Hemmer.ch – Générateur de feuilles dominicales

Documentation de développement

|  |  |  |
| --- | --- | --- |
| Version | Date | Information/modifications |
| Semestre d’hiver 2018-19 / Patrick Morattel | | |
| 1.0 | 12.04.2019 | Première version |
| 1.1 | 06.05.2019 | Ajout formatage, traduction des dates |
| 1.2 | 09.05.2019 | Correction lecture plusieurs icals |

Chef de projet : Olivier Neuhaus

Table des matières

[1 Introduction 3](#_Toc8216721)

[1.1 Feuille Dominicale 4](#_Toc8216722)

[1.1.1 Formats de tableau 5](#_Toc8216723)

[2 Présentation du webservice 5](#_Toc8216724)

[2.1 Requête POST 6](#_Toc8216725)

[2.2 Comment utiliser le service ? 6](#_Toc8216726)

[2.3 Comment tester le webservice ? 7](#_Toc8216727)

[2.3.1 Paramètres 8](#_Toc8216728)

[2.3.2 Formatage des dates et heures 9](#_Toc8216729)

[2.4 Diagramme d’activité 10](#_Toc8216730)

[2.5 Librairies utilisées 10](#_Toc8216731)

[2.5.1 PHPWord 11](#_Toc8216732)

[2.5.2 PHP iCal Parser 11](#_Toc8216733)

[2.5.3 HTMLtoOpenXML 11](#_Toc8216734)

[3 L’exportation en Word 11](#_Toc8216735)

[3.1 Balises 11](#_Toc8216736)

[3.1.1 Glossaire des balises 12](#_Toc8216737)

[3.2 Tableaux 13](#_Toc8216738)

[3.3 Template Google Drive 15](#_Toc8216739)

[4 Lecture de l’ICal 16](#_Toc8216740)

[4.1 Événements 17](#_Toc8216741)

[4.1.1 Contenu HTML 17](#_Toc8216742)

[5 Implémentation 18](#_Toc8216743)

