|  |  |
| --- | --- |
| 4D SUMMIT 2020  La Modernisation de l’Interface Facilitée  Présenté par : **Gabriel Inzirillo** | page1image27590656 |

Introduction

Les applications 4D peuvent paraître d'un autre temps du point de vue des jeunes développeurs et si nous examinons la conception de certaines applications, nous ne pouvons pas être totalement en désaccord avec ces derniers.

4D a un grand héritage qui peut être sa force mais peut aussi être sa faiblesse parfois. Nous pouvons construire de belles choses avec 4D en très peu de temps. Mais parfois, les fonctionnalités et la vitesse de développement ne suffisent pas à convaincre de nouvelles personnes d'utiliser votre produit. Les gens veulent quelque chose qui a l'air génial. Il n'y a pas d'application connue aujourd'hui qui ne soit pas entièrement remaniée de temps à autre. Nous avons de plus en plus d'applications web qui ont une interface compliquée et les gens veulent quelque chose qui a l'air moderne, même dans une application de bureau. Ce n'est pas parce qu'il s'agit d'une application professionnelle qu'elle n'a pas besoin d'avoir un look superbe.

State of today

Aujourd'hui, avec 4D, nous avons plus d'applications existantes que de nouvelles applications. Certaines existent depuis plus de 20 ans et les modifications en termes d'interface sont parfois proches de zéro. Ces applications ne survivront pas si un certain travail n'est pas effectué. Votre client peut être satisfait des fonctionnalités et aimer l'application, mais lorsque de nouvelles personnes arrivent dans l'entreprise et voient un ancien outil, elles le combattent et essaient de le remplacer. Les applications existantes doivent continuer à satisfaire les utilisateurs existants mais doivent attirer de nouveaux clients et utilisateurs à l'utiliser.

En ce sens, la modernisation de votre interface est un impératif.

Pour les nouvelles applications, il est important de commencer par une interface bien conçue afin de rendre votre application compétitive par rapport aux concurrents.

Voici quelques points que les développeurs 4D peuvent dire :

* Nous n'avons pas de temps à consacrer à la conception de nos formulaires.
  + Il faut laisser du temps aux développeurs pour concevoir leurs formulaires, ce n'est pas quelque chose qui prend 10 minutes et qui doit être fait rapidement, cela peut se faire en plusieurs étapes au cours du développement et vous ne devez pas avoir peur de repenser l'interface lorsque vous ajoutez des fonctionnalités.
* Il s'agit d'une application commerciale et n'a pas besoin d'être sexy
  + S'il s'agit d'une application commerciale, elle doit être sexy pour attirer des clients et poursuivre votre activité.
* Nous sommes des développeurs et non des designers
  + Il est important de pouvoir, en tant que développeur 4D, être capable de coder, mais aussi de construire une bonne interface. Les développeurs 4D sont multi-stack, il est très rare qu'un développeur 4D n'écrive que du code, cela existe mais c'est rare. Il est donc important de savoir comment construire une bonne interface.
* Il n'y avait rien dans les spécifications concernant le design et l'interface
  + Comme indiqué au point précédent, vous devez être capable de traduire les spécifications en une interface bien conçue qui rendra l'utilisateur final heureux.
* L'utilisateur final ne se plaint jamais du design
  + Il est trop tard lorsque l'utilisateur final commence à se plaindre du design, il cherche peut-être déjà une solution plus moderne pour remplacer votre produit.
* Les objets de formulaire 4D ne nous permettent pas de construire de beaux formulaires
  + Ce n'est pas totalement faux, 4D a un long héritage et certains objets de formulaire ne devraient plus être utilisés. Vous devez savoir comment utiliser les objets de formulaire modernes et comment construire une belle interface.

Pourquoi l’interface est importante?

L'interface est importante car, en 2020, les clients pourront aimer les fonctionnalités de votre application mais seront repoussés par un design pas très moderne.

Si votre application est déjà en place depuis de nombreuses années, les nouveaux arrivants dans l'entreprise n'aimeront peut-être pas un logiciel démodé et mettront un mauvais esprit sur votre logiciel.

