Email Validation App

Interface web permettant la validation d’une adresse email face à de multiples critères.

<https://github.com/LaurentSignorel/EmailValidationApp>

Contents

[Critères : 2](#_Toc55551750)

[Présentation de l’application 3](#_Toc55551751)

[Technologies choisies 3](#_Toc55551752)

[Index (HomeController) 3](#_Toc55551753)

[Index (EmailValidationController) 4](#_Toc55551754)

[Choix de conception 6](#_Toc55551755)

[Expression régulière 6](#_Toc55551756)

# Critères :

* La syntaxe de l’email doit répondre à la norme suivante :   
  [identifiant@[sous-domaines.]domaine.extension](mailto:identifiant@[sous-domaines.]domaine.extension)
* L'identifiant doit commencer par une lettre.
* L’identifiant doit contenir plus de lettres que de chiffres.
* Il ne peut y avoir plus de 3 sous-domaines.
* Le domaine doit appartenir à la France ou un de ses pays limitrophes (Italie ,Suisse ,Andorre ,Belgique ,Espagne ,Luxembourg ,Allemagne ,Monaco).
* La vérification du code pays se fait avec la réponse à l’appel de l’api : <https://freegeoip.app/>

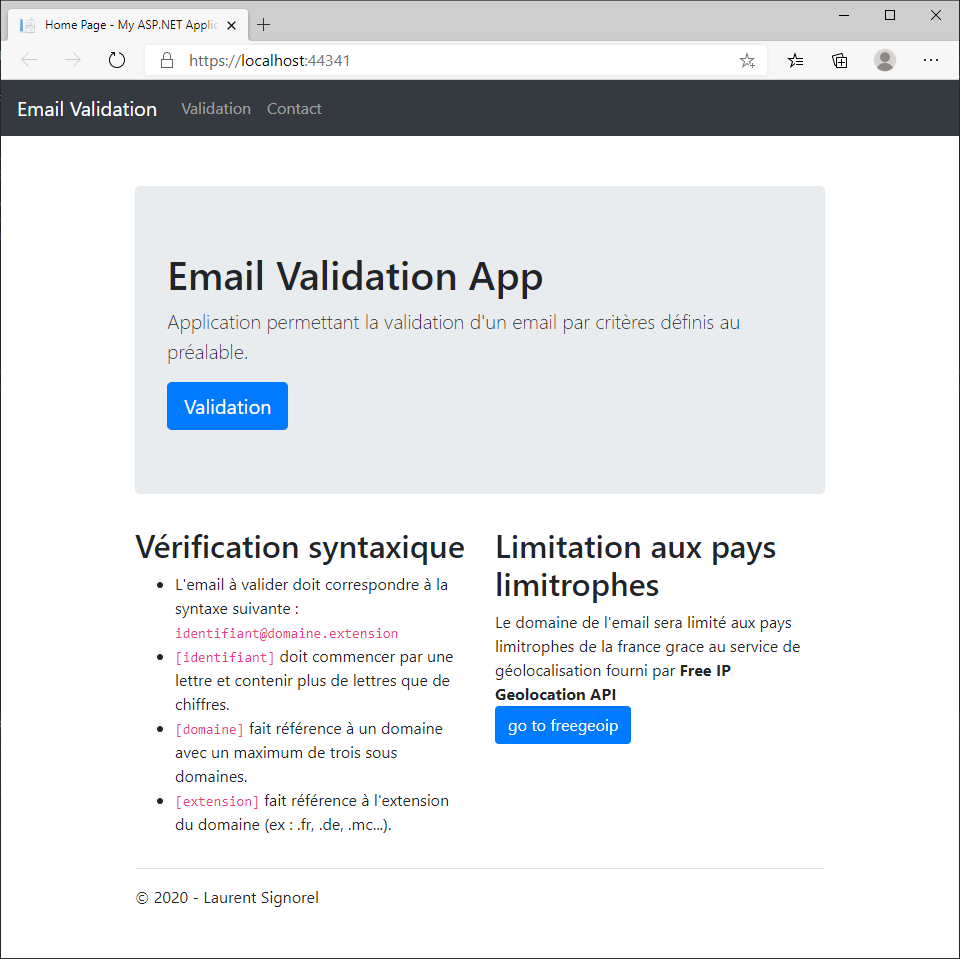
# Présentation de l’application

## Technologies choisies

* C#
* ASP.NET MVC
* Ninject
* Log4Net
* MsTest

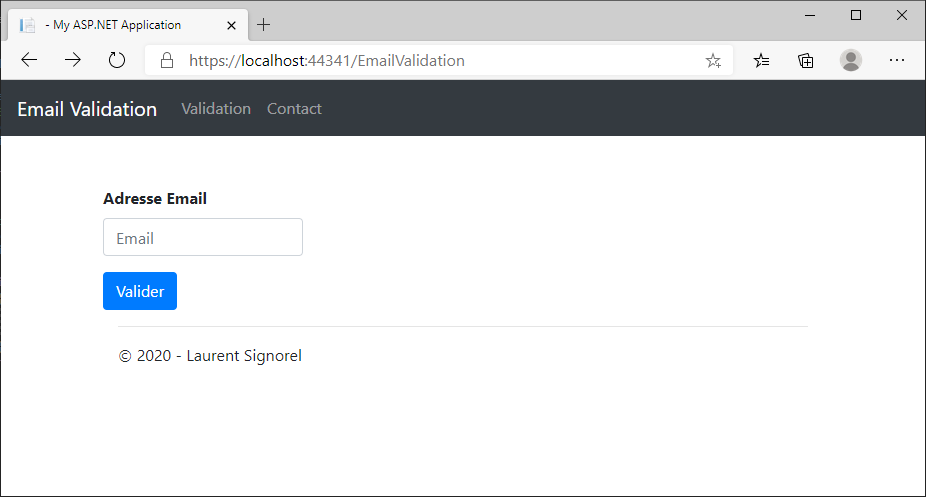
## Index (HomeController)

Page de démarrage de l’application récapitulant les critères à appliquer pour valider l’email.

