

# Rapport de projet

## Réplication de données

Alexis Grall

March 2018

Ce projet a été réalisé en python en utilisant le package *icalendar*. Il est composé de 5 fichiers dont 3 fichiers de code :

- *client\_secret.json* : fichier permettant à l'application d'utiliser les services google. Pour obtenir ce fichier il est nécessaire de le créer en suivant la procédure de ce lien : [https://developers.google.com/calendar/quickstart/python#step\\_1\\_turn\\_on\\_the\\_api\\_name](https://developers.google.com/calendar/quickstart/python#step_1_turn_on_the_api_name).
- *url.txt* : fichier devant contenir l'url de l'edt voulu, à modifier pour y mettre le sien.
- *get\_ade.py* : code source permettant d'obtenir l'emploi du temps ade depuis l'url fournie dans le fichier *url.txt*. On garde une copie locale de cet edt dans le fichier *ade.ics*. La principale fonction de ce code est *get\_modifs* qui permet de comparer l'edt téléchargé avec la copie locale pour obtenir les nouveaux évènements, les évènements supprimés et les modifications des évènements existants. Pour savoir si un évènement a été modifié, on comparera les valeurs de *LAST-MODIFIED*, puis celles des différents champs d'un évènement pour savoir lesquels ont été modifiés dans l'ade.
- *update\_google.py* Code permettant de se connecter au service google calendars. Lors de la première connexion, le programme créera un nouveau calendrier nommé "EDT Université" et sauvegardera l'id de ce calendrier dans le fichier *cal.id.txt*. La fonction *update\_google\_cal* permet de synchroniser le calendrier google à partir des modifications obtenues préalablement avec *get\_ade.py*. Si un event a été modifié dans l'ade, seuls les champs modifiés seront mis à jour sur le calendrier google. Ainsi les modifications sur le calendrier google seront maintenues sauf si elles concernent un champ qui doit être mis à jour.
- *program.py* lance simplement les différentes fonctions de *update\_google.py* et *get\_ade.py* pour synchroniser le calendrier google.