Comment tester sur différents supports, quels sont les résultats](#)

[Grille bootstrap ayant quelques composants Polymer](#)

[Comment réaliser l'exemple](#)

[Quels sont les résultats](#)

[Conclusion](#)

Business Name or Tagline

1920×400

First name	Name	Age
Jill	Smith	50
Eve	Jackson	94

Contact Us

Start Bootstrap
3481 Melrose Place
Beverly Hills, CA 90210

P: (123) 456-7890
E: name@example.com



Marketing Box #1

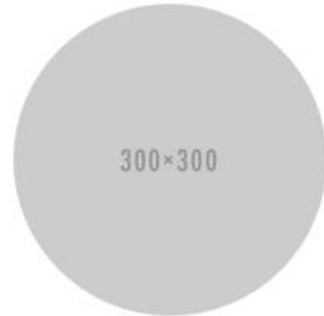
These marketing boxes are a great place to put some information. These can contain summaries of what the company does, promotional information, or anything else that is relevant to the company. These will usually be below-the-fold.

- Link 1
- Link 2
- Link 3
- Link 4
- Link 5
- Link 6
- Link 7
- Link 8
- Link 9
- Link 10
- Link 11
- Link 13
- Link 13



Marketing Box #2

The images are set to be circular and responsive. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui.



Marketing Box #3

Donec id elit non mi porta gravida at eget metus. Fusce dapibus, tellus ac cursus commodo, tortor mauris condimentum nibh, ut fermentum massa justo sit amet risus. Etiam porta sem malesuada magna mollis euismod. Donec sed odio dui.

2. L'exemple

Ci-dessus : L'interface que nous cherchons à réaliser avec plusieurs technologies différentes. (Bootstrap, Polymer, Polymer avec des composants embarquant Bootstrap, et bootstrap contenant des composants Polymer)

Cet exemple contient les éléments suivants :

- Une grande image 1920x400
- Un tableau
- Un encadré "contact us"
- 3 objets :
 - Une map issue de google maps
 - Une liste d'éléments
 - Un objet contenant une image et du texte

(Les 3 objets communiquent entre eux, l'élément de la liste sélectionné met à jour les 2 autres objets en conséquence).

Plusieurs phénomènes seront intéressants à étudier sur cette interface :

- La gestion de l'affichage de l'image sur des petits supports.
- Le redimensionnement du tableau
- L'agencement des différents objets
- La facilité ou difficulté d'intégration des objets à l'interface.

Environnement de développement

Un simple éditeur de texte type sublime text ou notepad++ est amplement suffisant. Des outils de développement et de test mobile sur chrome peuvent être une solution, cependant le mieux est de tester sur les différents supports cible. A noter que nous n'avons testé que sur **firefox**, chrome nous donnant des problèmes d'affichage.

3. Bootstrap

3.1. Introduction

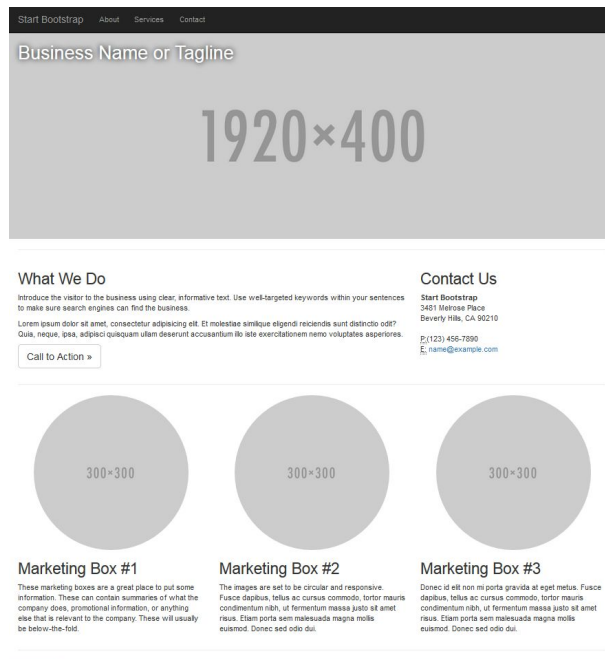
Bootstrap (anciennement Twitter BluePrint) est un framework développé dans sa première version en 2011 par Twitter pour encourager l'homogénéité dans les outils de développement interne et ainsi, faciliter la maintenance du code. Il contient un large ensemble de templates CSS, HTML et JavaScript permettant mettre facilement en forme un site ou application.

Bootstrap fonctionne à partir d'une grille découpant l'écran de manière régulière (12 cases de largeur par défaut) et permettant de placer facilement les éléments de la page dans l'espace et de réorganiser automatiquement la page lorsque l'on change de support. En effet, bootstrap est un framework responsive.



