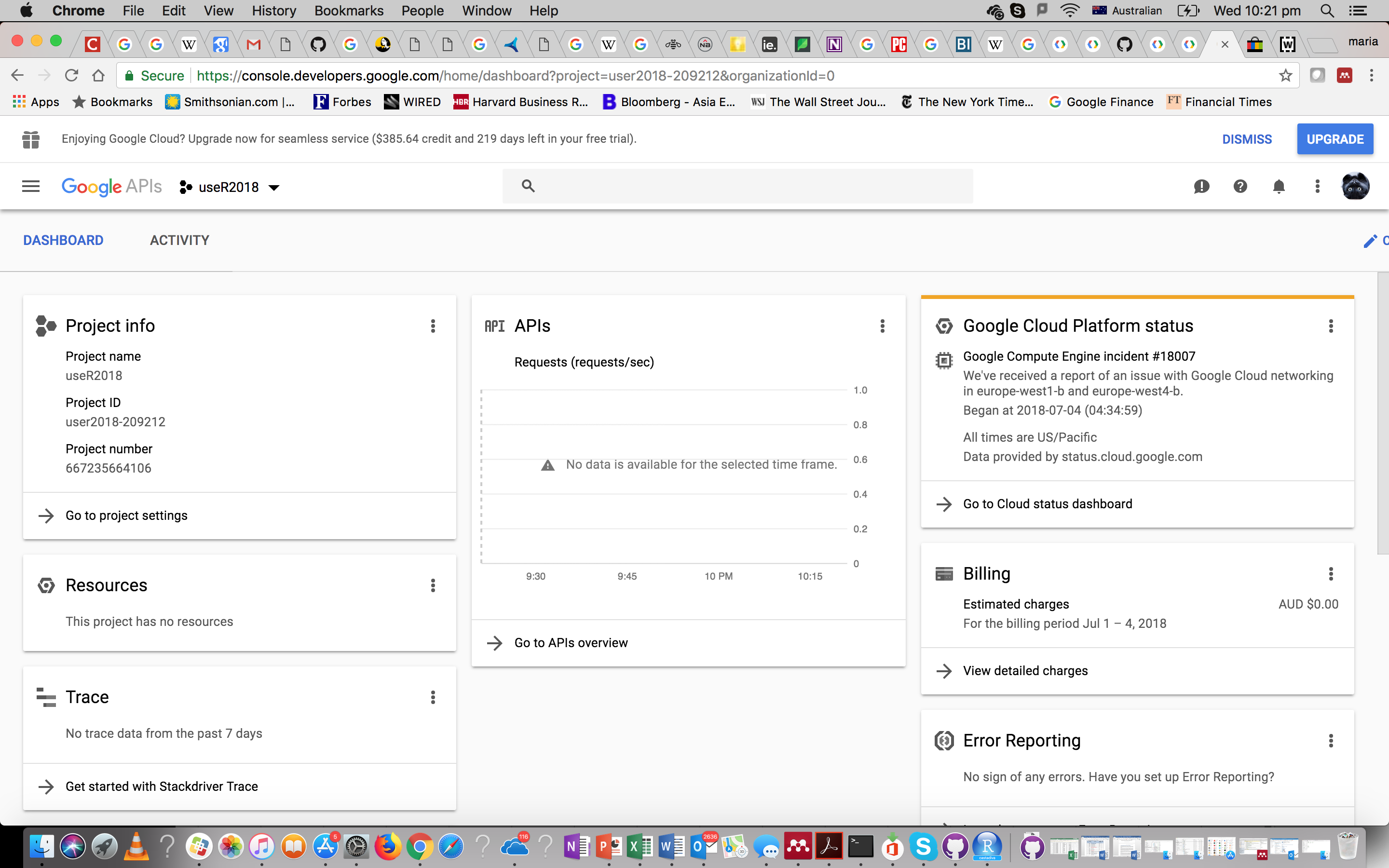
# YouTube

If you do not have a Google account, you can create one at <https://www.google.com/accounts/NewAccount>

1. Go to the Google Developers Console <https://console.developers.google.com/> and create a project there



2. Obtain authorisation credentials

To use YouTube data API you need to have authorization credentials.

The API supports API keys and OAuth 2.0 credentials.

The tuber package uses OAuth 2.0, but you will also need the API keys for some data extraction, e.g. obtaining YouTube channel ID.

To set up your credentials, you go to the Credentials section in your Google Developers Console.

