

Kiva Crowdfunding Dataset Analysis

Poojitha Katta
Department of Applied Data Science
San Jose State University
San Jose, United States
poojitha.katta@jsu.edu

Purnima Bhukya
Department of Applied Data Science
San Jose State University
San Jose, United States
purnima.bhukya@jsu.edu

Deepali Zutshi
Department of Applied Data Science
San Jose State University
San Jose, United States
deepali.zutshi@jsu.edu

Yasaman Emami
Department of Applied Data Science
San Jose State University
San Jose, United States
yasaman.emami@jsu.edu

Abstract—The main aim of this project is to create a database for the crowdfunding platform-Kiva, to analyze and evaluate factors which influence various aspects of a crowdfunded project and draw conclusions about them.

Keywords—*NoSQL, Crowdfunding, Database, Kiva*

I. INTRODUCTION

Crowdfunding is a system that enables individuals or ventures to seek small investments, contributions, or loans from a variety of funders online. Kiva is a crowdfunding platform that offers a new financing channel for small and micro businesses as well as individuals. The aim of the project is to build a database management system for Kiva platform to analyze the factors that influence crowdfunded projects by estimating the welfare level of partners in specific regions, based on shared financial and demographic aspects. Technologies such as MongoDB would be used to create the database management system and perform analysis of the data. Using Python, a powerful programming tool, we can observe, understand, and draw conclusions on better funded categories, borrowing patterns and regional analysis from the data. The system can be deployed on cloud-based platforms such as Atlas for better accessibility and security. Apart from that, Charts on MongoDB Compass allow detailed visualization of the data and give us deeper insights. This project can help improve access to crowdfunding, assess borrower welfare levels, by analyzing the growth of previously funded projects and benefit Kiva with a better database system to enhance their platform.

II. DATASET

A. Source

The data was obtained from Kaggle, an online community of data scientists and machine learning practitioners. It was made available by Kiva, an online crowdfunding platform, for the “Data Science for Good” Challenge and invited people to help them build a localized model to estimate various metrics in regions where Kiva has active loans.

B. Dataset Description

The dataset contains 4 csv files with 54 attributes total, which includes 30 string, 7 decimal, 4 datetime and 13 other data types. The first table consists of 20 columns, detailing the id, funded amount, loan amount, country code, country, currency, region, etc. Similarly, the second, third and fourth table outlines data snapshot and can be matched to the loan theme regions to get a loan's location and provides details for id, loan theme id, loan theme type, partner id and MPI (Multidimensional Poverty Index). Extracting several insights from the historical micro-loans over a period and correlating the regional averages by gender, sector, or borrowing behavior to estimate the welfare rate is to be followed.

C. Data Cleaning

The data required cleaning and correcting to be processed and stored into the database. Many of the tables had missing data which was replaced by the mode of the data in that column. Attributes with most of the data missing were removed from the tables altogether. Many of the tables also contained attributes which were duplicated in the same table as well as across multiple tables. These were removed and the data was pre-processed to remove redundancies.

```
In [1]: import pandas as pd
import numpy as np
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

In [2]: df = pd.read_csv('kiva_loans.csv')
data = pandas.read_csv('kiva_loans.csv', encoding='utf-8', quotechar='"', delimiter=',')
df.info()

In [3]: df.describe()

In [4]: null = df.isnull().sum().sort_values(ascending = False).reset_index()
null.columns = ['Column', 'Frequency']

In [5]: ## temp column consists of 10000 null values so remove the column.
df.drop('temp', axis=1, inplace=True)

In [6]: ## this column null values are replaced with the mode
df['funded_time'].mode()

In [7]: df['borrower_genders'].mode()

In [8]: df['funded_time'].fillna(df['funded_time'].mode(), inplace = True)
df['borrower_genders'].fillna(df['borrower_genders'].mode(), inplace = True)

In [9]: df.dropna(inplace = True)

In [10]: df.info()

In [11]: data.to_csv('kiva_mpi_region_locations', encoding='utf-8', index = False)
```

Fig. 1 Jupyter Notebook for Data Cleaning

III. DATA MODEL

A NoSQL database system such as MongoDB requires work differently than a traditional relational database system. Where an RDBMS is incapable of handling and storing unstructured and semi-structured data, NoSQL can store, process, and visualize such data with ease. MongoDB represents the information as a JSON document instead of the row and column format adopted by the RDBMS system. MongoDB stores data in structures called ‘collections’ and the data entries are called ‘documents’. The documents contain key-value pairs of various types and can also consist of arrays, nested documents, etc. Each document can have its own structure and are self-describing in nature.

A. Document Structure

The following diagram represents the various collections created by importing the data into MongoDB. Each attribute in the collections represents the various documents keys and is of variable types (string, integer, date, timestamp, etc.).

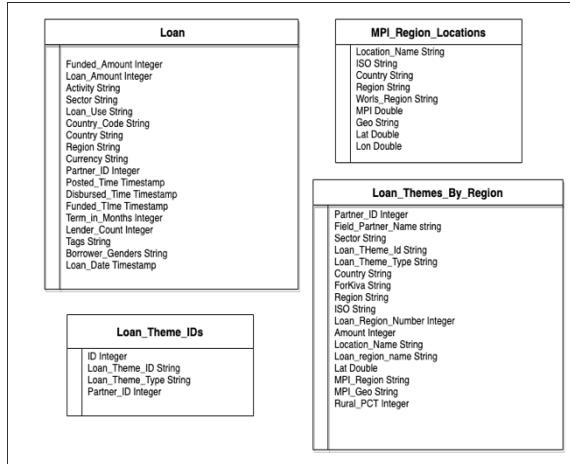


Fig. 2 Data Model

Each collection in the database contains documents with a JSON-like structure having key-value pairs for each of the entries in the collection. A total of four collections were created initially and imported into MongoDB for this project. The following are the structures of the documents in each of the collections. In the end, all of the four collections merge into one collection called the “all_kiva” collection.

Cluster0.kiva_loans_unicode

Document 1:

```

{
    "_id": "610e911c14a203007f000000",
    "id": 12345,
    "customer": "John Doe",
    "activity": "Business",
    "sector": "Retail",
    "loan_use": "To purchase pigs to breed and resell when they gain weight",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "pig, business",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Document 2:

```

{
    "_id": "610e911c14a203007f000001",
    "id": 12346,
    "customer": "Sarah Smith",
    "activity": "Personal Housing Expenses",
    "sector": "Housing",
    "loan_use": "To purchase a home and save to build a new house",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "house, family",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Fig. 3 Kiva_loans

Cluster0.kiva_mpi_region_locations_unicode

Document 1:

```

{
    "_id": "610e911c14a203007f000000",
    "id": 12345,
    "customer": "John Doe",
    "activity": "Business",
    "sector": "Retail",
    "loan_use": "To purchase pigs to breed and resell when they gain weight",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "pig, business",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Document 2:

```

{
    "_id": "610e911c14a203007f000001",
    "id": 12346,
    "customer": "Sarah Smith",
    "activity": "Personal Housing Expenses",
    "sector": "Housing",
    "loan_use": "To purchase a home and save to build a new house",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "house, family",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Fig. 4 Kiva_mpi_region

Cluster0.loan_themes_by_region_unicode

Document 1:

```

{
    "_id": "610e911c14a203007f000000",
    "id": 12345,
    "customer": "John Doe",
    "activity": "Business",
    "sector": "Retail",
    "loan_use": "To purchase pigs to breed and resell when they gain weight",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "pig, business",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Document 2:

```

{
    "_id": "610e911c14a203007f000001",
    "id": 12346,
    "customer": "Sarah Smith",
    "activity": "Personal Housing Expenses",
    "sector": "Housing",
    "loan_use": "To purchase a home and save to build a new house",
    "country": "United States",
    "currency": "USD",
    "posted_time": "2024-01-01T07:45:00Z",
    "funded_time": "2024-01-01T07:45:00Z",
    "due_time": "2024-01-15T07:45:00Z",
    "tags": "house, family",
    "lender_count": 12,
    "borrower_genders": "Female",
    "loan_date": "2024-01-01T07:45:00Z"
}

```

Fig. 5 Kiva_loan_themes_by_region

YASAMAN'S ORG - 2021-10-19 > PROJECT 0 > DATABASES

Data225

- Overview
- Real Time
- Metrics
- Collections**
- Search
- Profiler
- Performance Advisor
- Online Archive
- Command Line

DATABASES: 2 COLLECTIONS: 11

+ Create Database

NAMESPACES

kiva

- all_kiva
- firstAgg
- firstAgg
- firstAgg
- firstAgg
- firstAgg
- kiva_loans
- kiva_mpi_region_locations
- loan_theme_by_region
- loan_theme_ids
- sample

kiva.kiva_mpi_region_locations

COLLECTION SIZE: 165.1KB TOTAL DOCUMENTS: 892 INDEXES TOTAL SIZE: 24KB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

FILTER { field: 'value' }

QUERY RESULTS 1-20 OF MANY

```

_id: ObjectID("61a772d151cd99db7840bc38")
locationname:"Babushtan, Afghanistan"
ISO:"AF"
country:"Afghanistan"
region:"Babushtan"
world_region:"South Asia"
lat: 35.734725
lon: 63.781993
tso: 78.811993

