

KETAN MANUBHAI PANCHAL - 2509903

by Ketan Manubhai Panchal

Submission date: 16-Jan-2026 12:13PM (UTC+0000)

Submission ID: 273516347

File name: 66642_Ketan_Manubhai_Panchal_KETAN_MANUBHAI_PANCHAL_-
_2509903_717126_78833688.docx (1.89M)

Word count: 4348

Character count: 27100

Data storage investigation

Abstract: This report will assess the appropriate database technologies to use in PixelBay, an online digital art market, which deals with customer accounts, records of artworks and transactions. It contrasts the various models of NoSQL database such as document, graph, key-value and wide-column databases, with the traditional relational database systems to establish their applicability to the operational needs of PixelBay. The analysis brings out the strengths and weaknesses of each technique with respect to scalability, flexibility of data and integrity. According to the need of PixelBay to manage both transactional and content-based data in an effective manner, the report suggests a hybrid database system that incorporates both a relational database and a NoSQL database in order to provide reliability, performance, and scalability in the future.

Keywords:

Digital marketplace, NoSQL databases, Relational databases, Data management, Scalability, PixelBay

Table of Contents

I. PIXELBAY OVERVIEW AND DATABASE NEED.....	4
II. ADVANTAGES OF NOSQL FOR PIXELBAY.....	4
III. EVALUATION OF PIXELBAY'S DATA REQUIREMENTS	4
IV. TASKS	4
V. RECOMMENDED DATABASE SOLUTION	15
VI. JUSTIFICATION OF THE RECOMMENDED SOLUTION	15
VII. DEPLOYMENT STRATEGY	16
VIII. DATA SECURITY AND BACKUP	16
IX. CONCLUSION AND FINAL RECOMMENDATION.....	16
X. ALIGNMENT WITH PIXELBAY'S GOALS.....	16
XI. REFERENCES	17

I. PIXELBAY OVERVIEW AND DATABASE NEED

PixelBay is a digital art platform in Swansea which allows customers to browse, buy, and stream digital artworks across a broad selection of digital artworks of independent artists and studios, including its primary partner TaithArts. Since PixelBay will be expanding throughout the UK and abroad, it will be required to maintain high numbers of customer accounts, artwork records and financial transactions reliably and scalably [1]. It must have a solid database system to guarantee the proper storage of the user data, proper retrieval of the information about the artworks and safe processing of the purchases. A proper database system will facilitate the performance of the platform, data integrity and expansion of business in future.

II. ADVANTAGES OF NOSQL FOR PIXELBAY

NoSQLs have a number of strengths that they can bring to the online digital art marketplace of the PixelBay company. The data that PixelBay is likely to deal with will include customer profiles, metadata of artwork, and artist information, as well as user activity like browsing and streaming behaviour in large volumes and of constantly growing and varied volumes. The NoSQL databases offer a relaxed data structure in which these diverse types of data can be stored without a predefined schema, and more fields can be added to the platform (artwork tags, ratings or discount details) as the platform expands [2]. Besides this, NoSQL databases are horizontally scalable, thus allowing PixelBay to handle more users and more artworks as the company grows and opens more offices globally. The user experience is also enhanced by the fact that their performance is high, and they read quickly, allowing one to load an artwork gallery or user profile in a short time. In addition, NoSQL databases are highly appropriate when dealing with semi-structured data and high volumes of multimedia-related metadata that are prevalent in digital art sites such as PixelBay.

III. EVALUATION OF PIXELBAY'S DATA REQUIREMENTS

The online art marketplace of PixelBay needs a database system that is efficient, scalable, and should be able to deal with various kinds of data effectively and safely. Structured data has to be stored in the platform, customer accounts, artist information, prices of artwork, and a record of transactions, which require a high degree of accuracy and a high level of data integrity. Besides, PixelBay will handle semi-structured information, such as descriptions of artwork, categories, tags, and promotional information, which might be updated regularly as new artists and art pieces are introduced [3]. The system should also accommodate the growing data sizes as the platform grows throughout the UK and the world, indicating the need for high levels of scalability and performance. Another aspect that is vital is query complexity, as PixelBay should be able to get the listing of artwork, customer purchase history, and artist portfolios in a short time. Moreover, it must be easy to build so that the platform can be enhanced with new functionality, e.g., discounts and recommendations, and streaming, without disruptions to current data structures.

IV. TASKS

1. Comparative Analysis

Comparing Different Types of NoSQL Databases

NoSQL databases are made to handle a large amount of data that has loose structures and large scalability. MongoDB Document databases, such as MongoDB, are configured to store data in the form of JSON-like formats and suit well with user profiles and metadata of artwork. Graph databases, including Neo4j, are relationship-oriented, which is why they will suit the modelling of the relationships between customers, artists, and pieces. Key-value databases and other such as Redis store simple pairs and are very efficient in caching as well as managing sessions [4]. The wide-column databases like Cassandra are ideally structured to process large amounts of data as well as quick retrieval of data in the distributed systems, hence they are ideal for dealing with activity logs and streaming data.