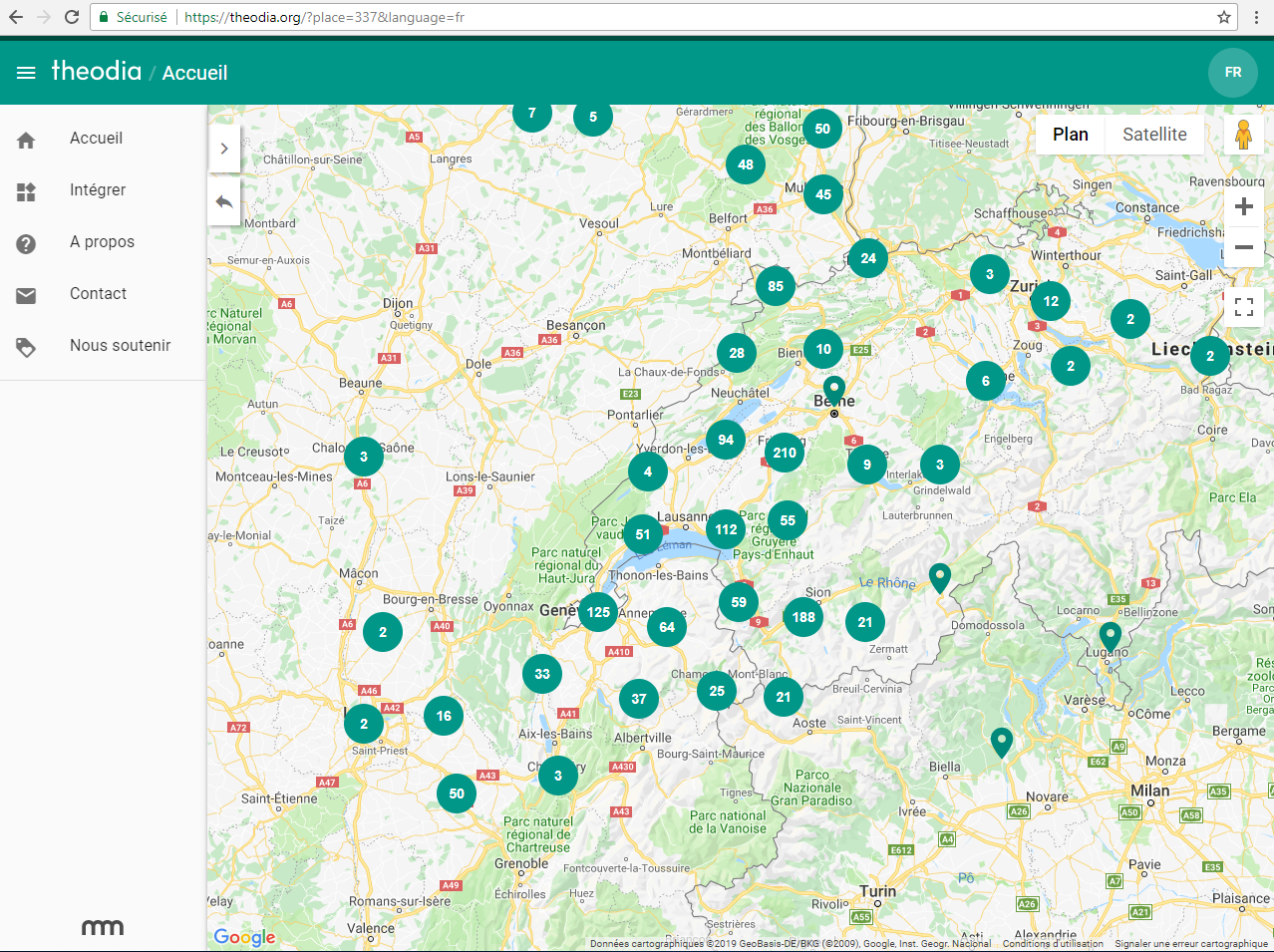
[5.1 Diagramme de classes 18](#_Toc8216744)

[5.2 Etat du développement 18](#_Toc8216745)

[6 Conclusion 19](#_Toc8216746)

# Introduction

L’entreprise Hemmer.ch est une entreprise fribourgeoise spécialisée dans la création de sites web utilisant le CMS PHP Typo3. Parmi ses projets, la plateforme THEODIA permet de gérer de façon centralisée la liste des différents offices religieux des paroisses du Suisse et du monde. Elle offre entre autre la possibilité de visualiser sur une carte la liste des paroisses et offices religieux qui se dérouleront dans les prochaines semaines.

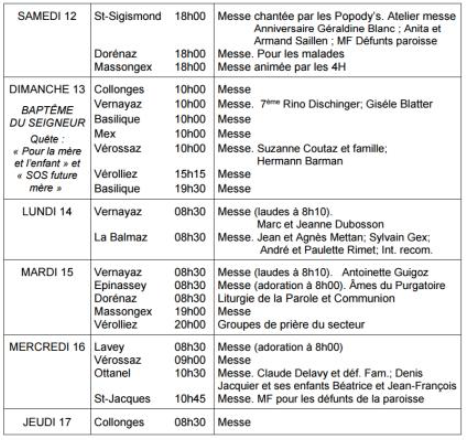
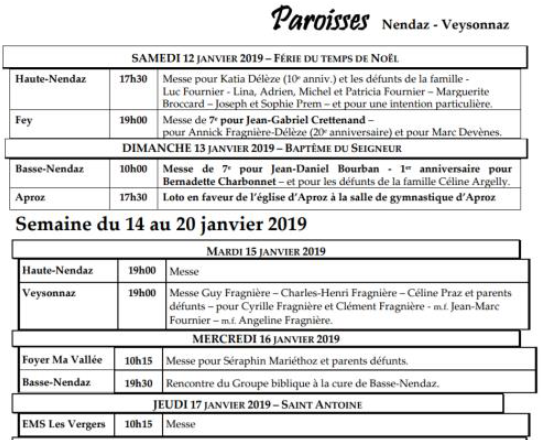


Actuellement cette plateforme contient déjà les informations de nombreuses paroisses mais toutes par contre réalisent manuellement les feuilles dominicales qui sont mises à disposition des paroissiens. L’idée du projet consiste ainsi à profiter de la plateforme pour reprendre sur les données s’y trouvant afin de générer de manière automatique ces feuilles dominicales.

