Document de conception technique

Projet : DiscoverIt

Étudiants : Dupont Alexandre, Khauv Thierry

[Considérations générales](#h.pi63ctjk8hse)

[Vocabulaire](#h.27hwrbkwqj8v)

[Nomenclature](#h.1jsr07c7fq77)

[MySQL](#h.x7t23pbg0j19)

[Php](#h.b2d7h1m6if8m)

[FTP et Versionning](#h.q1bvna3l77z7)

[Serveur FTP](#h.wc72r2diaf4k)

[Versionning](#h.hq295ewfpa86)

[Base de données](#h.r8parip8xa7f)

[Fonctionnalités :](#h.xlmgjjy528np)

[Fonctionnalités de carnet de voyage :](#h.iifcg3p9o87u)

[Fonctionnalités d’un itinéraire :](#h.31lmnrg6ziy1)

[Fonctionnalités d’annonce :](#h.ks36mgi2ni04)

[Fonctionnalités d’users :](#h.l635njjghn18)

[Partie technique :](#h.gdqv89lz78yv)

[Module carnet :](#h.pbkgbkz0d7jl)

[Schéma](#h.jx7zf7vc7voz)

[Action : createbook](#h.pj6fct96v0ym)

[Action : editbook](#h.9fyqkewam7im)

[Aciton : addstep](#h.fylkyvn7bqrr)

[Action : editstep](#h.9mox8bmg6mat)

[Action : deletebook](#h.e99nymm381as)

[Action : showbook](#h.cu2mt2oc2lhm)

[Action : showstep](#h.y8vo9phjmnup)

[Module itinéraire :](#h.umhulfr5qaea)

[Schéma](#h.6z5tpavbiqn8)

[Action : Create](#h.i471fe1zvuig)

[Aciton : Edit](#h.xol11rc8jmnn)

[Action : Delete](#h.m6xrxoengd5r)

[Action : Show](#h.lafmm4mjhjh7)

[Module annonce :](#h.ygrpt77pq1s8)

[Schéma](#h.cvivvaj36kz7)

[Action : Create](#h.qvyti5ooutip)

[Action : analysis](#h.hvt1io8nkiho)

[Concepts généraux :](#h.k9ux09q2qn4x)

[Vérification des formulaires :](#h.j3a04c6uf03k)

# Considérations générales

## Vocabulaire

* Carnet/Book : un utilisateurs doit pouvoir créer un carnet via le site web. Le carnet est constitué d’étapes/steps
* Etape/Step : plusieurs étapes constitue un carnet
* Annonce : une annonce est un encart publicitaire pouvant apparaître lors de l’affichage d’un contenu en relation avec l’annonce.

## Nomenclature

### MySQL

Le nomage de la base de données à été définit avec un préfixe ‘DT’ faisant référence au nom de projet. La suite du nom de la table est définit en majuscule et de préférence au pluriel.

Exemple : ‘DT\_USERS’

### Php

Le code Php doit respecter les norme PSR-1

# FTP et Versionning

## Serveur FTP

Les différents serveurs sont :

* Développement : local
* Test : <http://ns366377.ovh.net/dupont/perso/Discoverit/Projet3A/>
* Production : <http://ns366377.ovh.net/alexandr/projet3a/Discoverit/>

La migration vers le serveur de test se fera automatiquement au minimum une fois par semaine. La migration vers le serveur de production se fera elle manuellement lors de la période de mise en production (aux alentours du 01/02/2015).

## Versionning

Comme les consignes l’ont demandé le projet utilise un dépôt à l’adresse suivante : <https://github.com/Yamato93/Projet3A>

La branche développement est consacré à la phase de test. À la fin de cet dernière la branche sera émerger sur la branche master avec les livrables demandés.