3.2. Comment réaliser l'exemple ?

L'exemple que nous avons réalisé est particulièrement facile à reproduire étant donné qu'il se base sur un des templates disponibles sur la page <http://startbootstrap.com/>, proposant de nombreux modèles de sites libre de droits. Le template contient le code html, et les librairies css et JavaScript.



Une fois le template acquis, nous remplaçons les éléments du modèle par les notres.

Les différents éléments apparaissant sur la page ci-dessus sont séparés par des balises telles que celles-ci :

```
<div class="col-sm-4"> </div>
```

Le paramètre col-sm-4 spécifie le comportement du conteneur de l'élément :

col-sm indique que l'élément tolère un redimensionnement pour des périphériques d'une résolution supérieure à 768 pixels de largeur (ce qui est supérieur à la résolution du smartphone que nous utiliserons pour les tests).

Nous remplaçons le contenu entre les balises par le contenu de notre choix :

- Le tableau : On réalise entre des balises `<table>` `<tr>` et `<th>` un tableau tout comme en HTML, en précisant `<table class="table">`, nous chargeons automatiquement la mise en forme Bootstrap pour le tableau.
- La carte Google Map :

Un très simple script JavaScript, facilement récupérable.

-
- La liste :

Tout comme pour le tableau, on utilise les balises et HTML...

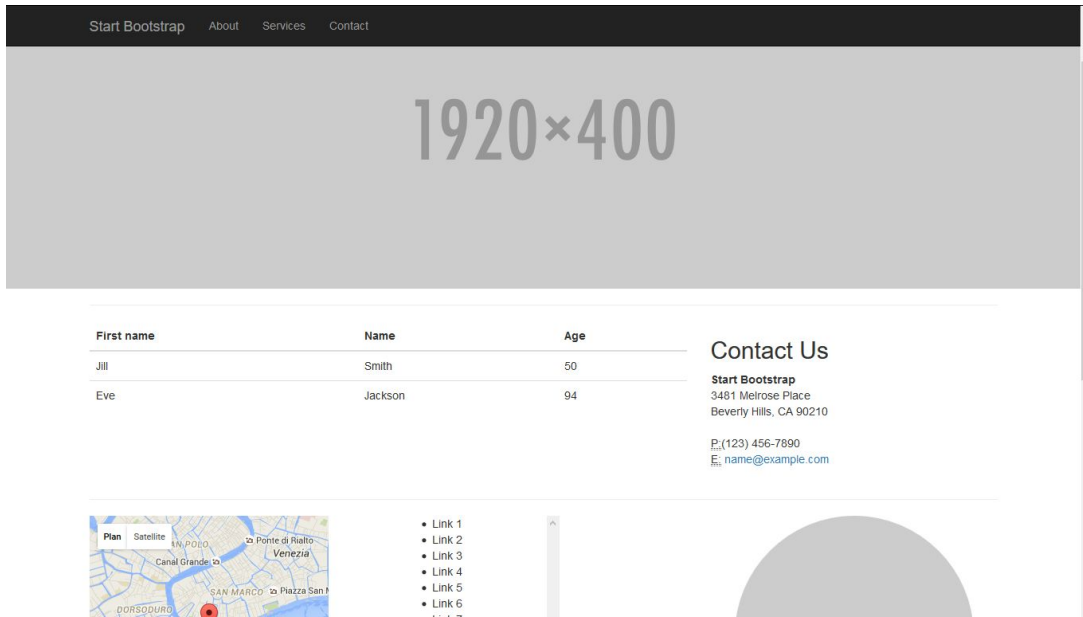
3.3. Comment tester sur les différents supports, quels sont les résultats.

Pour tester sur les différents de supports, il suffit de copier le dossier des projets (bootstrap, polymer, et bootstrap+polymer) et d'ouvrir le fichier index.html (avec ES explorateur de fichiers et Firefox installés)