For the OAuth 2.0 setup, set “Authorized redirect URIs” as http://localhost:1410/

Note down the following:

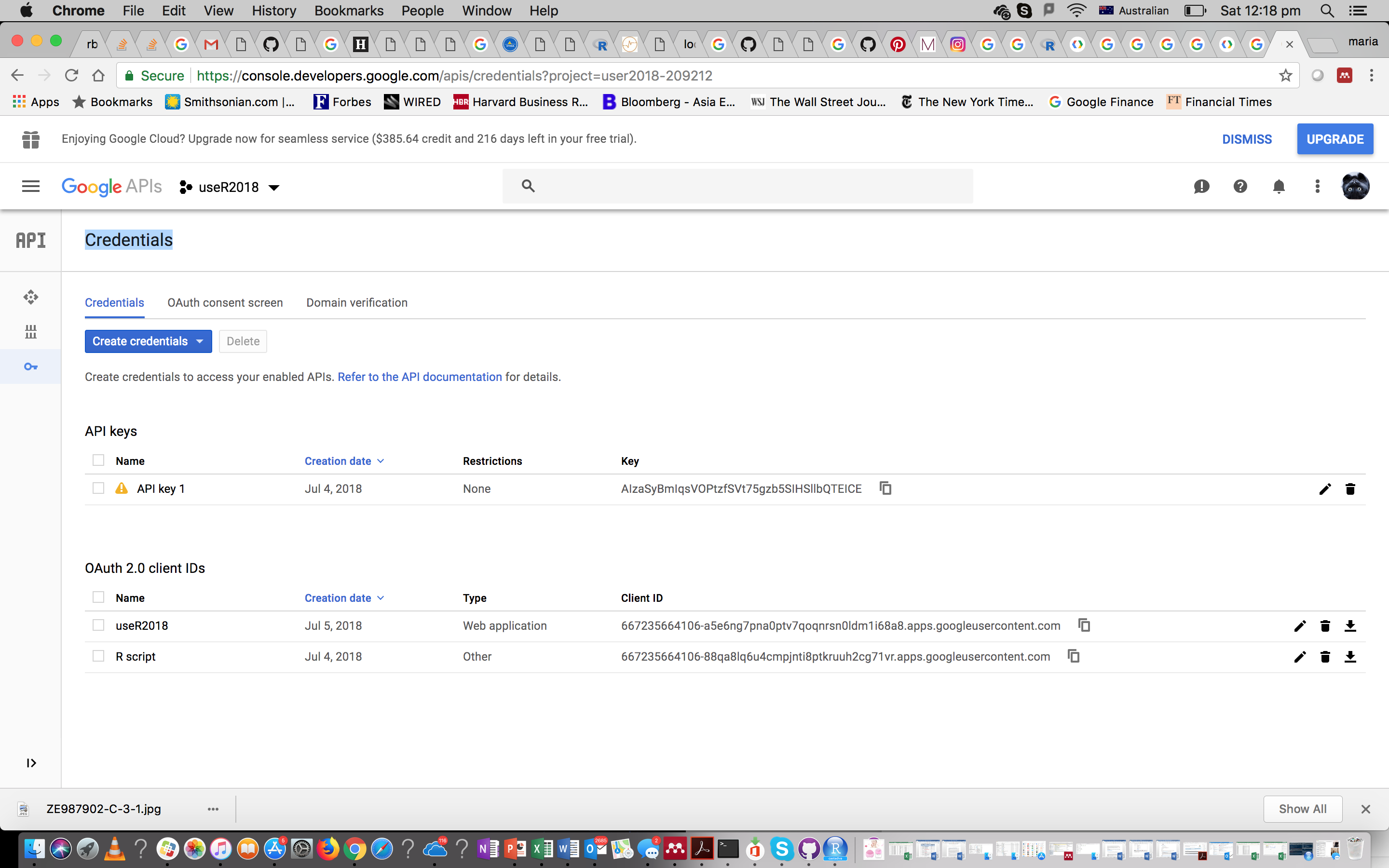
1. Your API key

2. OAuth 2.0 credentials:

Name: useR2018

Client ID:

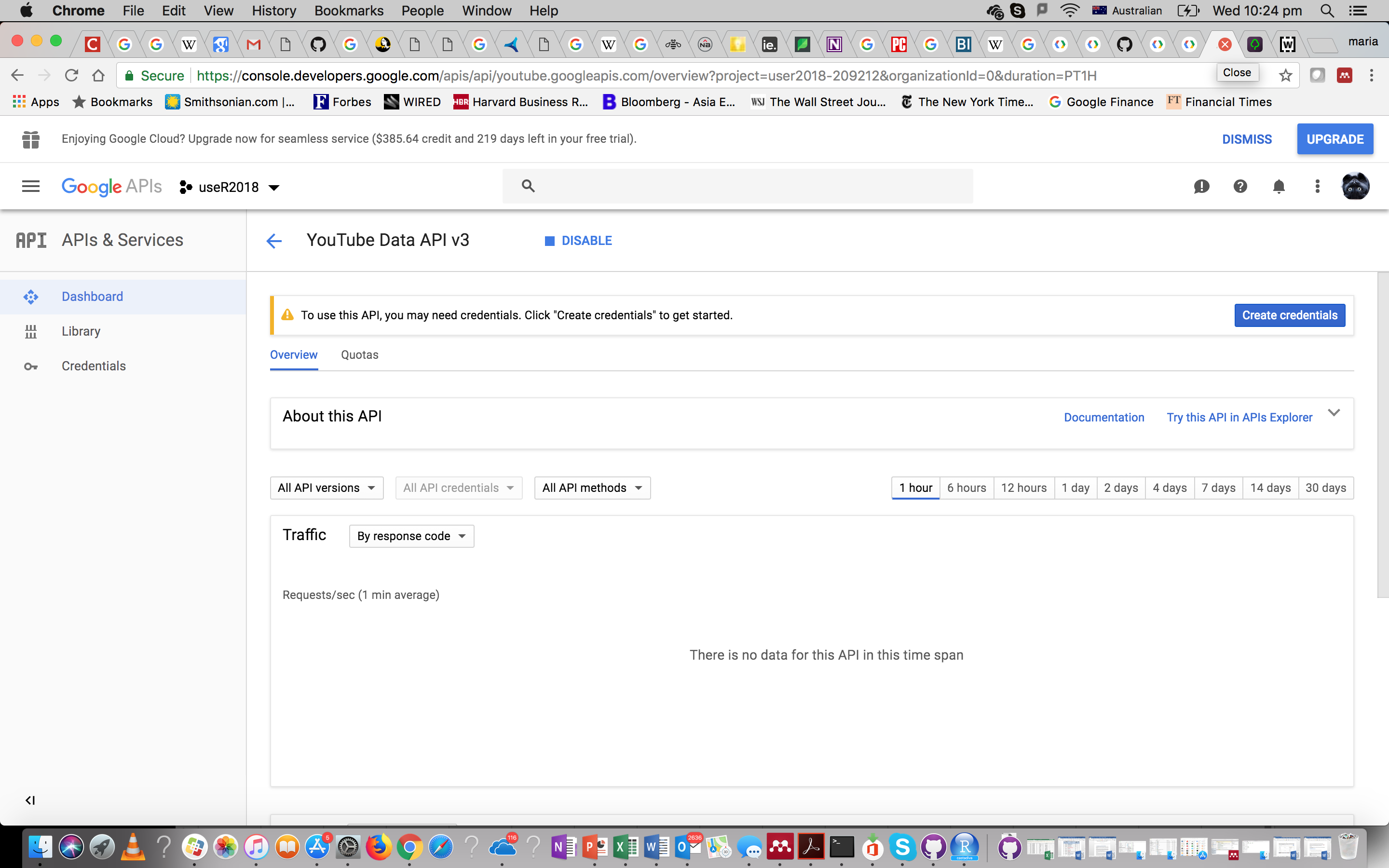
Client secret:



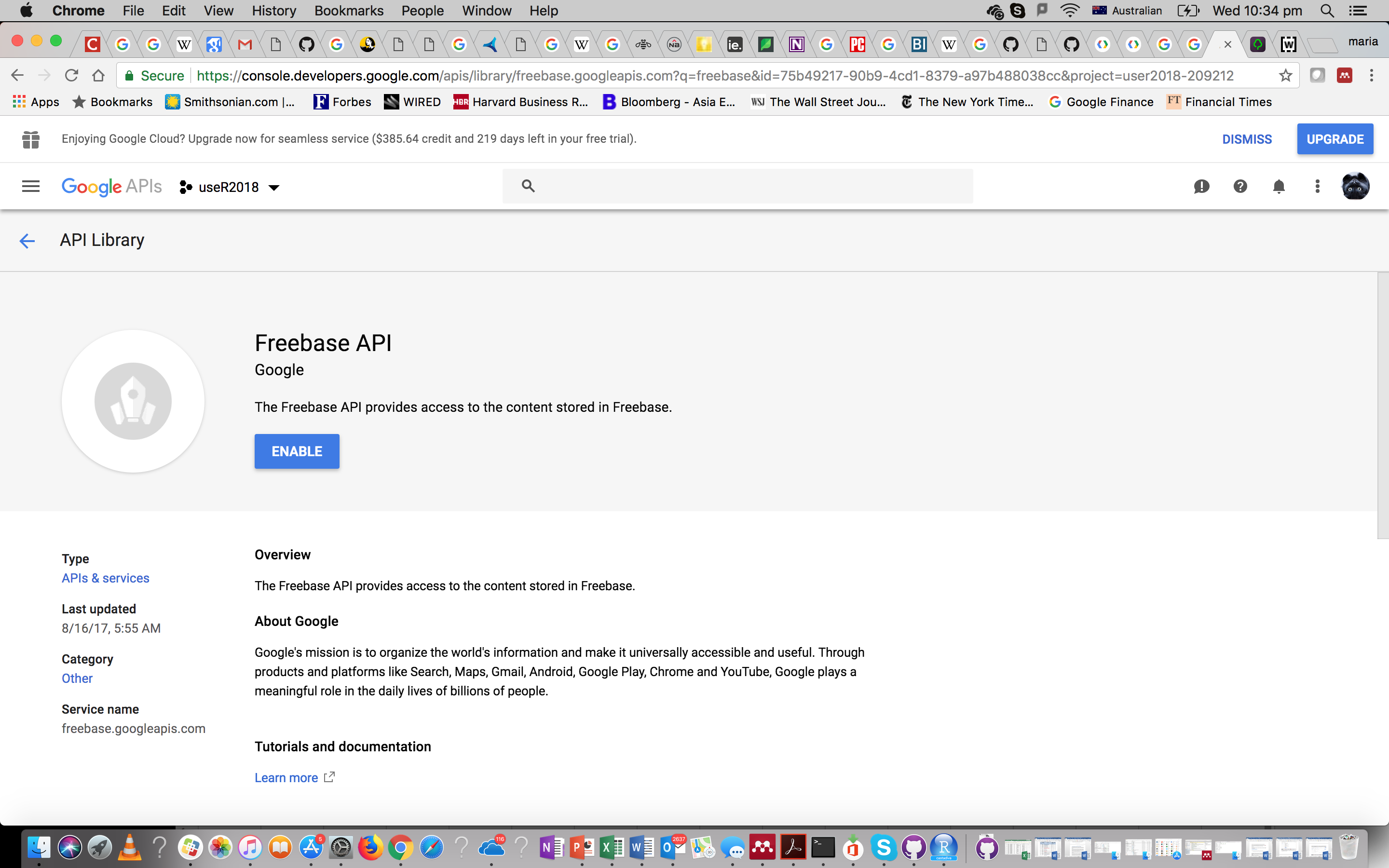
3. Add YouTube Data API to your application

- In the Developers Console, select your project and open the API library.

- select **YouTube Data API v3**



4. FreeBase API to your application as well



Video from Google on how to set up your app

<iframe width="412" height="232" src="https://www.youtube.com/embed/Im69kzhpR3I" frameborder="0" allow="autoplay; encrypted-media" allowfullscreen></iframe>

Further information on Google Developer Console and setting up your App is available at <https://developers.google.com/youtube/v3/getting-started>

## **- Connect your R session with YouTube**

To collect data from youTube directly into R use the tuber package

<https://cran.r-project.org/web/packages/tuber/README.html>

## Setup your R session to download **YouTube** data with the **tuber** package

The function tuber::**yt\_oauth** passes your credentials and connects you to youTube API.

app\_id <- ‘Client ID’

app\_secret = ‘Client secret’

#connect

yt\_oauth(app\_id=app\_id, app\_secret=app\_secret, token=' ', cache=FALSE)

## Waiting for authentication in browser...

## Press Esc/Ctrl + C to abort

## Authentication complete.

By default the function looks for .httr-oauth in the working directory in case you connected before. If it doesn't find it, it passes an application ID and a secret. If you do not want the function to use cache, set cache=FALSE.

The function launches a browser to allow you to authorize the application