

Rapport de Projet Licence 3 Informatique : Présentation de l'application RunTourist

Xavier Lacoudray 36005157
Sautron Samuel 36000990

Avril 2021

Résumé

La Réunion terre d'outre-mer située entre terre et mer, préparez-vous à un voyage ! Résident ou même Touriste la Réunion regorge de Lieu dont nous ignorons, mais grâce à notre application vous pourrez découvrir ou redécouvrir ces Lieux Emblématiques.

1 Introduction

Quel projet avons-nous choisi de réaliser ?

Pour notre projet, on a choisi de créer une application orientée pour les Touriste ou Résident de la Réunion qui veulent se renseigner sur les lieux de notre île.

Pourquoi ce Projet d'application ?

Au départ on a voulu s'orienter sur une application de randonnées, car n'ayant pas trouvé d'application ou de site regroupant toutes les Randonnées de l'île on avait cette idée, mais la mise en place de cette application nous semble trop compliqué pour nos débuts en Développement mobile du fait de la gestion de coordonnée et de l'utilisation de Google Maps pour se repérer sur la Carte. On a décidé de partir sur une application plus simple dans le procédé, mais qui se rapporte toujours à notre idée de faire découvrir ou redécouvrir la Réunion.

Quelle est le but de l'application ?

Le but de l'application est de pouvoir regarder les différents lieux par rapport aux cardinalités de l'île (Nord, Sud, Ouest et Est) si le lieu intéressé l'utilisateur, il peut la sauvegarder dans mes favoris afin de se retrouver plus facilement l'application est facile d'accès. L'utilisateur peut aussi supprimer ces favoris et grâce à la base donnée avoir une persistance des données.

Nous allons vous montrer les différents visuels de notre application l'utilisation sur Android, puis une description du Code, on continuera avec celle d'iOS ainsi nous finirons avec notre ressenti et l'avenir de notre application.

2 Description générale de l'application

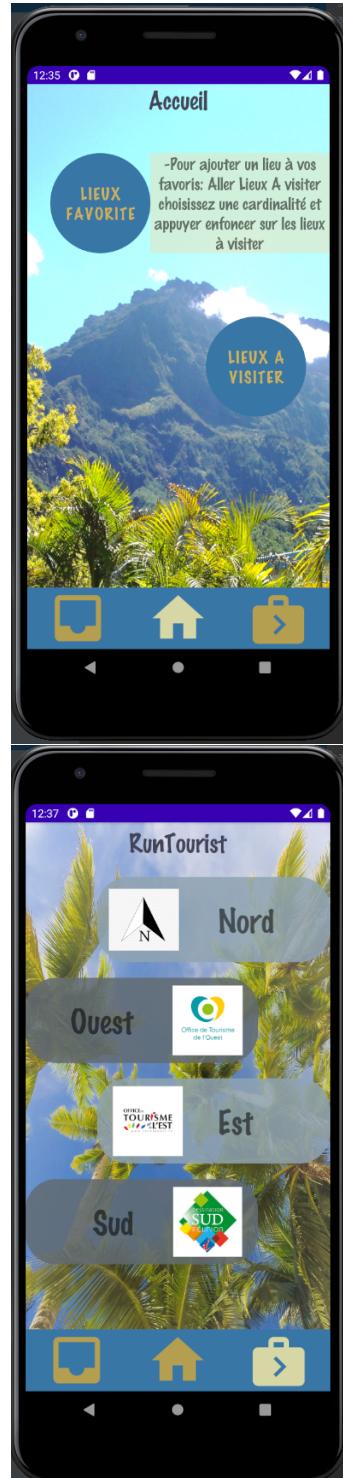
2.1 Mode Portrait

2.1.1 Page d'accueil

Voici l'écran d'accueil il comporte 5 boutons :

- Le bouton Lieux a visité qui va afficher une nouvelle page pour choisir les lieux en fonction de leur cardinalité.
- Le bouton Lieux Favorites qui permettra de stock sur une base de donnée locale les lieux qui nous intéresse.
- Le bouton Lieux Favoris en bas à gauche (un raccourci menant directement sur la page Lieux Favorites).
- Le bouton Home en tout en bas au milieu (un raccourci pour revenir à l'accueil).
- Le bouton Lieux a visité en bas à droite (un raccourci pour revenir sur la page Lieux a visités).