Use Case of NoSQL Types

All of the types of NoSQL databases have a designated role within a digital marketplace like PixelBay. The document databases are suitable for keeping customer accounts and artwork records as they support flexible fields, including tags, description and pricing choices. Graph databases can prove useful in relation analysis, such as determining who follows whom and what to recommend based on viewing patterns. The fast access speed allows the use of key-value stores to store data about login sessions, user authentication, and shopping carts [5]. Wide-column databases can be used to handle large amounts of performance and usage data, like streaming statistics and histories of user activity.

Comparison Between NoSQL and Relational Databases

Relational databases are organised into a structured table with a defined schema, and they are well-suited to applications where high data integrity is required, as is the case with financial transactions and user records. They aid complicated queries in SQL format and implement tight relationships in the form of keys and constraints. On the contrary, NoSQL databases have loosely-typed schemas, which enable data to be stored in various formats, which are simpler to scale and adjust to the changing requirements of an application [6]. NoSQL systems are

designed to be able to scale to high-performance and big data, whereas relational databases are focused on consistency and accuracy. In the case of PixelBay, both can be used together to achieve a balance between reliable processing of transactions and scalable content management.

2. NoSQL vs Relational Databases

Discussing the Usage of NoSQL database

NoSQL databases are best suited where an application needs to be scaled to a high degree, has a flexible data structure, and be able to perform quickly on high volumes of data. NoSQL would come in handy, specifically in PixelBay, when it comes to dynamic and fast-changing information like artwork descriptions, tags, user preferences, browsing history, and information related to streaming. Since NoSQL databases are not tied to a specific schema, the addition of new attributes is easy, as PixelBay introduces new features like personalised recommendations, discount offers, or artist collaboration [7]. They can also be extended horizontally to multiple servers, and as such, they are well-suited to the proposed expansion of PixelBay into the UK and international markets. Furthermore, the read and write latency of data in NoSQL systems is lower when dealing with considerable volumes of data, and this enhances the user experience as the customer views artwork galleries or retrieves their digital collections. Nevertheless, important functions such as payments and financial records might not be consistent in terms of transactions in relational databases, as an operation on the NoSQL.

3. Implementation Demonstration

Demonstrate NoSQL and Relational Database

In order to illustrate either database methods, the data of PixelBay was applied to a relational database and a NoSQL database. Tables of customers, artists, artworks, and transactions were created in the relational database, and primary and foreign keys were used to ensure data integrity and data relationships. Sample records were entered and queries executed by SQL, e.g., to get the purchase history of customers and sales of artwork [8]. MongoDB collections were developed to store the customers, artworks and transactions as flexible documents in the NoSQL implementation. The user profiles, artist portfolios, and transaction history were accessed using sample documents and queries which reflected the scalability and flexibility of the NoSQL approach.

Implement a NoSQL database

```
> db.customers.find()
< {
  _id: 'C001',
  name: 'John Smith',
  email: 'john@pixelbay.com',
  phone: '0711111111',
  address: 'London',
  dateJoined: '2024-01-10'
}
{
  _id: 'C002',
  name: 'Sara Khan',
  email: 'sara@pixelbay.com',
  phone: '0722222222',
  address: 'Cardiff',
  dateJoined: '2024-02-15'
}
{
  _id: 'C003',
  name: 'Michael Brown',
  email: 'michael@pixelbay.com',
  phone: '0733333333',
  address: 'Swansea',
  dateJoined: '2024-03-20'
}
```

Fig 1. View all customers

```

> db.artworks.find()
< {
  _id: 'ART001',
  title: 'Sunset Waves',
  artistId: 'A001',
  releaseDate: '2024-03-01',
  price: 15.99,
  discount: 2
}
{
  _id: 'ART002',
  title: 'Urban Lights',
  artistId: 'A002',
  releaseDate: '2024-03-15',
  price: 20.5,
  discount: 1.5
}
{
  _id: 'ART003',
  title: 'Digital Forest',
  artistId: 'A003',
  releaseDate: '2024-04-10',
  price: 18.75,
  discount: 0
}

```

Fig 2. View all artworks

```

> db.artworks.find({ artistId: "A001" })
< {
  _id: 'ART001',
  title: 'Sunset Waves',
  artistId: 'A001',
  releaseDate: '2024-03-01',
  price: 15.99,
  discount: 2
}

```

Fig 3. Find artworks by TaithArts

```

> db.transactions.find({ customerId: "C001" })
< {
  _id: 'T001',
  customerId: 'C001',
  artworkId: 'ART001',
  date: '2025-01-01',
  quantity: 1,
  totalPrice: 13.99
}

