# ETL Project: Crypto Mining Database

## Introduction

This ETL project was to set up a database of the required data to analyse the profitability of crypto mining. With each GPU has different efficiency to which algorithm they are using, there are 100s of combinations that could fit into your budget and see which combination of crypto and GPUs is the most profitable.

The main customer that would target this data are Current Crypto miner, starting up Miner, Starting up crypto investor, General investor.

Not only the coin prices fluctuates but GPUs also fluctuates in prices and technology. Mining requirements for each coins are also different e.g. GPU RTX 3060 is compatible to mine ETH, ARK but not CAD, BTC.

Refer to the scheme below:

Graphical user interface, text

Description automatically generated

Figure 1 Database scheme

## Part 1: Extract

COST

Data is scraped from a computers sale/supplier PLE website using beautiful soup and splinter. Splinter was used instead of just beautiful soup and the tiles needs to be loaded up.

CRYPTO

Data is taken from a free API which provide JSON format data from a website called ‘www.whattomine.com’.

GPU

From the website above, the JSON dictionary does not provide data on the GPUs. ‘www.whattomine.com/gpus’ is scraped using pandas html by using response library to provide the .html data

MINING

From the information accumulated, this will be the potential table that could be assembled to analyse your mining potential with the different type of GPUs and Crypto tags.

## Part 2: Transform

Refer to the notebook ‘whattomine.ipynb’ for more detail write up. Typically, raw data are inserted into dataframe using python library Pandas. As most data comes in as ‘str’ and they needed to split e.g. NVDIA RTX8900 will need to be split into two, Make: NVDIA and GPU\_name: RTX8900.

The units of hashrate was also not consistent, H/s and MH/s. We have decided to use MH/s as a default unit, those that have been scraped and shown H/s will be converted by dividing by 10^6.

Some columns were dropped from the raw data dataframe as they are not relevant for this database.

Renaming column names to relevant naming as per our scheme.

## Part 3: Load

Collection will be loaded to database hosted locally through postgresSQL. Postgres was chosen as these collections/table are relational. The naming convention is quite standard and will not be an issue to retrieve or manipulate given it’s a static database.