## Feuille Dominicale

Généralement chaque semaine les paroisses réalisent des feuilles dominicales sous format Word ou PDF qui sont ensuite transmises aux paroissiens via leur site internet pour les informer des offices prévus durant la semaine ainsi que des événements particuliers (activités, concerts, soupes de carême, …). Certaines paroisses font des feuilles dominicales hebdomadaires, d’autres bihebdomadaires ou mensuelles.

La forme exacte de ces feuilles dominicales dépend de la paroisse :



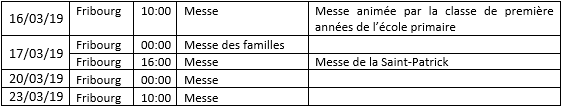
### Formats de tableau

Il y a trois formats de tableau que l’on souhaite pouvoir générer :

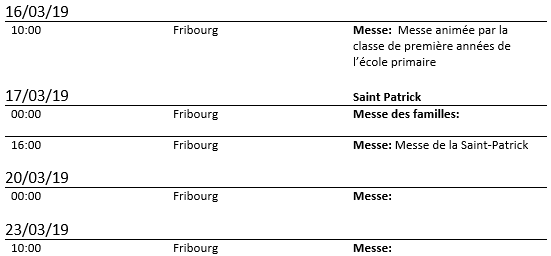
* Un tableau simple avec un évènement par ligne :



* Un tableau avec les événements regroupés dans la même ligne de date :



* Un tableau avec les événements organisés en liste, les événements d’une même date regroupés :



# Présentation du webservice

L’application prend les paramètres en entrée :

* Un ICS des offices religieux (URL vers un fichier ICS)
* Un autre ICS contenant des événements spéciaux et / ou les Saints (URL vers un fichier ICS)
* Le template DOCX de la paroisse (URL vers document sur Google Drive)

Un document DOCX est généré en sortie.

## Requête POST

Les paramètres sont transmis à l’application (appel http avec paramètres POST de la page ctrl.php) via une requête post contenant les paramètres suivants :

|  |  |
| --- | --- |
| Nom de la variable | Valeur correspondante |
| calendarMainURL\* | URL vers le fichier .ics principal |
| calendarOtherURL | URL vers le fichier .ics secondaire |
| templateURL\* | URL vers le fichier .docx template |
| logoURL | URL vers une image / logo d’organisation |
| dateBegin | Date de début de lecture |
| dateEnd | Date de fin de lecture |
| organisationName | Nom de l’organisation |
| organisationLocation | Lieu (localité) de l’organisation |
| organisationAddress | Adresse de l’organisation |
| organisationPhoneNbr | Numéro de téléphone de l’organisation |
| Locales\* | Code régional pour traduire dates/heures |
| datePattern\* | Format de date à utiliser |
| timePattern\* | Format d’heure à utiliser |

(\*Paramètres requis pour la génération du fichier)

D’autres paramètres seront ajoutés en fonctions des réunions avec le client.

## Comment utiliser le service ?

|  |  |  |
| --- | --- | --- |
| **Nom de la méthode** | **POST** | |
| **Description** | Méthode permettant de lire les événements d’un fichier .ics et de générer une feuille dominicale avec les information de ces événements | |
| **Appel HTTP** | POST | |
| **Input** | Paramètres listés dans le tableau du chapitre 2.1 | |
| **Output** | Code HTTP | 201 si le fichier a été créé correctement  400 si un paramètre requis est manquant  500 si un problème a empêché la création du fichier |
| result | Le fichier créé est téléchargé |
| message | Une description de l’erreur / du succès |

## Comment tester le webservice ?

Une page HTML a été conçue pour tester le webservice. Elle permet de donner tous les paramètres nécessaires pour tester les fonctionnalités de l’application. Le calendrier des messes, le template, et le format des dates et heures sont des paramètres obligatoires pour le fonctionnement du service.