On a en bas une surbrillance sur le bouton home en bas qui nous indique où on se trouve.



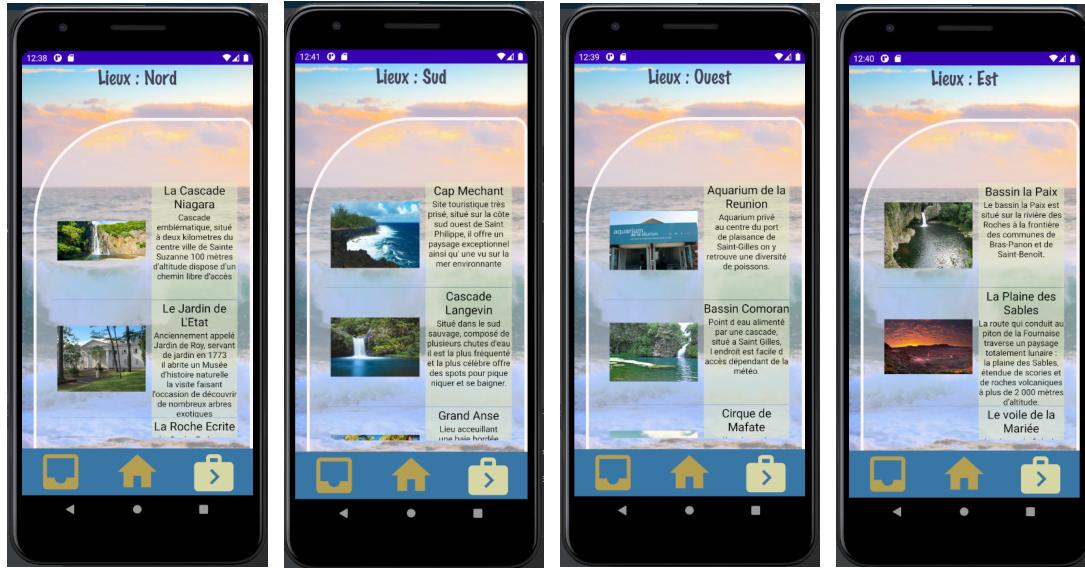
2.1.2 Page lieux a visiter

Voici l'écran des "Lieux à Visiter" il comporte 4 boutons :

- Le bouton Nord qui ouvre une nouvelle page sur les lieux dans le nord de l'île.
- Le bouton Sud qui ouvre une nouvelle page sur les lieux dans le sud de l'île.
- Le bouton Ouest qui ouvre une nouvelle page sur les lieux dans l'ouest de l'île.
- Le bouton Est qui ouvre une nouvelle page sur les lieux dans l'est de l'île.

On a toujours une surbrillance sur le bouton en bas à droite nous indiquant qu'on est sur la page "Lieux À visiter"

2.1.3 Pages des cardinalités

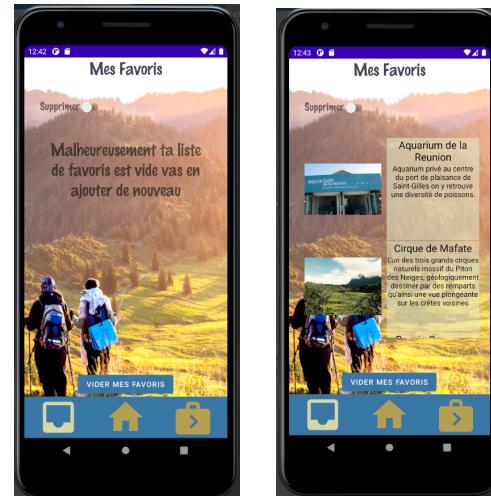


Nous avons ici les quatre cardinalités distinctes avec leur propre liste de lieux, ils ont cinq lieux au total. Pour sauvegarder un lieu intéressant il faut maintenir le clic sur la zone de la photo et du texte une réponse vous affichera qu'il à été bel et bien sauvegardé dans vos favoris.

