Rapport projet « Toilettes »

Programmation mobile et web services

Table des matières

Sujet et but de l'application

Toilettes est une application mobile faite avec le framework Ionic 2 qui interroge une API que nous avons créée à l'aide de l'environnement de développement NodeJS et de divers modules.

Le but de l'application est de permettre aux utilisateurs de trouver les toilettes les plus proches de sa localisation.

Notice de déploiement

Installation client

Pour que que l'application, il faut que le serveur soit lancé.

1. Lancer la commande 'npm install' dans le dossier Toilettes du dossier ionic_project
2. Lancer la commande 'ionic build'.
3. Lancer ensuite la commande : **ionic plugin add cordova-plugin-googlemaps --variable API_KEY_FOR_ANDROID="AIzaSyDE99utD1l0leasTivb7AuNw_Qk1DzSY2c " --variable API_KEY_FOR_IOS="AIzaSyDE99utD1l0leasTivb7AuNw_Qk1DzSY2c "**
4. Lancer la commande 'ionic run android' ou 'ionic run ios'

Installation serveur (API)

Tout d'abord, il faut que NodeJS soit installé puis il faut suivre les étapes suivantes :

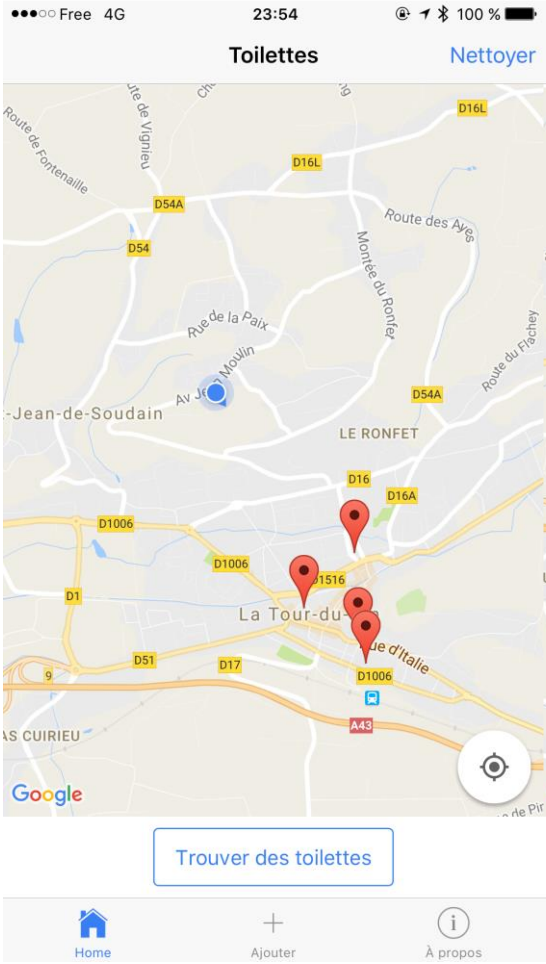
1. Cloner ou télécharger le projet https://github.com/AhmetCelikbas/Toilettes.git
2. Ouvrir une fenêtre de commandes à la racine du dossier nodejs_api
3. Lancer la commande 'npm install' pour récupérer les dépendances du projet
4. Lancer la commande 'sequelize db:migrate' pour créer la base de données
5. Enfin lancer la commande 'node app.js' pour démarrer le serveur

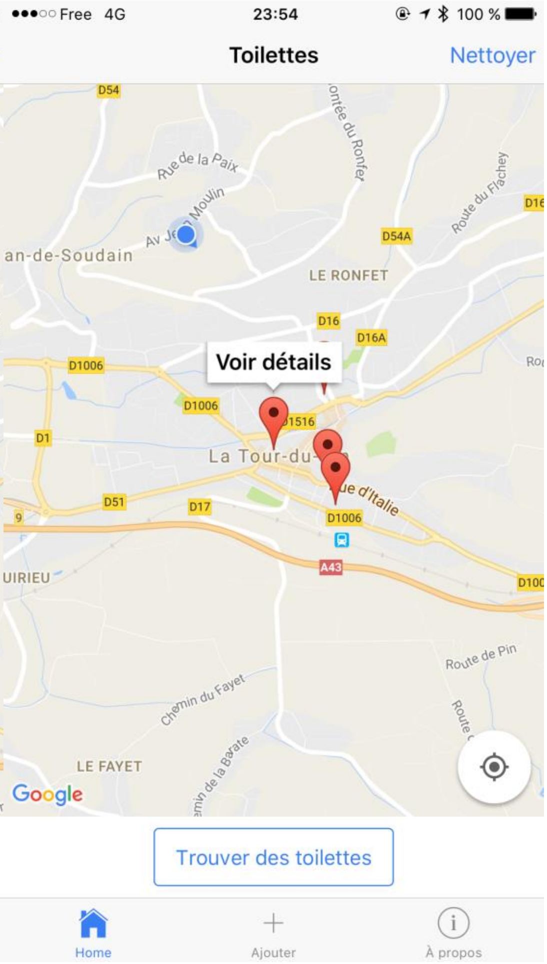
Guide d’utilisation

L'application est simple d'utilisation, elle dispose de trois onglets : Home, Ajouter et About

Home montre les toilettes proches de l'utilisateur sur une carte, Ajouter permet d'ajouter des nouvelles toilettes sur la position gps de l'utilisateur et About permet de se connecter ou modifier son compte

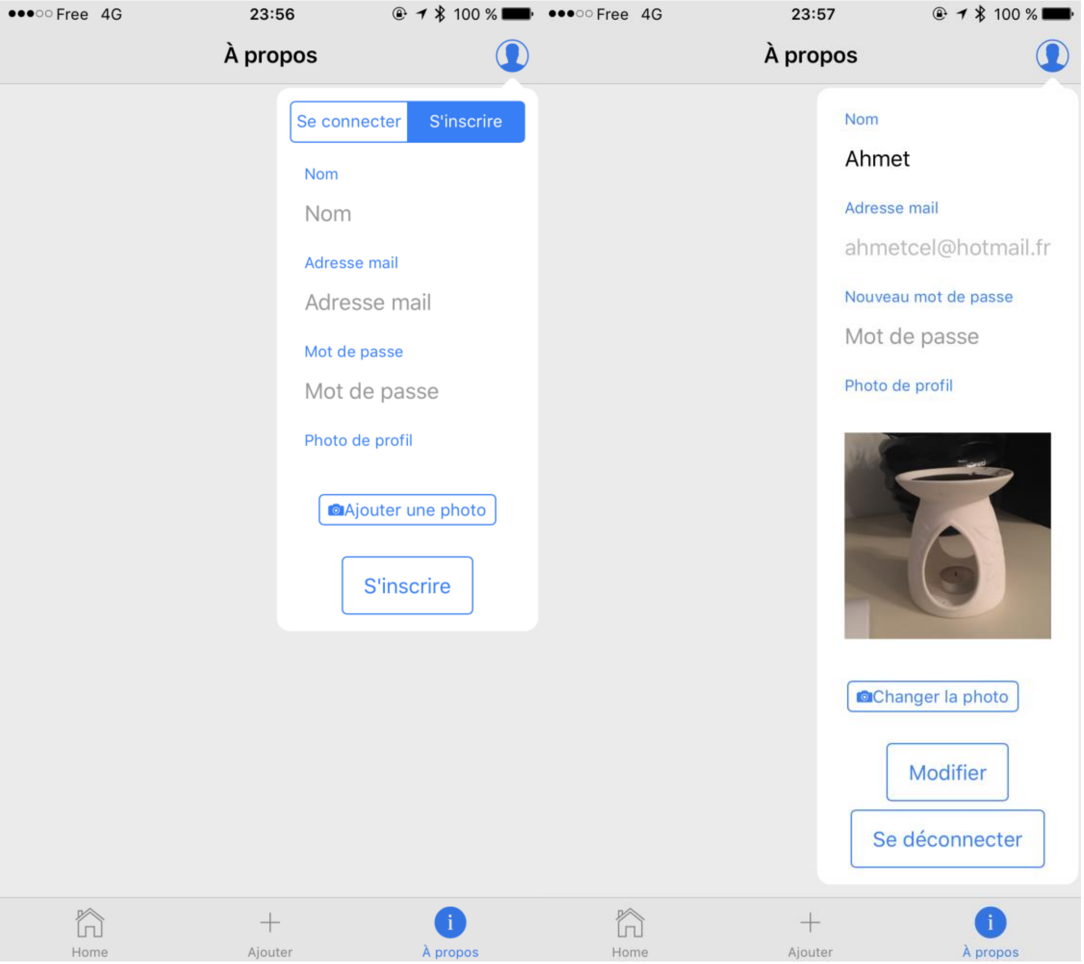
Sur l'onglet homme cliquer sur le bouton trouver des toilettes. Chaque marqueur correspond à des toilettes. Le point bleu indique la position de l'utilisateur. Cliquer sur un marqueur pour sélectionner les toilettes correspondant au marqueur puis cliquer sur 'Voir détails' pour afficher les détails de ces toilettes.