_id: ObjectID("61a772d151cd99db7840bc31")
locationname:"Badghis, Afghanistan"
ISO:"AF"
country:"Afghanistan"
region:"Badghis"
world_region:"South Asia"
lat: 35.167139
lon: 63.769334
tso: 69.287753

_id: ObjectID("61a772d151cd99db7840bc32")
locationname:"Baghlan, Afghanistan"
ISO:"AF"
country:"Afghanistan"
region:"Baghlan"
world_region:"South Asia"
lat: 35.167139
lon: 63.769334
tso: 69.287753

_id: ObjectID("61a772d151cd99db7840bc33")

```

Fig.6 kiva_mpi_region_locations

YASAMAN'S ORG - 2021-10-19 > PROJECT 0 > DATABASES

Data225

- Overview
- Real Time
- Metrics
- Collections**
- Search
- Profiler
- Performance Advisor
- Online Archive
- Command Line

DATABASES: 2 COLLECTIONS: 11

+ Create Database

NAMESPACES

kiva

- all_kiva
- firstAgg
- firstAgg
- firstAgg
- firstAgg
- firstAgg
- kiva_loans
- kiva_mpi_region_locations
- loan_theme_by_region
- loan_theme_ids
- sample

kiva.all_kiva

COLLECTION SIZE: 102.69MB TOTAL DOCUMENTS: 807542 INDEXES TOTAL SIZE: 9.59MB

Find Indexes Schema Anti-Patterns Aggregation Search Indexes

FILTER { field: 'value' }

QUERY RESULTS 1-20 OF MANY

```

_id: ObjectID("61a772d151cd99db7856993")
id: 653192
funded_amount: 100
loan_amount: 100
activity: "Pigs"
sector: "Agriculture"
use: "to purchase pigs to breed and resell when they gain weight"
country_code: "AF"
country_name: "Afghanistan"
region: "Takhar province"
partner: 1469
posted_time: 2014-01-07T01:13:09.000+00:00
disbursed_time: 2013-12-17T01:00.00-00:00
funded_time: 2014-01-07T15:58:16.000+00:00
term_in_months: 12
lender_count: 1
borrower_genders: "Female"
repayment_start_time: "2014-01-07T00:00:00+00:00"
date: 2014-01-07T00:00:00.000+00:00

_id: ObjectID("61a772d151cd99db7856998")
id: 653181
funded_amount: 1500
loan_amount: 1500
activity: "Personal Housing Expenses"
sector: "Housing"
use: "to buy timber and zinc to build a new house"
country_code: "AF"
country_name: "Afghanistan"
region: "Kangar Chahar Province"
current: 1000
partner: 1469
posted_time: 2014-01-07T01:55:54.000+00:00
disbursed_time: 2013-12-17T01:00.00-00:00
funded_time: 2014-01-07T15:58:16.000+00:00
term_in_months: 20
lender_count: 1
borrower_genders: "Female"
repayment_start_time: "2014-01-07T00:00:00+00:00"
date: 2014-01-07T00:00:00.000+00:00

```

Fig. 7 all_kiva

B. Denormalization

Denormalization is the process of improving the read performance of a database at the expense of some of the write performance by either making redundant copies of the data or by grouping the data. For this project, the previously normalized data from Project-1 was denormalized to improve database functionalities.

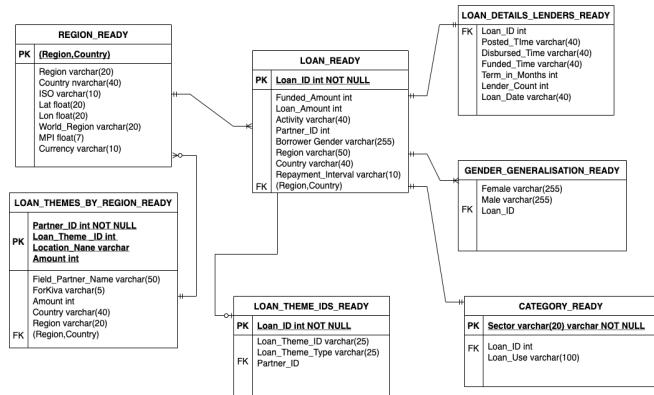


Fig. 8 Normalized Data (Project-1)

The diagram below represents the Kiva database and the collection ‘All_Kiva’ which is denormalized with the various keys to store the data.

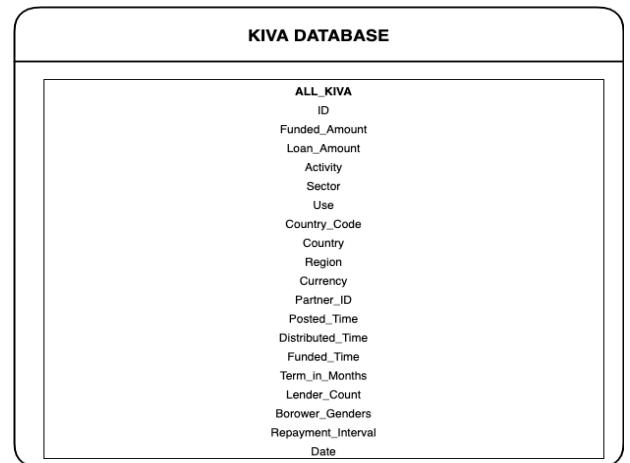


Fig. 9 Document Structure of Kiva- MongoDB

The figure below shows the records as in the collection of the denormalized data. MongoDB removes the fields where the key has a null value and displays only the pairs with valid data as opposed to the relational database which displays the missing values as ‘Null’.

```

[Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find()
{
  _id: ObjectId("61878d1c51c09adb7856b987"),
  id: 65317,
  funded_amount: 800,
  loan_amount: 800,
  activity: 'Pigs',
  sector: 'Agriculture',
  use: 'To purchase pigs to breed and resell when they gain weight',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Takeo province',
  currency: 'USD',
  partner_id: 9,
  posted_time: ISODate("2014-01-02T07:13:49.000Z"),
  disbursed_time: ISODate("2013-12-13T08:00:00.000Z"),
  funded_time: ISODate("2014-01-02T17:50:16.000Z"),
  term_in_months: 12,
  lender_count: 32,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b988"),
  id: 65318,
  funded_amount: 1500,
  loan_amount: 1500,
  activity: 'Personal Housing Expenses',
  sector: 'Housing',
  use: 'to purchase timber and zinc to build a new house',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Kampong Chhnang Province',
  currency: 'USD',
  partner_id: 9,
  posted_time: ISODate("2014-01-02T06:55:54.000Z"),
  disbursed_time: ISODate("2013-12-18T08:00:00.000Z"),
  funded_time: ISODate("2014-01-03T21:27:40.000Z"),
  term_in_months: 20,
  lender_count: 26,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b989"),
  id: 65319,
  funded_amount: 1500,
  loan_amount: 1500,
  activity: 'Motorcycle Transport',
  sector: 'Transportation',
  use: 'To buy a motorcycle for his son's commute to school.  ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Khsach Kandal district',
  currency: 'USD',
  partner_id: 61,
  posted_time: ISODate("2014-01-02T07:17:22.000Z"),
  disbursed_time: ISODate("2013-12-16T08:00:00.000Z"),
  funded_time: ISODate("2014-01-23T16:28:01.000Z"),
  term_in_months: 22,
  lender_count: 22,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b98a"),
  id: 65316,
  funded_amount: 600,
  loan_amount: 600,
  activity: 'Cattle',
  sector: 'Agriculture',
  use: 'To buy a calf to raise. ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Phnom Penh',
  currency: 'USD',
  partner_id: 60,
  posted_time: ISODate("2014-01-02T06:18:15.000Z"),
  disbursed_time: ISODate("2013-12-06T08:00:00.000Z"),
  funded_time: ISODate("2014-01-02T15:12:10.000Z"),
  term_in_months: 22,
  lender_count: 22,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b98b"),
  id: 65315,
  funded_amount: 3000,
  loan_amount: 3000,
  activity: 'Vehicle',
  sector: 'Personal Use',
  use: 'To buy a motorbike for family transportation. ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Phnom Penh',
  currency: 'USD',
  partner_id: 61,
  posted_time: ISODate("2014-01-02T07:05:02.000Z"),
  disbursed_time: ISODate("2013-12-13T08:00:00.000Z"),
  funded_time: ISODate("2014-01-10T03:22:29.000Z"),
  term_in_months: 22,
  lender_count: 33,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: null, count: 774185 },
  { _id: 125, count: 215 },
  { _id: 289, count: 3598 },
  { _id: 189, count: 3453 },
  { _id: 175, count: 3351 },
  { _id: 187, count: 1873 },
  { _id: 580, count: 1858 },
  { _id: 256, count: 1839 },
  { _id: 159, count: 1533 },
  { _id: 1680, count: 799 },
  { _id: 760, count: 527 },
  { _id: 121, count: 527 },
  { _id: 480, count: 519 },
  { _id: 656, count: 510 },
  { _id: 120, count: 479 },
  { _id: 158, count: 463 },
  { _id: 625, count: 455 },
  { _id: 100, count: 443 },
  { _id: 488, count: 393 },
  { _id: 525, count: 364 }
]

```

Fig. 10 all_kiva key-value pairs

IV. NoSQL QUERIES

In this section all the SQL queries performed on the data in relational database model have been applied to denormalized form of data stored in no sql format in MongoDB. The queries run on MongoShel connected to the cloud cluster of MongoDB. The queries considered to perform are similar so the comparison between NoSQL data format and RDBMS can be performed to get a better understanding of performance metrics and execution time as well as the syntax. Figure below shows the all key-value pairs from kiva where the load_amount is between 500\$ and 1500\$. Also Fig. 9 is displaying the count of each loan.