```

Fig 4. View the purchase history of John Smith

```

> db.transactions.aggregate([
  {
    $lookup: {
      from: "artworks",
      localField: "artworkId",
      foreignField: "_id",
      as: "artworkDetails"
    }
  }
])
< {
  _id: 'T001',
  customerId: 'C001',
  artworkId: 'ART001',
  date: '2025-01-01',
  quantity: 1,
  totalPrice: 13.99,
  artworkDetails: [
    {
      _id: 'ART001',
      title: 'Sunset Waves',
      artistId: 'A001',
      releaseDate: '2024-03-01',
      price: 15.99,
      discount: 2
    }
  ]
}

```

Fig 5. View the purchase history of John Smith

Implement a relational database

```

118
119 /* View all customers */
120 • SELECT * FROM Customers;
121
122 /* View all artworks with artist name */
123 • SELECT A.title, B.artist_name, A.price

```

customer_id	name	email	phone	address	date_joined
1	John Smith	john@pixelbay.com	0711111111	London	2024-01-10
2	Sara Khan	sara@pixelbay.com	0722222222	Cardiff	2024-02-15
3	Michael Brown	michael@pixelbay.com	0733333333	Swansea	2024-03-20
4	Emma Wilson	emma@pixelbay.com	0744444444	Bristol	2024-04-05
5	David Lee	david@pixelbay.com	0755555555	Manchester	2024-05-18

Fig 6. Join-like query (lookup)

```

121
122 /* View all artworks with artist name */
123 • SELECT A.title, B.artist_name, A.price
124 FROM Artworks A
125 JOIN Artists B ON A.artist_id = B.artist_id;
126

```

title	artist_name	price
Sunset Waves	TaithArts	15.99
Urban Lights	Pixel Studios	20.50
Digital Forest	Digital Dream	18.75
Cyber City	NeoVision	22.00
Abstract Dreams	ArtSphere	17.99

Fig 7. View all artworks with the artist's name

```

126
127 /* View customer purchase history */
128 • SELECT C.name, A.title, T.purchase_date, T.total_price
129 FROM Transactions T
130 JOIN Customers C ON T.customer_id = C.customer_id
131 JOIN Artworks A ON T.artwork_id = A.artwork_id;

```

name	title	purchase_date	total_price
John Smith	Sunset Waves	2025-01-01	13.99
Sara Khan	Urban Lights	2025-01-02	19.00
Michael Brown	Digital Forest	2025-01-03	37.50
Emma Wilson	Cyber City	2025-01-04	19.00
David Lee	Abstract Dreams	2025-01-05	16.99

Fig 8. View customer purchase history

SQL code for relational database

```

/* =====
PIXELBAY ONLINE ART MARKETPLACE
RELATIONAL DATABASE IMPLEMENTATION
===== */

/* =====
1. CREATE DATABASE
===== */
CREATE DATABASE PixelBayDB;
USE PixelBayDB;

/* =====
2. CREATE TABLES
===== */

/* ----- CUSTOMERS TABLE ----- */
CREATE TABLE Customers (
    customer_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(120) UNIQUE NOT NULL,
    phone VARCHAR(20),
    address VARCHAR(255),
    date_joined DATE NOT NULL
);

/* ----- ARTISTS TABLE ----- */
CREATE TABLE Artists (
    artist_id INT AUTO_INCREMENT PRIMARY KEY,
    artist_name VARCHAR(120) NOT NULL,
    country VARCHAR(50),
    email VARCHAR(120)
);

/* ----- ARTWORKS TABLE ----- */
CREATE TABLE Artworks (
    artwork_id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(150) NOT NULL,
    artist_id INT NOT NULL,
    release_date DATE,
    price DECIMAL(10,2) NOT NULL,
    discount DECIMAL(5,2) DEFAULT 0,