2.1.4 Page mes favoris

Nous avons maintenant 2 pages mes favoris l'un est vide pour l'instant et si on ajoute des lieux a sa base il deviendra comme la deuxième page une notification apparaîtra une fois qu'un lieu est enregistré. Il y a aussi 2 nouveaux boutons :

- Le bouton "vider mes favoris" qui supprime tous les lieux enregistrer dans la base de données locale.
- Le bouton actionnable "supprimer" qui permet de supprimer un lieu de la base en maintenant le clic sur le lieu.



2.2 Mode Paysage

2.2.1 Page accueil

Comparé a son mode normale, il n'y a vraiment pas de changement les boutons ont dus être réadapté en circonstance.

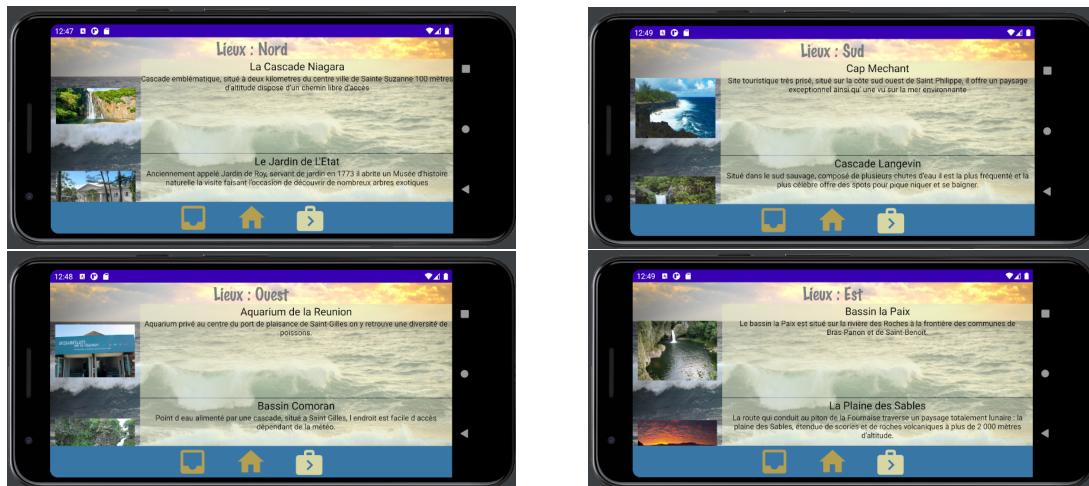


2.2.2 Page lieux a visiter

Comparé a son mode normale, les boutons ont changé de place et sont devenues comme des colonnes.

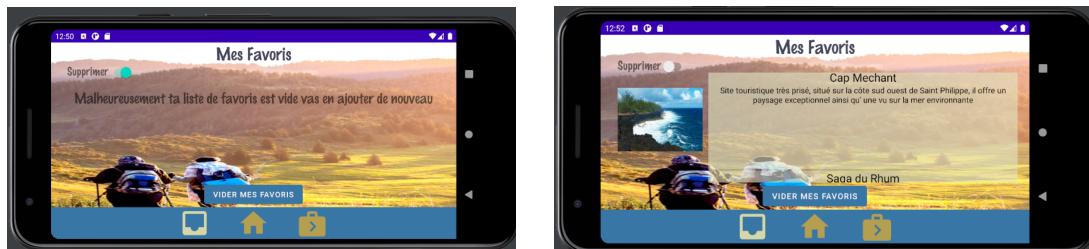


2.2.3 Pages des cardinalités



On peut apercevoir qu'ici, c'est devenu plus petit il y a une perte d'informations qu'on ne puisse pas voir, mais avec un scrollview on peut néanmoins accéder aux autre parti caché de l'écran comme son prédécesseur en mode normale.