```

[Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find({loan_amount:{$gt:500, $lt:1500}})
{
  _id: ObjectId("61878d1c51c09adb7856b987"),
  id: 65317,
  funded_amount: 800,
  loan_amount: 800,
  activity: 'Pigs',
  sector: 'Agriculture',
  use: 'To purchase pigs to breed and resell when they gain weight',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Takeo province',
  currency: 'USD',
  partner_id: 9,
  posted_time: ISODate("2014-01-02T07:13:49.000Z"),
  disbursed_time: ISODate("2013-12-13T08:00:00.000Z"),
  funded_time: ISODate("2014-01-02T17:50:16.000Z"),
  term_in_months: 12,
  lender_count: 32,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b988"),
  id: 65318,
  funded_amount: 1500,
  loan_amount: 1500,
  activity: 'Personal Housing Expenses',
  sector: 'Housing',
  use: 'to purchase timber and zinc to build a new house',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Kampong Chhnang Province',
  currency: 'USD',
  partner_id: 9,
  posted_time: ISODate("2014-01-02T06:55:54.000Z"),
  disbursed_time: ISODate("2013-12-18T08:00:00.000Z"),
  funded_time: ISODate("2014-01-03T21:27:40.000Z"),
  term_in_months: 20,
  lender_count: 26,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b989"),
  id: 65319,
  funded_amount: 1500,
  loan_amount: 1500,
  activity: 'Motorcycle Transport',
  sector: 'Transportation',
  use: 'To buy a motorcycle for his son's commute to school.  ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Khsach Kandal district',
  currency: 'USD',
  partner_id: 61,
  posted_time: ISODate("2014-01-02T07:17:22.000Z"),
  disbursed_time: ISODate("2013-12-16T08:00:00.000Z"),
  funded_time: ISODate("2014-01-23T16:28:01.000Z"),
  term_in_months: 22,
  lender_count: 22,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b98a"),
  id: 65316,
  funded_amount: 600,
  loan_amount: 600,
  activity: 'Cattle',
  sector: 'Agriculture',
  use: 'To buy a calf to raise. ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Phnom Penh',
  currency: 'USD',
  partner_id: 60,
  posted_time: ISODate("2014-01-02T06:18:15.000Z"),
  disbursed_time: ISODate("2013-12-06T08:00:00.000Z"),
  funded_time: ISODate("2014-01-02T15:12:10.000Z"),
  term_in_months: 22,
  lender_count: 22,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: ObjectId("61878d1c51c09adb7856b98b"),
  id: 65315,
  funded_amount: 3000,
  loan_amount: 3000,
  activity: 'Vehicle',
  sector: 'Personal Use',
  use: 'To buy a motorbike for family transportation. ',
  country_code: 'KH',
  country: 'Cambodia',
  region: 'Phnom Penh',
  currency: 'USD',
  partner_id: 61,
  posted_time: ISODate("2014-01-02T07:05:02.000Z"),
  disbursed_time: ISODate("2013-12-13T08:00:00.000Z"),
  funded_time: ISODate("2014-01-10T03:22:29.000Z"),
  term_in_months: 22,
  lender_count: 33,
  borrower_genders: 'female',
  repayment_interval: 'monthly',
  date: ISODate("2014-01-02T00:00:00.000Z")
},
{
  _id: null, count: 500 }
]

```

Fig. 11 loan_amount between 500\$,1500\$

```

[Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.aggregate([
  { $group: { _id: "$loan_amount", count:{$sum:1} } },
  { $sort:{'count':-1} }
])
{
  _id: null, count: 774185 },
  { _id: 125, count: 215 },
  { _id: 289, count: 3598 },
  { _id: 189, count: 3453 },
  { _id: 175, count: 3351 },
  { _id: 187, count: 1873 },
  { _id: 580, count: 1858 },
  { _id: 256, count: 1839 },
  { _id: 159, count: 1533 },
  { _id: 1680, count: 799 },
  { _id: 760, count: 527 },
  { _id: 121, count: 527 },
  { _id: 480, count: 519 },
  { _id: 656, count: 510 },
  { _id: 120, count: 479 },
  { _id: 158, count: 463 },
  { _id: 625, count: 455 },
  { _id: 100, count: 443 },
  { _id: 488, count: 393 },
  { _id: 525, count: 364 }
]
Type 'it' for more
Atlas atlas-Salle7-shard-0 [primary] kiva> it
{
  _id: 1280, count: 361 },
  { _id: 880, count: 349 },
  { _id: 275, count: 338 },
  { _id: 120, count: 333 },
  { _id: 1825, count: 277 },
  { _id: 456, count: 272 },
  { _id: 100, count: 271 },
  { _id: 875, count: 243 },
  { _id: 550, count: 224 },
  { _id: 386, count: 213 },
  { _id: 120, count: 203 },
  { _id: 1180, count: 207 },
  { _id: 1280, count: 195 },
  { _id: 120, count: 193 },
  { _id: 850, count: 189 },
  { _id: 1888, count: 167 },
  { _id: 120, count: 165 },
  { _id: 575, count: 149 },
  { _id: 1380, count: 148 },
  { _id: 1125, count: 148 }
]
Type 'it' for more
Atlas atlas-Salle7-shard-0 [primary] kiva> it
{
  _id: 725, count: 146 },
  { _id: 975, count: 146 },
  { _id: 120, count: 145 },
  { _id: 1475, count: 124 },
  { _id: 1380, count: 116 },
  { _id: 120, count: 115 },
  { _id: 1275, count: 108 },
  { _id: 675, count: 106 },
  { _id: 120, count: 105 },
  { _id: 1375, count: 99 },
  { _id: 475, count: 86 },
  { _id: 120, count: 85 },
  { _id: 1875, count: 83 },
  { _id: 1225, count: 77 },
  { _id: 1888, count: 77 },
  { _id: 120, count: 75 },
  { _id: 1975, count: 71 },
  { _id: 1525, count: 69 },
  { _id: 120, count: 68 }
]
Type 'it' for more
Atlas atlas-Salle7-shard-0 [primary] kiva> it

```

Fig. 12 Count of Each Loan

In Fig.13 below is a query to display information on members from specific regions.

```

    {
      "_id": ObjectId("63fb78d6551c94db78575b4e"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Toliara",
      "partner_id": 355,
      "partner_name": "Brioche Brioche & Partenaires",
      "loan_id": "10000000000000000000000000000000",
      "loan_theme_type": "Small Enterprise",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region": "Tolitara",
      "location_name": "Tolitara, Madagascar",
      "lat": Decimal128("26.3166737"),
      "lon": Decimal128("46.4628937")
    },
    {
      "_id": ObjectId("63fb78d6551c94db78575bc1"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Antananarivo",
      "partner_id": 356,
      "partner_name": "Vahatra",
      "loan_id": "10000000000000000000000000000001",
      "loan_theme_id": "100000000002XXX",
      "loan_theme_type": "Extreme Poverty",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region_number": "76-0",
      "amount": 300000,
      "location_name": "Antananarivo, Madagascar",
      "lat": Decimal128("-19.884268"),
      "lon": Decimal128("47.234267")
    },
    {
      "_id": ObjectId("63fb78d6551c94db78575bd2"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Antananarivo",
      "partner_id": 357,
      "partner_name": "Vahatra",
      "loan_id": "10000000000000000000000000000002",
      "loan_theme_id": "1000000000002XXX",
      "loan_theme_type": "Extreme Poverty",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region_number": "956-0",
      "amount": 300000,
      "location_name": "Antananarivo, Madagascar",
      "lat": Decimal128("19.884268"),
      "lon": Decimal128("47.234267")
    },
    {
      "_id": ObjectId("63fb78d6551c94db78575bd3"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Antananarivo",
      "partner_id": 358,
      "partner_name": "Vahatra",
      "loan_id": "10000000000000000000000000000003",
      "loan_theme_id": "1000000000002XXX",
      "loan_theme_type": "Extreme Poverty",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region_number": "956-0",
      "amount": 300000,
      "location_name": "Antananarivo, Madagascar",
      "lat": Decimal128("19.884268"),
      "lon": Decimal128("47.234267")
    },
    {
      "_id": ObjectId("63fb78d6551c94db78575bc4"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Antananarivo",
      "partner_id": 359,
      "partner_name": "Vahatra",
      "loan_id": "10000000000000000000000000000004",
      "loan_theme_id": "1000000000002XXX",
      "loan_theme_type": "Extreme Poverty",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region_number": "953-0",
      "amount": 300000,
      "location_name": "Antananarivo, Madagascar",
      "lat": Decimal128("19.862211"),
      "lon": Decimal128("47.059795")
    },
    {
      "_id": ObjectId("63fb78d6551c94db78575bd5"),
      "sector": "General Financial Inclusion",
      "country": "Madagascar",
      "region": "Antananarivo",
      "partner_id": 360,
      "partner_name": "Vahatra",
      "loan_id": "10000000000000000000000000000005",
      "loan_theme_id": "1000000000002XXX",
      "loan_theme_type": "Extreme Poverty",
      "forsave": "No",
      "loan_amount": "1000000",
      "loan_region_number": "953-0",
      "amount": 300000,
      "location_name": "Antananarivo, Madagascar",
      "lat": Decimal128("19.862211"),
      "lon": Decimal128("47.059795")
    }
  ]
}

```

Fig. 13. Members in Specific Region

Fig. 14 is showing the applicants from countries that their name contains “am” and their attribute of ‘forkiva’ is set to “yes”.

