Q1.

Develop a data flow diagram which shows the flow of data in an organisation's information system.

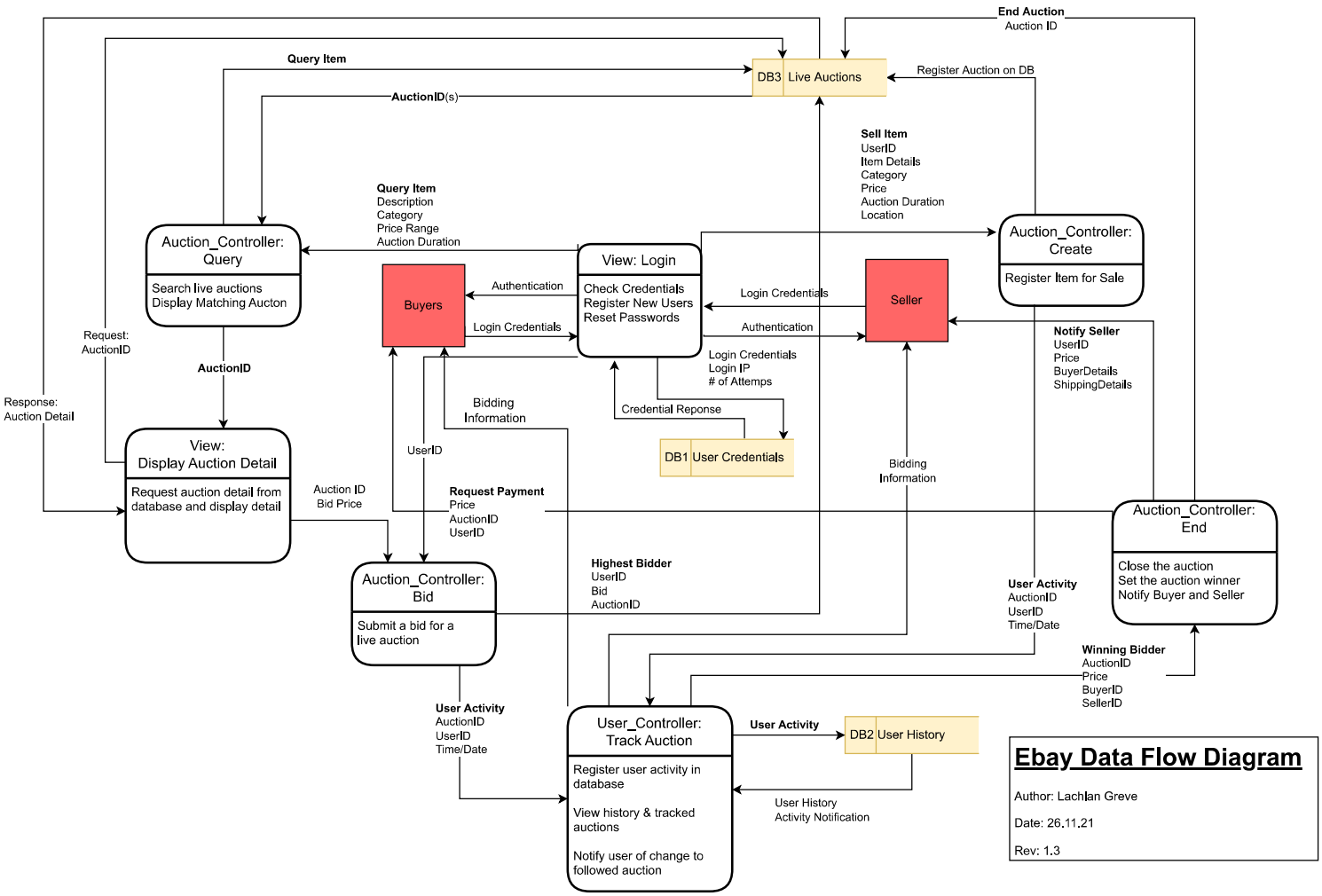

Figure 1: High-Level Dataflow Diagram of Ebay

The above figure describes at a high level the primary dataflow for the website Ebay. Ebay is a online auction house with a focus on consumer to consumer sales.

External Entities:

1. Buyers: Registered Users who are authorised to bid and participate in live auctions.
2. Sellers: Registered Users who are authorised to list items for sale on the site[1].

Processes:

1. View - Login: Receives login credentials attempts from existing users, registers new users or resets forgotten passwords by checking if the provided login credentials exist in the database `DB1: User Credentials`.
2. Create Auction: A process of the Auction Controller that is used by Sellers create new auction listings. The Seller provides details about the item they intend to sell, which are used to create a new auction listing in `DB3: Live Auctions`. This process also triggers the User Controller - track auction Process, to notify the seller of the status of the auction.
3. Query: A process of the Auction Controller that queries `DB3: Live Auctions` for live auctions that match search criteria. The database responds with matching auctions and displays a list of auctions for the user to browse.
4. View - Auction Detail: Following the Query process above, a user may select a specific auction to view further details and participate in the auction. A request containing the AuctionID is sent to `DB3: Live Auctions` and returns details about the AuctionID that are displayed to the user.
5. Bid: A process of the Auction Controller that is used by Buyers to place a bid for a live auction. Information about the bid is entered into `DB3: Live Auctions` and the Track Auction process is triggered, to notify the user of the status of the auction they are participating in.
6. Track Auction: A process of the User Controller that is triggered by user activity participating in an auction (either a buyer or a seller). The user activity is registered in `DB2: User History` which stores data about auctions that the user is currently participating in.