2.2.4 Page mes favoris



Comme son prédécesseur, c'est la même chose sauf que peut juste voir l'affichage d'un seul lieu favori toujours accessible avec un scrollview le reste des lieux enregistrer.

3 Code

3.1 MainActivity

```

//Choix de la vue selon ce que l'utilisateur a choisi
public void choixDeMaVue(){
    lieux = intent.getIntExtra( name: "lieux", defaultValue: 0 );
    if(lieux == 1){
        donneNord();
        texteH.setText("Lieux : Nord");
    }
    if(lieux == 2){
        donneEst();
        texteH.setText("Lieux : Est");
    }
    if(lieux == 3){
        donneOuest();
        texteH.setText("Lieux : Ouest");
    }
    if(lieux == 4){
        donneSud();
        texteH.setText("Lieux : Sud");
    }
}

```

L'utilisateur commence tout d'abord sur une page d'accueil si ce lui vas dans les lieux a visité et arrive au 4 points cardinaux. Il doit choisir une cardinalité qui sera interprété par la fonction choixDeMaVue dans le MainActivity2, on utilise ici un entier «lieu» selon ce que l'utilisateur décide de choisir il enverra un entier qui transféra les données a affiché dans la ListView.

3.2 AffCell

```

public class AffMyCell extends ArrayAdapter<MyCell> {
    private final Context context;
    //Methode pour implementer la lecture de ma Liste de Cellule
    public AffMyCell(@NotNull Context context, MyCell[] listeCell) {
        super(context, R.layout.cell_list, listeCell);
        this.context=context;
    }

    //Appeler automatiquement depuis Le MainActivity
    //Elle est utiliser comme une boucle de lecture et se termine quand elle a lu tout les donnée de ma liste de cellule
    public View getView(int position, View convertView, ViewGroup parent){
        View cellView=convertView;

        //Ici c'est lorsque la liste est null on arrete la lecture
        if(cellView == null){
            LayoutInflator inflater = (LayoutInflater)context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            cellView = inflater.inflate(R.layout.cell_list,parent, attachToRoot: false);
        }
        //Ici on lit ma liste de ma cellule
        //Pour chaque cellule on definit son id
        ImageView image=cellView.findViewById(R.id.imgLieux);
        TextView textNom=cellView.findViewById(R.id.textNom);
        TextView textInfo=cellView.findViewById(R.id.textInfo);
        //On indique ainsi la cellule choisi
        MyCell s = getItem(position);
        //On implemente selon la position choisi on indique le nom ,l'information et l'image associe donner par une cell
        textNom.setText(s.getIdLieux());
        textInfo.setText(s.getIdTextInfo());
        image.setImageResource(s.getIdImg());
        //On Retourne ma cellule lors de chaque repetition
        return cellView;
    }
}

```

Pour afficher correctement la listeraient d'envoyer les données choisies plus haut est affiché dans cette ListView, on a la fonction getView qui se démarre automatiquement pour lire la liste de nos lieus, celle-ci fonctionne comme une boucle et fini quand la liste sera vide ainsi a chaque passage on donne l'image, nom du lieu et les informations complémentaire à afficher dans la cellule.