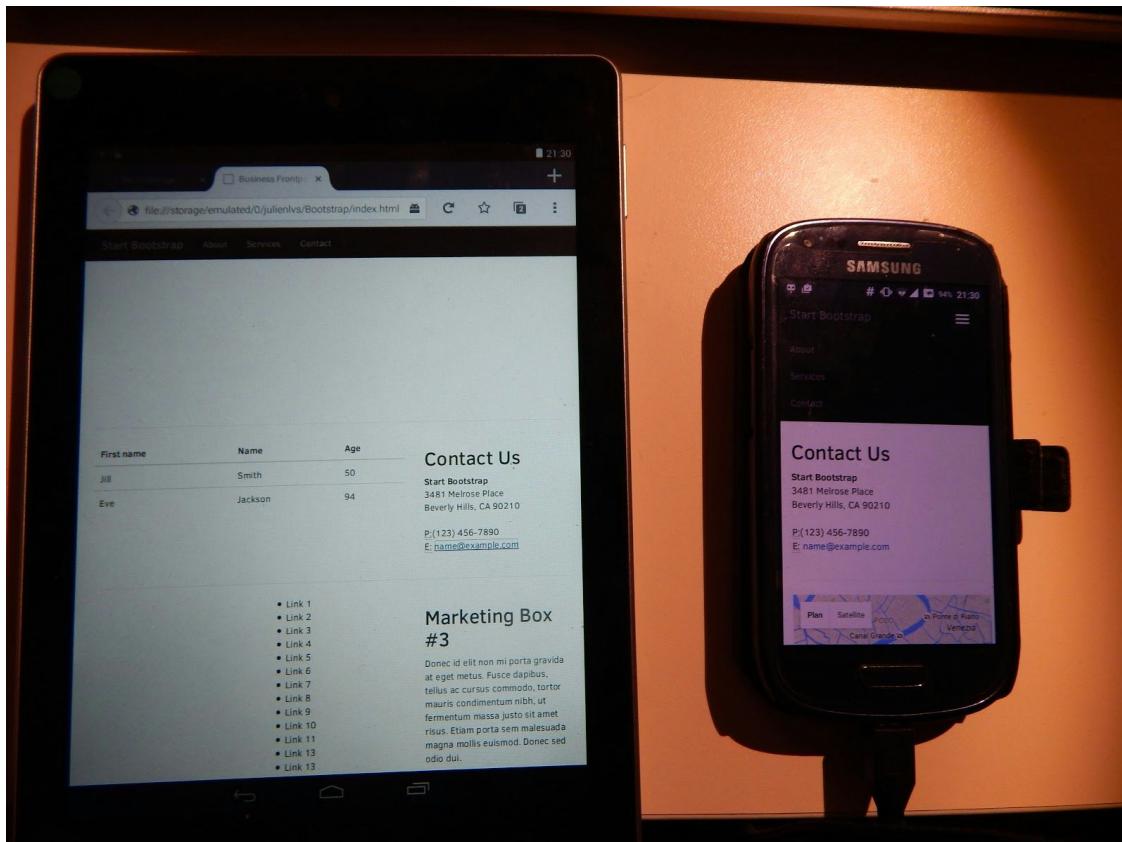
On peut constater que Bootstrap s'adapte très bien sur tous types de supports et navigateurs (la tablette avait quelques soucis pour afficher les images et la google map mais cela est du a une configuration erronée de la tablette).

En revanche, le site crée ressemble à un style "Bootstrap" , donc très classique et formaté .