Rendez-vous ensuite dans l'onglet About puis cliquer sur l'icone 'profile' en haut a droite.

Sélectionnez s'inscrire et renseignez le formulaire en y ajoutant une photo si vous le désirez. Cliquez ensuite sur le bouton s'inscrire. Lorsque vous cliquerez à nouveau sur l'icone profile vous pourrez modifier votre compte pour changer votre nom, votre mot de passe ou encore votre photo.



Nom: Nom

Adresse mail: Adresse mail

Mot de passe: Mot de passe

Photo de profil

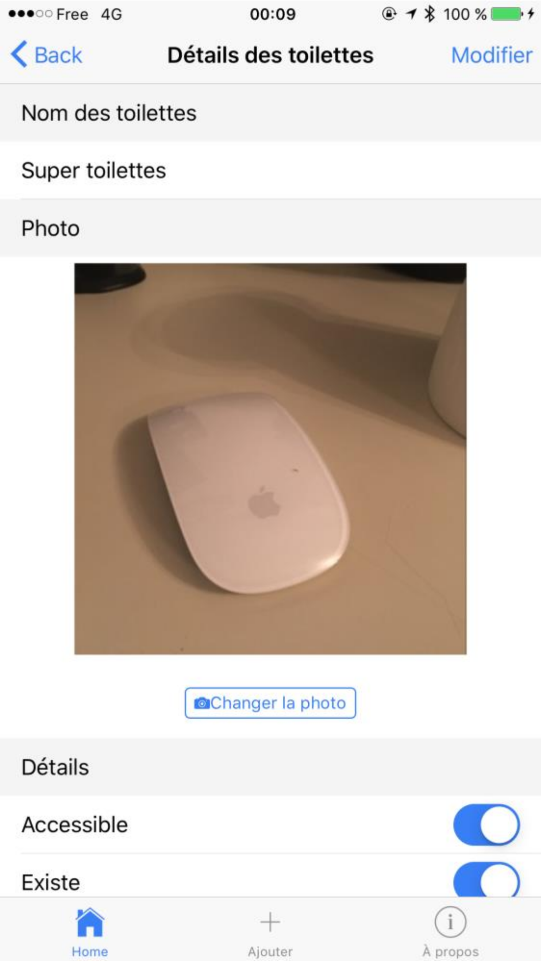
Nom: Ahmet

Adresse mail: ahmetcel@hotmail.fr

Nouveau mot de passe: Mot de passe

Photo de profil

Une fois connecté avec votre compte vous pourrez modifier des détails de toilettes ou encore ajouter de nouvelles toilettes avec l'onglet Ajouter



Détails des toilettes

Nom des toilettes

Super toilettes

Photo

Détails

Accessible

Existe

Documentation de l'API

API prefix : "/v1/api"

Get all toilets around user position

```
GET /toilets/:southWestLat/:southWestLon/:northEastLat/:northEastLon
```

Parameters

Name	Type	Description
southWestLat southWestLon	real	Location of the bottom left of the bounding box where are all toilets around our position
northEastLat northEastLon	real	Location of the top right of the bounding box where are all toilets around our position

Response

```
Status: 200 OK
```

```
[
  {
    "id": 1,
    "id_osm": 1966868061,
    "id_user": 1,
    "lat": 45.1974288,
    "lng": 5.776482,
```

```
    "Details": null,
    "Comment": []
```

```
  },
  {
    "id": 2,
    "id_osm": 773538141,
    "id_user": 1,
    "lat": 45.1991028,
    "lng": 5.7663749,
```

```
    "createdAt": "2017-04-12T21:31:36.869Z",
    "updatedAt": "2017-04-12T21:31:36.869Z",
    "Details": {
      "id_toilet": 1,
      "name": "",
      "access": true,
      "exist": true,
      "fee": false,
      "male": false,
      "female": false,
      "wheelchair": false,
      "drinking_water": false,
      "place_type": "restaurant",
      "address": "23 Chemin de l'Île d'Amour, 38240 Meylan, France"
      "createdAt": "2017-04-12T21:31:43.607Z",
      "updatedAt": "2017-04-12T21:31:43.607Z",
    },
    "Comment": [
      {
        "id": 1,
        "id_toilet": 3,
        "id_user": 1,
        "comment": "Mon premier commentaire.",
        "rating": 3
      },
      {
        "id": 2,
        "id_toilet": 3,
        "id_user": 2,
        "comment": "Un autre commentaire.",
        "rating": 4
      }
    ]
  }
]
```

Create a new toilet

```
POST /toilets
```

Header: Authorization : token

Input

Name	Type	Description
id_osm	int	The identifier of a toilet get from overpass API.
lat	double	Latitude of toilet location.
lng	double	Longitude of toilet location.
Object details		
name	string	The user's email.
access	boolean	Indicates whether the toilet is accessible
exist	boolean	Indicates whether it exists
fee	boolean	Indicates whether it's free
male	boolean	Indicates whether it's for male
female	boolean	Indicates whether it's for female
wheelchair	boolean	Indicates whether it's accessible for wheelchair
drinking_water	boolean	Indicates whether you can drink water
place_type	string	Indicates type of toilet, eg: 'restaurant'

Response

```
Status: 200 OK
```

```
{
  "id": 3,
  "id_osm": null,
  "id_user": 1,
  "lat": 45.1974276,
  "lng": 5.736442,
  "Details": null
}
```

Get a specific toilet

```
GET /toilet/:id
```

Response

```
Status: 200 OK
```

```
{
  "id": 2,
  "id_osm": 773538141,
  "id_user": 1,
  "lat": 45.1991028,
  "lng": 5.7663749,
  "createdAt": "2017-04-12T21:31:36.869Z",
  "updatedAt": "2017-04-12T21:31:36.869Z",
  "Details": {
    "id_toilet": 1,
    "name": "",
    "access": true,
    "exist": true,
    "fee": false,
    "male": false,
    "female": false,
    "wheelchair": false,
    "drinking_water": false,
    "place_type": "restaurant",
    "address": "23 Chemin de l'Île d'Amour, 38240 Meylan, France"
    "createdAt": "2017-04-12T21:31:43.607Z",
    "updatedAt": "2017-04-12T21:31:43.607Z",
  }
}
```

Update a specific toilet

```
PUT /toilet/:id
```

Header: Authorization : token

Input

Name	Type	Description
name	string	The user's email.
access	boolean	Indicates whether the toilet is accessible
exist	boolean	Indicates whether it exists
fee	boolean	Indicates whether it's free
male	boolean	Indicates whether it's for male
female	boolean	Indicates whether it's for female
wheelchair	boolean	Indicates whether it's accessible for wheelchair
drinking_water	boolean	Indicates whether you can drink water

Response

```
Status: 200 OK
```

```
{
  "id": 2,
  "id_osm": 773538141,
  "id_user": 1,
  "lat": 45.1991028,
  "lng": 5.7663749,
  "createdAt": "2017-04-12T21:31:36.869Z",
  "updatedAt": "2017-04-12T21:41:12.165Z",
  "Details": {
    "id_toilet": 1,
    "name": "",
    "access": true,
    "exist": true,
    "fee": false,
    "male": true,
    "female": true,
    "wheelchair": true,
    "drinking_water": false,
    "place_type": "restaurant",
    "address": "23 Chemin de l'Île d'Amour, 38240 Meylan, France"
    "createdAt": "2017-04-12T21:31:43.607Z",
    "updatedAt": "2017-04-12T21:41:14.632Z",
  }
}
```