```

```

FOREIGN KEY (artist_id) REFERENCES Artists(artist_id)
);

/* ----- TRANSACTIONS TABLE ----- */
CREATE TABLE Transactions (
    transaction_id INT AUTO INCREMENT PRIMARY KEY,
    customer_id INT NOT NULL,
    artwork_id INT NOT NULL,
    purchase_date DATE NOT NULL,
    quantity INT NOT NULL,
    total_price DECIMAL(10,2),
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id),
    FOREIGN KEY (artwork_id) REFERENCES Artworks(artwork_id)
);

/* =====
3. CHECK CONSTRAINTS
===== */

ALTER TABLE Artworks
ADD CONSTRAINT chk_price CHECK (price > 0);

ALTER TABLE Artworks
ADD CONSTRAINT chk_discount CHECK (discount >= 0);

ALTER TABLE Transactions
ADD CONSTRAINT chk_quantity CHECK (quantity > 0);

/* =====
4. INDEXES (Performance tuning)
===== */

CREATE INDEX idx_customer_email ON Customers(email);
CREATE INDEX idx_artist_name ON Artists(artist_name);
CREATE INDEX idx_artwork_title ON Artworks(title);
CREATE INDEX idx_transaction_date ON Transactions(purchase_date);

/* =====
5. INSERT SAMPLE DATA
===== */

/* ----- CUSTOMERS ----- */
INSERT INTO Customers (name, email, phone, address, date_joined) VALUES
('John Smith','john@pixelbay.com','0711111111','London','2024-01-10'),
('Sara Khan','sara@pixelbay.com','0722222222','Cardiff','2024-02-15'),
('Michael Brown','michael@pixelbay.com','0733333333','Swansea','2024-03-20'),
('Emma Wilson','emma@pixelbay.com','0744444444','Bristol','2024-04-05'),
('David Lee','david@pixelbay.com','0755555555','Manchester','2024-05-18');

/* ----- ARTISTS ----- */
INSERT INTO Artists (artist_name, country, email) VALUES
('TaithArts','UK','taith@arts.com'),
('Pixel Studios','USA','pixel@studio.com'),
('Digital Dream','Germany','dream@art.com'),
('NeoVision','Japan','neo@vision.com'),
('ArtSphere','France','artsphere@art.com');

/* ----- ARTWORKS ----- */
INSERT INTO Artworks (title, artist_id, release_date, price, discount) VALUES

```

```

('Sunset Waves',1,'2024-03-01',15.99,2.00),
('Urban Lights',2,'2024-03-15',20.50,1.50),
('Digital Forest',3,'2024-04-10',18.75,0),
('Cyber City',4,'2024-04-20',22.00,3.00),
('Abstract Dreams',5,'2024-05-05',17.99,1.00);

/* ----- TRANSACTIONS ----- */
INSERT INTO Transactions (customer_id, artwork_id, purchase_date, quantity, total_price) VALUES
(1,1,'2025-01-01',1,13.99),
(2,2,'2025-01-02',1,19.00),
(3,3,'2025-01-03',2,37.50),
(4,4,'2025-01-04',1,19.00),
(5,5,'2025-01-05',1,16.99);

/* =====
6. SAMPLE QUERIES
===== */

/* View all customers */
SELECT * FROM Customers;

/* View all artworks with artist name */
SELECT A.title, B.artist_name, A.price
FROM Artworks A
JOIN Artists B ON A.artist_id = B.artist_id;

/* View customer purchase history */
SELECT C.name, A.title, T.purchase_date, T.total_price
FROM Transactions T
JOIN Customers C ON T.customer_id = C.customer_id
JOIN Artworks A ON T.artwork_id = A.artwork_id;

```

Queries for NoSQL database

```

db.createCollection("customers")
{ ok: 1 }
db.createCollection("artists")
{ ok: 1 }
db.createCollection("artworks")
{ ok: 1 }
db.createCollection("transactions")
{ ok: 1 }
db.customers.insertMany([
{
  _id: "C001",
  name: "John Smith",
  email: "john@pixelbay.com",
  phone: "071 1111111",
  address: "London",
  dateJoined: "2024-01-10"
},
{
  _id: "C002",
  name: "Sara Khan",
  email: "sara@pixelbay.com",
  phone: "072 2222222",
  address: "Cardiff",
  dateJoined: "2024-02-15"
},
{

```

```

    _id: "C003",
    name: "Michael Brown",
    email: "michael@pixelbay.com",
    phone: "0733333333",
    address: "Swansea",
    dateJoined: "2024-03-20"
  },
  {
    _id: "C004",
    name: "Emma Wilson",
    email: "emma@pixelbay.com",
    phone: "0744444444",
    address: "Bristol",
    dateJoined: "2024-04-05"
  },
  {
    _id: "C005",
    name: "David Lee",
    email: "david@pixelbay.com",
    phone: "0755555555",
    address: "Manchester",
    dateJoined: "2024-05-18"
  }
])
{
  acknowledged: true,
  insertedIds: {
    '0': 'C001',
    '1': 'C002',
    '2': 'C003',
    '3': 'C004',
    '4': 'C005'
  }
}
}
db.artists.insertMany([
  { _id: "A001", name: "TaithArts", country: "UK", email: "taith@arts.com" },
  { _id: "A002", name: "Pixel Studios", country: "USA", email: "pixel@studio.com" },
  { _id: "A003", name: "Digital Dream", country: "Germany", email: "dream@art.com" },
  { _id: "A004", name: "NeoVision", country: "Japan", email: "neo@vision.com" },
  { _id: "A005", name: "ArtSphere", country: "France", email: "artsphere@art.com" }
])
{
  acknowledged: true,
  insertedIds: {
    '0': 'A001',
    '1': 'A002',
    '2': 'A003',
    '3': 'A004',
    '4': 'A005'
  }
}
}
db.artworks.insertMany([
  {
    _id: "ART001",
    title: "Sunset Waves",
    artistId: "A001",
    releaseDate: "2024-03-01",
    price: 15.99,
    discount: 2.00
  },
  {
    _id: "ART002",
    title: "Ocean Breeze",
    artistId: "A002",
    releaseDate: "2024-04-15",
    price: 12.50,
    discount: 1.50
  },
  {
    _id: "ART003",
    title: "Urban Dreams",
    artistId: "A003",
    releaseDate: "2024-05-01",
    price: 18.75,
    discount: 3.00
  },
  {
    _id: "ART004",
    title: "Mountain Peaks",
    artistId: "A004",
    releaseDate: "2024-06-10",
    price: 20.00,
    discount: 2.50
  },
  {
    _id: "ART005",
    title: "City Lights",
    artistId: "A005",
    releaseDate: "2024-07-01",
    price: 14.99,
    discount: 1.99
  }
])
{
  acknowledged: true,
  insertedIds: {
    '0': 'ART001',
    '1': 'ART002',
    '2': 'ART003',
    '3': 'ART004',
    '4': 'ART005'
  }
}
}