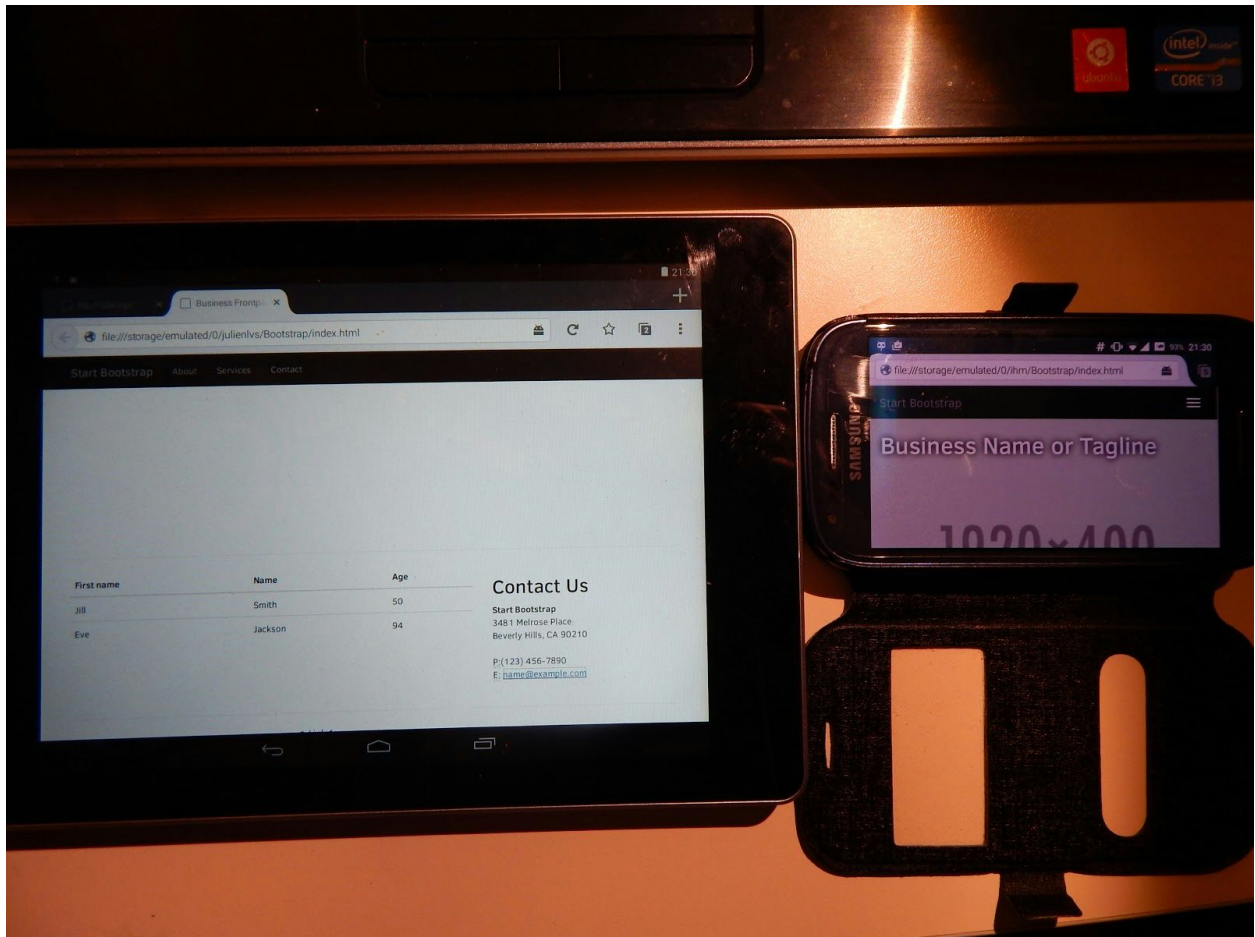
Bootstrap	Paysage	Portrait
Ordinateur	OK	OK
Tablette	OK (- Google Map)	OK (- Google Map)
Mobile	OK	OK



Ci dessus : rendu sur ordinateur



Ci dessus : rendu sur tablette et téléphone en mode portrait



ci dessus : rendu sur tablette et téléphone en mode paysage.

4. Polymer

4.1. Introduction

Polymer est un framework assez récent (environ 2 ans), et se base sur des technologies nouvelles : les webcomponents. La programmation par composants de manière générale est très "en vogue" et pourrait devenir la norme. Elle permet de s'abstraire des objets (qui sont dépendant entre eux) , et de créer des entités non interdépendantes, qui peuvent être réutilisés dans un autre contexte sans aucun effort.

4.2. Tutorial : Comment réaliser l'exemple

La documentation de Polymer étant légère, nous avons du par maintes reprises faire des déductions.

Pour installer Polymer, la prodédure est très simple. Il suffit d'installer NodeJS :

<https://nodejs.org/en/>

Puis d'installer bower : *npm -g install bower*

Ensuite, dans chaque projet, on utilise la commande : *bower install --save*

Polymer/polymer#^1.1.0

puis *bower init* (donne les paramètres initiaux du projet, les auteurs, ...)

Une fois ces opérations réalisées, on peut choisir d'installer certaines classes d'éléments polymer (iron, paper, ...) , ou certains éléments en particulier (iron-icon, google-map, ...)

Pour l'exemple choisi, nous avons eu besoin des éléments iron, des paper et de la google map : *bower install Polymer/iron-elements , ...*

Ces éléments permettent d'avoir une interface utilisateur agréable en peu de temps.

Ensuite, pour recréer la page de bootstrap, nous avons fait usage de ces composants et nous avons recrée la page de haut en bas.

Tout d'abord, nous avons mis une `<paper-toolbar>` au sommet de la page. Celle ci n'a aucun problème de redimensionnement sur téléphone. Elle peut de plus être aisément personnalisée (couleur , titre, images, ...).

Ensuite, en dessous, nous avons utilisé une `<iron-image>`. Celle ci peut se centrer, avoir une image de chargement (ce que ne permet pas bootstrap). Finalement, ce composant "image" n'apporte pas grand chose par rapport à une balise `img` standard.

Puis, nous avons crée notre propre composant `<pol-table>` . Celui ci est une démonstration de la composition. Le principe est simple, on écrit du HTML standard entre les balises et celui ci est inséré au sein du composant,entre les deux balises `<content>`. Cela peut permettra par exemple d'appliquer un style personnalisé. C'est ce qu'on a fait pour créer une table qui possède un design convenable.