Get a specific toilet's picture

```
GET /toilet/:id/picture
```

Response

```
Status: 200 OK
```

```
{
```

```
  "picture":
"data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD/2wBDAAMCAgMCAgMDAwMEAwMEBQgFBQQE
BQoHBwYIDAoMDAsKCwsNDhIQDQ4RDgsLEBYQERMUFRUVDA8XGBYUGBIUFRT/2wBDAQMEBAUEBQkFBQkUDQs
NFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBT/wgARCADcANwDAS
IAAhEBAxEB/8QAHQABAAEFAQEBAAAAAAAAAAAAAAYCAwQFBwgBCf/EABsBAAEFAQEAAAAAAAAAAAAAAAADB
AUGBwIB/9oADAMBAAIQAxAAAAH1SAAFu55/DSem/wAzP0yD6AAAAAAAAAAABRXQFYAAt/nP7agEI55J7b5n
01DoLE0AAAAAAAAADC58wU6dRRW/TrAAEDtY2Tjljo6JCprosME81ACxEmvczj0SyqfI7DC+q698/wAHiOz
1GB9B+kdDvnfDU0QyBdMsyewbqSQ2ZbPWqw/SHwOXY2u32J2a7l0JFL7cwMA8llWntyCNrMKnIjGQ6yYHNN
1bo7y17B2bUINo/sOr7v7lGS2EEuvk/wCbdI1evXBaGSLyjjMaruczkfUsrn8oQzrn3BfQnHtJhPQ+fbvZz
NRyiSW59HMkW5anXAdJtRk8/hXH3MU5BY6mHh8q7hHMBbrf9P5L1vTq7cFnj7fCZTz2p2HPubvxvW3HsO35
K7iu67jFJXyusrxeawLYWm490wM7Hp2fdJ+6nbbtVQ9OWa7YRTM7Tm4lKGkyiYSbWL9KlGRe60orol2VYDi
kK9Bc+plv33A++Y1EPOU3yNxpc/P9fsGS1/mMukB46Bi1zZpzDpGu1y+LE00fGPQfG6zL6S7ayqfPdW35ql
LDrxRXQFbE0CXsqhGBYpsl8zdTsM8mLVGc86BP1iZcbV41+ZKMNdO/dxYw/wCe5IFPdhriM5/J49zq6b2Nm
9t9EMbJ1ujgCiugINqtvhZjNXhTpQWPSmP7ulx1pNxg77ruvTbnWot9kiOaotIcTTSJPzVSrWYVljYVclMK
cyOd1aHdYt8EE7GgCiugMLQS82755ruqfK485Nh9Q1bB5zCcU/K85rFcdAGLqZAV8wc449Dk5vpp15R1FGj
3N5N9JWNv1MO6+AKK6ArAAABh856RzehyuxGeTIAAAAA57PeU2RL0vdhsy1qrh74AKK6ArAAABZ5j1TmtQk
Mk+ZfPfWt066kqc8z1iaIvfS9kNMd1nXuZsY1064spkNErAAAKK6ArAAAA1e0cnIor6H5TTpqBdC0vavOtB
c3i4wnM4z3JEPvOtr0ZAoWQ5D6Z450/tHci4wwAAKK6ArWwXFsFxbBcWwXI5v3HvN+mWyXtxbOObi2C5r8x
4cF6nss+GXyltNIXFsFxbBco+Uh//8QAKBAAAQQCAQMFAAMBAQAAAAAAAwECBAUABiAQERMHEhQVMCMxQDQ
h/9oACAEBAAEFAuLnIxG+pJLTfP8AIvL1b2hauq18iivv8i8XuRjdoti7RsdJ6eSo/wDlXjvMg7Ne1fT4uu
COnvf/AIZMoUMM+xmyGCI0w142r/Nb5Hb5bj8D30MLn3kl+Om2Dfb2kuk9U9TPTVhxEh2lm+Y2ex/a/ek1E
ZdPTXh0HUjG8aq6aZEGBx4WJVVrLTkcEWiTeLBsf0eDhZKg7V/SGlMgXXaQ5zcXGQqAXsXYgXjlrAbJ1TV
9T0Gzp0ZJnHLHAz7hzdwmqGG+UwOV4xJFz9nhJ8q8v6lGP6Bl8UVuHj9jecG+p5SzHpR+tFZrVgkLRPRtD
c/RVlYYvWGN6ZGs+OoHdbWsPYYnO7WbJxfvO2Ax76yZ1tD+Aya92IsRW6O9xAi0g9tXaTgIl3TEpnkLPWd8
KOsNoRfkIT3XyoJXhkaGbrYdWYQgqAvq/BzhE5tKXMwbHkuf0ScgGq5ndqo5Os1PUI5LWCOtpJbCrOqdI
0cq3zDBnVPOt4zbHQJl3PVSbm2TYqhQ==
RyV6Nh3HCXKHBjt8so+KvbHSwsx1mNMdZvwtkRrYdNMt2r1e9Btvql+0U+o6kzWQP22kj2TtpqGtq70d8XL
63WsE6XKI6gvCkkZLd4k4W9ihrR9m7HTDPxVV3WKA9gWs10EF2L13yxWPWavOIyXe1CW8AjHCeKK5+em/wD
CTNsGrbDKtji22SBeePVyfmV3XaIvgn9XP9q1+sEkYELI4+i9fUMDvJrIVLcZvOtL88Wtvc3T691da5Ngis
I7tRN7qqmDVN6UJfEbrcQPsq8b/IzAAfMmVtNGq28F63NWO5raiGKADDgZJCeOSCegZ7rDlIc+O8JmSA9b2
H8G1yC/xWvFeF8wIliShzQ5IiimCiwxQhdZMhIw1nGkqAcQ7xt8bKqV9dK67BB+dWjIhRlf4k4rh5IYrZF+
BMIabLwMQIFkRXeUb/IzjG880+SIoZSRQ/HjbWP31GqX63ELLCyBWBsbI9uuSE8go/v8HBcvyoc6SkWVwP7
/AB/YmTFsDLkF7yDwX8Fl1lRmTI469sWN9uWRV+VZb8CMss9RSCq04rmytciqJil6ma54okn5QZENh8SrXu
xqMbk2MskQrcqP+0XPnmKqf1hPJGNYV458cswIY+nW0XyclyZCFPCTX0bj40+Pj7AQc+3iuwtgR+QirFsOM
iMKUz6ONkeGKL1/vNEuFgSvU8deLIcw1fL1bYw7LV8V4mq4cjHaxVuy01mAGvY9CN/IFBGBlhDfXTqbRLi6
zUtGHq5OK85ad4td/wC1/wCci3r9f2tj2kZyXnK/5q//AIPyI9BstaAm0zdYoF1yv5LzKzyDrXe6v/Fzkak
yX511kXkuOa/gUXwLHoWcMWEsirhb1QYG0eVEsipn2i46zeuEM8yqqNTWIKxq/mv4T4I7AEwdnXYW2E9X2D
EZF1U5ciUUGEsylhT1k6mRmHgTouLLC1XWEZuTEJLijd72c1/Lah+GwbGaWZxc1HZs0JFw9V7h05fNU81/K
6rPtIlLQzY9jynRGzobikjzdYehKPmud8753zvnfO+d8753zvnfO+d8753zvnfO+Ta2PYMExAj753zvnfO+
d874q5//xAAvEQACAQMDAgUCBQUAAAAAAAABAgMABBEQEiEFMRMgIjJBMGEUUXGRoSQzQLHw/9oACAEDAQE
/Af8AAhgaY/armNYmCr9FI2k9vktV2xCrlt0p1VGc4UUliT7zQs4lp8bjjSCAzH7UiLGNq1Ku1yNYj6BihZ
xfPNG2g7V+BXd34pEVBhaJxU90oUqnfSCAzH7UqhBgaXgxLpEoZvV2qCbxVzjGk8Ehl3LQ7c1cuyxZTg08j
v7jpBCZmxSqEG0VtJrYavXR39JzpZdOkktmvD7e1JGB6BxintpVTxD2q0iWe4SN+xrqVjbLasyqFIqRdyka
2KoYQRWAKnuY4B6u9T3ck/HYa2PUv6A2OOc/x3/3Q4YN+VXcqy2u5fnGktzPMu2RyRpdR+HJ+unTptknhn5
q4LLExXvRJJydQC3AqK0kzuzigfg1jRjjgVsP61LD4y7aZSjFWq3gkmb0fvUn9s7vJZezjVGHzTFfimoKTy
KZcd6ubfeNy96tJklTCcY+KvZ5HcxtwB5Fdl9ppb2Qd+aXqCjun81DcLOTtGNcaxwbrnxQdv8A3NdTe3k9p
9XnsO7eaVEeM+J2o9+PPZOFfB+dAhNbc8Zrw2oRmuoOI4dn5/Rtr5twR/3q9unBCoayc5qO+nj+c/rUfUi7
BSK6ljKkfRBxzTMWOT5JJfEGPL//xAA3EQABAgQDAwoFAwUAAAAAAAABAgMABAUREiExEBNRBiAiMkFxgcH
R8BQwYZGhFrHhM0BCUvH/2gAIAQIBAT8B/sKlVWqcnpZqOgiiTj08yp57j6fJmp1iTALyrX5lae30+4eGX2
iiM7iQbB7c/v8AxtfmWZVON5VhEzyobTlLIv8AU+/SHeUE+5orD3CJTH8Oje9awv37KpVW6ci2qzoPWJiYd
mnC66bmJF74iVbd4gbZqyZlxTnE5eML5QTyskEJH0HreEVmqWxhVx3D0j9TzG6KSgYuP8Q8+7ML3jqrmACo
2EUuhvuupdmE2QOOp8NlVqqKeiwzWdB5mHXVvrLjhuTs5OOY5AJ/1JHn57Kg6tmXJa6xyHecoqcgZB4N4sR