3.3 MainActivity2

```

//Ici on écoute l'utilisateur appuyer longtemps sur un lieu pour l'ajouter dans ses favoris
list.setOnItemLongClickListener(new AdapterView.OnItemLongClickListener() {
    @Override
    public boolean onItemLongClick(AdapterView<?> parent, View view, int position, long id) {
        mesDonnes = maBase.getData();
        //test du doublon avant ajout
        if(mesDonnes.length!=0) {
            if (pasDeDoublon(mesDonnes, mesCell[position])) {
                //Ajout a ma base
                maBase.addIN(mesCell[position].getIdLieux(), mesCell[position].getIdTextInfo(), mesCell[position].getIdImg());
                Toast.makeText(context, text: "Ajout de l'élément comme préféré", Toast.LENGTH_SHORT).show();
                //Mise à jour de mes données
                mesDonnes = maBase.getData();

            } else {
                Toast.makeText(context, text: "Déjà ajouter dans ma liste favorite", Toast.LENGTH_SHORT).show();
            }
        }else{
            maBase.addIN(mesCell[position].getIdLieux(),mesCell[position].getIdTextInfo(),mesCell[position].getIdImg());
            Toast.makeText(context, text: "Ajout de l'élément comme préféré",Toast.LENGTH_SHORT).show();
        }
        return true;
    }
});

```

Lorsque l'utilisateur appuie enfoncé sur un lieu celle-ci envoie un message et Ajoute à la base de donnée notre élément choisi. On envoie dans la base donnée la position «servant à supprimer», et les données propres aux lieux Image, Nom et Info. On remarque aussi la Fonction pas de doublon pour éviter les doublons dans ma liste Favorite.

3.4 Base de Donnée

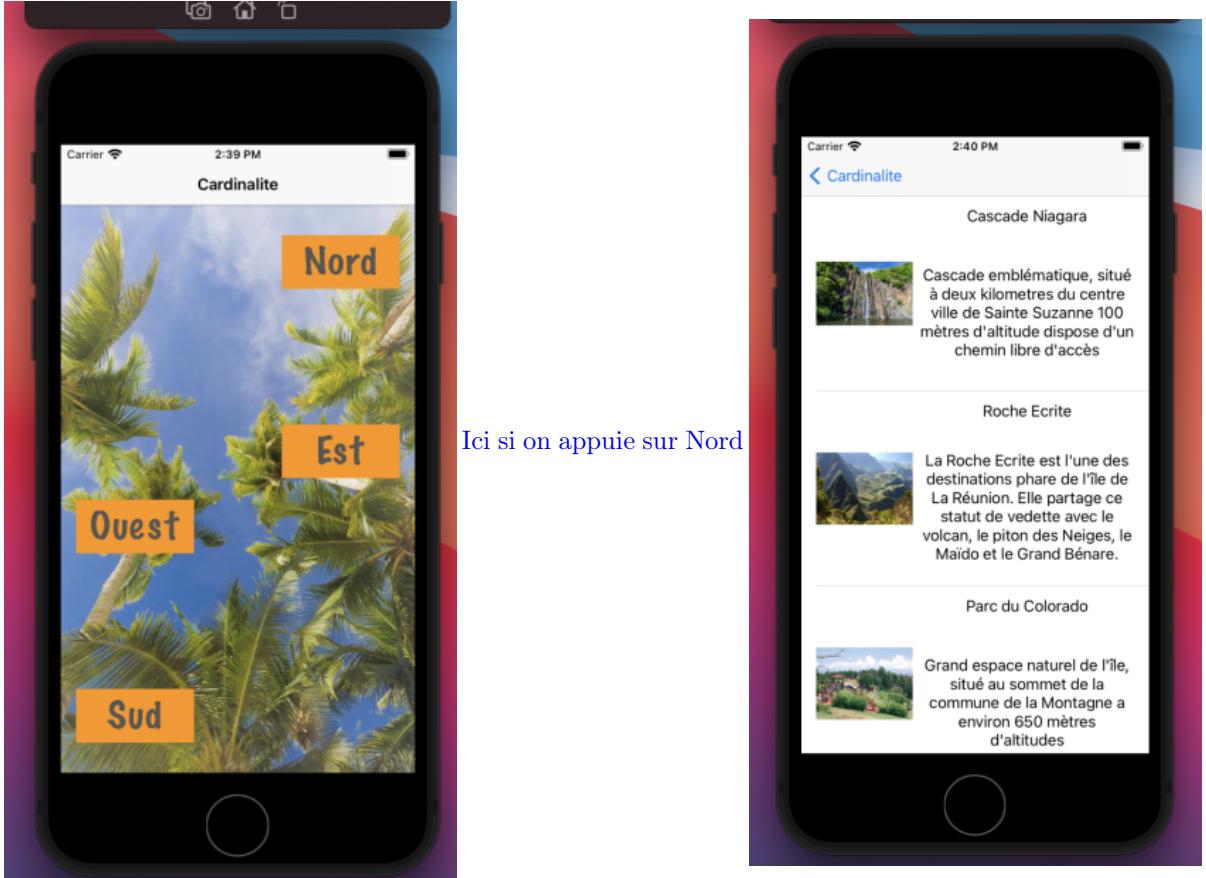
```

public void addIN(int name,int info ,int img){
    SQLiteDatabase db = this.getWritableDatabase();
    String strSq = "insert into my_table (name, info,img) values ("+
        + name +", "+ info +", "+ img +")";
    this.getWritableDatabase().execSQL(strSq);
    Log.i( tag: "ici", msg: "Ajout dans ma base de donnee");
}
//Envie d'un String
public MyCell[] getData(){
    SQLiteDatabase db =this.getWritableDatabase();
    //Valeur a Retourner
    String donne =" ";
    //Ici on va se déplacer dans la base en selectionnat tout les colonne
    String query = "select * from my_table";
    // Cursor data pour parcourir la table
    Cursor data = this.getWritableDatabase().rawQuery(query, selectionArgs: null);
    //Creation d'une celule
    MyCell[] cells = new MyCell[data.getCount()];
    int i = 0;
    data.moveToFirst();
    //donne = data.getInt(0)+" "+data.getInt(1)+" "+data.getInt(2)+" "+data.getInt(3);
    int count = data.getCount();
    Log.i( tag: "ici", msg: data.getCount()+"");
    while (i!=count){
        data.moveToPosition(i);
        //Log.i("ik1",donne);
        //Log.i("ik2",donne);
        Log.i( tag: "iko",data.getString( columnIndex: 0));
        cells[i] = new MyCell(data.getInt( columnIndex: 0),
            data.getInt( columnIndex: 1),data.getInt( columnIndex: 2),data.getInt( columnIndex: 3));
        i++;
    }
    return cells;
}

