Ride Along

DAR report for Vehicle Marketplace

Team Specs

Jesus Cerda (Lead)

Jason Barber

Giovanni Contreras

Rainier Marlone Getuaban

Vi Nguyen

March 12, 2024

VPM-7 - Request Vehicle Profile Market Value
    Technologies Options:
1. Rapid API (https://rapidapi.com/dfskGT/api/car-utils/)
2. VinAudit (https://www.vinaudit.com/vehicle-ownership-cost-api)
3. CarsXE API (https://api.carsxe.com/vehicle-market-value)
4. Zyla API Hub (Car Market Value API) (https://zylalabs.com/api-marketplace/data/car+market+value+api/2061)
5. Vehicle Database - Vehicle Market value API (https://vehicledatabases.com/vehicle-market-value-api)

Metrics:

Evaluation on scale from 1-5 with 5 being the best and 1 being the worst.
Weights from 1 to 1.2 with 1.2 being the most important

1. **Max number of request per month**
   a. **Scoring guidelines**:
      i. [5]: services provide unlimited requests per month for free
      ii. [4]: services provide over 1000 requests per month for free
      iii. [3]: services provide less than 1000 request for free
      iv. [2]: services does not provide free request but charge less than 100$ for a month subscription that allowed more than 100,000 requests (0.1$ for 100 requests)
      v. [1]: services does not provide free request but charge more than 100$ for a month subscription that allowed more than 100,000 requests (0.1$ for 100 requests)
   b. **Weights: 1.2** due to this metrics affect tremendously to the cost and performance of our product
2. **Average response time over 1000 iterations**
   a. **Scoring guidelines:**
      i. [5]: services provide result in less than 1 second for 15 iterations
      ii. [4]: services provide result in less than 2 second for 15 iterations
      iii. [3]: services provide result in less than 3 second for 15 iterations
      iv. [2]: services provide result in less than 4 second for 15 iterations
      v. [1]: services provide result in more than 5 second for 15 iterations
   b. **Weights: 1.0** even though this metric is importance since it's affect our performance, but it's not our major concern at the moment
3. **Highest range of vehicle production year**
   a. **Scoring guidelines:**
      i. [5]: Support all vehicles before and after 1980
      ii. [4]: Support vehicles from latest to 1912
      iii. [3]: Support vehicles from latest to 1981
      iv. [2]: Support vehicles from latest to early 2000s

v. [1]: Does not support latest vehicles
b. **Weights: 1.0** even though this metric affect our range of supported vehicles, but it's not our major concern at the moment

4. **Number of items per request**
   a. **Scoring guidelines:**
      i. [5]: Allow more than 100 items per request
      ii. [4]: Allow more than 70 items per request
      iii. [3]: Allow less than 50 items per request
      iv. [2]: Allow less than 20 items per request
      v. [1]: Allow 1 item per request
   b. **Weights: 1.1** this metrics also affect both the cost and performance of our product, due to the more items per request, we can make less queries for both testing and in deployment.

5. **Request per minute - How many request can we make to the API in a minute (Rate Limiting)**
   a. **Scoring guidelines:**
      i. [5]: More than 100 requests per minute
      ii. [4]: More than 60 but under 100 requests per minute
      iii. [3]: 60 requests per minute
      iv. [2]: Less than 60 requests per minute
      v. [1]: Less than 10 requests per minute
   b. **Weights: 1.1** this metric affect the performance of our product

	Rapid API	VinAudit	CarsXE	Zyla API Hub	Vehicle Databases
Max number of request per month	500,000 calls per month	NO FREE TIER, student trial for 20 requests then basic tier is 30$ a month for 1000 requests	NO FREE TIER only 7 days trial after that 0.10$ for 1000 call for Vehicle Specification	No FREE TIER only 7 days trial, after that it's 25$ a month for 500 requests	15 request per month
[Weight: 1.2]	[5]	[2]	[2]	[3]	[3]
Average response time over 15 iterations	3.8101072311401367 seconds	2.628866672515869 seconds	3.30351053298123 seconds	2.5773184299468994 seconds	2.486091375350952 seconds

[Weight: 1.0]	[2]	[3]	[2]	[3]	[3]
Highest range of vehicle production year	2024 - 1981	2024 - 1912	2024 - 1981	2024 - 1981	2023-1912
[Weight: 1.0]	[3]	[4]	[3]	[3]	[4]
Number of items per request	1 VIN per request	1 VIN per request	1 VIN per request	1 VIN per request	1 VIN per request
[Weight: 1.1]	[1]	[1]	[1]	[1]	[1]
Request per minute - Rate Limiting	50 requests per minute	60 requests per minute	60 requests per minute	50 requests per minute	50 requests per minute
[Weight: 1.1]	[2]	[3]	[3]	[2]	[2]
TOTAL	14.3/27	13.8/27	11.8/27	12.9/27	13.9/27

Analysis:
- **Max number of request per month**: RapidAPI dominated this metrics with 500,000 requests per month for free.
- **Average response time over 15 iterations:** We use the same black box testing methods, running a python script to make request to the API 15 times and record the time it takes to complete all 15 requests. Then we compare the time accordingly.
- **Highest range of vehicle production year:** Both Vehicle Database API and VinAudit has the highest range as they also allowed VIN number before 1981 for classic vehicles. As for the rest, they all support the 17 characters VIN which is essentially from 1981 to current.
- **Number of items per request:** While the rest only allow the standard one 1 VIN number per request, NHTSA does have a bulk request feature which allow up to 50 VIN numbers for one request.
- **Request per minute - Rate Limiting:** Since NHTSA that does not mention rate limit in their document nor answer our email, we try to run a python script to query to the API as many time as possible in one minute until either the time run out or the API reject the request. The number of time we get a response back from the API is recorded and display at the end in order for us to compared.

Conclusion
In conclusion, RapidAPI is our choice for this user feature due to the large volume of requests allowed per month, even though once again, we do compromised that the feature won't be able to support classic cars with VIN number before 1980 which is not as important, thanks to the allowed volume of calls make things easier for testing and deploying our product. Our team second options would be the VehicleDatabase, even though it will cost us money for using their services, it is still cheaper compared to the rest of our options, with this we also won't have to compromise our support for classic vehicles since the supported VIN ranged are wider.

Reference:

1. Rapid API
   a. Pricing: (https://rapidapi.com/products/pricing/#pricing-features)
   b. API Request Documentation (https://echo.paw.cloud/)
   c. Supported Vehicle Year Range:
      (https://www.autocheck.com/vehiclehistory/vin-basics)
   d. Rate Limiting: (https://docs.rapidapi.com/docs/rate-limiting)
2. Zyla
   a. Pricing: (https://zylalabs.com/api-marketplace/data/car+market+value+api/2061
3. Vehicle Databases:
   a. Supported Year:(https://vehicledatabases.com/vin-decode-api)
4. CarsXE
   a. Pricing: (https://api.carsxe.com/pricing)
   b. API Request Documentation:
      (https://docs.carsxe.com/specifications#vehicle-specifications)
5. VinAudit
   a. Documentation: (https://www.vinaudit.com/vehicle-ownership-cost-api)