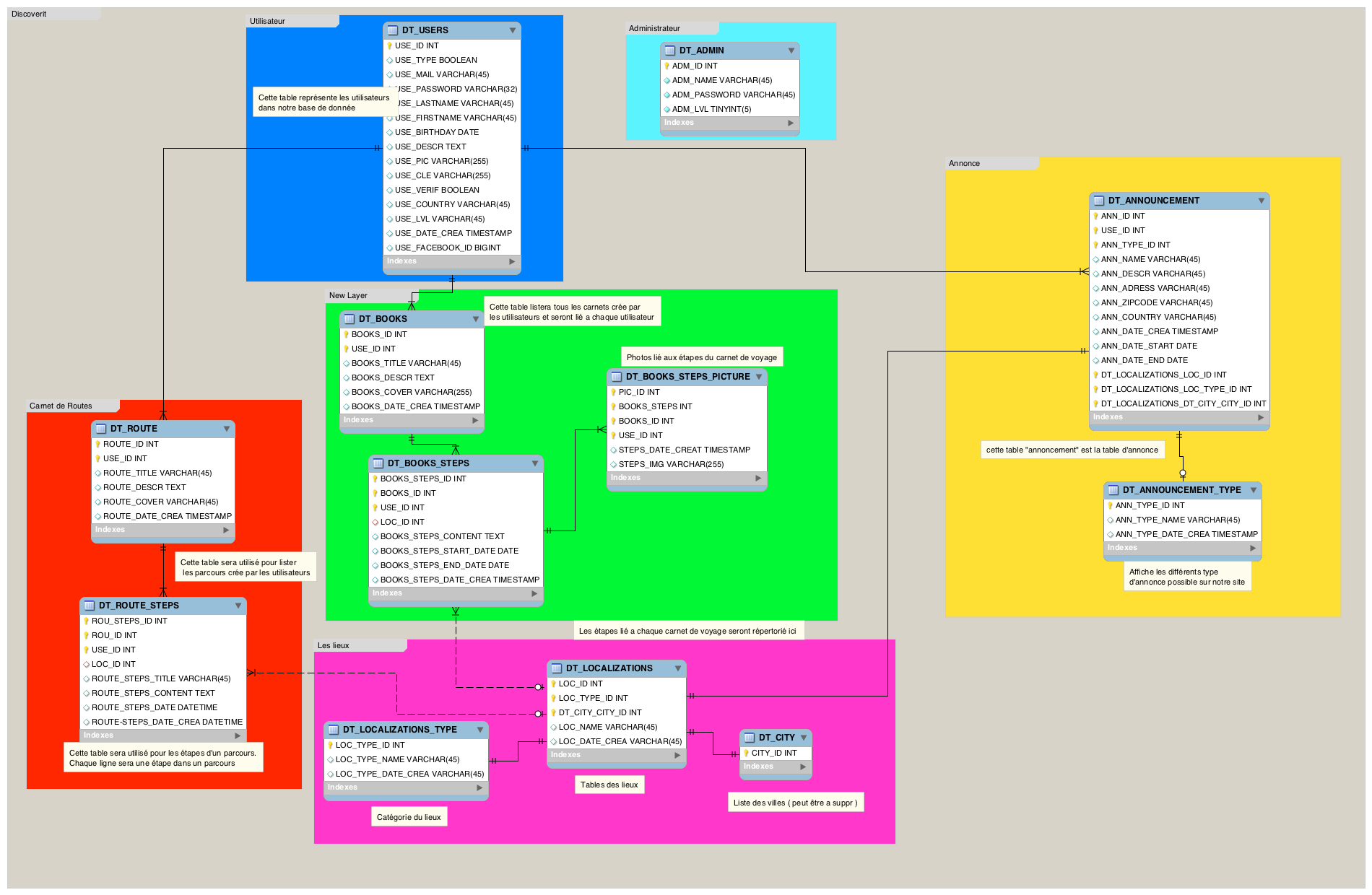
Le dépôt est aussi doté d’une branche pour l’AB testing.

# 

# Base de données

La base de données doit être modélisé via MySQLWorkBench. Le fichier suivi de l'extension .mwb doit être disponible sur le dépôt mentionné au dessus ainsi qu’un fichier test .sql dès que cela est possible. Les différentes bases données et leurs codes sont disponible sur un document externe.

|  |  |
| --- | --- |
| Nom de la table | Description |
| DT\_USERS | Table des utilisateurs |
| DT\_ROUTES | Itinéraire d’un utilisateur suite à sa recherche |
| DT\_BOOKS | Voyage d’un utilisateur suite à la création d’un carnet |
| DT\_CITIES\_WORLD | Table de toutes les villes du monde entiers |
| DT\_ANNOUNCEMENT | Les annonces des annonceurs liés à leurs lieux |
| DT\_LOCALIZATIONS | Les lieux par ou passe un ou plusieurs utilisateurs |
| DT\_CONFIG | Toutes les configurations |
| DT\_ANNOUNCEMENT\_TYPE | Type d’annonce |
| DT\_LOCALIZATIONS\_TYPE | Type de lieux |



# Fonctionnalités :

1. créer un carnet de voyage
2. créer un itinéraire
3. créer une annonce
4. s’enregistrer et se connecter au site

**Page d’accueil :**

La page d’accueil sera constitué, d’un bouton principal pour créer un carnet, une bar de recherche, et 6 vignettes des carnet de voyage sélectionnés dans le BO ordonné par date.