Cette page ainsi que le projet sont hébergés à l’adresse :

<http://hemmer.emf-infopro.ch/fiche_dominicale_1.0/src/>



### Paramètres

* Calendrier des messes **(obligatoire)** :



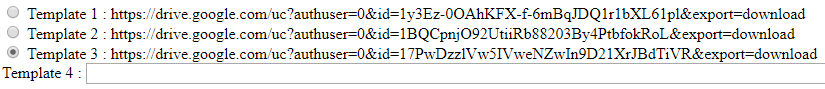
L’URL vers le calendrier principal contenant les événements.

* Calendrier d’autres événements :



L’URL vers un calendrier secondaire, qui sera utilisé pour donner par exemple des événements qu’on ne retrouverait pas sur theodia.org.

* Choix du template **(obligatoire)** :



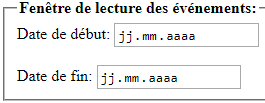
Choix entre trois URL de templates qui correspondent aux trois formats de tableau que l’on peut générer, ou une 4ème option qui permet d’entrer une URL vers un document template de son choix.

* Logo :



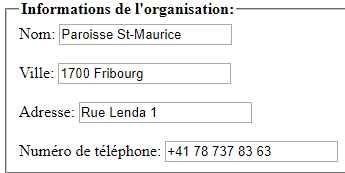
URL vers une image à insérer dans le document.

* Fenêtre de lecture des événements :



Date de début et date de fin pour définir une fenêtre de lecture des événements. Seuls les événements se déroulant entre les deux dates (elles y compris) seront utilisés. Si on entre qu’une seule ou aucune des deux dates, tous les événements seront utilisés. Elles sont automatiquement remplies pour lire la semaine courante.

* Informations de l’organisation :



Informations concernant l’organisation à insérer dans le document.

* Formats des dates et heure des événements **(obligatoire)** :



Choix de la langue et des paternes de formatage des dates et des heures (voir chapitre suivant).

### Formatage des dates et heures

Le web service prend en paramètre ce qu’on appelle des « locales », qui sont composées :

* D’un code de langage (Arabe = ar, Anglais = en, …)
* D’un code de pays (Arabie Saoudite = sa, Etats-Unis = us, …).

Exemple : locale de l’anglais des États-Unis = « en-us ».

Voir ICU User Guide – locale : <http://userguide.icu-project.org/locale>.

Liste de codes de langages : <https://www.science.co.il/language/Locale-codes.php>.

Cela permet d’afficher un élément de la date ou de l’heure (comme le mois ou le jour de la semaine) dans la langue correspondant à la locale donnée.

Il y a encore deux paramètres nécessaires au formatage des dates et heure, les paternes de formatage. Ces paternes permette de définir quelles informations de la date ou de l’heure à afficher, et dans quel ordre. Par exemple :

* « dd MMMM » affichera « 01 Mars » en français.
* « HH:mm » affichera « 10:00 » en français.

Comme dans le deuxième exemple, il est possible d’utiliser des caractères qui ne doivent pas être interprétés pour le formatage. Pour les lettres et chiffres il faut les entourer d’apostrophes, mais les caractères spéciaux comme ‘:’, ‘.’, ‘ ’, ‘#’, ‘@’ seront automatiquement considérés comme entourés.