Le shadow DOM, utilisé dans polymer, est un principe assez particulier : c'est un DOM qui n'est visible que d'une partie du code. Cela implique que les balises dans `<content>` ne sont pas accessible via un selecteur css classique. Il faut utiliser `::content` mabalise.

Soulignons que la composition n'est pas obligatoire : on peut ne pas avoir de balise `<content>` : dans ce cas l'utilisation du composant ressemble fort à un `include php` (division du code).

Pour voir l'interactivité de polymer, nous avons crée un composant modulaire. Le composant `<pol-contact>` permet d'avoir un composant affichant des informations de contact dont on peut changer le numéro de téléphone et le mail.

Du coté de l'appel, il suffit de mettre des paramètres : `<pol-contact class="custom" mail="mailto:mailserieux@serieux.com"></pol-contact>`

Du coté du composant, cela se fait dans son initialisation, on definit un tableau propriétés qui contient les différents "paramètres". Ensuite, on peut lire ceux ci dans le composant grâce à des doubles crochets `<code HTML {{var polymer}} code HTML >`. A

noter que tous les composants doivent avoir un nom de la forme xxxx-xxxx (tiret) pour être reconnus.

Pour finir, nous avons ajouté un composant google map et une liste d'amis (dont les données sont récupérées en ligne), qui pourraient interagir ensemble selon le principe ci dessus.

4.3. Comment tester sur les différents supports, quels sont les résultats

Pour tester sur les différents de supports, il suffit de copier le dossier des projets (bootstrap, polymer, et bootstrap+polymer) et d'ouvrir le fichier index.html (avec ES explorateur de fichiers et Firefox installés)

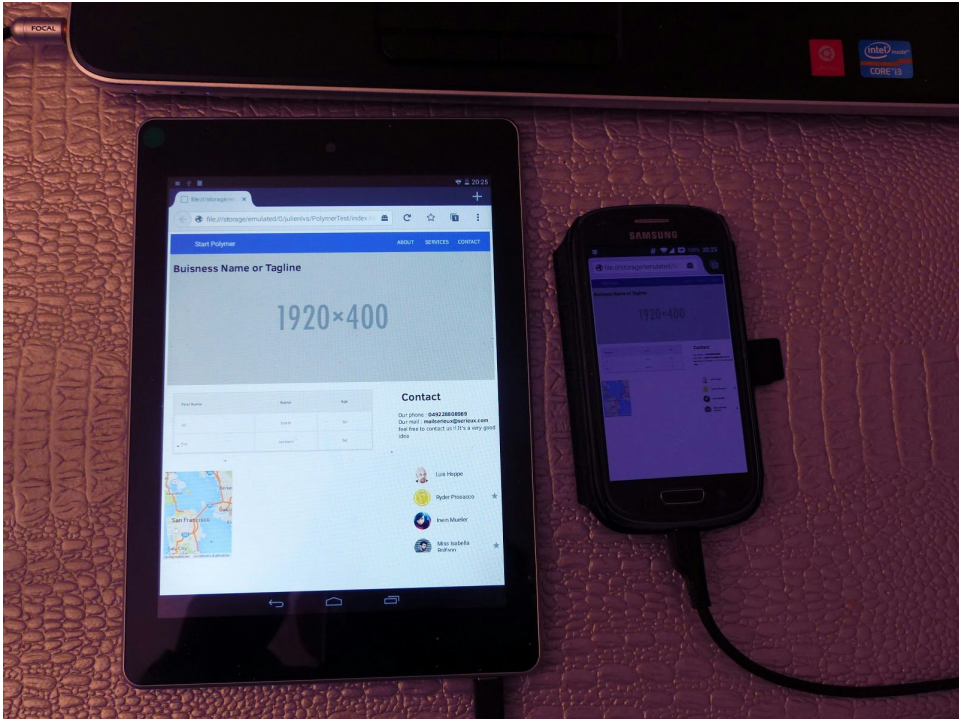
Les résultats montrent que polymer s'adapte bien moins que Bootstrap aux mobiles; une intervention à base de float été nécessaire dans tous les cas. De plus, la page est mise à l'échelle.