## Fonctionnalités de carnet de voyage :

L’utilisateur doit pouvoir :

* créer un carnet puis s’identifier ou s’enregistrer
* éditer un de ses carnets
* supprimer un de ses carnets
* partager un de ses carnets
* rechercher des carnets

## Fonctionnalités d’un itinéraire :

L’utilisateur doit pouvoir :

* s’enregistrer puis commencer à créer son itinéraire
* éditer son itinéraire
* supprimer son itinéraire
* partager son itinéraire

## Fonctionnalités d’annonce :

L’annonceur doit pouvoir :

* s’enregistrer en tant que pro\*
* créer son annonce à l’aide de :
  + titre de l’annonce
  + description de l’annonce
  + choisir le nombre de mois où son annonce
* payer la mise en ligne

## Fonctionnalités d’users :

L’internaute doit pouvoir :

* S’enregistrer à l’aide de :
  + un compte facebook ou une adresse e-mail
  + un nom
  + un prénom
  + une date de naissance (facultatif)
  + profession
  + un auto-portrait
  + un mot de passe
  + il peut avoir un compre pro\*
* Se connecter à l’aide de :
  + L’API facebook sans mot de passe
  + Une adresse e-mail et un mot de passe

# 

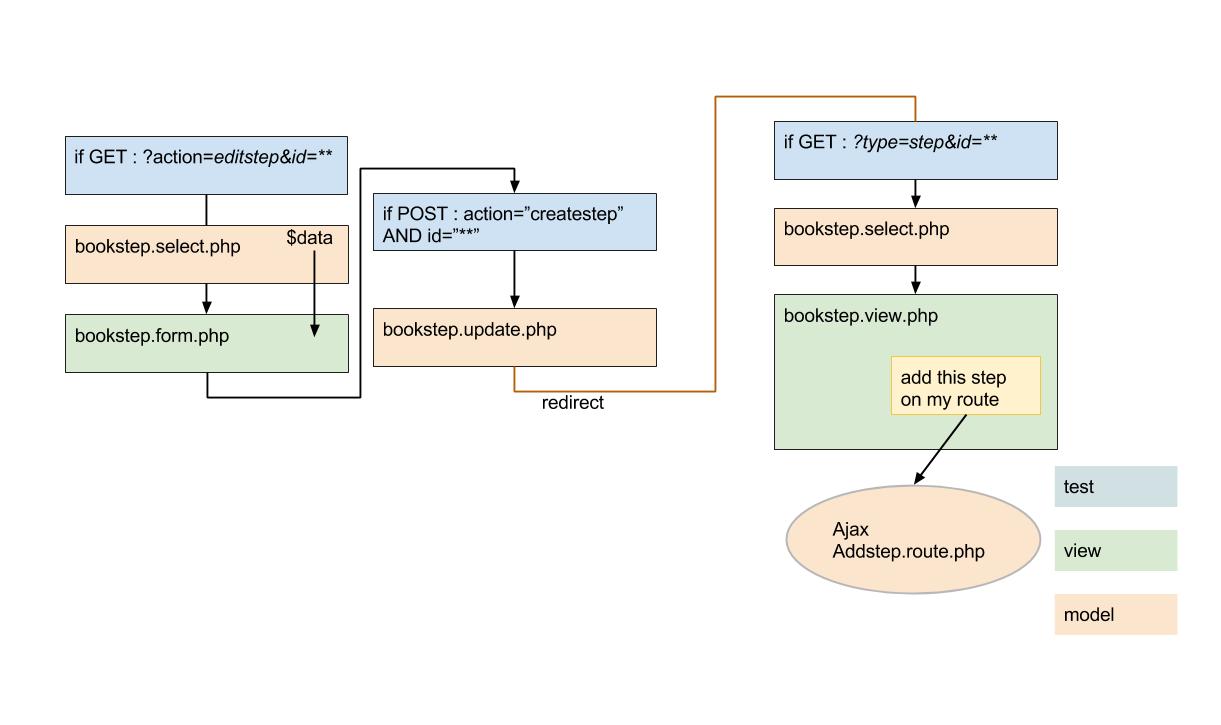
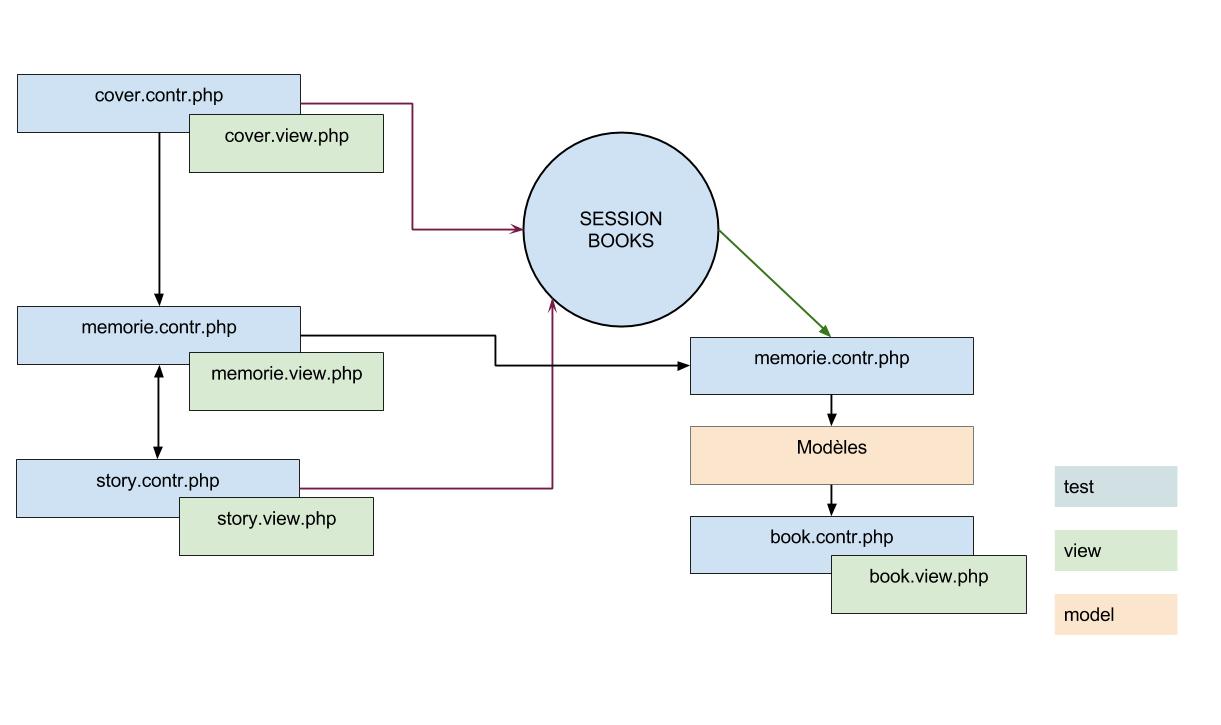
# 

# Partie technique :

## Module carnet :

*discover.it?module=books*

*discover.it/books*

Schéma 

### Action : createbook

Permet la création d’un carnet de voyage. Avec les différents attributs défini ci-dessus.

Tout doit être mémorisé en base de donnée et relié à une session. Une fois crée la page suivante est une page de la vue de se carnet. Cependant en tant qu’auteur l’utilisateur vois un bouton pour ajouter des étapes à son carnet.

### Action : editbook

Permet l'édition des informations d’un carnet de voyage. Donc il faut retourner un formulaire identique à celui de la création, avec les informations au pré-chargé dans les champs. Renvoi à la vue de la consultation de son carnet après sauvegarde.

### Aciton : addstep

Permet la création d’une étape par rapport à un carnet. Il nous faut l’*id* du carnet dans lequel il faut insérer une étape. L’utilisateur va pouvoir rédiger une description ainsi que toutes les informations mentionné ci-dessus. Suite à ça l’utilisateur retournera sur la page de la vue de son carnet.

### Action : editstep

Similaire à l’action *editbook.* Renvoi à la page de vue de cette étape.