Non seulement pour le design, mais aussi parce que les designs à l'ancienne ne répondent pas aux priorités d'aujourd'hui. Un formulaire comportant des dizaines de champs superposés n'aide pas l'utilisateur à se concentrer sur les informations importantes. De plus, ouvrir trop de fenêtres rend l'utilisateur confus, il est important de pouvoir naviguer via une seule fenêtre, dans la plupart des cas cela suffit.

Une interface bien conçue doit aider l'utilisateur final à se concentrer sur les informations importantes, nous ne devons pas surcharger le formulaire avec des informations qui perdront l'attention de l'utilisateur. Parfois, nous devons modifier la conception en cours de développement parce que nous recevons des commentaires de l'utilisateur final et que les spécifications n'étaient pas assez bonnes pour couvrir tous les détails afin de fournir la meilleure conception au départ.

Comment pouvons-nous vous aider?

Chez AJAR SA, nous développons une série de composants qui vous aideront à construire de jolis formulaires. Cette suite de composants est regroupée sous le nom de AJUI\_Suite.

Voici la liste des composants :

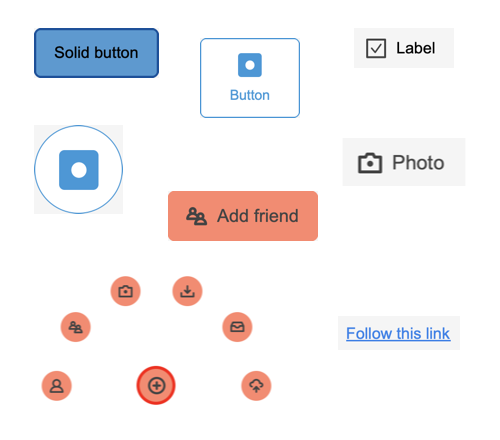
AJUI\_Button

Créer des boutons entièrement personnalisables. Contrôlez la couleur des bordures, du fond, des labels et des icônes, contrôlez le rayon et la taille des bordures et bien d'autres options pour concevoir vos boutons d'application en fonction de votre design. Vous pouvez définir un design spécifique pour 5 états : default, hover, activated, disable and focus. Ces états suivent la même logique qu'en CSS. Chaque état a les mêmes valeurs que l'état “default", vous pouvez remplacer les valeurs en les spécifiant. Par exemple, vous ne voulez qu'une couleur de fond plus claire pour l'état "on hover", vous pouvez définir uniquement la couleur de fond pour l'état "hover".

Le bouton est ensuite rendu via SVG et affiché à l'intérieur d'une variable image que vous avez dessinée dans votre formulaire. Il peut gérer tous les événements de la même manière qu'un bouton standard.

Une application Lab contient de nombreux How Do I et un labo complexe qui vous aide à créer des modèles à réutiliser facilement dans vos applications. Il montre un exemple sur la façon de rendre votre bouton tabulaire et d'utiliser la barre d'espace (ou autre touche) pour déclencher le clic.

La dernière fonctionnalité ajoutée est appelée " Floating Action Button " (FAB), elle vous permet de définir certains boutons qui ne seront affichés que lorsque vous cliquerez sur un bouton principal. Ces nouveaux types de boutons peuvent vous permettre d'économiser de l'espace dans votre formulaire pour n'afficher les boutons que lorsque cela est nécessaire.



1 Button Examples

AJUI\_Tip

Comme pour AJUI\_Button, AJUI\_Tip vous aide à gérer les tips dans votre application en contrôlant chaque aspect. Vous pouvez même aller plus loin en incluant votre propre formulaire à l'intérieur du tip pour fournir une toute nouvelle façon d'entrer les données. Certaines animations peuvent être utilisées pour forcer l'utilisateur à se concentrer sur un élément. Un effet de fondu entrant, de fondu sortant, de saut et de clignement peut être défini. Un minuteur de masquage automatique peut être réglé pour se masquer automatiquement au bout d'un certain temps.

Le tip sera créée via un objet de sous-formulaire que vous devrez mettre dans votre formulaire principal. Il sera utilisé pour créer autant de tip que vous le souhaitez. Le tip est rendue via SVG.

Un Lab est également inclus qui contient de nombreux HDI et un Lab qui vous aide à configurer les tips et à créer des modèles.