```
Type: "for nose"
Atlas atlas-Sale1 shard0 [primary] kiva_db.all_kiva.find({country:/^vn.+/, forklive:"Yes"})
{
  "_id": ObjectId("63bf78d9f51c09ad078573fa"),
  "sector": "General Financial Inclusion",
  "country": "Cambodia",
  "region": "Battambang",
  "partner_id": "10000000000000000000000000000001",
  "field_partner_name": "Hutha Kaksekar Limited (HKL)",
  "lat": Decimal128(13.96523),
  "lon": Decimal128(-104.897596),
  "is_vn": "Yes",
  "is_km": "No",
  "lon_region_number": "1.8",
  "sector_vn": "General Financial Inclusion",
  "location_name": "Bantay Meanchey, Cambodia",
  "lat1": Decimal128(13.96523),
  "lon1": Decimal128(-104.897596)
},
{
  "_id": ObjectId("63bf78d9f51c09ad078573fa"),
  "sector": "General Financial Inclusion",
  "country": "Cambodia",
  "region": "Battambang",
  "partner_id": "10000000000000000000000000000002",
  "field_partner_name": "Hutha Kaksekar Limited (HKL)",
  "lat": Decimal128(13.96523),
  "lon": Decimal128(-104.897596),
  "is_vn": "Yes",
  "is_km": "No",
  "lon_region_number": "21.0",
  "sector_vn": "General Financial Inclusion",
  "location_name": "Bantay Meanchey, Cambodia",
  "lat1": Decimal128(13.96523),
  "lon1": Decimal128(-104.897596)
},
{
  "_id": ObjectId("63bf78d9f51c09ad078573fa"),
  "sector": "General Financial Inclusion",
  "country": "Cambodia",
  "region": "Battambang",
  "partner_id": "10000000000000000000000000000003",
  "field_partner_name": "Hutha Kaksekar Limited (HKL)",
  "lat": Decimal128(13.96523),
  "lon": Decimal128(-104.897596),
  "is_vn": "Yes",
  "is_km": "No",
  "lon_region_number": "12.0",
  "sector_vn": "General Financial Inclusion",
  "location_name": "Battambang, Cambodia",
  "lat1": Decimal128(13.96523),
  "lon1": Decimal128(-104.292285)
},
{
  "_id": ObjectId("63bf78d9f51c09ad078573fa"),
  "sector": "General Financial Inclusion",
  "country": "Cambodia",
  "region": "Battambang",
  "partner_id": "10000000000000000000000000000004",
  "field_partner_name": "Hutha Kaksekar Limited (HKL)",
  "lat": Decimal128(13.96523),
  "lon": Decimal128(-104.292285),
  "is_vn": "Yes",
  "is_km": "No",
  "lon_region_number": "2.0",
  "sector_vn": "General Financial Inclusion",
  "location_name": "Battambang, Cambodia",
  "lat1": Decimal128(13.96523),
  "lon1": Decimal128(-104.292285)
},
{
  "_id": ObjectId("63bf78d9f51c09ad078573fa"),
  "sector": "General Financial Inclusion",
  "country": "Cambodia",
  "region": "Kampot Cham",
  "partner_id": "10000000000000000000000000000005",
  "field_partner_name": "Hutha Kaksekar Limited (HKL)",
  "lat": Decimal128(13.96523),
  "lon": Decimal128(-104.292285),
  "is_vn": "Yes",
  "is_km": "No",
  "lon_region_number": "2.0",
  "sector_vn": "General Financial Inclusion",
  "location_name": "Kampot Cham, Cambodia",
  "lat1": Decimal128(13.96523),
  "lon1": Decimal128(-104.292285)
}
```

Fig. 14. Countries Like '/*am.*/'

Fig. 15 below is displaying the query and the returned result for the applicants that are active in the food industry and they have specialty and their loan_theme_type is not “General” .

```

Atlas atlas-Salinity shard=0 [primary] kive.$all_kiva.find({activity_id:{$ne:"Food"}, loan_theme_type:{$ne:"General"} })
{
  {
    "_id": ObjectId("61878d15c9a9eb7856b9e8"),
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "funded_amount": 200,
    "loan_amount": 1000,
    "activity": "Food Production/Sales",
    "sector": "Food",
    "use": "To buy a sugarcane Juicing machine.",
    "country_code": "MM",
    "partner": "Kiven",
    "region": "Phone Peing",
    "partner_id": 51,
    "posted_time": ISODate("2013-05-07T18:10:00.000Z"),
    "disbursed_time": ISODate("2013-12-23T10:00:00.000Z"),
    "funded_time": ISODate("2014-05-02T16:31:03.000Z"),
    "term_in_months": 14,
    "lender_count": 26,
    "borrower": "female",
    "repayment_interval": "monthly",
    "date": ISODate("2014-05-02T22:00:00.000Z")
  },
  {
    "_id": ObjectId("61878d15c9a9eb7856b9e9"),
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "funded_amount": 1000,
    "loan_amount": 1000,
    "activity": "Food",
    "sector": "Food",
    "use": "Buy ingredients for making rice soup and Chinese noodle.",
    "country_code": "MM",
    "partner": "Kiven",
    "region": "Battambang province, Moun Russey district",
    "partner_id": 280,
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "disbursed_time": ISODate("2013-12-06T18:00:00.000Z"),
    "funded_time": ISODate("2014-03-04T21:52:37.000Z"),
    "term_in_months": 12,
    "lender_count": 26,
    "borrower": "female, female, female",
    "repayment_interval": "monthly",
    "date": ISODate("2014-03-03T00:00:00.000Z")
  },
  {
    "_id": ObjectId("61878d15c9a9eb7856b9e8"),
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "funded_amount": 400,
    "loan_amount": 400,
    "activity": "Food",
    "sector": "Food",
    "use": "Buy vegetables for resale",
    "country_code": "MM",
    "partner": "Kiven",
    "region": "Battambang province, Moun Russey district",
    "partner_id": 280,
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "disbursed_time": ISODate("2013-12-06T18:00:00.000Z"),
    "funded_time": ISODate("2014-03-04T17:32:46.000Z"),
    "term_in_months": 12,
    "lender_count": 15,
    "borrower": "female, female",
    "repayment_interval": "monthly",
    "date": ISODate("2014-03-03T00:00:00.000Z")
  },
  {
    "_id": ObjectId("61878d15c9a9eb7856b9e9"),
    "posted_time": ISODate("2013-05-07T18:00:00.000Z"),
    "funded_amount": 1000,
    "loan_amount": 1000,
    "activity": "Food",
    "sector": "Food",
    "use": "Buy vegetables and fish to resell",
    "country_code": "MM",
    "partner": "Kiven"
  }
}

```

Fig. 15 Show the loans in food industry-specific type (exclude general types)

Last Query from Report 1 is the complex query that joins and groups by having clauses in Relational data format to return a count of activity along with other attributes for where the term in months is bigger than 10 and the count of activity is also bigger than 10 and it's ordered by count.

Fig. 16 depicts the query in MongoDB shell for NoSQLI kiva database.

```
        { _id: "Fish Selling", _ctn: 72 },
        { _id: "Fruit Selling", _ctn: 10 },
        { _id: "Livestock", _ctn: 76 },
        { _id: "Construction", _ctn: 77 },
        { _id: "Hardware", _ctn: 10 },
        { _id: "Poultry", _ctn: 91 },
        { _id: "Food Production/Sales", _ctn: 92 },
        { _id: "Food", _ctn: 93 },
        { _id: "Furniture", _ctn: 99 },
        { _id: "General Store", _ctn: 114 },
        { _id: "Fruits & Vegetables", _ctn: 134 },
        { _id: "Grocery Store", _ctn: 140 },
        { _id: "Motorcycle Transport", _ctn: 166 },
        { _id: "Fishing", _ctn: 157 },
        { _id: "Farm Supplies", _ctn: 269 }
    ]
}

Type "it" for more
Atlas alias=Sale->shard-0 [primary] kiva> it
{
    "_id": "Cattle", "cnt": 314 },
    { "_id": "Grocery Store", "cnt": 348 },
    { "_id": "Furniture", "cnt": 380 },
    { "_id": "Pigs", "cnt": 567 },
    { "_id": "Vehicle", "cnt": 644 },
    { "_id": "Clothing", "cnt": 1030 },
    { "_id": "Home Appliances", "cnt": 1812 },
    { "_id": "Higher education costs", "cnt": 1264 },
    { "_id": "Personal Housing Expenses", "cnt": 1970 },
    { "_id": "Farming", "cnt": 7682 }
}
Atlas alias=Sale->shard-0 [primary] kiva> it
```

Fig. 16 Display number of times each activity gets funded with terms of bigger
10 months and if they are funded more than 10 times

Figures 17 to 19 below demonstrates pdate Delete and Insert commands in all_kiva collection in MongoDB using Mongo shellupdate commands performed in NoSQL.