### Action : deletebook

Permet la suppression total d’un carnet. Et donc par un **trigger** la suppression des étapes qui le constitue. Il faut demander par javascript, pop-up, ou une autre page la suppression de se carnet. *On peut se demander l'utilité d’un archivage*

### Action : showbook

Permet l’affichage d’un carnet avec en paramètre GET l’*id* et aussi le *type*(ici book). Affiche le carnet avec les différentes étapes qui le compose. Cela sera l’action par défaut de se controleur.

### Action : showstep

Permet l’affichage d’une étape. Via GET on doit récupérer la l’*id* et le *type* (ici step). Permet l’affichage d’une étape.

**Action : search**

Permet la recherche au sein de tout les carnets enregistrer sur le site. Par mot-clés, au sein de tout le contenu d’un carnet.

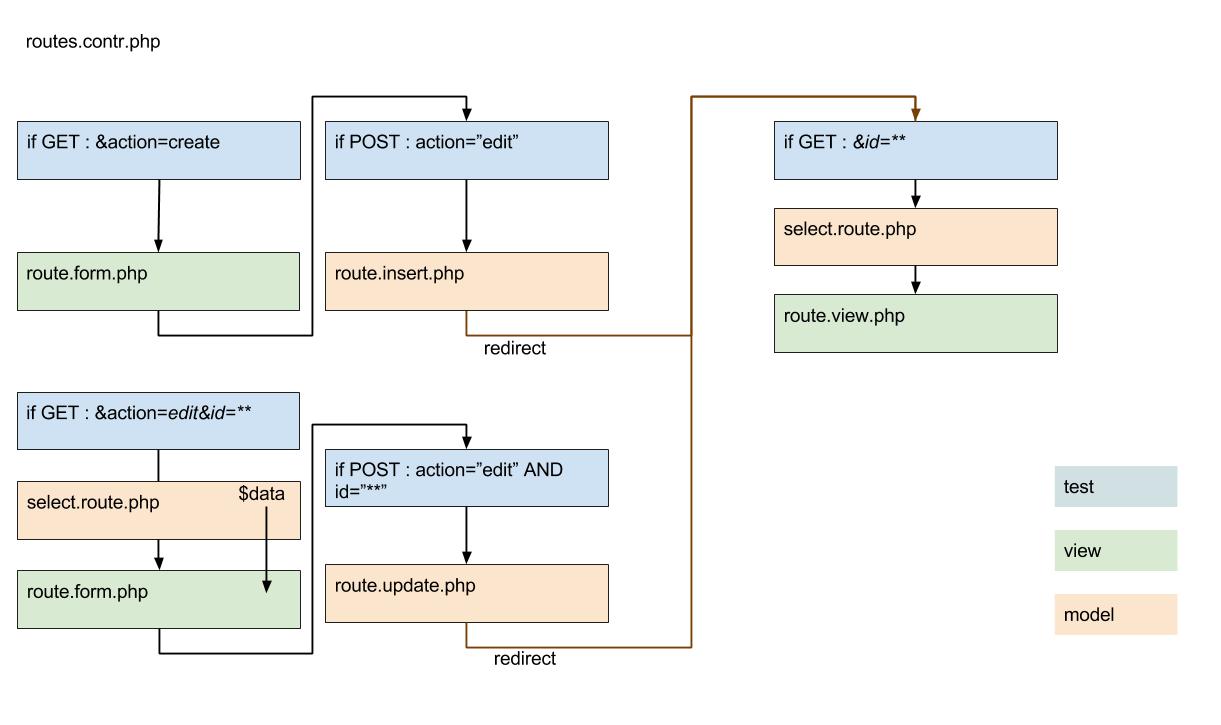
## 

## Module itinéraire :

*discover.it?module=routes*

*discover.it/routes*

### Schéma



### Action : Create

Permet de créer pour tout utilisateur connu (donc identifier) de créer un itinéraire. Il déterminera dans un formulaire les différents aspects de celui-ci. A la validation du formulaire le model entrera en base de donnée les différentes information puis l’utilisateur sera dirigé vers la vue de se nouvel itinéraire (*routes&id=\*\**) .

### Aciton : Edit

Permet à l’utilisateur de modifier les informations de son itinéraire. Réplique du formulaire avec les informations pré-chargé. A la validation : enregistrement puis redirection vers la page de la vue de cet itinéraire.