tc7KVVZJuRDTxsU6jj/2FEFRKdIpHwz0yll9F7wzLMS/9JAGyqVJFOaxHNR0EPPLmHC44bkwG1HQQJdaooE
m9JsKDotc32V511KE7k5oOL7RISqaupx6YcOL7RPURqSlVvpJJHHviQaQ6+Er0iZYbcaUCNBEu7uHkO8CDG
uytNEzqi7nw7oCQNIlJB6cPQGXGJOmsyeYzVx28obstlwDrZRITipF8OjTt7oqzzT1KW62bg2/cQlRScSYd
nX3k4Fqy2UOc+LlAD1k5Hy2VyW3jIeTqn9okkoXMoS4MiYSkIGFO1x1DKcbhsIn+UEmpBZSjeA+AhYTqnSN
4vBu75cNiEg9JWgjfJ0Aw90UqbVKTaSk5HIwhYWMSYnpxiWbIdzv2cYYJDySgdvM5SX+LGPS2Xvt2CEEIOF
wQ6pu1kQ2Rex7YMu4IKVNkGKNVlSrm6ePQV+Dx9YqMo+w/0+lfQxS5JlhsOpzJ7eY9LMzAs6gGHuTcm5mi6
ff1j9MJT1XPx/MVinqkVIKje/ltS6tGSTBJUbnZTqk3OpLNrlPv8aQyFDu5/KrqM+PlzqAVJnUkd3yOUsup
2WS6n/E/g7EsrVBlHALxuFwmXUdYoMteYxjRPyCAoWMVCjS7Dan2h4RRZNiYbLy03z7YwJthtlD1IlHs8Nu
6JihoZbU6lZNheOTzyXEuJH08/krQlxJQrQwyw3LpwNiw2nOJKnJknFqScj79+PN//8QASBAAAQIDAgcMCA
IJBAMAAAAAAQIDAAQREiETICIxQVFxECMyQlJhgZGhscHRBRQkMDNTYnJAkjRDY3OCorLC8BVUk+HS4vL/2
gAIAQEABj8CxSSaAaTEjKyq6ei8IWf3pN1rrzfi0+jWF0mJsZdOK3/35x6OWOLMtn+YfiipRoBeTEzMoCl4
RdhlAFckZoZnp1wMKQ4hSWBerhDPq/FPS8qCqam/Z2wOfP2Vi1c9OkZTx0cydUSrfLmG+w18PwSnXlhCBpM
KfbKpVtrfENcZdL8ryhK0mqVCoOM2jQw1a6VHySevck08gLdPVZ/u9yUB3DuDiMC2eyN5krA1vuU7BWPjst
fu2a95hcpLzwsJSk3soreNkfpw/wCFHlCg/gVSLQ313BUNdAHPu2l1UpVyG08JZ1CBMTVC4OA2OC1s1nn3G
```

```
2yaqZJZP8JoOymNPu63rH5QB313JuY0CjCei89p7MYrcWEIGdSjQCKSTNofPeyUdAzmPa31zH0cFH5R41gJ
SAlI0DdnphJqguUTsFwiwyMHLIO+zBFyfMw1JyiLDSOtR1nn3BdhHl3NtDOr/rngvvqwkwq6uhI5Kebdnmd
dh4dIof6cYr5bjiutZjlvEZDQvKuiGWGZVS1AVW4+qxaUbyaXnPHDlm9jaleMfpDH/Af/ACi9Es8OYqR5x7
Uy7K/WcpHWPGMHJ0m3daVZCdp8IDk056w4LwCMhOxP+HFdUk+0Pb00Oc6YRM+k7UnJ5w3mcX5QiWlWUsMIu
CE7gycI+u5tocbyHPCnnlYWYXwl+A1DElz81tbfcR44yWWMgIKkuPEZjU3J1mCUDLVwnFXqVtO6XX3EtNjj
LNIIlZtqYIzhCr4I16oS20gNoTmSnEq6tLY+ox7PLlCPmzGSPy5z2QibmPa5tAolxwcD7Ro790XYR5dzbQz
qPlzwp55WFmV8JejYNQxZNzkzCe3J8cVDDKrL75shQ4qeMr/NJgS7akAoHwwbwMRhPpHIlDK725ZqELtXnq
FIPpD0IqwEhJwqRZtL0mJaYpZwraV02iLoNyTq54dCFsS2DNFBWUc2e6PafSD7v0tb2PPtirLCUr5ZvV1m/
EtqFtZNlDac61aoVMTCguZXnIzJHJTzYyl8gpX1KBxfSE5nVLMJbbB5aj/89UehpGTbUucTV+cfWm+tk2r9
NSYUyHUF5IqW7WUBs3UoUrBPt/Ddpm5tkD1uZZEuDfgqlSuyEoQLKEiyBqG5QiohooFlL6C2doyh/dirfdN
lCBfBmpkUdIohv5SdW3XuX3Rwx0RckmLkAQVKcCE64q+pctKnlfEX0cXpxCpRokXkmJtNrAuTLgeQVjMBwa
9AhZUsPTTvDcAuA1CHW1PIS9wVPhFQea0Itf6jL05l1hwyavZmVUWtQvWebm59xCGgDMO8GuZI1xaVNzBVr
whHdCZSaVhCr4bpz7DuNPfJdQvorf2E4qm1g4OVIso1rIra6jd0xkpA2xwyNkXmu6W5VvCEXKWbkI2nwhLz
p9ZmR+sULk/aNHfiolEmipo2VfYOF4DpgyilqW0pBUi0a2SIeaC1tvFBCFJWRfCkLFhSTQg6Iqo2R2x6QaB
NCEKv6dyXc4imigbQa7kklOcOWzsGfccb5aSmJZ7StAJ26cRmZAyXhglfcL09lcRKb1LVclCRVStggLnt6b
/26DefuPgIS20gNtpuCUigGNIP8TLaO00I7jFvQ00Sem4eO436QYRVDqgl1I5Wg9MVceCTqArEwkrCwtmoP
8W4WXk2knrB1iMicRY+tq/vglJLjyuE6ru5huzMmdBwzf2qz/zV68R5gXLIqg6lDNAVSydKdR0jcZlkKDZc
rlqFaU5oJbSVOq4Tq71K9w9KrNm0MlY4qtBhbSGy06lW/BRqq1t7ubcW04m02sUIjAOm18tz5g84mF6ENJT
0k18MdqaaFpxg1sjjp4w/zSBCHW1W0LFpKhpGIogb1M74Pv4w8evckF/trP5gR7oTIfaYmUillxdkOp5PlA
daNUnQbiDqO4W3kBxB0GLDKLKa1N9anEtkFV4SEpzknRClOImGWEGjiWrijac5/h7YISZltVLQUp1abQ1ip
gC0VU4ys8eqrNJZ9W8nkr0o6c46cRYTQOt762Tyh/lIStPBUKiEu/LWlfUoHHtPOoaSbqrVSLMr7a5+yOSN
qs0b9MYFPy5a7+bP3RVDYCuVnUemPWJchL/GB4Lg1HzgEpKDyVaMZDruDDbK10QmtbXBv6O/c31pDtM1tNY
aarasJCa64Kr8h1tdRoyoCX7ppu5V3xALrQ3MI+umhKRepR1ARv29S+iXB/q17M23cLSQVuOgoQhOdRhvCX
OWRa24zMqluryCl+2rMkVp06RBl0IUSgAqOhNc2LvZoqNHVGcDogqWa1N2483xXkh1O0XK/txHGXBVCxQww
iWOCXL/AAlm+m3XXTD7ssxbnWhRUsTmV46xrj1lx0zDqv1h1agNGzcDEujCPG/mSNajogrJw0yoZTxHYNQx
5R+xVlpdVrRwxW4DYdPRAcsjCJFArTTEUG1WF6DzxapYUMlaOSoZxFeCrWIvXdzCAkZhuDBrwbyDabXqPlG
CeQEPjO2rw1iPh9sBKEgEwNO4mbYFXUChR8xPJ8oT6W9Gb8h4W1to4/1D6ubTtjDqWMDdlQ96MCkesIOEBG
dwHXrI9xgnklSKg3Ei8bIrLTLrJ1LOFSevzjLlkvjlML8Fecb9blv3zZTGQ8l37DWKITYECvw5nJV9+g+HV
jWHm0uJ1KEZOGRzJeV5xvab9ajU4g9HuE4B6tn6F1p29+2B6vOITNFdXZRF9fq+nxhqZYWW3mjaSqEzCKIe
TkutchXl7vfZRlw/U2DH6E0Ptuh91hiw82kuINtVyheNMBQzKFfdzvCV61WteKM9B03w/LOcNtZSTr54CkS
3q7J/WzGSOrOYU/624++tNlXFR1e9e+w90Sv7pHd7xt2clWiiZbA9aKaqaUPClICkkKSoVBGn3zv2mJb90n
u92VHMIlZdDyWF1WsrUK3XR6p647NoBqnCCgRzD3yk6xSJfWEAHaLvdVJoIsp4A7YWvQyz2qP/AK/gHWD8N
0l5k/1J67+nduyzzRdROyN+W4wnlOIonrirbyXBzUMcU9EfDHXFyQIylVgkmgGcxhXBZdmDhCDoHFHV+Awb
lRfaSpOdJ1iN8QX2/nMIr1pziKOTKa8lRp2QVNocfpyE3dZis3MBlPy5fP8AmPlFpqWRb5aspXWYq9LoUvl
i5XWL4rKzVf2cwK/zC/vjfZJ2nKZ3wdl/ZFFOBB1Lye+PjoJ1JNo9kPVSWmAkmyeEvyEJOsfgkTBzFin5VD
wUYbbvyZxDdDpAovGvFdsSTqcgJdwaiBoUPOzCkBwVUkihiTXpLKe78FYSQHEmqbWbUQdoJjDTa2yhulmzn
WoJs2ua7HdYXcFppXVzwQ4B602su4PWQKOJG0UUIlVDg32btFo07PxakvNA1FLYuUNhhLaBRKBQe5//xAAp