POLYMER	Paysage	Portrait
Ordinateur (15")	OK	N/C
Tablette (7,9")	OK (LENT)	OK (LENT)
Téléphone portable (4")	MISE A L'ECHELLE (LENT)	MISE A L'ECHELLE (LENT)

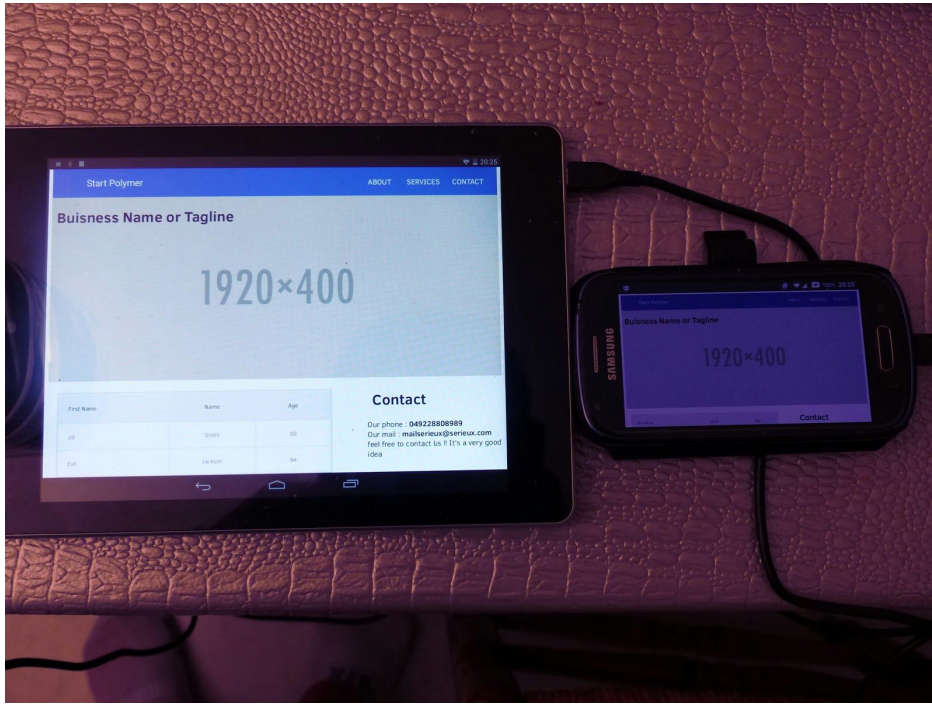
A noter que redimensionner ne donne pas du tout le même résultat que lancer directement sur mobile (le redimensionnement n'engendre pas de mise à l'échelle mais certains éléments s'adaptent).



Ci dessus Rendu sur grand écran



Ci dessus rendu sur tablette et mobile en portrait



Ci dessus rendu sur tablette et mobile en format paysage

5. Polymer avec des composants qui embarquent bootstrap

5.1. Comment réaliser l'exemple

Imposer la mise en page bootstrap à l'intérieur d'un composant polymer se fait en quelque étapes :

1. Importer les librairies bootstrap disponibles sur le site officiel.
2. Ajouter dans le composant polymer le lien vers la librairie

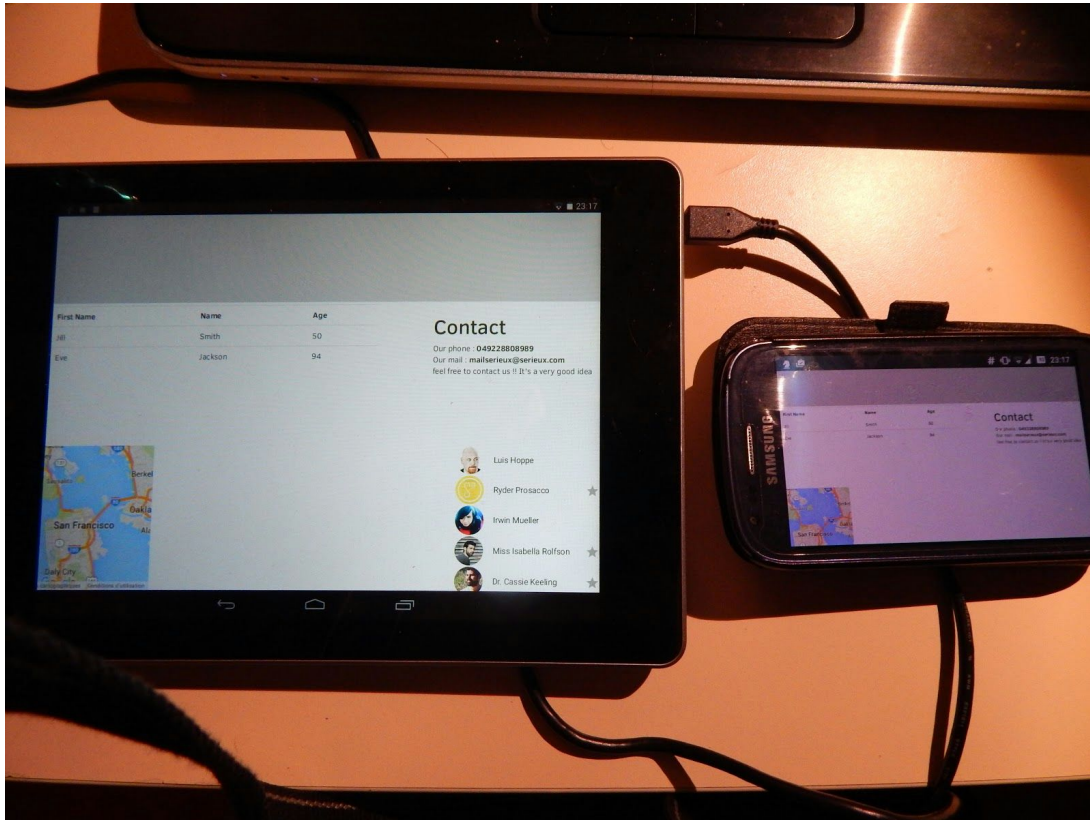
```
<link rel="stylesheet" href="../../bower_components/bootstrap/bootstrap.css">
```