### Action : Delete

Permet la suppression d’un itinéraire. Après confirmation de la surpression par l’utilisateur. (javascript et/ou page de confirmation) le model fera un delete dans le base de donnée qui pourra alors entraîner un **trigger** de suppression sur les étapes de l’itinéraire.

### Action : Show

Affichage de l’itinéraire à travers les différentes étapes qui le compose. Via un model récupérant les différents étapes de ce dernier à partir de l’*id* passer en GET.

**Action : Addstep**

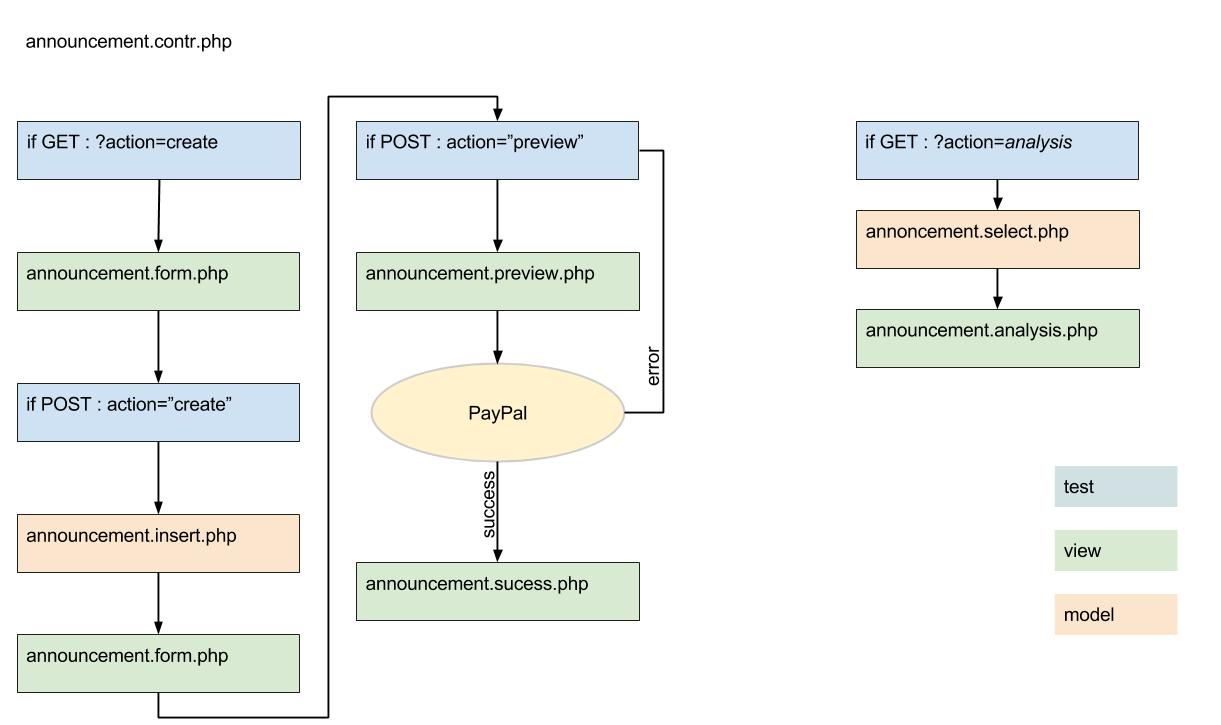
Fonctionnera pour l’ajout d’une étape via un appel ajax. Permettra d’afficher les différents itinéraires dont un utilisateurs possède et de choisir auquel il ajoute.

## Module annonce :

*discover.it?module=announcement*

*discovert.it/announcement*

### Schéma



### Action : Create

Permet à un utilisateur pro de créer une annonce avec ses différentes informations. Dont la date de départ de la mise en ligne de l’annonce et la date de fin. L’utilisateur remplira alors un formulaire. Puis en verra un aperçu, puis validera en fonction de ses dates et prix et devra payer via paypal.

### 

### Action : analysis

Permet de suivre le nombre de clic sur une annonce et le nombre d’affichage. Cela via un model qui cherchera les données.

## Concepts généraux :

### Vérification des formulaires :

Chaque formulaire doit être muni d’un champs caché “action” avec comme valeur ce sur quoi il sera testé après.