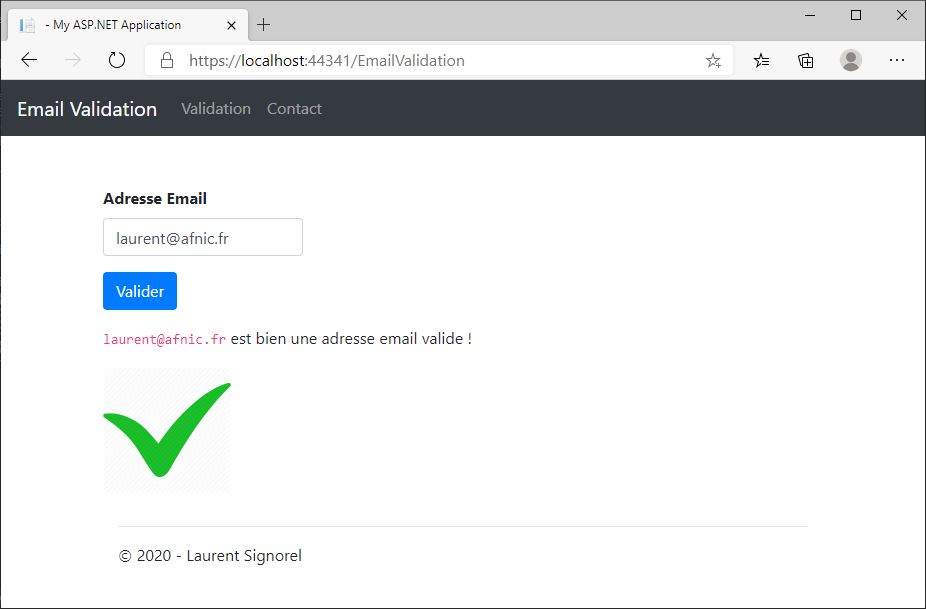


## Index (EmailValidationController)

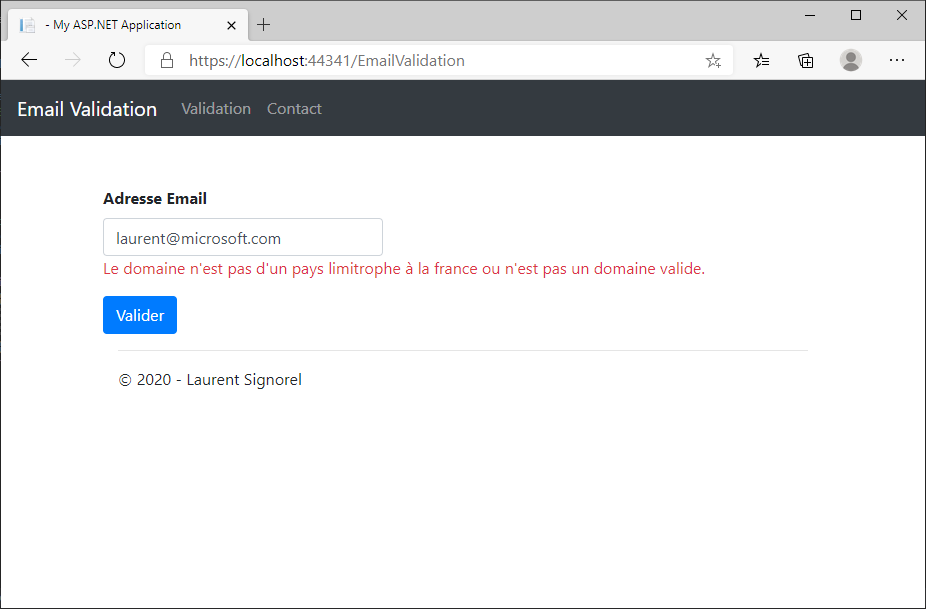
Formulaire de saisie d’un email à valider.



Une saisie valide affiche correctement le mail.

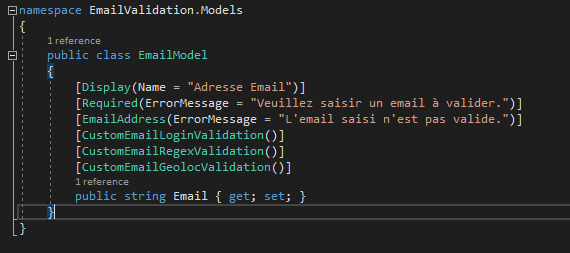


Une saisie invalide renvoie une information en fonction de la validation échouée.



## Choix de conception

Afin d’appliquer les critères à la validation, le choix à été fait de se reposer sur des attributs personnalisés.



CustomEmailLoginValidationAttribute se charge de vérifier que l’identifiant de l’email contient plus de lettres que de chiffres.

CustomEmailRegexValidationAttribute se charge de vérifier la validité de la syntaxe de l’email et de la limitation à trois sous-domaines pour le domaine.

CustomEmailGeolocValidationAttribute se charge de valider auprès de l’api <https://freegeoip.app/> la présence du domaine dans un des pays autorisés.

L’api teste un nom de domaine en résolvant son IP, donc un domaine dont l’extension appartient à code pays valide, peut se voir invalidé par l’application, dans le cas ou le domaine est hébergé dans un pays autre que ceux autorisés. Ex : [email@outlook.fr](mailto:email@outlook.fr) ne sera pas valide car le domaine n’est pas reconnu comme ayant une IP dans les pays autorisés.