```

```

{
  _id: "ART002",
  title: "Urban Lights",
  artistId: "A002",
  releaseDate: "2024-03-15",
  price: 20.50,
  discount: 1.50
},
{
  _id: "ART003",
  title: "Digital Forest",
  artistId: "A003",
  releaseDate: "2024-04-10",
  price: 18.75,
  discount: 0
},
{
  _id: "ART004",
  title: "Cyber City",
  artistId: "A004",
  releaseDate: "2024-04-20",
  price: 22.00,
  discount: 3.00
},
{
  _id: "ART005",
  title: "Abstract Dreams",
  artistId: "A005",
  releaseDate: "2024-05-05",
  price: 17.99,
  discount: 1.00
}
])
{
  acknowledged: true,
  insertedIds: {
    '0': 'ART001',
    '1': 'ART002',
    '2': 'ART003',
    '3': 'ART004',
    '4': 'ART005'
  }
}
}
db.transactions.insertMany([
{
  _id: "T001",
  customerId: "C001",
  artworkId: "ART001",
  date: "2025-01-01",
  quantity: 1,
  totalPrice: 13.99
},
{
  _id: "T002",
  customerId: "C002",
  artworkId: "ART002",
  date: "2025-01-02",
  quantity: 1,
  totalPrice: 19.00
},

```

```

{
  _id: "T003",
  customerId: "C003",
  artworkId: "ART003",
  date: "2025-01-03",
  quantity: 2,
  totalPrice: 37.50
},
{
  _id: "T004",
  customerId: "C004",
  artworkId: "ART004",
  date: "2025-01-04",
  quantity: 1,
  totalPrice: 19.00
},
{
  _id: "T005",
  customerId: "C005",
  artworkId: "ART005",
  date: "2025-01-05",
  quantity: 1,
  totalPrice: 16.99
}
])
{
  acknowledged: true,
  insertedIds: {
    '0': 'T001',
    '1': 'T002',
    '2': 'T003',
    '3': 'T004',
    '4': 'T005'
  }
}
}
db.customers.find()
{
  _id: 'C001',
  name: 'John Smith',
  email: 'john@pixelbay.com',
  phone: '0711111111',
  address: 'London',
  dateJoined: '2024-01-10'
}
{
  _id: 'C002',
  name: 'Sara Khan',
  email: 'sara@pixelbay.com',
  phone: '0722222222',
  address: 'Cardiff',
  dateJoined: '2024-02-15'
}
{
  _id: 'C003',
  name: 'Michael Brown',
  email: 'michael@pixelbay.com',
  phone: '0733333333',
  address: 'Swansea',
  dateJoined: '2024-03-20'
}
}

```

```

{
  _id: 'C004',
  name: 'Emma Wilson',
  email: 'emma@pixelbay.com',
  phone: '0744444444',
  address: 'Bristol',
  dateJoined: '2024-04-05'
}
{
  _id: 'C005',
  name: 'David Lee',
  email: 'david@pixelbay.com',
  phone: '0755555555',
  address: 'Manchester',
  dateJoined: '2024-05-18'
}
db.artworks.find()
{
  _id: 'ART001',
  title: 'Sunset Waves',
  artistId: 'A001',
  releaseDate: '2024-03-01',
  price: 15.99,
  discount: 2
}
{
  _id: 'ART002',
  title: 'Urban Lights',
  artistId: 'A002',
  releaseDate: '2024-03-15',
  price: 20.5,
  discount: 1.5
}
{
  _id: 'ART003',
  title: 'Digital Forest',
  artistId: 'A003',
  releaseDate: '2024-04-10',
  price: 18.75,
  discount: 0
}
{
  _id: 'ART004',
  title: 'Cyber City',
  artistId: 'A004',
  releaseDate: '2024-04-20',
  price: 22,
  discount: 3
}
{
  _id: 'ART005',
  title: 'Abstract Dreams',
  artistId: 'A005',
  releaseDate: '2024-05-05',
  price: 17.99,
  discount: 1
}
db.artworks.find({ artistId: "A001" })
{
  id: 'ART001',