Voici quelques symboles utilisables pour créer des paternes de formatage :

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Field Symbol Table** | | | |
| **Symbol** | **Signification** | **Example(s)** | |
| y | year | yy y or yyyy | 96 1996 |
| M | month in year | M MM MMM MMMM MMMMM | 9 09 Sep September S |
| d | day in month | d dd | 2 02 |
| E | day of week | E, EE, *or* EEE EEEE EEEEE EEEEEE | Tue Tuesday T Tu |
| H | hour in day (0~23) | H HH | 0 00 |
| h | hour in am/pm (1~12) | h hh | 7 07 |
| a | am/pm marker | a | pm |
| m | minute in hour | m mm | 4 04 |
| s | second in minute | s ss | 5 05 |

Source (liste complète des symboles de paternes) : <http://userguide.icu-project.org/formatparse/datetime>.

## Diagramme d’activité

Le diagramme d’activité décrivant les actions réalisées par l’application.



## Librairies utilisées

Trois librairies trouvées sur GitHub sont utilisées dans l’application :

* phpOffice/phpWord : <https://github.com/PHPOffice/PHPWord>
* om/icalParser : <https://github.com/OzzyCzech/icalparser>
* rkorebrits/htmlToOpenxml : <https://github.com/rkorebrits/HTMLtoOpenXML>

Toutes les librairies ont été installées à l’aide de « Composer » un manager de dépendances PHP.

Composer : <https://getcomposer.org/>

### PHPWord

PHPWord est une librairie qui met à disposition un set de classes pour écrire et lire depuis différents formats de fichiers différents. Nous l’utilisons dans cette application pour écrire dans un document Office Open XML (OOXML).

### PHP iCal Parser

Cette librairie permet de lire des fichiers .ical / .ics et d’en extraire les événements. Il gère également les événements récurrents.

### HTMLtoOpenXML

Cette librairie permet de convertir du code HTML en code OXML pour les fichiers .docx.

# L’exportation en Word

## Balises

Les feuilles dominicales sont générées à partir de fichiers .docx contenant les « balises » au format obligatoire « **${nomVariable}** ». Ces balises permettent à la librairie PHPWord de les cibler et les remplacer par le contenu souhaité.

Des balises sont aussi utilisées afin de « cloner » des lignes de tableau dans le document.

Afin que les utilisateurs puissent créer des templates intuitivement, une règle de nommage a été définie pour les balises des templates :

Toute les balises utilisées par l’application commenceront par « **t.** ».

La suite du nom dépend de la valeur représentée par la balise :

* Les balises en « **t.event.[suite]** » correspondent aux informations des événements de calendrier.
  + **Exemple** : ${t.event.location} – le lieu où se déroulera un événement.
* Les balises en « **t.organisation.[suite]** » correspondent aux informations concernant l’organisation (paroisse, unité pastorale, …).
  + **Exemple** : ${t.organisation.location} – le lieu où est basée l’organisation.
* Les balises en « **t.table.[suite]** » sont utilisées pour cloner les cellules de tableau.
  + **Exemple** : ${t.table.nocellmerge} – tableau sans cellule fusionnée.

### Glossaire des balises

1. **Balises d’organisation :**

|  |  |
| --- | --- |
| Nom de la balise | Valeur correspondante |
| t.organisation.name | Nom de l’organisation |
| t.organisation.location | Lieu (localité) de l’organisation |
| t.organisation.address | Adresse de l’organisation |
| t.organisation.phonenumber | Numéro de téléphone de l’organisation |
| t.organisation.logo | Image / logo de l’organisation |

1. **Balises de tableau :**

|  |  |
| --- | --- |
| Nom de la balise | Valeur correspondante |
| t.table.nocellmerge | Tableau sans cellules fusionnées |
| t.table.cellmerge.date | Tableau avec cellules fusionnées, à cloner en fonction du nombre de jours |
| t.table.cellmerge.event | Tableau avec cellules fusionnées, à cloner en fonction du nombre d’événements dans le jour |

1. **Balises d’évènement :**

**NOTE** : L’utilisation de ces balises n’a été prévue que dans un tableau (voir exemples chapitre suivant)

|  |  |
| --- | --- |
| Nom de la balise | Valeur correspondante |
| t.event.date | Date de l’événement |
| t.event.time | Heure de l’événement |
| t.event.location | Lieu (localité) de l’événement |
| t.event.summary | Titre de l’événement |
| t.event.description | Description de l’événement |
| t.event.eventoftheday | Nom de l’événement du jour |