```
Atlas atlas-5allie7-shard-0 [primary] kiva> db.all_kiva.updateMany({loan_theme_type:"General"},{$set:{loan_theme_type:"Unique"}})
{
  acknowledged: true,
  insertedId: null,
  matchedCount: 384876,
  modifiedCount: 384876,
  upsertedCount: 0
}
```

Fig. 17 Update documents set a value on condition

```
Atlas atlas-5allie7-shard-0 [primary] kiva> db.all_kiva.deleteMany({loan_theme_type:"Unique"})
{
  acknowledged: true,
  deletedCount: 384876
}
Atlas atlas-5allie7-shard-0 [primary] kiva>
```

Fig. 18 Delete documents based on conditions



Fig. 19 Insert 133 documents to all_kiva collection

V. DATA VISUALIZATION

The analysis of the Kiva Crowdfunding Dataset gives an insight into the various factors that affect and influence the sustainability of crowdfunded projects. It also reveals the distribution of these projects in the world, gives information about the people involved in the creation of the projects and how the projects are distributed among the different sectors.

- The following pie chart shows the distribution of the number of loans taken by various crowdfunding project owners and their distribution in the different regions.

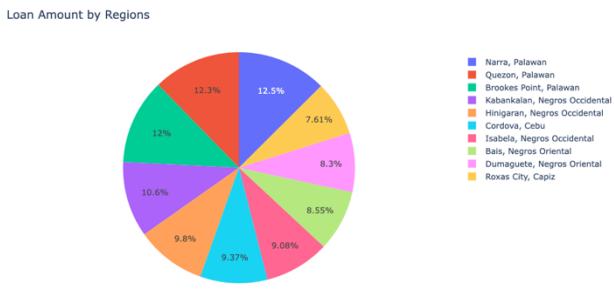


Fig. 20 Loan distribution pie chart

- Analysis of the data reveals the relation between the different projects and the amount that was funded to them with the gender of the borrowers and we see that a greater number of female borrowers as compared to

male borrowers and that as the funding amount increases, the number of projects funded for that amount decreases.

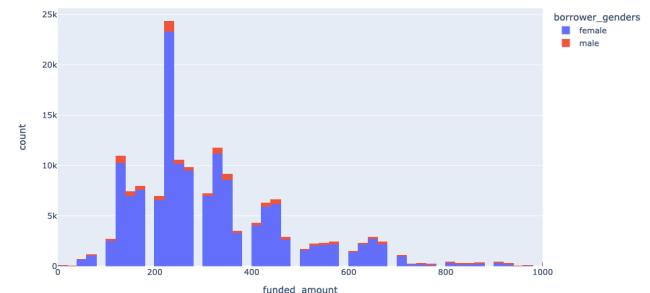


Fig. 21 Funded amounthistogram color-coded by gender

- The below boxplot shows the various sectors of crowdfunded projects and the distribution of the loan amount for each of those sectors as well as the quartiles for the loan amount in those sectors. We see that the wholesale sector has the greatest amount of loans whereas the personal use sector has the lowest.

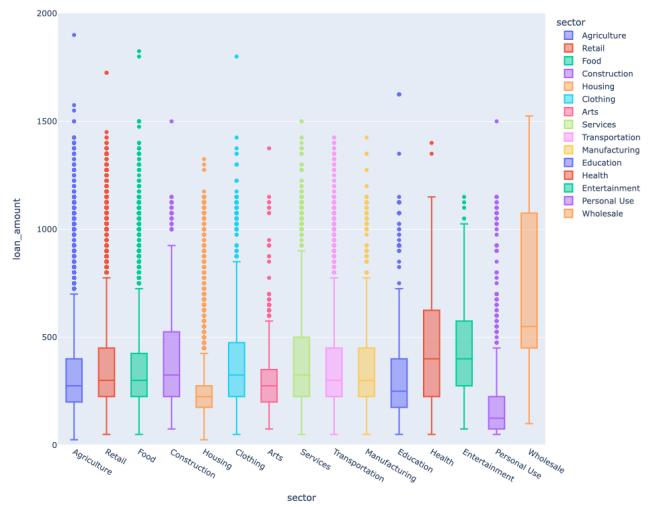


Fig. 22 Project sectors box plot

- The scatter plot shows the cumulative funding of projects in different sectors with respect to the amount of funding they receive. Projects belonging to the retail sector receive the highest funding closely followed by food and agriculture whereas manufacturing projects are the lowest funded.

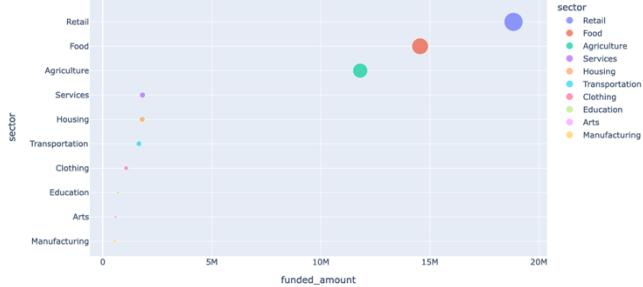


Fig. 23 Funding of projects in different sectors with respect to the amount of funding scatter plot

- The bar graph represents the top ten uses for loans taken by borrowers and their counts. There are about 70,000 different uses for the loans with different numbers of occurrences and we see that the use case ‘to build sanitation facilities’ is the most seen use.

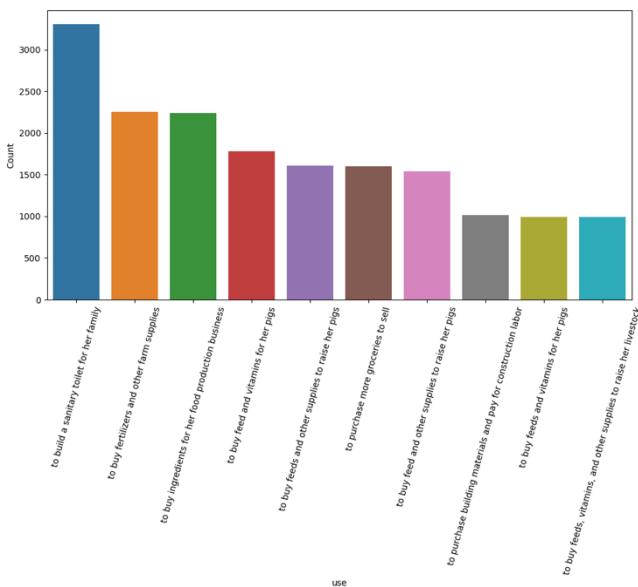


Fig. 24 Top 10 loan use bar chart

- The following heat map shows the degree of relation between the funded amount, loan amount, the term of the loan(in months), and the lender count for each of the projects in the Kiva database.

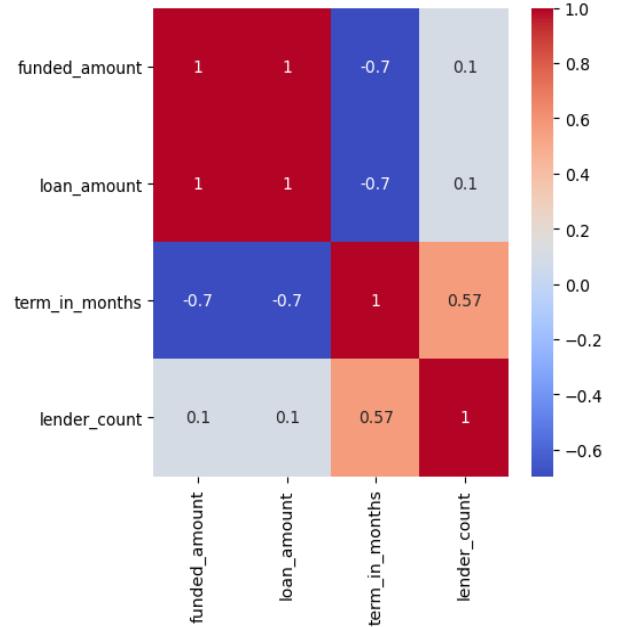


Fig. 25 funded amount, loan amount, the term of the loan(in months), and the lender count heatmap

- The following map shows the distribution of the crowdfunding projects based on their location. Hovering over the location indicates their latitude, longitude, country name, and region name. The heat map helps us visualize areas with a concentration of these projects. It can be interpreted that South America and West Africa have the highest concentration of crowdfunding projects than anywhere else in the world.



Fig. 26. Distribution of projects by location

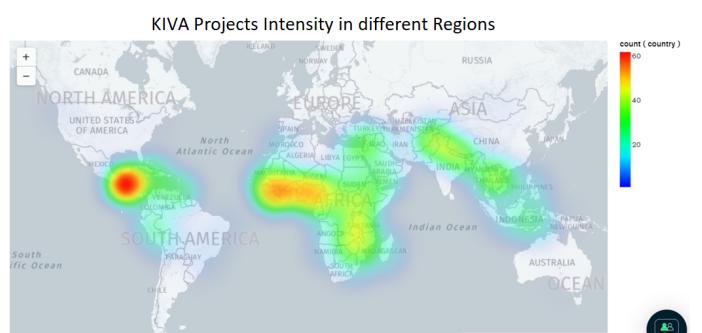


Fig. 27 Kiva project intensity in regions

- The pie chart shows how much funding was received by the projects each month over a period of three years. The greater the area of the sector the more the funding was dispensed during that particular month.



Fig. 28 Monthly funded projects

- Each partner involved in the crowdfunding process contributed to the success of the project and the below scatter plot shows funds contributed by each of the partners to all the projects they were involved with. We can see that contributions by partners range from a few hundred thousand to a few million.

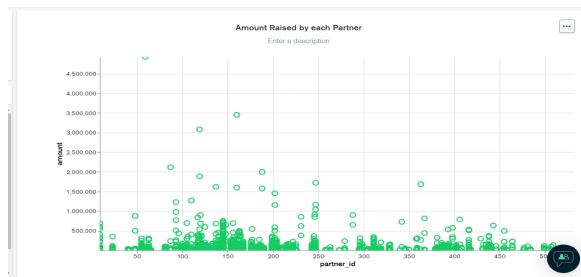


Fig. 29 Partners vs Projects

- The bar graph indicates the loan themes used in each of the sectors. The most different uses for the loan amount are in the General financial Inclusion Sector and the least in the DSE Direct sector. Another graph reveals the distribution of these sectors with their uses in different countries.

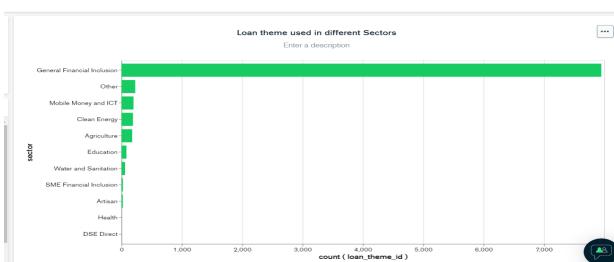


Fig. 30 Kiva loan themes by different sectors

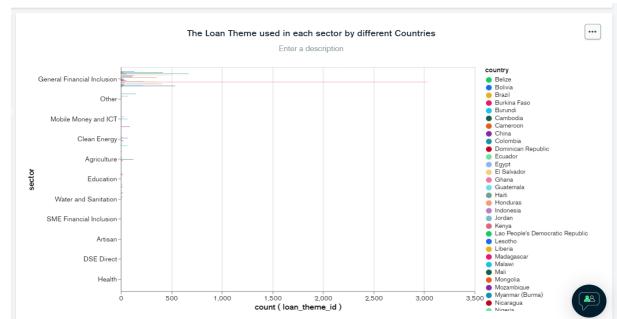


Fig. 31 Kiva loan themes by different countries

VI. CONNECTIVITY TO ATLAS

MongoDB Atlas is a Database-as-a-Service(DBaaS) that gives users the power to create, scale, manage NoSQL databases on the cloud with the flexibility of choosing their own provider(Azure, AWS, and GCP). Deployment of the database on the cloud ensures access, security, timely software updates, and physical hardware independence to database users. For this project, a database ‘Kiva’ was created on MongoDB and deployed on Atlas.

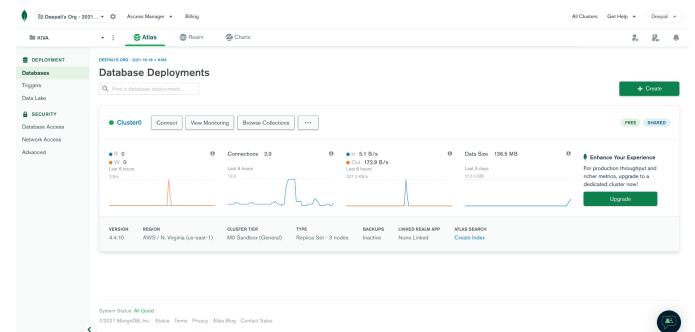


Fig. 32 MongoDB Atlas Connectivity-1

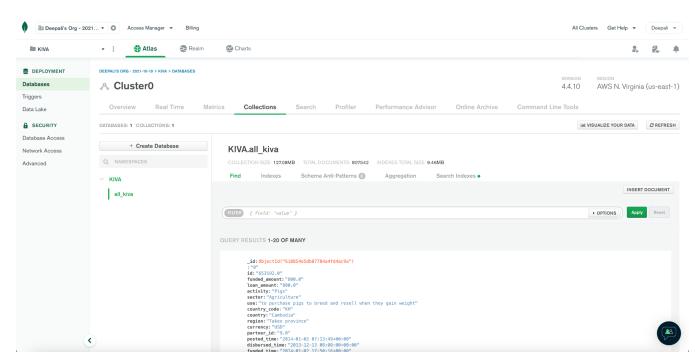


Fig. 33 MongoDB Atlas Connectivity-2

VII. NOSQL PERFORMANCE MEASUREMENT

NoSQL is a non-relational database that can handle semi-structured and unstructured data and enables the user to query the data at speed. The prioritisation of speed, adaptability

and availability over consistency makes NoSQL well suited for big data manipulation over other querying languages. The query performance here is measured using the explain() function available in MongoDB. This method returns a document containing the query plan as well as the execution statistics.

Fig. 34 below shows loan_amounts between \$500-\$1500 performance statistics. Also, the count of each loan_amount and then ordering the results in descending format is shown in Fig. 35 the '_id' here representing the loan_amount value.