```

```

    title: 'Sunset Waves',
    artistId: 'A001',
    releaseDate: '2024-03-01',
    price: 15.99,
    discount: 2
  }
  db.transactions.find({ customerId: "C001" })
  {
    _id: 'T001',
    customerId: 'C001',
    artworkId: 'ART001',
    date: '2025-01-01',
    quantity: 1,
    totalPrice: 13.99
  }
  db.transactions.aggregate([
    {
      $lookup: {
        from: "artworks",
        localField: "artworkId",
        foreignField: "_id",
        as: "artworkDetails"
      }
    }
  ])

```

V. RECOMMENDED DATABASE SOLUTION

According to the analysis of PixelBay data needs and features of various database technologies, a hybrid database architecture should be chosen and implemented, which is the combination of the relational database and the NoSQL database. This can enable PixelBay to capitalise on the strengths of the two systems with minimal limitations on the same. Highly structured and important data like customer accounts, artist records, artwork pricing and transaction histories should be stored in a relational database (such as MySQL or PostgreSQL) [9]. The types of data involved here demand a high level of consistency, referential integrity and consistency in carrying out transactions, especially in purchasing and financial transactions.

Simultaneously, a flexible and large-scale data set, which includes artwork metadata, user preferences, browsing behaviour, and streaming-related information, should be managed with the help of a NoSQL database like MongoDB. The flexibility of the schema of NoSQL has allowed PixelBay to add new features like tags on the artwork, rating and other promotional information with ease without having to redesign the entire database. The fact that it is scalable horizontally also means that PixelBay will be able to accommodate more users and artworks as the platform becomes global.

The combination of both technologies using application programming interfaces (APIs) can give PixelBay a secure way of processing payments in addition to getting fast and scalable content access, which can lead to a robust and future-proof database solution.

VI. JUSTIFICATION OF THE RECOMMENDED SOLUTION

The justification of the hybrid database solution is that it is viable to meet the varied data management needs of PixelBay that are varied. The type of data that will need to be processed by PixelBay includes highly structured and transactional data, including customer details, the price of artwork and purchase history, and semi-structured and dynamic data, such as the description of artwork, tags, user preferences and viewing behaviour. Strong data organisation and integrity are maintained through primary keys, foreign keys, and constraints, which are an important part of relational databases to ensure accurate financial transactions by eliminating data inconsistency. NoSQL component offers the scalability and flexibility to handle large and quickly growing volumes of data without any degradation in performance [10]. NoSQL enables the data to be spread across several servers as PixelBay grows and additional users and artists sign up with the site, which results in high accessibility and low response times. The balance of query complexity is also matched through this method since SQL can accept complex transactional queries, and NoSQL can support fast searches and content retrieval well.

In terms of development, the flexible schema on NoSQL means that a developer does not need to make significant structural adjustments when adding new functionality, like a recommendation system and promotion campaigns.

The hybrid model has all the benefits of being the most reliable solution, more scalable, and efficient for PixelBay on its changing online marketplace.

VII. DEPLOYMENT STRATEGY

The hybrid database system suggested by PixelBay is to be deployed on a cloud-based architecture to guarantee scalability, reliability, and global accessibility. The relational database, which holds customer records, artist details and transaction data, can be hosted in a managed relational database service like Amazon RDS, Azure SQL database or Google Cloud SQL [11]. These platforms offer built-in backup, replication and security services, which are highly available and protect the data.

Meanwhile, a controlled NoSQL database like MongoDB Atlas or Amazon DynamoDB can be utilised to store malleable information like artwork metadata, user interests and browsing history. NoSQL database must be set to scale horizontally and support load balancing to support a high number of users concurrently, as well as to support large amounts of read and write operations.

The two databases must be linked to the web platform of PixelBay using secure APIs, which means that the application will be able to send financial and transactional queries to the relational database and content and user activity queries to the NoSQL database [12]. This division enhances performance and reliability. Non-stop monitoring devices are also needed to monitor the health and performance of systems, as well as storage usage, so that the database infrastructure at PixelBay can be easily expanded with the upcoming demand.

VIII. DATA SECURITY AND BACKUP

The department of data security and backup is an important attribute of the online art marketplace at PixelBay because the site deals with sensitive information about customers, digital assets and financial transactions. Individual information, including names, addresses, email addresses and transactional information, should be secured to keep the users' trust and in accordance with the data protection laws [13]. The relational and the NoSQL databases must have strong access controls, whereby only the authorised users and applications can access or modify the data. Data must be encrypted during rest and transmission to ensure that it is not intercepted or compromised by an unauthorised party.