The databases involved in operation of the website are:

1. `DB1: User Credentials` : Stores users login credentials.
2. `DB2: User History` : Stores details about auctions that users have participated in.
3. `DB3: Live Auctions` : Stores details about live auctions.

Q2.

Develop an application architecture diagram which describes relationships between technologies and applications within an organisation’s information system.

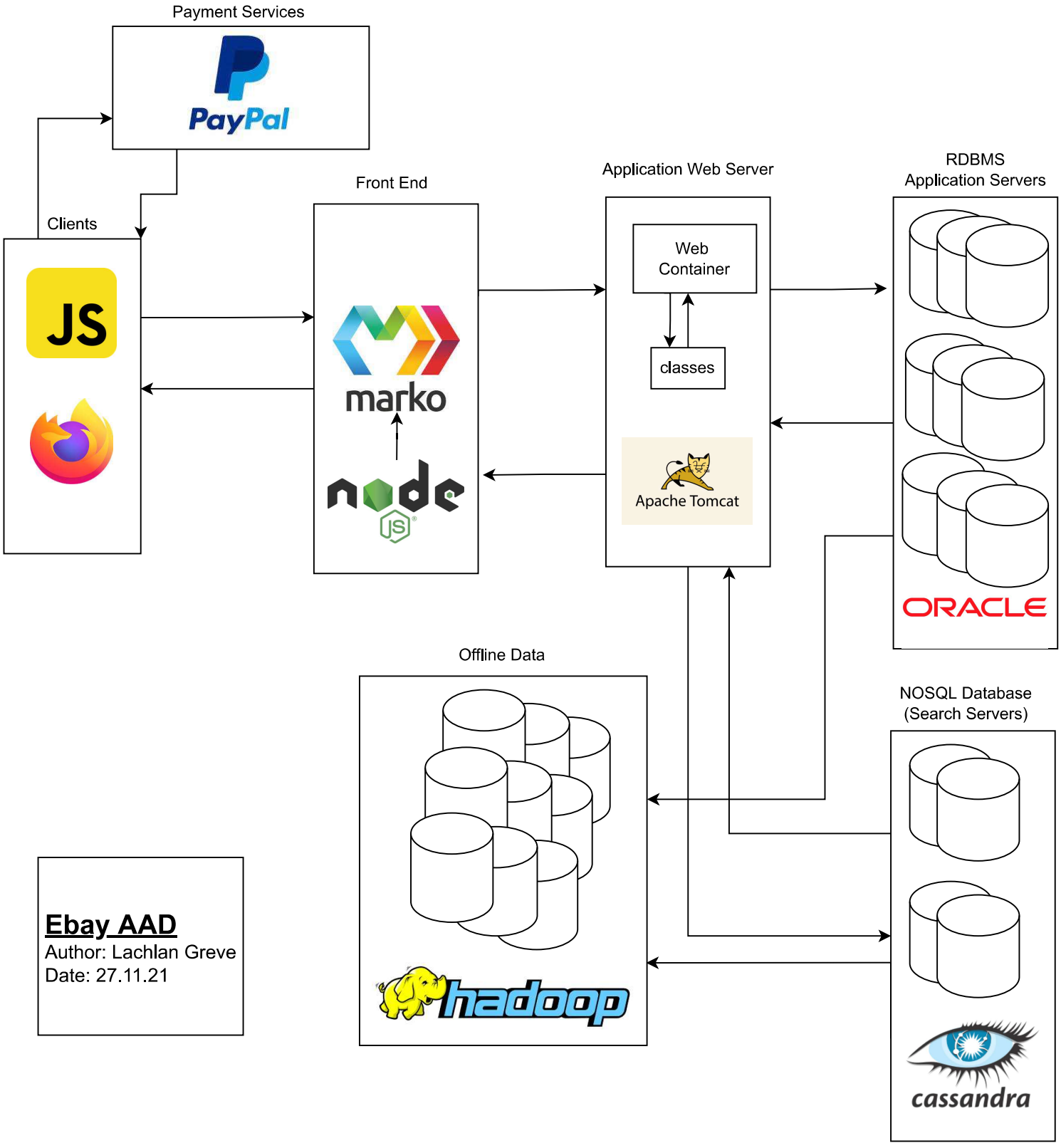

Figure 2: Application Architecture Diagram for Ebay

The Ebay application architecture ebay is strongly based on Java and Javascript, utilising a partial J2EE technology stack in order to streamline development and prioritise latency and scalability [2].