```
Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.aggregate([ { $group: { _id: "$loan_amount", count:{$sum:1} } }, { $sort:{'count':-1} } ])
{
  "ok": 1,
  "errmsg": "for now",
  "atlas": {
    "atlas": "atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find({loan_amount:{$gt:500, $lt:1500}}).explain('executionStats')",
    "queryPlanner": {
      "plannerVersion": 1,
      "namespace": "kiva.all_kiva",
      "indexFilterSet": false,
      "parsedQuery": {
        "$and": [
          {"loan_amount": {"$gt": 500}},
          {"loan_amount": {"$lt": 1500}}
        ]
      },
      "winningPlan": {
        "stage": "COLLSCAN",
        "filter": {
          "$and": [
            {"loan_amount": {"$gt": 1500}},
            {"loan_amount": {"$gt": 500}}
          ]
        },
        "direction": "forward"
      },
      "rejectedPlans": []
    },
    "executionStats": {
      "executionSuccess": true,
      "nReturned": 10023,
      "executionTimeMillisEstimate": 317,
      "totalKeysExamined": 0,
      "totalDocsExamined": 887542,
      "executionStages": {
        "stage": "COLLSCAN",
        "filter": {
          "$and": [
            {"loan_amount": {"$gt": 1500}},
            {"loan_amount": {"$gt": 500}}
          ]
        },
        "direction": "forward"
      },
      "rejectedPlans": []
    }
  },
  "serverInfo": {
    "host": "data225-shard-00-01.5qd6p4.mongodb.net",
    "port": 27017,
    "version": "4.4.10",
    "gitVersion": "58971da1ef93435a9f62bf4788a81713defe88c"
  },
  "ok": 1,
  " errmsg": "Timestamp({ t: 1636314302, i: 1 })",
  "signature": {
    "hash": "Binary(Buffer.from(\"6a26ddfa8b7be994acbf8daf15893c867769c03\", \"hex\"), 0)",
    "keyId": Long("699458689132449738")
  },
  "operationTime": Timestamp({ t: 1636314302, i: 1 })
}
Atlas atlas-Salle7-shard-0 [primary] kiva> 
```

Fig. 34 Loans between \$50-\$1500 query performance statistics