EAEAAQMCBgEFAQEBAAAAAAABEQAhMUFRIGFxgZGhEDCxwdHwQPHh/9oACAEBAAE/IeECwpbAVG+2RpELkQh
3c2/yZ8UibpK42fO3QowkP+ULPhFwaRgKysUMxEOZfqtQcCnMxg8ZYJaP8mfCiszfM66AXOlEYwo3dsPbrt
VtJgRyl+n/AIszZ/QG67F2rbQDG2PQQSO93QhPNvhuVnw6oNXbHwDE3O0gPv8AH0JrUhBJdbo7xWBNA78P4
```

```
1keWRefw1eG63CRotM/WoCc/wAiZYhJvjB0+RU4ZnYH8GWkiyFkv4PaQVbUk1KbygcPuHes+E6AHT4RkqGW
dH43iDc2QAc1pG+9zqfoTnV8Bddn7lGKmhgHY+ZmBN/RYokzTC9j0HmCom/y31Dar4uPmIwuvQaqx1gQfmk
SNsaezl0D4uYQwfxX96z4d61e5pjqyWXpAu3qOIcowkLichT6CUO6ftQWvyas22sz7rDHJA93HYUJQi3yPC
c+EvIzS5mWQv8AV2ebhOOyhZguOhfrFAO6EkPpp5t+WtH6mFgP28278B4mqkK1V0NdHNQXwkYIA0PR0O7Lw
bcKdR9IrnwNMLfBIJicGq2ObYVmcg7ou9MfOajhRRirkdBvFC5ICGSPkxR5KgEBwFWjFiehr2qAkWm9c+67
qMC2Pj1+u/N83dvX4/CGXR4GA+DAQDQ9HQ1y3eFg9K9JquXAxxrJFcOYQHIpMBsFdIpnu8Gv2WmVYByEjHR
amCwLBzCWs2Gc33r1jbQY90FUodGJpYLIbSDucUVSGyxsgACG+uRq/dCH8TRvb699Zl54JTOxNB910BWxQ2
HDptsD23eXDFWRP4NKy4Lfi6UJT+zS1WtARk5o3KINAoqEDB3DkHzLCylK7KNVak38k4GxIid2hOhjwBAeP
iMJNGo1Jcw/iffhv20WV2A1VgDWaWWlWRdHNZXbAfAmUDm1nGdrq/QXSvdM0TNbAKMZXmJODQ53cjNZ/IXC
qFg1aEJfAiAEZJKdlq3cZIBjlarr2oYDBzC2zMeKk62mwvBerTZRNBIFxzL2gNfiWk1mQZZrEhGq8mtc9wj
9BAeKgg1tBISx2bEjyZ+B30XolAcHQvUZk5AHMral+dU1mCbW05KLmz8ypvJflLyS9M1psNC/tXvzfGfywZ
ZDreU9BUB4TSQJBdEZjROdAKGzmRIMItmTC0jFa3LLJ2oC8sWu/VEQHk4/r4SxdDqB4fT8GjJYNDVfjv8AH
/ewEqeWeVsLOzJwWzp71E7w9jgAQRkSNgz/AE1NtTHh3/OVoekQ4HIPnP5FDcht7F6qgAvpzYHrw+AbIo2P
qDzDeiCny7vUAAoIbDJ3+LUpIjAsJolEijQ/ggfVAIWAkpsC3I8z8zxtCcy06ejwSYvn10vyHur/AFMmSsn
RE7fG0fvESxqdpdO1ECKH906HIg5cOfznDLIK49GoYpVbXKskQrcfB9mX6jUjyZV/iOp3LNkzsh5iHgeeKL
UgVB5Fw9S5QJALgEJHgtPG7YIO8R8fJsrp93Tiz4LayBbg55lo8lpbFghuQOE2+GK0MOuibPMvTaZK8W4rd
cZ24JOhgSUgE7tQm9Bm9di6WgjNBPoupWNYJ3NTcrCbRJK5rQGBhSw0vQu5kNuDXoYwGguySuTU30ObMJNQ
5n+8NB4s6NvyBiu0tLpO2f290u8qnbjvebu7UXv+Y9w7vmsGUHRrZTTM5lqZF2ci2tnqcQWTcdkoutZLJs+
BBHGzjOc0iU/T4CfVRIhmGGAJHRJmatXZRgI9sh2eSfFmisB7JNRUXZlh2Xq8Hl8OWhDLBsHtcBdigI2ATM
Qvfrw50f6WOJ5QZSWDE3S1H4wcEZcmWY0HhDTA3JJnlUlEmWZphOio320ly+FIW7IReJd3gS8vFZ6jvrSh7
q6LXkXQ1ncIBRyw/wACMEYY0lxDEMRLpI9U5l+IS2kLHjR7dCrLAopjZ9PlXizp1NbN0Bbs2F2LKUTIi2LJ
O1uC1AZckNEmpvRHfShltE6Pkh1pX9m61ifzKFmAgPjTgyzbi5qhRNmuQygeq9h3htUWma6CQQTRQlkF3f4
eUTKDOrnlWjyWukKvYw7NdiMCoFly2S7EvKW+1X6cS15VuAnUiOXFnWpACokiKERrqQv6gc+BWOHop9XpVr
+292ke6B3rY6EqNzLV12taA9wK6cTVxwNh3Nu1OqDYzw0OZJF2t3X5jDRtTZiYLH4BwdlFN0Ohs2ZbjzZyc
Mo9H8mialY1gTf9jI7dHiz4IpeW2r3sq/8Aa37UqET5psZhkrAdD0SfpuSRWETryMJlmY2KemXp49wj3qcz
fTcz7AjnVuz8YZnfKOq6vFnxkRwi+VKQ5fsfqOUYTQxPUpRcjapl7HkGiOpx58f9/Zowf8x+nZVCaQwFDGE
gCL3Nal61AbgS4awrx58ew7eRFcpnbBL2P0kZgytaEN/1SR8xV2HHp+foZ/QW1NHDLPWE9HK/M8Lk4+akEh
sJfdXXnrHsVzHG/sVmjwqIb9innvb1OG2mh2qNtpTAGWpPYfLxgvQHdfoZ/QV2wJQfCaJ/5caYQzi9nKlOg
nMxSgIZkjuh9UYO3JsMBmovrBd5fYda6GD/AH5raXBx+L2qWUf+LB5FFGZyT/xrTcB86O0KiRUxMOhKa0kt
rASSciYy6wZAXAP0M/pQylMZjKb9R2ofXMJMSTweqOEaCOwmrTmsAhHr5U70MFEyR+aZu6r1jP0M/pZiUyr
kYK8AW3nSmmvAvZuVjMm3G+ytBlaDmMPahYEM4bimnPfZqRFZMom8w7w+hnU6nU6nU6nU6nU6nU6nU6nU6n
U6nScvaCuQ3O1QYEPYC1TqdTqdTqdTq5X/2gAMAwEAAgADAAAAEPPPDPPPPPPPPPPKPPOk/PPPPMPPO6POAV
vPOrktzNYQvLMcsGoR1tOAgz/K7Q2iFZnNxwxfPHIAwtn/vJf8Ae8jzAYsGIdHyG5TyjlK3MNcXJWltTyjE
MDnlURLLanzyjw66kMMEEOY3zyjzzzYMMMMMx7zyjzzyUcwcIbbzzyjzzz9XY2vFPzzyjzzzzx/zy1vzzyz
/xAApEQEAAQMDAgUFAQEAAAAAAAABEQAhMRBBYVGBIHGRobEwwdHh8PFA/9oACAEDAQE/EP8AgtKwy10yr/
RnoTHg8539aN5tb01vINX6DgvWaJ82uKyxpLMDLUJIK4PdURFgL9qJcq5fxU22HzZ+atkXCoUQUAlaeCV7a
X7YZfxR04DSXeoP20AmC75F6JLSYNHwSOHpRQGVILZwY7lZXf7ppBGDLR8YCjCKty2oDiEPT96CYRbd7wKc
Sh69K2lXG/8Ac/NBhiZxniZ+Ka+Gv6LHdI63tehmjIgDJsxmcXnNr0Du40kWdD5vv5/5FGAVdWdgZ/XenZP
CffrqY3cDtJFzPoh4pWiZDHXqPDU+pF6pFHkvV5EoUYbL89e86KiYuPvpMYMfM/J7xTlwBfSkSSur0MtHCl
O7TcnzUNFhkaiuvdRGrNo4o4IRho62w+Q/fFERVoZe3gue6b/22iwTSSySokGg2TJQOQpIlCirPeioCy6P9
```

```
/2jxkY+718FxQq1xFf3T4pD2FvPWEzQRTejCGAUN0A4Q5aHGc9LkdFx8+P2x4nBINzb8x0oAhSeNDtNA5Ci
aAmuClZtUI5fsXX6AoyUgEjhfHFQ1yLxbd7lTXWeu9WC13e+feitksTL8fuglpzPt9FGDJUmZdRi5Sii54f
/xAApEQEAAQMCBQQDAAMAAAAAAAABEQAhMUFREGFxgaEgkbHwMMHRQOHx/9oACAECAQE/EP8AALw+oXY+dO
SWkzACAAsNcuqv4bqggNW8Y2NXB1j0NpSh2Q+ZpwFyvuk+HGInnOehleRTC3tDsF3u0SsjYPlF80DKm6PIT
54Qx9ROw8uDVFRr9gNA0CpgbyusQ+Z4i9av7mXTy9LNWorAEB7nmhNo5Zju4e9CbQRMBvJb7XjlpTpT6v62
ORajByuhU4qDFkFw1BOZi2J4W/i6H6OWVtuiV1Sr99jBpwljgBp23XQe0z2pAzi0YVRm7tMu9+AySIg5Li1
7rmZnSFKCJMHLQqIsTBuXiTCZiI3aEjpID754bFFvO7yNd8FJOVK/cGxgLFZMp4Au/dKwFkNcBfbHXhCWAX
w9rqOY90zHAhHQ6iWgIIjpUJEosgkDYMw2v2oH5uY3gWP7yoKZIiARBNo0tc1xTAfyEaEEmOGhELsGwOTPe
