

Rapport de projet

Architecture n-tiers

Picture-In-The-Box

Picture In The Box


Email

Password

Connexion


Inscription ▾


GALLERIE D'IMAGE





CONTACT

INFORMATIONS

 Mardi - Mardi : 18:30 à 18:31

 25 Rue Dr Rémy Annino, 42000 Saint-Étienne

 04 77 91 58 88

 contact@pictureinthebox.com

LAISSEZ NOUS VOTRE MESSAGE

Nom

Email

Telephone

Message

Samed KARDJILOV
Maxime MARCHET
Marion ROCHER
Aurélien TOUZET

Sommaire

I. Présentation de l'application

II. Readme du projet

III. Descriptions des fonctionnalités

I. Présentation de l'application

Notre application se présente sous la forme d'une application web. Le but de notre application est de permettre à chaque utilisateur de stocker des photos, et de les partager avec les personnes de leur choix. De plus les photos peuvent avoir des labels afin de mieux les classer, et enfin chaque utilisateur peut créer son arborescence de dossiers dans lesquels il pourra ranger ses images.

Les images publiques sont accessibles à tous les utilisateurs de l'application à travers la fonction recherche. Pour les images privées, seuls les personnes invitées par l'utilisateur postant les images pourront les voir.

La page d'accueil permet de se connecter et de créer un compte. Ensuite une fois connecté, on peut ajouter des images et les classer dans des dossiers.

La page Photos Personnelles permet d'ajouter de nouvelles photos et dossiers, ainsi que de les gérer. On peut déplacer les images dans les dossiers en drag & drop. Les images ont la propriété "public" ou "privé" du dossier dans lequel elles sont. On peut gérer les propriétés du dossier dans lequel on se trouve sur cette page.

La page Photos Partagées permet d'afficher toutes les photos qui sont partagées avec un utilisateur, il peut donc les visualiser ici.

La page Photo Publiques permet d'afficher toutes les photos classées "publiques" et de rechercher par mot clef des images, ou par nom d'utilisateur.

II. Readme du projet

Install

- Install nodejs v4.1.1
- Install mongodb v3.0.7
- run the command: `$npm install`
- launch mongodb: `$mongod`
- run the command: `$node createClient.js`, it will install the client used for the authentication
- create a folder named ***photos*** in the root folder where app.js is

Endpoints

Signup

POST /signup

body :

- email
- password

Login

POST /authenticate

body :

- grant_type : password OU refresh_token
- client_id : webClient
- client_secret : secret
- username : email de l'utilisateur
- password : password

Return :

- access_token
- refresh_token
- expires_in
- token_type

Profile

GET /profile

params :

- access_token

Return :

- email
- password
- albumRoot
- albums

Albums

GET /albums

params :

- access_token

Return :

- List of albums

GET /albums/{id}

params :

- access_token

Return :

- nom
- description
- tags
- owner : User id
- parentAlbum : Album id
- permissions : User id

GET /albums/{id}/content**params :**

- access_token

Return :

- current : current album specified by {id}
- albums : list of children albums
- photos : list of photos contained in that album

POST /albums/**params :**

- access_token

body :

- nom : string
- description : string
- tags : string
- parentAlbum : Album id
- permissions : list of User id (not mandatory)

Return :

- nom
- description
- tags
- owner : User id
- parentAlbum : Album id
- permissions : User id

PUT /albums/{id}**params :**

- access_token

body :

Any of these

- nom : string
- description : string
- tags : string
- parentAlbum : Album id
- permissions : list of User id

Return :

- nom
- description
- tags
- owner : User id
- parentAlbum : Album id
- permissions : User id

DELETE /albums/{id}**params :**

- access_token

Return :

- nom
- description
- tags
- owner : User id
- parentAlbum : Album id
- permissions : User id

Upload**POST /upload**

enctype : multipart/form-data

params :

- access_token

body :

- files[] : files to upload, must be of type image
- tags : string
- album : Album id => parent album

Return :