```
Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.aggregate([ { $group: { _id: "$loan_amount", count:{$sum:1} } }, { $sort:{'count':-1} } ])
{
  "ok": 1,
  "errmsg": "for now",
  "atlas": {
    "atlas": "atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find({loan_amount:{$gt:500, $lt:1500}}).explain('executionStats')",
    "queryPlanner": {
      "plannerVersion": 1,
      "namespace": "kiva.all_kiva",
      "indexFilterSet": false,
      "parsedQuery": {
        "queryHash": "BFBFA7BC",
        "plannedHash": "BFBFA7BC",
        "winningPlan": {
          "stage": "PROJECTION_SIMPLE",
          "transformBy": { loan_amount: { _id: 0 } },
          "inputStage": { stage: "COLLSCAN", direction: "forward" },
          "rejectedPlans": []
        },
        "executionStats": {
          "executionSuccess": true,
          "nReturned": 887542,
          "executionTimeMillis": 402,
          "totalKeysExamined": 0,
          "totalDocsExamined": 887542,
          "executionStages": {
            "stage": "PROJECTION_SIMPLE",
            "transformBy": { loan_amount: { _id: 0 } },
            "inputStage": {
              "stage": "COLLSCAN",
              "filter": {
                "$and": [
                  {"loan_amount": {"$gt": 1500}},
                  {"loan_amount": {"$gt": 500}}
                ]
              },
              "direction": "forward"
            },
            "rejectedPlans": []
          }
        }
      },
      "executionStats": {
        "executionSuccess": true,
        "nReturned": 887542,
        "executionTimeMillis": 402,
        "totalKeysExamined": 0,
        "totalDocsExamined": 887542,
        "executionStages": {
          "stage": "PROJECTION_SIMPLE",
          "transformBy": { loan_amount: { _id: 0 } },
          "inputStage": {
            "stage": "COLLSCAN",
            "filter": {
              "$and": [
                {"loan_amount": {"$gt": 1500}},
                {"loan_amount": {"$gt": 500}}
              ]
            },
            "direction": "forward"
          },
          "rejectedPlans": []
        }
      }
    },
    "executionStats": {
      "executionSuccess": true,
      "nReturned": 887542,
      "executionTimeMillisEstimate": 133,
      "totalKeysExamined": 0,
      "totalDocsExamined": 887542,
      "executionStages": {
        "stage": "PROJECTION_SIMPLE",
        "transformBy": { loan_amount: { _id: 0 } },
        "inputStage": {
          "stage": "COLLSCAN",
          "filter": {
            "$and": [
              {"loan_amount": {"$gt": 1500}},
              {"loan_amount": {"$gt": 500}}
            ]
          },
          "direction": "forward"
        },
        "rejectedPlans": []
      }
    }
  },
  "serverInfo": {
    "host": "data225-shard-00-01.5qd6p4.mongodb.net",
    "port": 27017,
    "version": "4.4.10",
    "gitVersion": "58971da1ef93435a9f62bf4788a81713defe88c"
  },
  "ok": 1,
  " errmsg": "Timestamp({ t: 1636311298, i: 1 })",
  "signature": {
    "hash": "Binary(Buffer.from(\"6806dd4d3d9686d4fe919bf37a1f287cf8d81383a\", \"hex\"), 0)",
    "keyId": Long("699458689132449738")
  },
  "operationTime": Timestamp({ t: 1636311298, i: 1 })
}
Atlas atlas-Salle7-shard-0 [primary] kiva> 
```

Fig. 35 Loan counts performance statistics

Fig. 36 represents the performance statistics for a query of countries whose name has “am” and they are applying for kiva.

```
Type "it" for more
Atlas atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find({country: /.*am.*/, forkiva:"Yes" }).explain("executionStats")
{
  "ok": 1,
  "errmsg": "Timestamp({ t: 1636314302, i: 1 })",
  "atlas": {
    "atlas": "atlas-Salle7-shard-0 [primary] kiva> db.all_kiva.find({country: /.*am.*/, forkiva:"Yes" }).explain("executionStats")",
    "queryPlanner": {
      "plannerVersion": 1,
      "namespace": "kiva.all_kiva",
      "indexFilterSet": false,
      "parsedQuery": {
        "$and": [
          {"forkiva": {"$eq": "Yes"}},
          {"country": {"$regex": ".+am.+"}}
        ]
      },
      "winningPlan": {
        "stage": "COLLSCAN",
        "filter": {
          "$and": [
            {"forkiva": {"$eq": "Yes"}},
            {"country": {"$regex": ".+am.+"}}
          ]
        },
        "direction": "forward"
      },
      "rejectedPlans": []
    },
    "executionStats": {
      "executionSuccess": true,
      "nReturned": 264,
      "executionTimeMillisEstimate": 69,
      "totalKeysExamined": 362,
      "totalDocsExamined": 0,
      "executionTimeMillis": 279,
      "totalDocsExamined": 887542,
      "executionStages": {
        "stage": "COLLSCAN",
        "filter": {
          "$and": [
            {"forkiva": {"$eq": "Yes"}},
            {"country": {"$regex": ".+am.+"}}
          ]
        },
        "direction": "forward"
      },
      "rejectedPlans": []
    }
  },
  "serverInfo": {
    "host": "data225-shard-00-01.5qd6p4.mongodb.net",
    "port": 27017,
    "version": "4.4.10",
    "gitVersion": "58971da1ef93435a9f62bf4788a81713defe88c"
  },
  "ok": 1,
  " errmsg": "Timestamp({ t: 1636349484, i: 7 })",
  "signature": {
    "hash": "Binary(Buffer.from(\"03e209837ee528bfad68c88579d52f3c0ba9bdd\", \"hex\"), 0)",
    "keyId": Long("699458689132449738")
  },
  "operationTime": Timestamp({ t: 1636349484, i: 7 })
}
Atlas atlas-Salle7-shard-0 [primary] kiva> 
```

Fig. 36 Country like “.*am.*” statistics

Fig. 37 shows the performance for specific countries and their ordering desc by country performance.

```
Atlas: atlas-GalleT-shard-0 [primary] kiva> db.all_kiva.find({country: { $in: ["Yeman", "Egypt", "Madagascar"] }}).sort({country:-1}).explain("queryPlanner")
{
  queryPlanner: {
    planCacheVersion: 1,
    namespace: "kiva.all_kiva",
    indexFilterSet: false,
    pipeline: [
      { stage: "COLLSCAN", filter: { country: { $in: ['Egypt', 'Madagascar', 'Yeman'] } } },
      { stage: "SORT", direction: -1 },
      { stage: "LIMIT", count: 1 }
    ],
    winningPlan: {
      stage: "COLLSCAN",
      filter: { country: { '$in': [ 'Egypt', 'Madagascar', 'Yeman' ] } },
      direction: "forward"
    },
    rejectedPlans: []
  },
  executionStats: {
    executionSuccess: true,
    nReturned: 52,
    executionTimeMillis: 348,
    totalKeysExamined: 0,
    totalDocsExamined: 887542,
    executionStages: {
      stage: "COLLSCAN",
      nReturned: 57,
      executionTimeMillisEstimate: 54,
      works: 887544,
      advanced: 57,
      needTime: 887544,
      needKeyFile: 0,
      saveState: 887,
      restoreState: 887,
      isEOF: 1,
      isEOForEmpty: 1,
      sortPattern: { country: -1 },
      memLimit: 33854432,
      type: "SIMPLE",
      totalDataSizeSorted: 14821,
      usedDisk: false,
      inputStage: {
        stage: "COLLSCAN",
        filter: { country: { '$in': [ 'Egypt', 'Madagascar', 'Yeman' ] } },
        direction: "forward",
        docsExamined: 887542
      }
    },
    serverInfo: {
      host: "data226-shard-00-01.5qpq4.mongodb.net",
      port: 27017,
      version: "4.4.18",
      gitVersion: "58977d1def93435a9ff62bf4708a81713def6e88c"
    },
    ok: 1,
    lastErrorTime: {
      clusterTime: Timestamp({ t: 1636354413, i: 7 }),
      signature: {
        hash: Binary(Buffer.from("912fe4672b01ef7ddca85628c52699c689f307cc6"), "hex"),
        keyId: Long("6994a5868913244897938")
      }
    },
    operationTime: Timestamp({ t: 1636354413, i: 7 })
  }
}

```

Fig. 37 Country specific query performance

Figure 38 is showing the applications in the food industry excluding General purpose applicants' query performance.