Backups should be taken regularly to defend PixelBay against failure of the system, cyber attacks or even unintentional deletions of the data [14]. The two databases should be configured to have automated daily backups, with the copies being saved in geographically different locations to provide disaster recovery. Besides, the relational database should also be provided with point-in-time recovery capability to allow restoration of transaction data to a given point in time as the need arises.

Database activity should also be tracked and suspicious behaviour identified through audit logs and monitoring tools. All these security and backup systems can help PixelBay to provide integrity, availability, and continuity of the business in the long term.

IX. CONCLUSION AND FINAL RECOMMENDATION

The database needs of the PixelBay online art marketplace have been discussed in this report, and the relational and NoSQL database technologies have been analysed against the operational and strategic objectives of the company. PixelBay must handle well-organised and highly sensitive information like customer data, prices of artwork, and financial information, and also accommodate large amounts of flexible and dynamically changing information in the form of artwork metadata, user preferences, and prior browsing history. The discussion of various NoSQL models and their comparisons with traditional relational databases proves that none of the technologies can be used to support all the requirements of PixelBay.

According to this analysis, the most appropriate solution to be used by PixelBay is a hybrid database. Transaction processing and core business data must be stored in a relational database to have high levels of consistency, data integrity and dependable reporting. Parallel to this, there needs to be a NoSQL database that can manage scalable, large-scale, and high-performance data, including artwork catalogues, user-interactions, and information associated with streaming.

This is a hybrid solution that enables PixelBay to enjoy the stability of relational systems and the scalability of NoSQL technologies. This architecture will enable PixelBay to sustain future growth, provide a responsive user experience, and have sound, uninterrupted and correct business operations.

X. ALIGNMENT WITH PIXELBAY'S GOALS

The proposed hybrid database platform is in line with the strategic and operational objectives of PixelBay that are to establish a scalable, reliable, and user-oriented digital art marketplace. PixelBay should extend its online business operations to the UK and other global locations, as well as offer an easier experience to customers by browsing, buying, and streaming digital artworks. Following the relational database of customer accounts, prices of artwork, and records of transactions, PixelBay will provide high accuracy, consistency, and reliability of all the

financial and business-related operations [15]. This is in line with the objectives of the company to offer a safe and professional space to both artists and customers. Similarly, the NoSQL database is capable of facilitating the flexibility, novelty and speed of growth required by PixelBay. It enables the platform to save and handle various data, including the description of artworks, tags, user preferences, and data on engagement, without strict structural constraints. This will also allow PixelBay to roll out new features like personalised recommendations, promotional campaigns and advanced search features easily and fast. A combination of the two systems will provide a strong yet flexible technology base, which will enable PixelBay to grow over the long term, improve user experience, and enable the company to be competitive in the digital art market around the globe.