2 Tip Examples

AJUI\_Breadcrumb

AJUI\_Breadcrumb est un objet complexe qui peut contenir différentes sections individuelles (étapes), chaque section peut avoir son propre type. 5 types différents peuvent être utilisés (standard, current, first, next and previous) chaque type a 4 états (default, hover, active and disable). Grâce à ces configurations, vous pouvez créer des breadcrumbs complexes qui peuvent couvrir de nombreuses situations différentes.

Nous avons 3 modèles de breadcrumb disponibles :

Simple

Il s'agit d'un simple breadcrumb avec un séparateur comme texte.

A close up of a logo

Description automatically generated

3 Simple Breadcrumb

Arrow

Une flèche sera utilisée comme séparateur.



4 Arrow Breadcrumb

Grouped buttons

Les boutons groupés sont spéciaux et peuvent être utilisés pour se comporter comme une sorte de bouton radio avec un rendu plus avancé. Il affectera automatiquement le type “current" à la section sélectionnée.

A screenshot of a cell phone

Description automatically generated

5 Grouped Buttons

Un Lab est également inclus qui contient de nombreux HDI et un Lab qui vous aide à configurer des breadcrumbs et à créer des modèles.

AJUI\_Banner

AJUI\_Banner vous aidera à dessiner 2 types de bannières dans votre formulaire.

### Window Banner

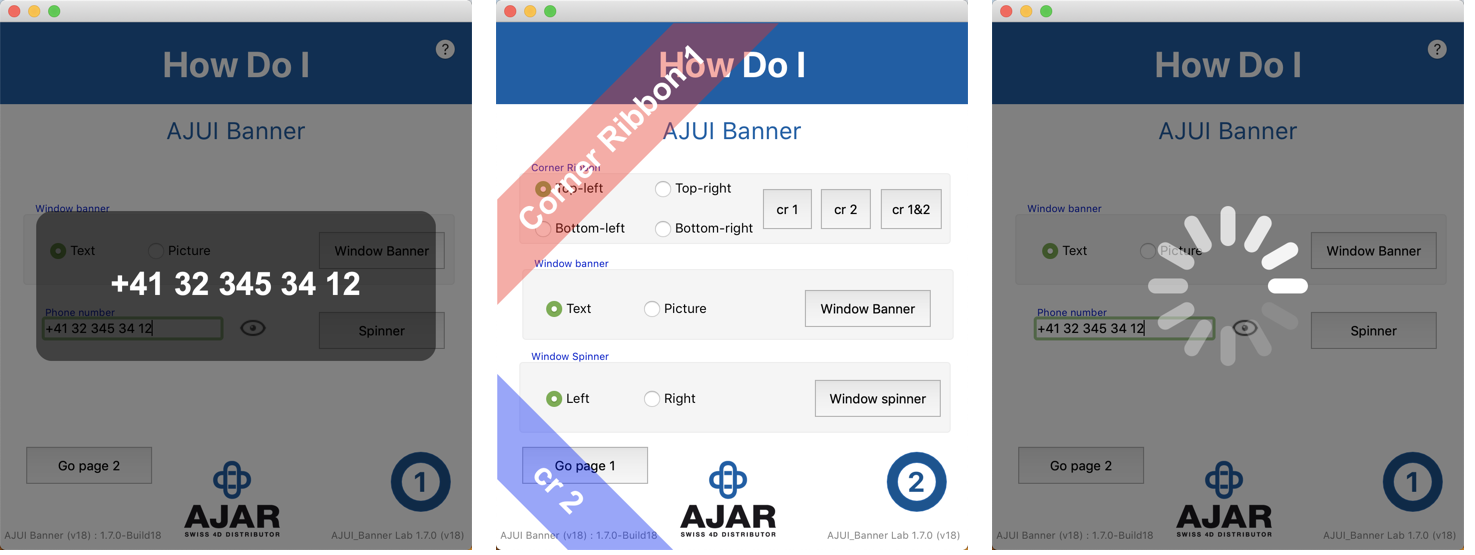
Une bannière de fenêtre couvrira votre formulaire avec une certaine opacité, et vous pourrez afficher un message au centre. Ceci est utile pour attirer l'attention sur une information spécifique

### Corner Ribbon.

Un ruban de coin sera un ruban affiché dans un coin de votre formulaire, il peut également être utile pour attirer l'attention de l'utilisateur sur quelque chose d'important.