```
type -tux more
[Atlas atlas-5alle7-shard-0 [primary] kiva> db.all_kiva.find({activity: {$regex: '+Food.'}, loan_theme_type:{$ne:'General'}}).explain("executionStats")
{
  queryPlanner: {
    planCacheVersion: 1,
    namespace: 'kiva.all_kiva',
    indexFilterSet: false,
    parsedQuery: {
      '$and': [
        { activity: { '$regex': '+Food.' } },
        { loan_theme_type: { '$not': { '$eq': 'General' } } }
      ]
    },
    winningPlan: {
      stage: 'COLLSCAN',
      filter: {
        '$and': [
          { activity: { '$regex': '+Food.' } },
          { loan_theme_type: { '$not': { '$eq': 'General' } } }
        ]
      },
      direction: 'forward'
    },
    rejectedPlans: []
  },
  executionStats: {
    executionSuccess: true,
    nReturned: 232,
    executionTimeMillis: 385,
    totalKeysExamined: 0,
    totalDocsExamined: 807542,
    executionStages: {
      stage: 'COLLSCAN',
      filter: {
        '$and': [
          { activity: { '$regex': '+Food.' } },
          { loan_theme_type: { '$not': { '$eq': 'General' } } }
        ]
      },
      nReturned: 232,
      executionTimeMillisEstimate: 37,
      works: 807542,
      advanced: 232,
      needMoreResults: 313,
      needToRead: 8,
      saveState: 807,
      restoreState: 807,
      isEOF: 1,
      direction: 'forward',
      docsExamined: 807542
    }
  },
  serverInfo: {
    host: 'data225-shard-00-01.5q6p4.mongodb.net',
    port: 27017,
    version: '4.4.18',
    gmtOffset: 60,
    pvtVersion: '69971daef93435a9f62bf708a81713defe88c'
  },
  ok: 1
}
{
  clusterTime: {
    clusterTime: Timestamp({ t: 1436349925, i: 5 }),
    signature: {
      keyId: 16000000000000000000,
      keySha256: '00000000000000000000000000000000'
    },
    keyId: Long('69971daef93435a9f62bf708a81713defe88c')
  }
},
operationTime: Timestamp({ t: 1436349925, i: 5 })
}
[Atlas atlas-5alle7-shard-0 [primary] kiva> ]
```

Fig. 38 Food industry applicants query performance

Fig. 39 shows the performance for the query of the loan activities group by where they have more than 10 months terms.

and also the count of the group is more than 10 at the end the result is ordered by count.

Fig. 39 Display number of times each activity gets funded with terms of bigger 10 months and if they are funded more than 10 times query performance

Up to this point, all of the queries were equivalent to select queries in Relational databases and the statistics display that queries run faster on MySQL in comparison with NoSQL takes more time.

Now we perform Update and Delete commands on both MySQL and MongoDB which shows this type of data manipulation takes more time on MySQL than MongoDB.

Fig40 and Fig41 illustrate the results of performing the same commands on MongoDB and MySQL workbench.

Action	Time	Action	Response	Duration / Fetch Time
287. update loan_theme_ids ready set loan_theme_type='Unique' where loan_theme_type='General';	100%	C - 1:287		

Fig 40 Update on 380693records MySQL

```
Atlas-atlas-Salle1-shard-0 [primary] kiva> db.all_kiva.updateMany({loan_theme_type:"General"}, {$set:{loan_theme_type:"Unique" }} )  
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 384876,  
  modifiedCount: 384876,  
  upsertedCount: 0  
}
```

Fig. 41 Update 384876 records MongoDB. Execution Time for Query

Query\Database	SQL	NoSQL
Loan between \$50-\$500 Query	2.6 ms	317 ms
Loan Counts	32 ms	682 ms
Country Like “.*am.*” Query	17 ms	362 ms
Country-Specific Query	20 ms	340 ms
Food Industry Applicants Query	72 ms	305 ms
Funding of Activity With Term >10 Months	81 ms	347 ms
Insert Query	102 ms	20 ms
Update Query	10.685s	20 ms
Delete Query	3.327 s	30 ms

Table 1 Execution Time: SQL vs. NoSQL

Both MySQL and MongoDB have their advantages and disadvantages. Being relational-oriented, MySQL is scalable and has consistency, it supports ACID transactions. Whereas, MongoDB is flexible and faster. MySQL DBM systems need to maintain a consistent state and impact the performance a bit. Whereas, MongoDB stores data as documents or key-value pairs and thus performs data manipulation operations better.

The performance of the database systems depends on the context of the data. NoSQL performs better with unstructured data such as document-oriented, graph-oriented, etc. whereas SQL works best with structured data. In this project, the data was structured and the structure was considered unlikely to be changed, hence SQL takes a shorter time to execute than NoSQL. Also, NoSQL works faster with big data as well as rapid growth data, however, the data used here is not huge. Selecting between NoSQL databases versus Relational databases depends on the priority of the project. For instance, if there are a great number of connection calls at the same time to a database on one server, it's better to use NoSQL structures to be able to handle the connections since Relational databases are capable of handling a few calls at once. Therefore it can be said that NoSQL is not faster than MySQL and MySQL is not faster than NoSQL.

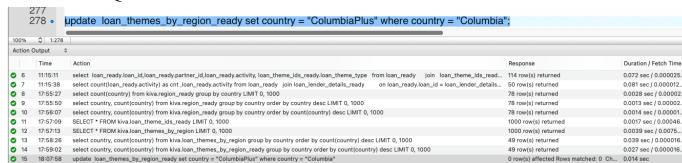


Fig. 38 Update Command MySQL

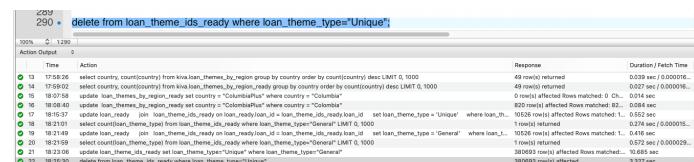


Fig. 39 Delete Command MySQL

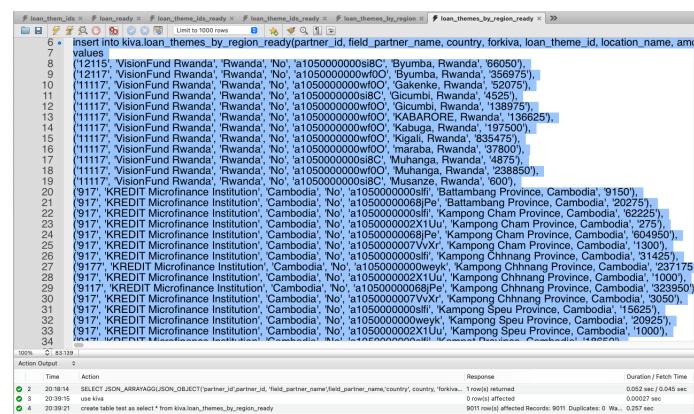


Fig. 40 Insert Command MySQL

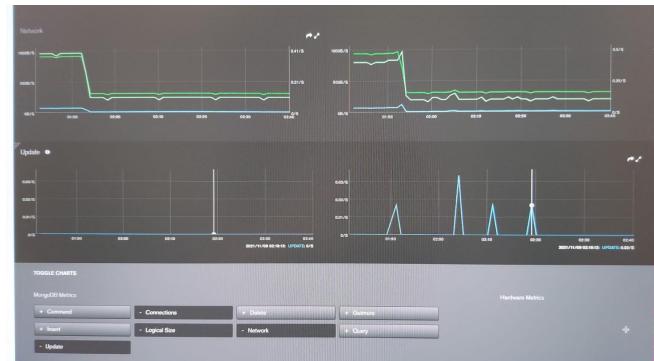


Fig. 41 Update Command NoSQL

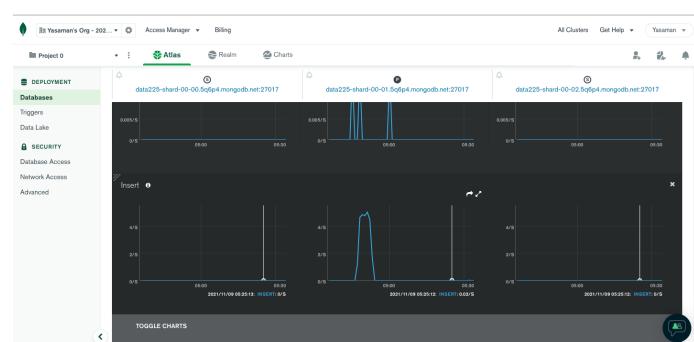


Fig. 42 Insert Command NoSQL

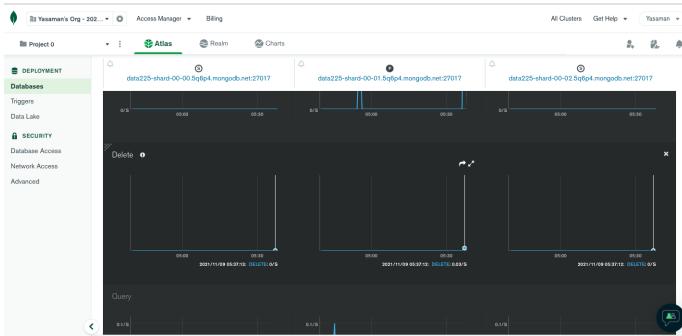


Fig. 43 Delete Command NoSQL

Acknowledgment

We would like to express our sincere gratitude to Professor Simon Shim for his guidance that aided in the completion of this project and helped us understand database concepts using practical examples. We would also like to thank Shiva Abhishek Varma Penmetsa and Rushikesh Jagtap for their continued help and support throughout the development of the project.

REFERENCES

- [1] S. Yu, Crowdfunding and regional entrepreneurial investment: an application of the CrowdBerkeley database, *Research Policy*, Volume 46, Issue 10, 2017, Pages 1723-1737, ISSN 0048-7333
- [2] Burtch, G., Ghose, A., Wattal, S., 2014. Cultural Differences and geography as determinants of online prosocial lending. *MIS Q.* 38 (3), 773–794.
- [3] Mollick, E., Nanda, R., 2015. Wisdom or Madness? Comparing Crowds with Expert Evaluation in Funding the Arts. *Manage. Sci.* 62 (6), 1533–1553. Mollick, E.R., 2014. The Dynamics of Crowdfunding: An Exploratory Study. *J. Bus. Venturing* 29 (January (1)), 1–16.
- [4] Smith, Tim. “Crowdfunding.” *Investopedia*, Investopedia, 13Sept.2021, www.investopedia.com/terms/c/crowdfunding