## Tableaux

Il y a trois types de tableau différents qui ont été prévus pour fonctionner avec l’application pour le moment :

1. **Tableau sans cellules fusionnées :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ${t.table.nocellmerge}${t.event.date} | ${t.event.location} | ${t.event.time} | ${t.event.summary} | ${t.event.description} |

Tableau dans le template :

Tableau une fois la feuille générée :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 16/03/19 | Fribourg | 10:00 | Messe | Messe animée par la classe de première années de l’école primaire |
| 17/03/19 | Fribourg | 00:00 | Messe des familles |  |
| 17/03/19 | Fribourg | 16:00 | Messe | Messe de la Saint-Patrick |
| 20/03/19 | Fribourg | 00:00 | Messe |  |
| 23/03/19 | Fribourg | 10:00 | Messe |  |

Ce format de tableau est le plus simple et est organisé de façon à utiliser une ligne pour chaque évènement.

1. **Tableau avec cellules fusionnées, simple :**

Tableau dans le template :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ${t.table.cellmerge.date} ${t.event.date}  26/03/19 |  |  |  |  |
| ${t.table.cellmerge.event}${t.event.location} | ${t.event.time} | ${t.event.summary} | ${t.event.description} |

Tableau une fois la feuille générée :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 16/03/1926/03/19 |  |  |  |  |
| Fribourg | 10:00 | Messe | Messe animée par la classe de première années de l’école primaire |
| 17/03/1926/03/19 |  |  |  |  |
| Fribourg | 00:00 | Messe des familles |  |
| Fribourg | 16:00 | Messe | Messe de la Saint-Patrick |
| 20/03/1926/03/19 |  |  |  |  |
| Fribourg | 00:00 | Messe |  |
| 23/03/1926/03/19 |  |  |  |  |
| Fribourg | 10:00 | Messe |  |

Ce type de tableau regroupe les événements se déroulant à la même date. Cela permet une lecture plus agréable que pour le premier type de tableau.

1. **Tableau avec cellules fusionnées, sous forme de liste :**

|  |  |  |  |
| --- | --- | --- | --- |
| ${t.table.cellmerge.date} | ${t.event.date} | | **${t.event.eventoftheday}** |
| ${t.table.cellmerge.event}${t.event.time} | ${t.event.location} | **${t.event.summary}:** ${t.event.description} |

Tableau dans le template :

|  |  |  |
| --- | --- | --- |
| 16/03/19 | |  |
| 10:00 | Fribourg | **Messe:**  Messe animée par la classe de première années de l’école primaire |
| 17/03/19 | | **Saint Patrick** |
| 00:00 | Fribourg | **Messe des familles:** |
| 16:00 | Fribourg | **Messe:** Messe de la Saint-Patrick |
| 20/03/19 | |  |
| 00:00 | Fribourg | **Messe:** |
| 23/03/19 | |  |
| 10:00 | Fribourg | **Messe:** |

Tableau une fois la feuille générée :

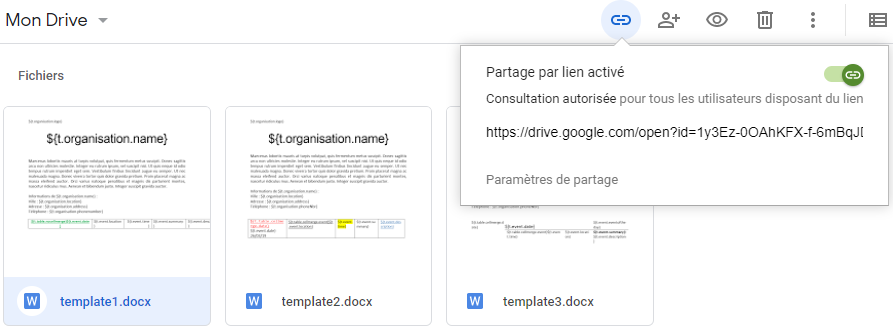
Ce type de tableau utilise les mêmes balises que le deuxième, mais les événements sont organisés sous forme de liste sous leur date.