- tags : string
- uri : string
- url : string
- urlThumb : string
- private : boolean
- owner : User id
- album : Album id

Photos**GET /photos****params :**

- access_token

Return :

- list of Photos

GET /photos/{id}**params :**

- access_token

Return :

- tags : string
- uri : string
- url : string
- urlThumb : string
- private : boolean
- owner : User id
- album : Album id

GET /photos/{id}/img**params :**

- access_token

Return :

- image file specified in the Photo uri

PUT /photos/{id}**params :**

- access_token

body :

- tags : string
- album : Album id

Return :

- tags : string
- uri : string
- owner : User id
- album : Album id

DELETE /photos/{id}**params :**

- access_token

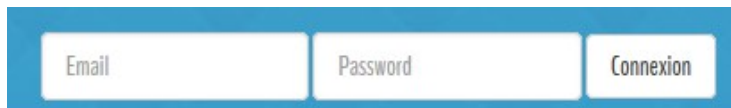
Return :

- tags : string
- uri : string
- owner : User id
- album : Album id

III. Descriptions des fonctionnalités

L'application propose diverses fonctionnalités listées ci-dessous :

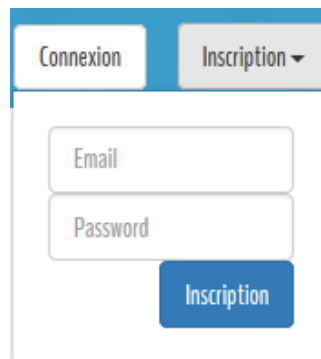
- Un bouton de connexion :

A horizontal login form with a blue border. It contains three white input fields: 'Email', 'Password', and 'Connexion'.

L'utilisateur ayant déjà un compte doit saisir son email et son mot de passe pour se connecter.

- Un bouton d'inscription :

L'utilisateur entre un email et un mot de passe qui lui serviront d'identifiants par la suite.

A registration form with a blue border. At the top, there are two buttons: 'Connexion' and 'Inscription' with a dropdown arrow. Below them are two white input fields: 'Email' and 'Password'. At the bottom right is a blue button labeled 'Inscription'.

- Une galerie des images publiques sur la page d'accueil :

GALLERIE D'IMAGE

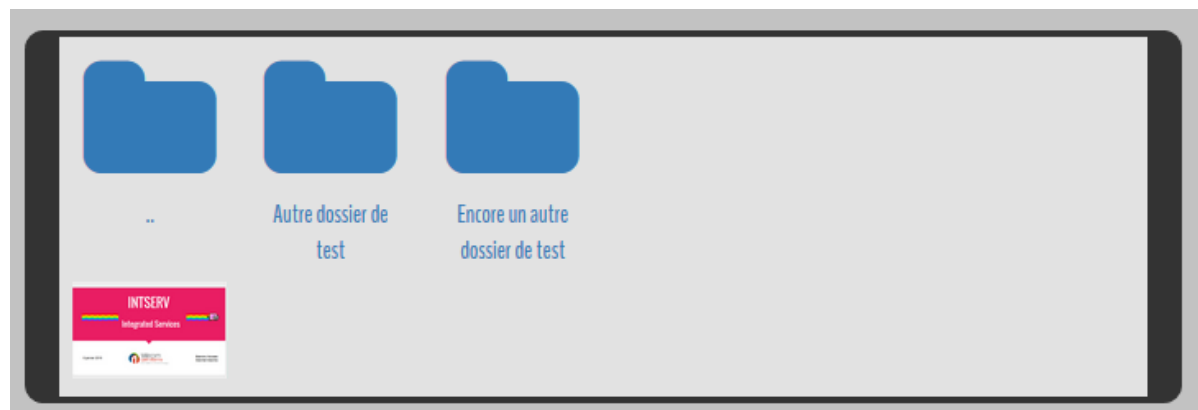


- Une page Photos Personnelles :