XI. REFERENCES

- [1] Ticovan, I.V. and Sebestyen, G., 2024, May. Forensic Investigation of Data Storage Systems. In 2024 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR) (pp. 1-7). IEEE.
- [2] Doricchi, A., Platnich, C.M., Gimpel, A., Horn, F., Earle, M., Lanzavecchia, G., Cortajarena, A.L., Liz-Marzán, L.M., Liu, N., Heckel, R. and Grass, R.N., 2022. Emerging approaches to DNA data storage: challenges and prospects. *ACS nano*, 16(11), pp.17552-17571.
- [3] Alshurideh, M.T., El Khatib, M., Al-Sulaiti, K., Sukkari, L., Al Kurdi, B., Nawaiseh, A.K., Alrawabdeh, W.A., Alshurideh, A., Shwede, F. and Alzoubi, H.M., 2024, June. Investigating the Influence of IoT on Optimizing Customer Data Warehousing, Data Mining, and Data Analysis on Categorizing Energy Consumption Behavior Data. In International Scientific Conference Management and Engineering (pp. 45-53). Cham: Springer Nature Switzerland.
- [4] Mevada, D., Panchal, H., Ahmadein, M., Zayed, M.E., Alsaleh, N.A., Djuansjah, J., Moustafa, E.B., Elsheikh, A.H. and Sadasivuni, K.K., 2022. Investigation and performance analysis of solar still with energy storage materials: An energy-exergy efficiency analysis. *Case Studies in Thermal Engineering*, 29, p.101687.
- [5] Balduf, L., Henningsen, S., Florian, M., Rust, S. and Scheuermann, B., 2022, July. Monitoring data requests in decentralized data storage systems: A case study of IPFS. In 2022 IEEE 42nd International Conference on Distributed Computing Systems (ICDCS) (pp. 658-668). IEEE.
- [6] Rahman, T. and Alharbi, T., 2024. Exploring lithium-Ion battery degradation: A concise review of critical factors, impacts, data-driven degradation estimation techniques, and sustainable directions for energy storage systems. *Batteries*, 10(7), p.220.
- [7] Wensel, C.R., Pluznick, J.L., Salzberg, S.L. and Sears, C.L., 2022. Next-generation sequencing: insights to advance clinical investigations of the microbiome. *The Journal of clinical investigation*, 132(7).
- [8] Kor, M., Yitmen, I. and Alizadehsalehi, S., 2023. An investigation for integration of deep learning and digital twins towards Construction 4.0. *Smart and Sustainable Built Environment*, 12(3), pp.461-487.
- [9] Khan, A.A., Laghari, A.A., Gadekallu, T.R., Shaikh, Z.A., Javed, A.R., Rashid, M., Estrela, V.V. and Mikhaylov, A., 2022. A drone-based data management and optimization using metaheuristic algorithms and blockchain smart contracts in a secure fog environment. *Computers and Electrical Engineering*, 102, p.108234.
- [10] Hanaysha, J.R. and Alzoubi, H.M., 2022. Investigating the impact of benefits and challenges of IOT adoption on supply chain performance and organizational performance: An empirical study in Malaysia. *Uncertain Supply Chain Management*, 10(2), pp.537-550.
- [11] Hanaysha, J.R. and Alzoubi, H.M., 2022. Investigating the impact of benefits and challenges of IOT adoption on supply chain performance and organizational performance: An empirical study in Malaysia. *Uncertain Supply Chain Management*, 10(2), pp.537-550.
- [12] Yathiraju, N., 2022. Investigating the use of an artificial intelligence model in an ERP cloud-based system. *International Journal of Electrical, Electronics and Computers*, 7(2), pp.1-26.
- [13] Scott, B., 2024. "Everyone freaks out when the leaks are made": data leaks, investigative journalism and intelligence practice. *Journal of Financial Crime*, 31(3), pp.545-557.

[14] de González, A.B., Gibson, T.M., Lee, C., Albert, P.S., Griffin, K.T., Kitahara, C.M., Liu, D., Mille, M.M., Shin, J., Bajaj, B.V. and Flood, T.E., 2023. The pediatric proton and photon therapy comparison cohort: study design for a multicenter retrospective cohort to investigate subsequent cancers after pediatric radiation therapy. *Advances in Radiation Oncology*, 8(6), p.101273.

[15] Inal, O.B., Zincir, B. and Deniz, C., 2022. Investigation on the decarbonization of shipping: An approach to hydrogen and ammonia. *International Journal of Hydrogen Energy*, 47(45), pp.19888-19900.

FINAL GRADE

GENERAL COMMENTS

The submission shows a satisfactory level of work and covers the main topics. The primary shortcoming is detailed explanation and justification

57 /100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16



Task 1 (Comparative Analysis): Partially done. Descriptions are provided, but a direct, point-by-point contrast with relational databases is limited.

Task 2 (NoSQL vs Relational): Partially done. A discussion is present but could be more detailed with concrete examples.

Report Structure: Partially complete. All required sections are mentioned but some lack development.

PAGE 17

PAGE 18

GRADING FORM: DC-DBS_CW2

KETAN MANUBHAI PANCHAL

57

ENGAGEMENT WITH LITERATURE(25)



A reference list is provided. In-text citations are included in some sections, but the IEEE citation style is not followed correctly; citations are placed within square brackets instead of using numerical superscripts. There is limited critical engagement with the sources cited. The references are listed but their content is not discussed or compared within the main body to support analysis. Current trends are mentioned in a general manner without being tied to specific literature.

15

KNOWLEDGE & UNDERSTANDING(15)



Different database types are identified and described. Relational principles and their contrast with NoSQL are outlined. Theoretical knowledge is applied to the PixelBay scenario at a general level. Data modelling is shown through the provided SQL code and database discussion, indicating an understanding of design principles for the platform.

10

COGNITIVE & INTELLECTUAL (10)



Analytical thinking is shown through the comparison of database technologies. Justification for the hybrid recommendation is provided, with reasons given for each component. The needs of PixelBay are interpreted in terms of structure, scalability, and integrity. Research, implementation results, and platform needs are combined to form a recommendation, though the connection between these elements could be more detailed.

14

PRACTICAL APPLICATION(40)



SQL code for a relational schema is provided in the report, including table creation, constraints, indexes, and sample data. A NoSQL implementation is described and example queries are given. Therefore, the implementation tasks are only partially met.

13

TRANSFERABLE SKILLS(10)



The report contains an abstract, table of contents, and uses headings. Required sections are included but some, such as the outline of deployment strategies and the importance of data security and backup, are only briefly addressed. The report is structured, but the formatting and presentation have room for improvement. In-text citations do not follow the specified IEEE format.