```

Ici ce sont les fonctions servant à ajouter les informations d'un lieu choisi par l'utilisateur dans une base donnée est ainsi de la sauvegarder. La fonction addIN ajoute un lieu et pour afficher les lieux on a getData. La Fonction getData renvoie un une liste de MyCell contenant tous les Lieus choisis par l'utilisateur. Pour parcourir la Table on écrit en SQL "select * from my table" que je vais directement transfère chaque information qui sont classés par colonne dans ma cells selon l'indice qui est incrémenté au fur et a mesure par «i». À la fin on retournera cells comportant les lieus choisis par l'utilisateur.

4 Parti iOS



On a donc la même fonctionnalité que la version Android selon la cardinalité choisie on a accès à la liste des différents lieux. On a rencontré des problèmes avec la version iOS dû au fait de l'affichage soit celle de régler les contraintes pour avoir le même affichage que celui d'Android. On a aussi eu du mal avec iOS ne pas tout coder simplement comme sur Android studio et n'ayant pas le même type de langage nous a posé problème.

Certes malgré les problèmes on a réussi avoir une application basique sans les fonctionnalités favoris.

5 Conclusion

Notre Application est donc fonctionnelle que ce soit sur Android ou sur iOS, on est fiers de vous la présenter. On a appris vis-à-vis de ce projet que créer une application n'est pas simple et demande beaucoup de rigueur en termes de travail et aussi en termes de travail personnel. On a su malgré tout structurer notre projet. Seulement on n'a pas pu passer assez de temps sur la partie iOS ou on voulait avoir les mêmes fonctionnalités comme "mes favoris" ou "la persistance de donnée". Le Monde peut donc découvrir les lieux visités à la Réunion grâce à notre application et c'est ça qui nous satisfait le plus d'être arrivé au bout de ce projet.