WsGioLFrWD+vI7xRwXOZ7GnzuvFLBBfoPPqY50Z84DfUddTmUE0cHrY6mppDNB0hLiZK6tCAJ6wE8DT4egZ
dzyPCOW71fxh6TUowgnWx5oKABgLBxaBLKoHmk1Ahn5Em2Rg3Gsru0cnJxPU8Nqgub5uYneMTz4TibnPMHe
InSprgc3yWl5rjMt2Xo8G4vzNzbGJKDpI/f8AuzariIJHIfg5vzSCFAgXbI+iFBK0oi95Z1TtElmGaEoLFJ
cTxVhielATwCF21HshOsTTMJ5KVDDkjSNadLUuup8H+txQLZdyeXUxHtVlzzIiJ0DTnq+PRAE4uD7OTtT6q
8mT5Pmpkydf4/SjEwpIcn6Q7cQpIbSxU4ZefBTWBF2dYzK+FZ8u9flfH1NVMeqSY8UYv6xIm8+wvZDtLwvI
Qc6l0tvePeIpPBPeltJUEFte6Qft7fgcHI2Rw0kYSLIi6FpFImdS2lSfxAy0HGHO1AWnJFvamW53s8XPFTe
ChBLBOf8AVY1RUdebOM/o/CAMgicmukWfenEAhqcuABsCsc40cqp09P8A/8QAJxABAAEDAwQDAQEBAQEAAA
AAAREAITFBUWEQIHGBMJGxocFA0fD/2gAIAQEAAT8Q7WKIcAhKq2AJVq2b5IXwNxjsW1AUME/8n49yKLE5i
Hhsf8FM8M0yRQf7/wAn49qVTowIlV0AFodLNoUjhKhIC6qi/wCttgBLl3IQ2rB8v/J+PauovobpqhhWBLUG
89Wiyb6K8iCBFFwdwKgMg7/HJUnacFydlWAEqEAFEAVigofqbiugoLJdNQAQkMCE9iV+PaZsQScMn5PQ5dI
Qgtsf1KHl8CDKWzSOZQpW0IfnSVHB9tJz0qoy3Og3vD9PFNqTbgQlVa9uGhNn0gtVF+mlB1jAlAiwsjHS05
VyFmWksXVQBQBafFyrMEJIJSyAtwsMMglshZNT6q4ebZBQ+TU/HsWBdqCMYm52/wD08rSxeiAI+alPIrnj3
c5r30IAPLTANpDE4YfIPo6aXZKuVMDAPNc0USoPfYAD66RNpibTUQw7ZExjhF90WcA8uvoSmF2Vco2KE3Ej
egSvgAACpuUg4pdLaaNoJlUJjk3SMyLmwq3SReoZcNczLF6F5r+PYsU7mUt4b+JTCXc4UsAgwJCL8NOzUkQ
QoRIEQhpRLB3HoYb9KWmz0EfqVBBzLP4QPuosKLjbykPIafiAgQCQQlGQVIsMOPYIHXEE0bgYrKrdbq5eyW
kWaxCF775BrSN81UYJF/YkYEgtfgrWquUbqKXVehrmCimoXYsQEAoAM4wSBVRmSuosoi9gR75u/wDtjlr8e
zD2ftRZuAJolCtU9okocOImmbyuCQNALUWx0ikuEFcEuVhgJXQrlX2QpahzEUp7NoWkMIKhyImlQ6KPPrAa
rdcqqqs9lqkiNvZLrgLUe2288LfIFoY54LJZQczm4sLAoAx0SAkLDJZWxUVYbrQYPEMBqrSpXUSoo9tj1Rb
D+bU4TmOzEM3RTkBsF3BpSg25jGxmYLrKqz2KytoyuyBYAotZVMov7boRIWgmJG5madPdKOBQcUTC5cgHRj
WiVBEa9ASE84o0E5eJyQINq0pmymWKNRbXwKmsMwGRmV7FADHWKLThOFG2LCiyAI1NaUmCyJuq8sM15DuOs
RU2Z6n+369hRQjUtxmpMDUhVuh0NKqyuRQbEBm6g5RBGRJCgMm/VNelLa9DASCIgmoxAuQatwAxaRiZhrB/
NUQOAB66LshBEfTU8RvwCqmJAChjsx2cqhQAugBdALtTJAhHgzFmIPKAbqubEEH9qdCG6/5Vs2tQD9y1OhO
7L/CoYXiAlYCYVVsBKtgWocxM0jmXZYwLMWqCzx1PjkQEKi4AFfFShTEGBrCkBU2Yij+kIDpC3uuIUFgFT2
SxmYSAa5sVBkYUvmwp8SJ4iaamLJWbhazAmAJ6HAUcYMciECRAlihS4NnFwi4AUq+CJKRQRQILBJhaVCgAa
CjkADa3YxucRBfCt/VADA0G0Q/ogqcGGwf5epctq366XWAldDWiBTq0ahDBt7oKODui2OSZNslNWWoAxX49
VbnmiABIyTT+CndiXJ7hTMkJiJU+IkhbB4gkKAvYCkSg0UJ6oem+BIN2bf0+KMitkMI2IDR6G7OiMb5CLGp
sdFKIDVAbER5E1oIAobWQ66Jj6UfVWdHesYTkU5HsZ8g8WDY5SfcduzyScNQCjWLGUF6DucWU2XYdUjQUlH
D2BBgAAeOv49TKhtFiE9C9KKKQXg8gEXwqSSGmDisy4OzEYkNSpdKqbZSC+PugvdRWuiTHMXpN5S3iA3kMJ
uiIov5buSTmYcgHYoxvjAzJZA3hKt0gSvDFDBDHNlCLiTsbvZGuBoQTdAmRkRrTKhUzSeQQc9DS/aghxCTi
wAJLMSAWkIYvEIA4MdNXb+PV0TIQWbpqIxqSYacsCJABuDwABgBc6WtaimRcuOES4glwoFCgsA6sWAg1sVA
SHDoW+QjwN+1tmlEkY8UBSA5KA2qAH9VILtGRAOER6pIm9LbCiWUD0W6rTo5dht4DSLg9v49VDNW9gKkrcZ
gqEVERAqnBRibinErcSRFpoSBlhkEIaIDRo408kYBfYAJTYGAOxKPJkcMQSmVABVAWkLlCnkTeEDWGKmjsB
```

```
SbFEKGSIRVNxAUfQALzBSr45MkSwgXTh6EesvYkaBUQgKY2FdKxrhEAAHmG9O0gG7T1Ax2/jQiGquCwUFgW
M2aFoC7smFH0mG46YSeMOxJuT41cTa8xvIbyqKpUpQOhFkMQyQiNDuFBCJlKTgiJCZ7TJeCbvFQ5yw7KkoA
QStAEspELbF8UuiI5OIRBiYJjMFIzVGQq2JYm6JpThQx4BLoIGEHSjcv2nsbCKCYUgS0Sv6dByAu2xYJVAU
V2oER2Q7a0L24OiO1SWUDzEiIAoCaIDghAgRBZyvr2/jUG9ZIF0GVwBCogjJY4yQQJIWg5RE7IihVrhARyp
3/YpEa0CiOyTX3hDP8AZrCnEAAIYjmfrp4ckxEbmHtNHsJsduEcaAIB0QaCbl2RnbZYtQSQik872CELCAq5
LGApaD7neVk4ZiIiBmC9B7kJHmAA4WAGFEukYXjDSm9Qm4VVBuHd+NQYxAVjSjCZCBIlHhw4ShBlJDGJJz2
DFaxjX3AkDMLF4pJsXJRLa6DqBYUsviwT6NfOai54bySPdih1mA0Oj8A6oAUPYRli4IqPN5KbKzaG2wkDZI
7Ff+UPr8SF5ltbeKgRIbiLvvoEZ+zeqWAMlstkxsDx6vdpiItXFCEWcSFCAQLgAkTdIQ0RjJpZNLJwUVwBK
GO78atdChuBoAIiUeGcTy6WLz7tOt1lMiarM8exQcbwL9TfpVMNJf0A01IWiKnmIPVN0KEBhZuau2rTm/bC
Q2AO8N1yh5oLY+DHwiPVRKTBtmipjgg6xYyUl4bVOYAqZFlxlDiuqY5OYx0m8wrQgAgtL60IUtE1CUWRGzQ
vqE9CShN0OUoyB2/j2ISEttSjMxw9yUsyrsv1URdBvBkEYJc1r+BoIH8T4rjbJUcC6STRgMHJ3FTDs8JDO4
8wa2GHSFHVCEYZFG0MmjxCAqQhCWAl7vx7wLEsdRkpVEpLz8mCd3EfZooTQIDcIEPAVdBAUQREYR7/AMe9R
RXDx/HpNKLng5cUXiJ4cwJSgEFs4osgMTCZIlwAViJe/wDHvJlhe7Sf6qaoiUyALyC9ds1PYeBpXAUwyzBN
lbjbYomTHSQl5meB8H496TRtzyWsQ0hk8WlSgKoBlcFOQraaQ84fU0GUBQ78Tq+qufzArxCfYlQfguZPOj3
UBFcr8NHg93UPygIe3F/4VaiMlvALFHCSYgCU4AX1Sq1IR2pAqpp8D/H4Bizh/re3ToiKhAy7htzQbAi+5m
AGUJ9loD3ooPIF2SoGeUBCsuKl810fVYnRmbaqK4Mz0d1fpYqKU5XxEChOfZCeBgcso0IqRhGsWTw0/NGJD
bIo+qXkHi4OkormDUp+zxnoBNtCTIAJCl4oS7iD/vwfj8MDpQJRsLjKYKQJ57NQZxwzwViJS1yOVYHPZAtw
mtR2wB9NFQhBBOTEaKkvTfaaovpDS00xPgvYvsfg/H4riN70F4UtwgSgqx9KzAs2Ty5E8qO+DMZ5zuIA5FF
gUBHBVMFURcWtKVN/JjCCAMEwnwfjXAVwFcBXAVwFcBXAVwFcBXAVwFcBXAVwFcBRRmmzVgDiqSM
u7IdJ3VhAS7AVwFcBXAVwFcBXAVwFM6MV//9k="
}
```