Sur la même ligne que celle des dates, on peut voir un événement durant toute la journée, la « **Saint Patrick** ». Avec ce type de tableau, on peut afficher les événements ou fêtes qui durent la journée entière, et qui n’ont donc pas de lieu ni d’heure exacts.

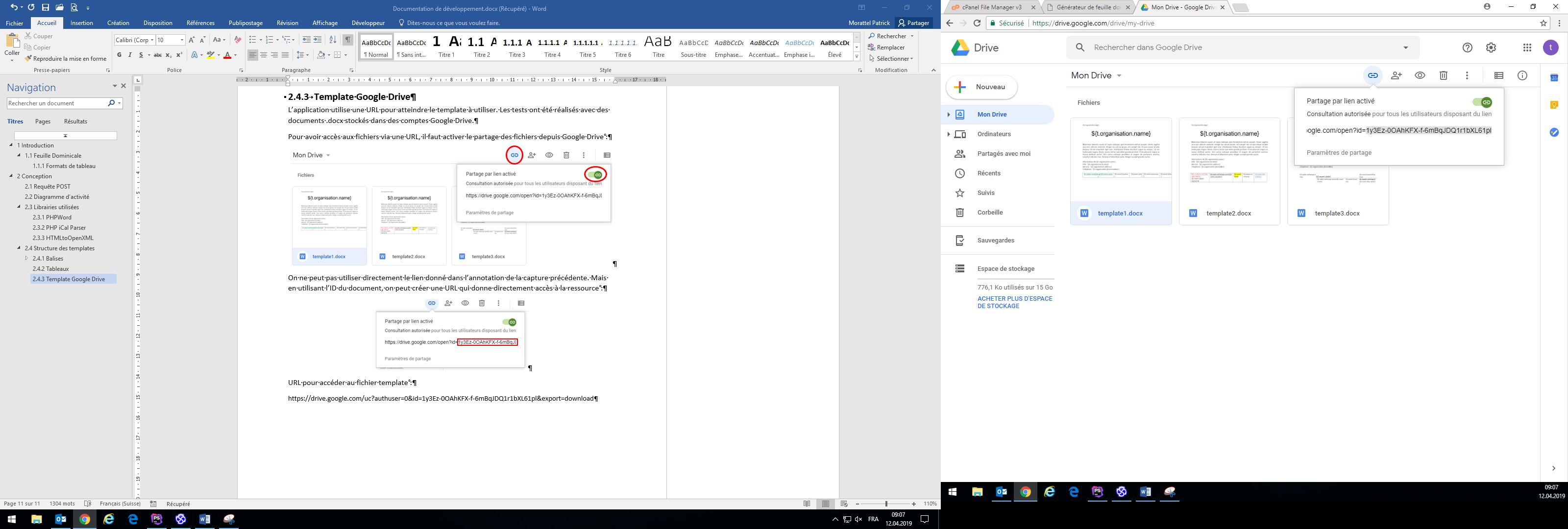
## Template Google Drive

L’application utilise une URL pour atteindre le template à utiliser. Les tests ont été réalisés avec des documents .docx stockés dans des comptes Google Drive.

Pour avoir accès aux fichiers via une URL, il faut activer le partage des fichiers depuis Google Drive :



On ne peut pas utiliser directement le lien donné dans l’annotation de la capture précédente. Mais en utilisant l’ID du document, on peut créer une URL qui donne directement accès à la ressource :



URL pour accéder au fichier template :

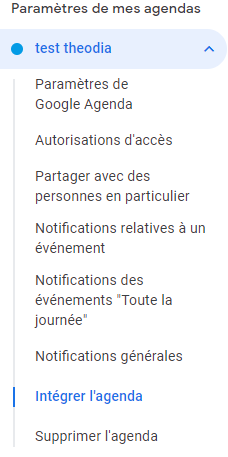
<https://drive.google.com/uc?authuser=0&id=1y3Ez-0OAhKFX-f-6mBqJDQ1r1bXL61pl&export=download>