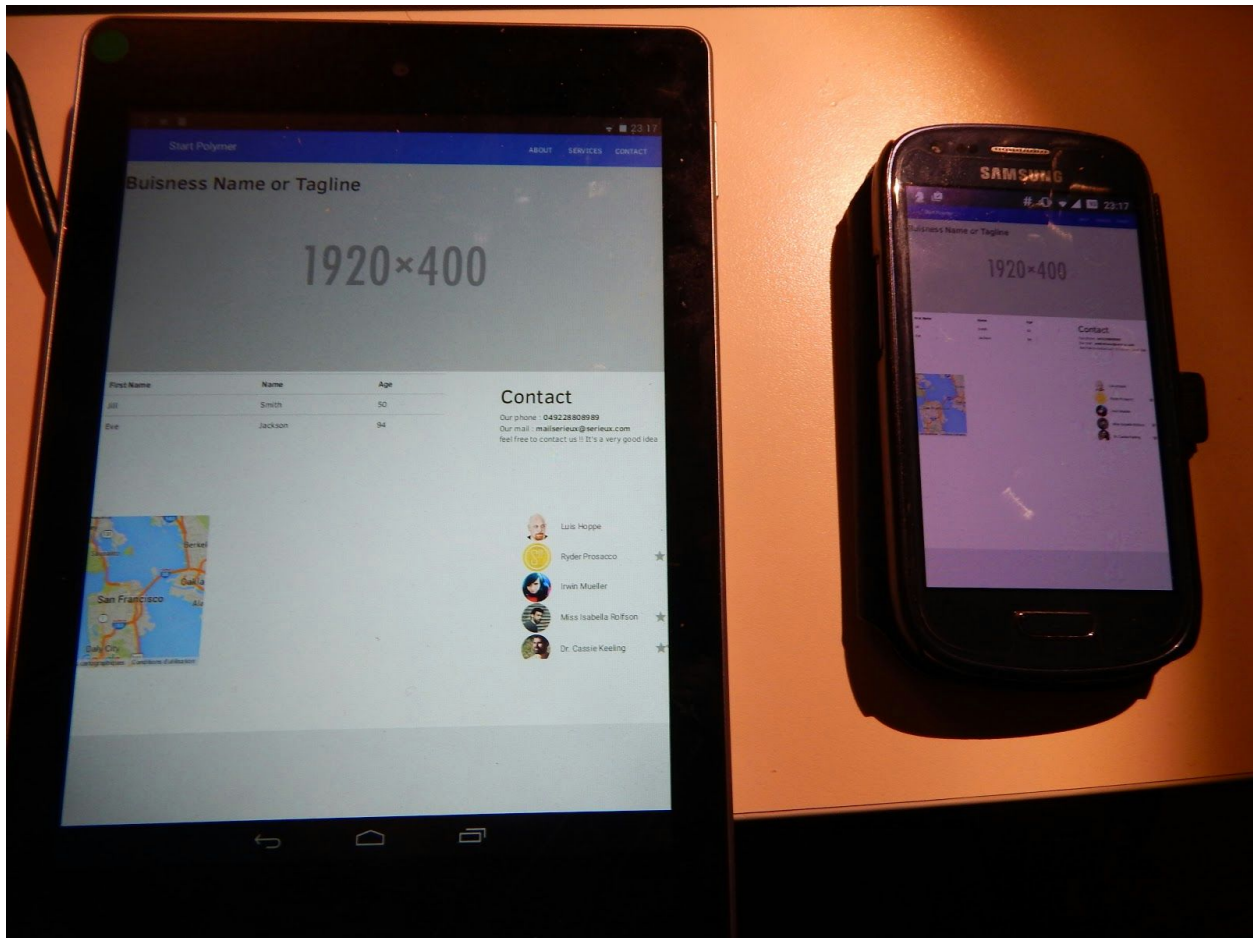
(par exemple)

3. Faire appel aux classes bootstrap dans les éléments du composant, par exemple, pour le tableau, utiliser `< table class = "table" >` appliquera la mise en forme bootstrap de tableaux.