Nos idées dans le futur ? Continuer l'application la rendre responsive et compatible à tout les téléphones. Mettre l'application iOS au même niveau que l'application Android. Pouvoir aussi utiliser Google Maps et la géolocalisation de son téléphone pour rechercher directement sur la Carte les Lieux. Ajout aussi de Compte permettant à chaque Utilisateur de se connecter et avoir sa propre base sauvegarder en Ligne.

6 References

Photos prises sur ces sites :

- <https://www.cartedelareunion.fr/listings/roche-ecrire>
- <https://www.zotcar.com/blog/cascade-la-reunion/>
- <https://www.istockphoto.com/fr/search/2/image?phrase=cascade+niagara+%C3%AEle+de+la+r%C3%A9union>
- <https://laterreestunjardin.com/jardin-etat-reunion/>
- <https://www.cartedelareunion.fr/listings/parc-du-colorado>
- <https://randopitons.re/tourisme/364-parc-loisirs-colorado>
- <https://blog.tropicalhome.fr/vanilleraie-a-la-reunion/>
- <https://www.cartedelareunion.fr/listings/piton-des-neiges>
- <https://www.lechotouristique.com/article/crise-a-la-reunion-les-recommandations-du-seto/ile-de-la-reunion-cascade-langevin-2>
- <http://espirituaventura.canalblog.com/archives/2012/10/30/25461287.html>
- <https://www.lrphotographies.com/media/6295da75-6710-498d-a0db-effdc94fc776-explosion-de-vague>
- <https://www.cartedelareunion.fr/listings/cap-mechant>
- <https://orkymel.fr/petite-ile/la-plage-a-voir-a-la-reunion/>
- <https://www.cartedelareunion.fr/listings/plage-de-grande-anse>
- <https://www.cartedelareunion.fr/listings/saga-du-rhum>
- <https://www.cartedelareunion.fr/listings/cirque-de-mafate>
- <https://www.airfrance.fr/FR/fr/common/travel-guide/le-fantastique-piton-maido.htm>
- <https://www.cartedelareunion.fr/listings/le-maido>
- <https://blog.tropicalhome.fr/kelonia/>
- <https://www.cartedelareunion.fr/listings/kelonia>
- http://gorgebleue.canalblog.com/albums/a1__la_reunion__paysages/photos/60011707-bassin_cormoran_tres_joli.html
- <https://blog.tropicalhome.fr/les-bassins-de-saint-gilles/>
- https://www.tripadvisor.fr/LocationPhotoDirectLink-g298470-d3349640-i77462278-Aquarium_de_la_Reunion-Saint_Gilles_Les_Bains_Arrondissement_of_Saint_Pau.html
- <https://www.cartedelareunion.fr/listings/piton-de-la-fournaise>
- <https://dailygeekshow.com/cascade-legende-reunion/>
- <https://www.rentiles.fr/blog-voyage/decouvrez-le-voile-de-la-mariee-a-la-reunion.html>
- <https://www.cartedelareunion.fr/listings/voile-de-la-mariee>
- https://www.geocaching.com/geocache/GC7XX0Z_ec-la-plaine-des-sables?guid=3b9bc1e0-9e42-437f-bb
- <https://www.cartedelareunion.fr/listings/plaine-des-sables>
- <http://www.lareunionparadis.com/les-10-choses-a-faire-absolument-dans-l-est-a-la-reunion/>
- <https://www.cartedelareunion.fr/listings/le-bassin-la-paix-et-bassin-la-mer-depuis-abondance>
- <https://www.cartedelareunion.fr/listings/eglise-notre-dame-des-laves>
- <https://www.wlaps.com/notre-dame-des-laves-la-reunion/>