Comments endpoints:

Get all comments of a specific toilet

```
GET /toilet/:id/comments
```

Response

```
Status: 200 OK
```

```
[
  {
    "id": 1,
    "id_toilet": 2,
    "id_user": 1,
    "comment": "Mon premier commentaire.",
    "rating": 3
  },
  {
    "id": 2,
    "id_toilet": 2,
    "id_user": 2,
    "comment": "Un autre commentaire.",
    "rating": 4
  }
]
```

Add new comment for a specific toilet

```
POST /toilet/:id/comments
```

Header: Authorization : token

Input

Name	Type	Description
comment	string	A comment about the toilet.
rating	int	Give a note for the toilet.

Response

```
Status: 200 OK
```

```
{
  "id": 1,
  "id_toilet": 2,
  "id_user": 1,
  "comment": "Mon premier commentaire.",
  "rating": 3
}
```

Get a specific comment of a toilet

```
GET /toilet/:id_toilet/comment/id
```

Response

```
Status: 200 OK
```

```
{
  "id": 1,
  "id_toilet": 2,
  "id_user": 1,
  "comment": "Mon premier commentaire.",
  "rating": 3
}
```

Update a specific comment of a toilet

```
PUT /toilet/:id_toilet/comment/:id
```

Header: Authorization : token

Input

Name	Type	Description
comment	text	A comment for the toilet.
rating	int	Give a note for the toilet.

Response

```
Status: 200 OK
```

```
{
  "id": 1,
  "id_toilet": 2,
  "id_user": 1,
  "comment": "Mon nouveau commentaire modifié.",
  "rating": 4
}
```

User endpoints:

Register a new user

```
POST /signup
```

Input

Name	Type	Description
name	string	The user's name.
email	string	The user's email.
password	string	The user's password.