5.2. Comment tester sur différents supports, quels sont les résultats

On obtient strictement la même adaptation que pour Polymer, cependant les tableaux ont une apparence plus agréable (tout comme les formulaires, non testés ici). Le gain paraît vraiment minime par rapport à l'effort dépensé (en termes de charge machine et humaine). Il faudrait, pour avoir l'usage de cette technique, utiliser une grille dans des composants Polymer. Or, la page web ne charge que très lentement sur téléphone : cela prouve qu'il y a une réelle lenteur apportée par le css et javascript bootstrap dans chaque composant (parfois pour un résultat nul) .





Nous nous sommes donc dit qu'il valait mieux utiliser une grille bootstrap dans laquelle on utilise des composants Polymer.

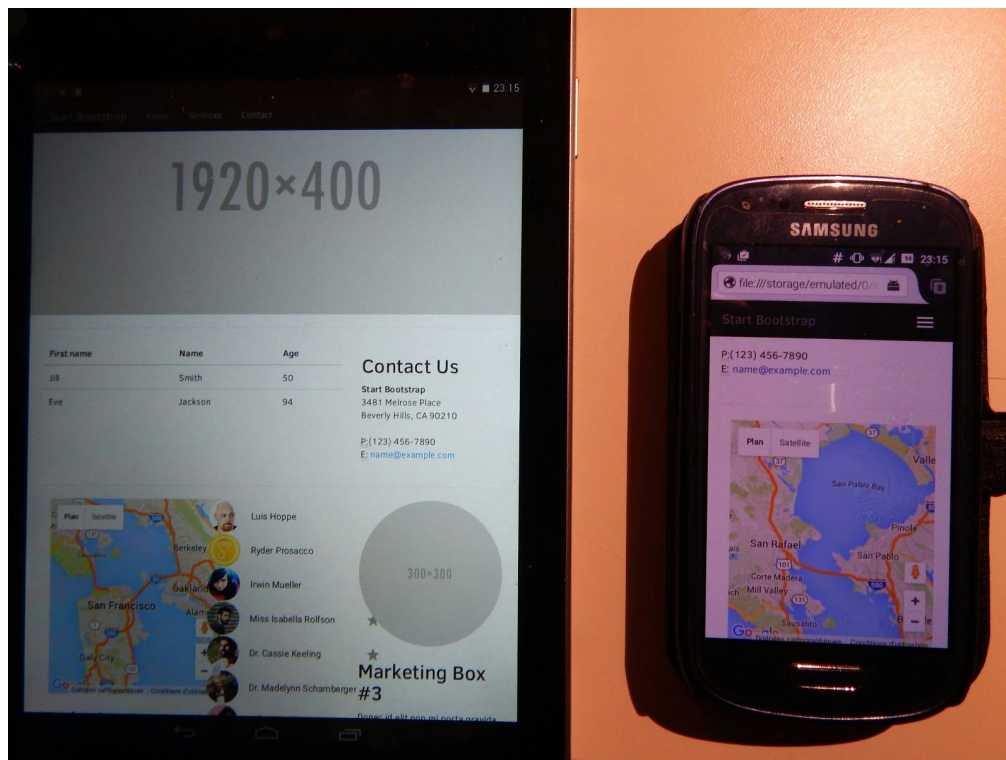
6. Grille bootstrap ayant quelques composants Polymer

6.1. Comment réaliser l'exemple

Pour réaliser cette exemple, nous avons copié collé des composants polymer dans du code ne comprenant que bootstrap a la base.

6.2. Quels sont les résultats

Les résultats sont assez bons : on peut intégrer en une balise une google map (a comparer avec le nombre de lignes dans la version full bootstrap). En revanche, comme le css de bootstrap définit certaines règles en termes de dimensionnement et de hauteur, la friend list ne s'affiche pas correctement par défaut (en mode portrait). Il faut définir un overflow, et des largeurs et hauteurs relatives pour éviter ce dépassement.



7. Conclusion

Nous pouvons constater qu'en termes d'adaptation aux interfaces pur, Bootstrap, en l'état actuel, possède beaucoup d'avantages sur Polymer. Sa simplicité d'utilisation et d'installation est d'ailleurs une de ses grandes forces, outre le fait qu'il soit bien plus adapté à des écrans réduits. Cependant, l'avenir est du côté de Polymer avec une architecture moderne et modulaire. Il permet déjà d'intégrer des composants complexes qu'on ne peut reproduire que difficilement avec bootstrap.