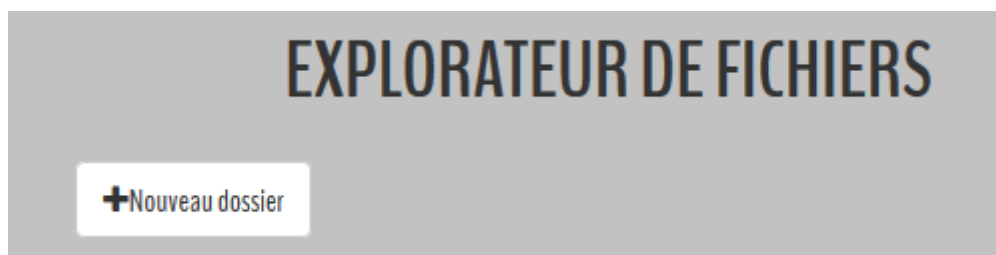
- un breadcrumb



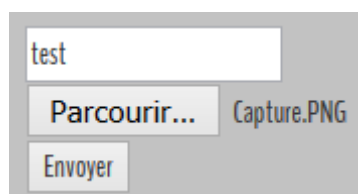
- Une zone de drag & drop explorateur de fichiers :



- Un bouton pour ajouter un nouveau dossier :



- Un bouton pour ajouter une image :



- Des champs de gestion du dossier dans lequel on se trouve :

Propriétés du dossier :

Nom du dossier :

Dossier de test

Description :

Ajoutez votre description

Partager avec :

Faudra modifier ici le défile

Visibilité :

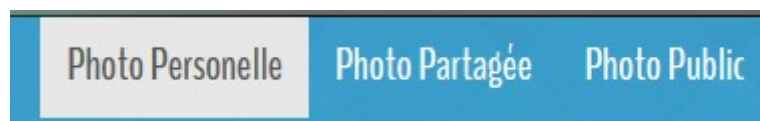
Public

Sauvegarder

- Un bouton de deconnexion :



- Une barre de navigation pour les onglets :

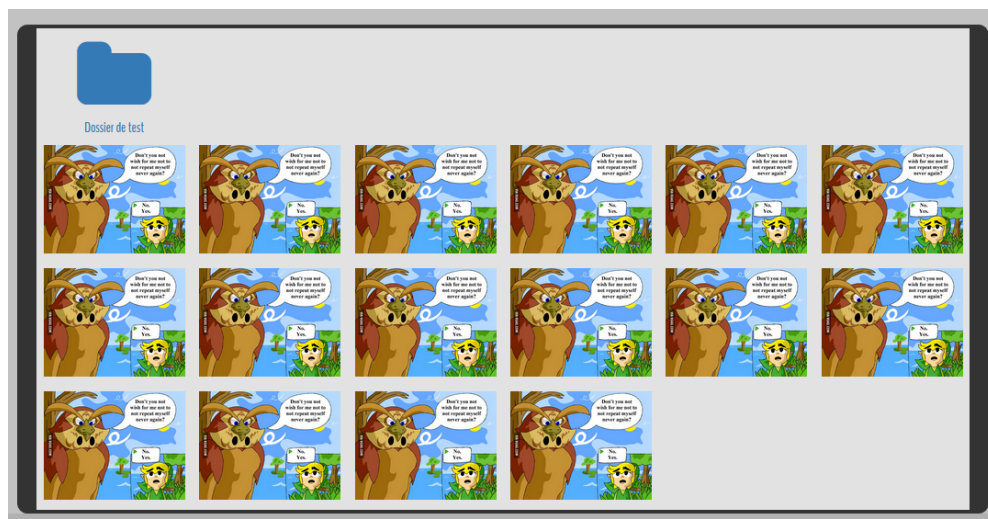


- Une page Photos Partagées :

- Un breadcrumb :



- Un explorateur de fichiers :



- Un bouton de deconnexion :



- Une barre de navigation pour les onglets :



- Une page Photos Publiques :

- Des champs de recherche :



- Des miniatures des images publiques sous forme de galerie :



- Une galerie des images :



- Un bouton de deconnexion :



- Une barre de navigation pour les onglets :