### Expression régulière

L’expression régulière utilisée est la suivante :

^((?:[a-zA-Z]{1})(?:[a-zA-Z0-9-\_.]{0,61})(?:[a-zA-Z0-9]{1}))@(?'domain'(?:[a-zA-Z0-9][a-zA-Z0-9-\.][a-zA-Z0-9]{1,63}){1,4})(?'tld'(?:\.[a-zA-Z]{1,63}){1})$

Elle se découpe en 3 groupes de captures, dont deux nommés.

* Groupe 1 : ((?:[a-zA-Z]{1})(?:[a-zA-Z0-9-\_.]{0,61})(?:[a-zA-Z0-9]{1}))  
  Ce groupe contenant trois sous-groupes non capturant, permet de valider l’identifiant de l’email.
* Entre les groupes 1 et 2 la regex teste la présence du caractère @.
* Groupe 2 : (?'domain'(?:[a-zA-Z0-9][a-zA-Z0-9-\.][a-zA-Z0-9]{1,63}){1,4})  
  Ce groupe permet de valider le domaine et d’empêcher de saisir plus de trois sous-domaines. De plus le groupe est nommé, afin de pouvoir réutiliser la valeur extraite dans la validation par GeoIP.
* Groupe 3 : (?'tld'(?:\.[a-zA-Z]{1,63}){1})  
  Ce groupe permet de valider l’extension du domaine. Il est aussi nommé afin de réutiliser la valeur dans la validation par GeoIp.