For Database Systems:

Ebay utilise a variety of different databases to address the requirements of different applications.

- Oracle RDS: RDS databases from Oracle are used to store and process ordered transactions where reliability, consistency and availability a priority - such as user login details and auction bid orders.
- Cassandra: NoSQL distributed databases are used for processing large datasets at high speed. Data stored in these databases are accessed by time critical processes such as search queries.
- Hadoop: Non time-critical, processing of historical big data is preformed offline on hadoop databases to preform process such as data analytics and machine learning optimization.

Back End:

- Apache Tomcat: Is a Java based web server environment that provides clustering support and load balancing, allowing individual instances of Ebay web applications to dynamically scale with demand.

Front End:

- Node.js: Node.js is used in the frontend layer due to its capability to stream chunks of data & html to the client (browser). This in effect improves performance latency by enabling progressive html rendering and asyncronous loading of webpages that are displayed as the data of the page is received.
- Marko: Marko is an open-source HTML Rendering, templating engine that was designed by Ebay. It operates with node.js to allow pages to allow progressive rendering that updates the view as the data of the page is received.

Q3.

Identify TWO products, services or priorities of an organisation and for each:

- explain how their application stack contributes to the delivery of the service or priority (2 marks)
- speculate why they chose the particular application or combination of applications (1 mark)

As explained by Randy Shoup [3], Ebay experiences an enormous amount of user activity:

- 250 million registered users
- 2 billion image files
- $2,000 in transactions per second

- Over 1 billion page views per day
- 100 million items in over 50,000 categories
- Over 2 petabytes of data
- 5.5 billion API calls per day

Two major priorities for the organisation are to optimise the latency and scalability of their systems.[4]

1. Latency

To increase performance and reduce user latency, Ebay partition their databases in two ways:

Functionally: Data is stored by its function on the site, for example, buying, selling and search data are all in different databases. As some of the data is common across different databases this can cause loss in efficiency as some data is redundant. However, this approach allows flexibility in the database architecture to be optimised for the specific application of the data it contains. This also allows the functionality of input data to be priortised.

Horizontally: To further improve latency Ebay's databases are segmented, or 'sharded' by set criteria such as key, range or a lookup and routing table.

The benefits of this approach is demonstrated by the following example; search data, which is time critical, is stored in Cassandra Apache databases- a readonly non-transactional NOSQL database which provides lower latency, whereas operations where consistency is critical, such as the order of bids at the end of an auction, are stored in Oracle RDBMS relational & transactional databases.

Unlike a traditional search engine, where search results can be cached periodically to provide performant response to user queries - Ebay users want events to be reflected in their search queries as soon as possible. Users sort their results by price or auction duration - variables which are constantly changing. To address this, each search database shard has a search feeder application, which polls for 'events' - changes of interest to primary databases. When events are identified this data gets injected into the search database at reliable intervals, this provides a type of intermediate persistance for the search database that is similar to a cache but that can be updated regularly at frequent polling interval.

2. Scalability

Ebay is a dynamic website where content is constantly changing. Every 2 weeks over 100,000 lines of new code is published to the site. This relates to over 48 billion SQL statements each day.

Ebay's strategy to provide scalability is to partition all components of their infrastructure whenever possible. When infrastructure components can be split, it can more easily be scaled and provides

greater cost/performance by increasing performance only of components are that are currently required. I have already described above how Ebay partition their database infrastructure for performance above.

For the application layer, Ebay utilise a Java Apache Tomcat for their web server. Apache Tomcat provides load balancing and clustering of components in web applications that can dynamically react to changes in content or demand. Java servelets are configured to operate in a stateless form, where all user persistent information is stored in a cookie, allowing each web application component to operate and scale independently.

To further enhance scalability Ebay priorities asynchronous processing whenever possible. At the front end Ebay utilise node.js which can stream chunks of html and data to the user as it becomes available from the infrastructure. Marko is an open source tool developed at Ebay that can render HTML views dynamically as it is streamed to the client from node.js.

Bibliography

1. Kalodikis, C. K. (2019, June 28). Context & Data Flow Diagram Request: The eBay Marketplace [Video]. Youtube. https://www.youtube.com/watch?v=BaTsZAYrxsQ ↩
2. ebay - ebay Tech Stack. StackShare. (2021). Retrieved 28 November 2021, from https://stackshare.io/ebay/ebay. ↩
3. IEEE C.S. [ieeeComputerSociety]. (2019, September 14). Episode 109: eBay's Architecture Principles with Randy Shoup [Video]. Youtube. https://www.youtube.com/watch?v=OYY3XR2JT3o ↩
4. Shoup, R. (2021). Scalability Best Practices: Lessons from eBay. InfoQ. Retrieved 28 November 2021, from https://www.infoq.com/articles/ebay-scalability-best-practices/. ↩