Response

```
Status: 200 OK
```

```
{
  "token":
 "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwiZW1haWwiOiJhQGIuY29tIiwicGFzc3dvcmQiOiJhem
 UiLCJpYXQiOjE0OTIwMDIyMDYsImV4cCI6MTQ5MjA4ODYwNn0.0zF9r6v4dOJjqxQaV7BOATQ6j5U64zbsEca
 LdR5K22g"
}
```

Authenticate a user

```
POST /authenticate
```

Input

Name	Type	Description
email	string	The user's email.
password	string	The user's password.

Response

```
Status: 200 OK
```

```
{
  "token":
 "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MSwiZW1haWwiOiJhQGIuY29tIiwicGFzc3dvcmQiOiJhem
 UiLCJpYXQiOjE0OTIwMDIyMDYsImV4cCI6MTQ5MjA4ODYwNn0.0zF9r6v4dOJjqxQaV7BOATQ6j5U64zbsEca
 LdR5K22g"
}
```

Get data of specific user account

```
GET /user
```

Header: Authorization : token

Response

```
Status: 200 OK
```

```
{
  "id": 1,
  "name": "Batman",
  "email": "mail@gmail.com",
  "password": "azerty",
  "createdAt": "2017-04-12 10:43:51.000 +00:00",
  "updatedAt": "2017-04-12 10:43:51.000 +00:00"
}
```

Update data of specific user account

```
PUT /user
```

Header: Authorization : token

Input

Name	Type	Description
name	string	The user's name.
password	string	The user's password.

Response

```
Status: 200 OK
```

```
{
  "id": 1,
  "name": "Superman",
  "email": "mail@gmail.com",
  "password": "qwerty",
  "createdAt": "2017-04-12 10:43:51.000 +00:00",
  "updatedAt": "2017-04-12 11:03:22.177 +00:00"
}
```

Diagramme de use cases

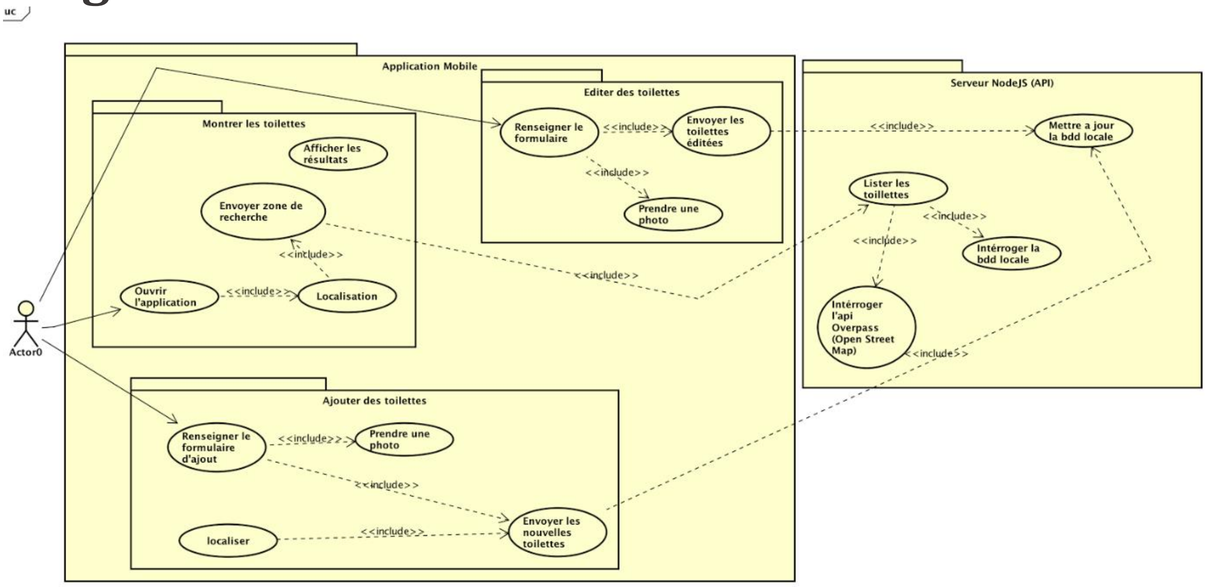
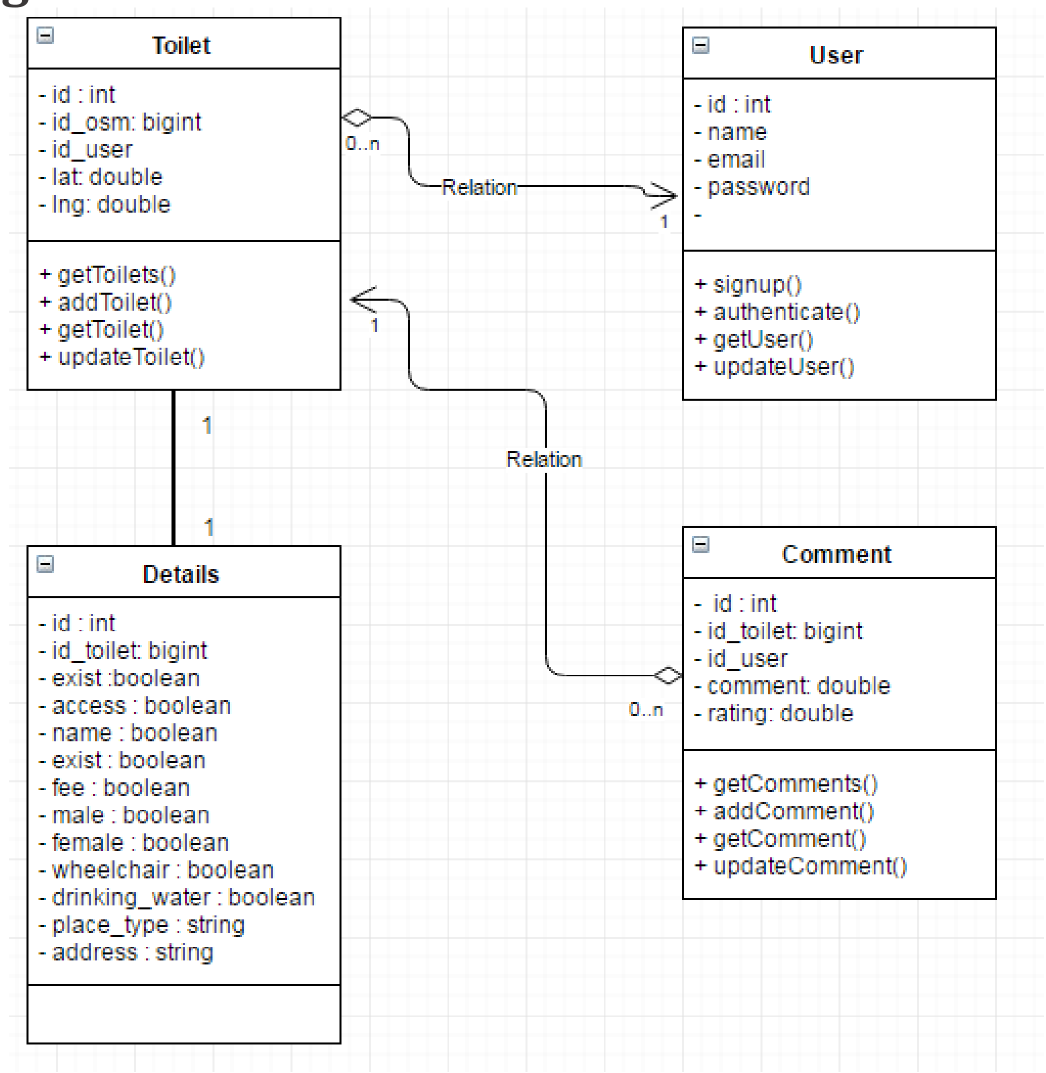


Diagramme de classes


Toilet
- id : int
- id_osm: bigint
- id_user
- lat: double
- lng: double
+ getToilets()
+ addToilet()
+ getToilet()
+ updateToilet()
User
- id : int
- name
- email
- password
-
+ signup()
+ authenticate()
+ getUser()
+ updateUser()
0..n
Relation
1
1
Relation
1
1
0..n
Details
- id : int
- id_toilet: bigint
- exist :boolean
- access : boolean
- name : boolean
- exist : boolean
- fee : boolean
- male : boolean
- female : boolean
- wheelchair : boolean
- drinking_water : boolean
- place_type : string
- address : string
Comment
- id : int
- id_toilet: bigint
- id_user
- comment: double
- rating: double
+ getComments()
+ addComment()
+ getComment()
+ updateComment()