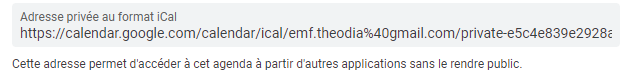
Format des URL : https://drive.google.com/uc?authuser=0&id=**[ID]**&export=download

**NOTE** : Lorsqu’un document Google Drive est remplacé par un autre document de même nom, son ID ne change pas.

# Lecture de l’ICal

Pour tester l’application, un calendrier Google Calendar a été utilisé. Pour obtenir l’URL permettant d’y accéder, il faut aller dans les paramètres du calendrier, sous « Intégrer l’agenda » :





On peut ensuite ajouter / modifier des événements du calendrier pour voir le résultat changer à la génération de la feuille dominicale.

L’application ne gère pas la limitation des événements à lire dans un calendrier, TOUS les événements du calendrier donné seront affichés dans le résultat.

Il reste à définir si theodia.org peut donner en paramètre un calendrier contenant uniquement les événements d’une tranche horaire spécifique, ou si c’est une fonctionnalité à implémenter dans l’application.

## Événements

L’application est capable de gérer différents types d’événements :

* **Les événements non-récursifs :**

Ce sont des événements qui ne se répètent pas et se déroulent à une date et une heure précise.

* **Les événements récursifs :**

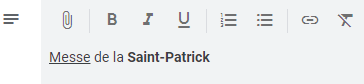
Ce sont des événements qui se répètent **selon une règle définie dans le calendrier** (une fois par semaine le mercredi / une semaine sur deux le dimanche / …). Ils se déroulent aussi à une date et heure précise

* **Les événements durant la journée entière :**

Ce sont des événements spéciaux, comme des fêtes ou des Saints. L’application ne gère pas l’affichage du lieu de ces événements. Ceux-ci ne s’affichent dans le tableau que lorsqu’un autre évènement normal (messe) se déroule le même jour.

### Contenu HTML

Les événements d’un calendrier Google permettent de mettre en forme le texte de leur description.



Cette mise en forme est sauvée en HTML dans le fichier .ics et il est possible de récupérer ce code HTML pour retranscrire la mise en forme dans la feuille dominicale.



# Implémentation

## Diagramme de classes



* **EventReader**

Cette classe lit le contenu d’un fichier .ics à l’aide de son URL, puis crée des objets Event avec les informations récupérées, et renvoie la liste d’Event.

* **Event**

Cette classe est la structure de données des événements.

* **DocGenerator**

Cette classe génère un document Word à l’aide d’un template et d’une liste d’objets Event.

## Etat du développement

Voici l’état des fonctionnalités de l’application :

|  |  |
| --- | --- |
| Fonctionnalité | État actuel |
| Lecture des événements d’un fichier .ics à partir de son URL (événements récursifs compris) | OK |
| Gestion des 3 formats de tableau | OK |
| Gestion des événements spéciaux (Saints / fêtes) | OK |
| Génération d’une feuille dominicale avec les informations des événements | OK |
| Gestion des 3 formats de tableau | OK |
| Gestion d’informations d’organisation en paramètres | OK |
| Gestion du contenu HTML – (texte en gras, italique, souligné) | OK |
| Gestion de plusieurs fichiers .ical | OK |
| Gestion des formats de date et heure | OK |

# Conclusion

Le but de ce document est de présenter l’état du projet « Génération des feuilles dominicales » pour faciliter sa poursuite et sa reprise.

En cas de question, n’hésitez pas à nous contacter :

* Patrick Morattel ([morattelp@studentfr.ch](mailto:morattelp@studentfr.ch))
* Olivier Neuhaus ([neuhauso@edufr.ch](mailto:neuhauso@edufr.ch))