### Spinner

Affichez un spinner pour montrer que quelque chose se passe en arrière-plan.

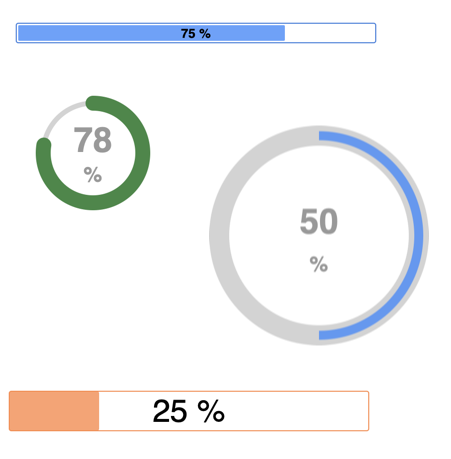


6 Banner Examples

AJUI\_Progressbar

Fournit tous les outils pour dessiner une barre de progression entièrement personnalisable. Deux types de barre de progression sont disponibles, linéaire et circulaire.

Comme les composants précédents, elle est rendue via SVG à l'intérieur d'une variable d'image.



7 Progressbar Examples

AJUI\_FloatingLabel

Floating label est un petit composant qui vous permettra d'afficher un label uniquement lorsque les données sont déjà saisies dans un champ de formulaire. Si le champ n'est pas encore saisi, le label sera un placeholder.

Lorsque la valeur est saisie, le placeholder n'est plus visible et le label est donc affiché.

Toutes les applications Lab sont disponibles et vous pouvez consulter le code pour voir comment les HDI sont mis en œuvre.

## Utilisation de la notation à point

Grâce à la notation objet, nous avons conçu tous nos composants pour travailler de manière orientée objet. Chaque type d'élément sera créé grâce à une méthode "New AJUI\_..." qui retournera un objet contenant toutes les propriétés et formules disponibles. Vous pourrez alors manipuler et afficher votre élément via l'API disponible.

Voici un exemple avec l'élément AJUI\_Button.

**Case of**

**:** (**Form event code**=On Load)

**Form**.btn3:=***New AJUI\_Button***

**Form**.btn3.***Name***("btn3")

*//default*

**Form**.btn3.***BGColor***(AJUI\_btn\_default;"lightgrey")

**Form**.btn3.***Label***(AJUI\_btn\_default;"BTN 3")

**Form**.btn3.***BorderSize***(AJUI\_btn\_default;2)

*//hover*

**Form**.btn3.***BGColor***(AJUI\_btn\_hover;"darkgrey")

**Form**.btn3.***Label***(AJUI\_btn\_hover;"On Hover")

*//active*

**Form**.btn3.***BGColor***(AJUI\_btn\_active;"grey")

**Form**.btn3.***Label***(AJUI\_btn\_active;"On Click")

**End case**

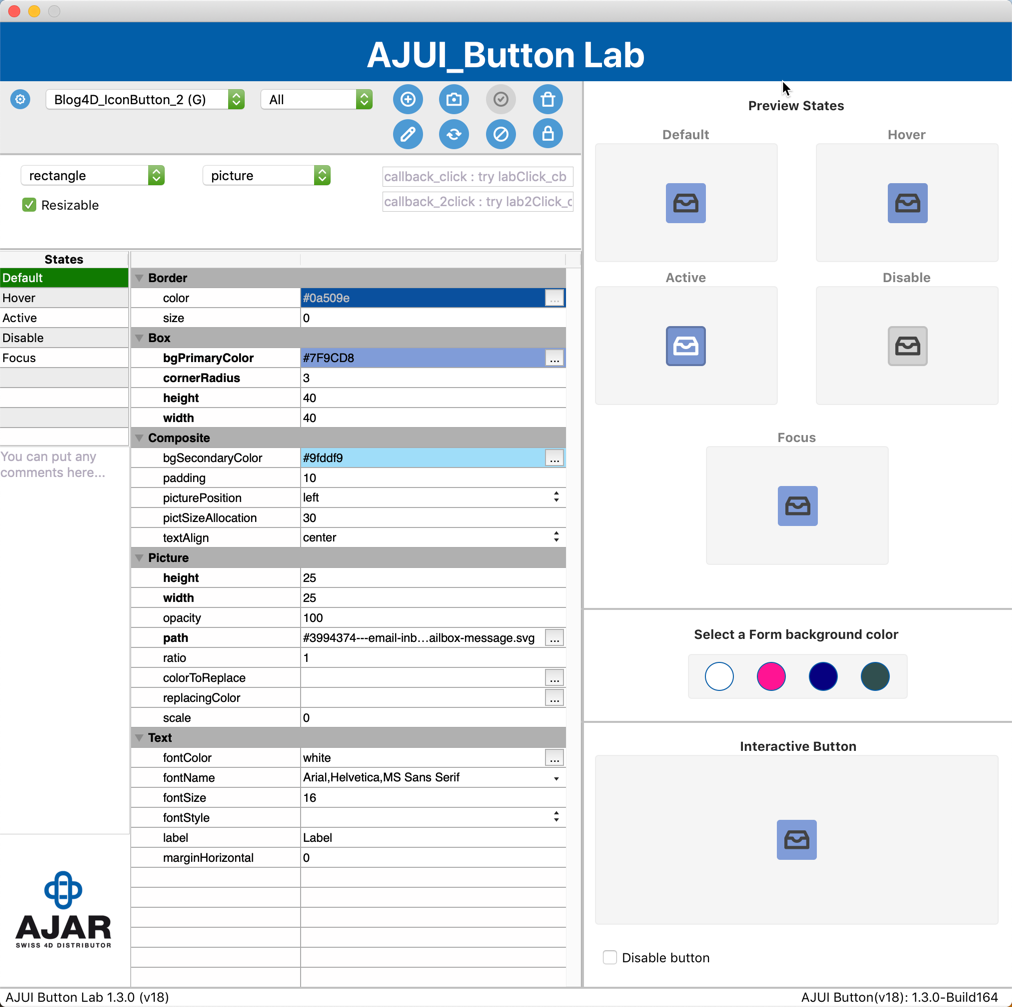
**Form**.btn3.***Draw***()

Vous pouvez voir que nous stockons notre objet bouton dans "Form.btn3" et que nous définissons certaines propriétés via des formules. Nous fournissons des formules car il est plus facile à utiliser que de passer par plusieurs niveaux de propriétés, certaines formules effectuent des tâches et des contrôles plus lourds.

## Templates

AJUI\_Tip, AJUI\_Button et AJUI\_Breadcrumb Labs offrent une interface sophistiquée qui vous aide à créer votre design avec un retour d'information direct. Vous pouvez enregistrer des modèles et les importer directement dans votre application. Il est très utile de ne pas avoir à définir chaque propriété par code, car certains designs peuvent devenir complexes. Le modèle sera enregistré en json et vous pourrez le placer dans le dossier par défaut ou dans votre dossier spécifique dans les ressources de votre application. Si vous utilisez des images dans votre modèle, un dossier sera créé avec toutes les images et le modèle. Le modèle va enregistrer le chemin de l'image pour qu'il soit relatif à l'emplacement du modèle dans votre application finale.

Vous pouvez enregistrer chaque modèle sur lequel vous travaillez pour le réutiliser plus tard. Grâce à la séparation du laboratoire et du composant, vous pourrez simplement glisser et déposer le nouveau modèle dans votre application pour modifier instantanément tous les boutons de votre application.



8 AJUI\_Button's Lab

Demonstration

Pendant la démonstration, nous prenons un exemple de base de données avec quelques formulaires existants :

## OldForm

Ce formulaire est un vieux "faux" formulaire qui fonctionne très bien et n'est pas si laid que ça, mais nous examinons ensemble comment nous pouvons l'améliorer en utilisant les différents composants de la suite AJ et comment les utiliser.

## Main Explorer

Voici un autre exemple d'explorateur construit avec la suite AJ.

## Pay Station

Cet exemple montre une borne de paiement dans un magasin. Il tente d'imiter ce que nous avons dans le magasin d'aujourd'hui, où vous pouvez scanner vos propres articles.